Generalization Bounds for Neural Belief Propagation Decoders

Sudarshan Adiga, Xin Xiao, Ravi Tandon, Bane Vasić, Tamal Bose

***Abstract*—Machine learning based approaches are being increasingly used for designing decoders for next generation communication systems. One widely used framework is neural belief propagation (NBP), which unfolds the belief propagation (BP) iterations into a deep neural network and the parameters are trained in a data-driven manner. NBP decoders have been shown to improve upon classical decoding algorithms. In this paper, we investigate the generalization capabilities of NBP decoders. Specifically, the generalization gap of a decoder is the difference between empirical and expected bit-error-rate(s). We present new theoretical results which bound this gap and show the dependence on the *decoder complexity*, in terms of code parameters (blocklength, message length, variable/check node degrees), decoding iterations, and the training dataset size. Results are presented for both regular and irregular parity-check matrices. To the best of our knowledge, this is the first set of theoretical results on generalization performance of neural network based decoders. We present experimental results to show the dependence of generalization gap on the training dataset size, and decoding iterations for different codes.**

***Index Terms*—Machine Learning, neural belief propagation, generalization gap, decoder complexity, code parameters, regular and irregular parity-check matrices.**

I. Introduction

DEEP neural networks have emerged as an important tool in 5G and beyond for hybrid beamforming [2]–[4], channel encoding, decoding, and estimation [5]–[19], modulation classification [20]–[22], and physical layer algorithms [23]–[25]. Within the context of channel decoding, prior works have demonstrated that deep neural network based decoders achieve lower bit/frame error rates than conventional iterative decoding algorithms such as belief propagation in several signal-to-noise ratio (SNR) regimes [5], [6], [9], [14]–[16]. In another line of works [26]–[28], deep neural networks have been used to jointly design *both* encoder and decoder. Given the expansive applicability of deep neural networks for channel encoding and decoding, we note here that determining neural network architectures that generalize well to large block length codewords is an active area of research.

Iterative decoding algorithms (such as belief propagation (BP)) are commonly deployed for decoding linear codes; and are known to be equivalent to maximum aposteriori (MAP) decoding when the Tanner graph does not contain short cycles [29]. However, if the Tanner graph contains short cycles, then BP can be sub-optimal i.e., the messages passed between the variable nodes and parity check nodes cannot correctly recover the transmitted codeword [5], [30], [31]. One approach to mitigate the effect of short cycles is by generalizing the BP algorithm by means of a deep learning based approach [5]–[13]. It is shown that the weights learnt by optimizing over the training data ensure that any message repetition between the variable nodes and parity check nodes do not adversely impact the performance of BP based decoders [5], [6]. We refer to this class of belief propagation decoders as *Neural Belief Propagation* (NBP) decoders. The salient aspect of NBP decoders is that its structure is determined from the corresponding Tanner graph, and therefore its architecture is a function of the code parameters itself. Several variants of NBP decoders have been a subject of recent study [7]–[13]. In [7], the authors propose a hardware efficient implementation of the NBP decoder by reducing the number of matrix multiplications. The authors in [9] implement message passing on graph neural networks wherein the output of each variable node is computed using a sub-network. An interesting variant of NBP decoder was proposed in [11] in which the unimportant check nodes were pruned in each decoding iteration thereby resulting in a architecture that corresponds to a different parity check matrix at each iteration. The authors in [12] propose correcting the output of conventional decoding algorithms using a NBP decoder thereby combining the desirable features of both conventional and NBP decoders. A knowledge distillation based technique to learn the node activations in NBP decoder was proposed in [13].

Post-training, it is important that the NBP decoder achieves low bit-error-rate (BER) on unseen noisy codewords. Prior works on NBP decoders [7]–[13] are empirical; to the best of our knowledge there are no theoretical guarantees on the performance of NBP decoders on unseen data. To this end, given a NBP decoder, our goal is to understand how its architecture impacts its generalization gap [32], defined as the difference between empirical and expected BER(s). Motivated by the above discussion, we ask the following fundamental question: *Given a NBP decoder, what is the expected performance on unseen noisy codewords? And how is the generalization gap*

This work was supported by NSF grants CAREER 1651492, CCF-2100013, CNS-2209951, CNS-1822071, CNS-2317192, CIF-1855879, CCSS-2027844, CCSS-2052751, and NSF-ERC 1941583. Bane Vasić was also supported by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration and funded through JPL's Strategic University Research Partnerships (SURP) Program. Bane Vasić has disclosed an outside interest in Codelucida to the University of Arizona. Conflicts of interest resulting from this interest are being managed by The University of Arizona in accordance with its policies. This work was presented in part at the 2023 IEEE International Symposium on Information Theory [1]. The associate editor coordinating the review of this article and approving it for publication was Varun Jog. (Corresponding author: Ravi Tandon.)

The authors are with the Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721 USA (e-mail: adiga@arizona.edu; 7xinxiao7@arizona.edu; tandonr@arizona.edu; vasic@arizona.edu; tbose@arizona.edu).

related to code parameters, neural decoder architecture and training dataset size?

There are several approaches to obtaining generalization gap bounds in the theoretical machine learning literature, which can be classified into two primary categories. The first category comprises data-independent approaches, such as VC-dimension, Rademacher complexity of the function class, and PAC-Bayes [32]–[41]. The second category focuses on data-dependent approaches, which analyze the mutual information between the input dataset and the algorithm output [42], [43]. VC-dimension is a measure of the number of samples required to find a probably approximately correct (PAC) hypothesis from the entire hypothesis class [44]–[46]. Rademacher complexity measures the correlation between the function class and the random labels [32]; it is known that generalization bounds obtained via Rademacher complexity (and it's variants) are tighter than the bounds obtained using the VC-dimension approach [37]. Another method is the PAC-Bayes analysis, where generalization gap is bounded by the Kullback-Leibler divergence between the prior and the posterior on the learned weights. The prior is chosen to be a multi-variate normal distribution centred around the initial weights [35], [47]–[49]. In [50], the authors propose the measuring the change in the training error with respect to perturbations in the model weights as a measure of its generalizability. Recent literature has increasingly focused on analyzing machine learning-based communication systems, particularly in terms of the generalization gap. For instance, [51] investigates generalization bounds in the context of codebook design and decoder selection in both uncoded and coded communication systems. This paper specifically examines a setting where the encoder and decoder are learned in a data-driven manner. In uncoded systems, the authors consider minimum distance decoders, while in coded systems, they focus on learning decoders by maximizing the mutual information between the input and the channel output. The study highlights how the generalization gap scales with codebook size, the number of noise samples, and input dimensionality. Additionally, [52] addresses the learning of decoders for predetermined codebooks. Drawing inspiration from the support vector machine paradigm, the authors propose a nearest neighbor decoder, the parameters of which are learned in a data-driven manner. The paper also derives bounds for the generalization gap of this learned decoder. For the scope of this paper, we adopt the PAC learning framework and use Rademacher Complexity to understand the generalization gap of NBP decoders. The notations introduced throughout the paper are summarized in Table I. We next summarize our main contributions.

Main Contributions:

1) *Generalization gap as a function of the covering number of the NBP decoder*: In this paper, we first upper bound the generalization gap of a generic deep learning decoder as a function of the Rademacher complexity of the individual bits of the decoder output (which we denote as the bit-wise Rademacher complexity). We next consider NBP decoders which belong to the class of belief propagation decoders whose architecture is a function of the code parameters. We upper bound the bit-wise Rademacher complexity as a function of the *covering number* of the NBP decoder, which is the cardinality of the set of all decoders that can closely approximate the NBP decoder. The covering number analysis provides an upper bound with a linear dependence of the generalization gap on spectral norm of the weight matrices and polynomial dependence on the decoding iterations. The bound we obtain is tighter than the other approaches such as VC-dimension and PAC-Bayes approaches in which the upper bound exponentially depends on the decoding iterations.
2) *Upper bounds on bit-wise Rademacher complexity for regular and irregular parity check matrices*: We upper bound the covering number of NBP decoder in terms of the code-parameters (blocklength, variable node degree, check node degree), and the training dataset size for both regular and irregular parity check matrices. From our results, we show that the generalization gap scales with the inverse of the square root of the dataset size, linearly with the variable node degree and the decoding iterations, and the square-root of the blocklength. To the best of our knowledge, this is the first result that determines upper bounds on the generalization gap as a function of the code-parameters.
3) *Experimental evaluation of the generalization gap bounds*: We also present simulation results to validate our theoretical findings. To the best of our knowledge, this is the first work that empirically studies the generalization gap of NBP decoders. In the experimental results, we consider binary phase shift keying (BPSK) modulation and additive white Gaussian noise (AWGN) channel. We use Tanner code to illustrate the dependence of the generalization gap on the decoding iterations, and training dataset size. To study the dependence of the generalization gap on the blocklength, we consider two QC-LDPC parent codes, and generate descendent codes with smaller blocklengths by puncturing the parent code. We assume that all-zero codewords are transmitted, and the NBP decoder is trained on the noisy realizations generated for a given channel signal-to-noise ratio (SNR). In our empirical results, we observe that the generalization gap has a linear dependence on the decoding iterations, and it increases with the blocklength, thereby agreeing with the theoretically derived bounds.

II. Preliminaries and Problem Statement

In Fig. 1, we consider a linear block code denoted by $\mathcal{C}$ of blocklength n and message length k. Let the code $\mathcal{C}$ be characterized by a *regular* parity check matrix $\mathbf{H} \in \{0,1\}^{(n-k)\times n}$, and we denote the Tanner graph as $\mathcal{G} = (\mathcal{V}, \mathcal{P}, \mathcal{E})$; where $\mathcal{V} = \{v_1, \cdots, v_n\}$ is the set of variable nodes, $\mathcal{P} = \{p_1, \cdots, p_{n-k}\}$ is the set of parity check nodes, and $\mathcal{E} = \{e_1, \cdots, e_{nd_v}\}$ is the set of edges. Here, d_v represents the variable node degree, i.e., the number of parity checks a variable node participates in. Let $\{v_i, p_j\}$ denote the edge in the Tanner graph $\mathcal{G}$ connecting variable node v_i to parity check node p_j. $\mathcal{V}(v_j) = \{p_i | \mathbf{H}[i,j] = 1\}$ denote the set of parity check

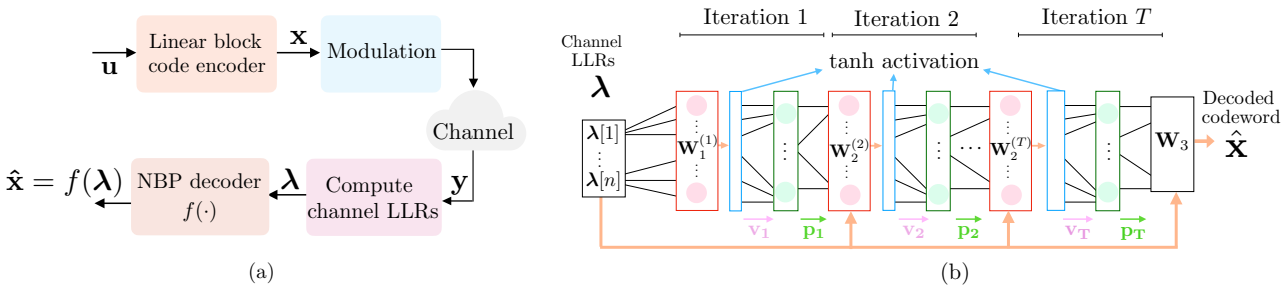

Fig. 1: (a) End-to-End block diagram for communication using neural belief propagation (NBP) decoders for linear block codes; (b) Architecture of the NBP decoder for T decoding iterations where each decoding iteration corresponds to 2 hidden layers: (1) variable node layer, (2) parity check node layer.

nodes adjacent to the variable node v_j in the Tanner graph $\mathcal{G}$. Similarly, $\mathcal{P}(p_i) = \{v_j | \mathbf{H}[i,j] = 1\}$ denote the set of variable nodes adjacent to the parity check node p_i in $\mathcal{G}$.

Let $\mathcal{Y} \subseteq \mathbb{R}^n$ be the space of n dimensional channel outputs, $\mathcal{X} \subseteq \{0,1\}^n$ be the space of n dimensional codewords, $\mathcal{U} \subseteq \{0,1\}^k$ be the space of k dimensional messages, and $\mathcal{Z} \subseteq \mathbb{R}^n$ be the space of n dimensional channel noise. The message $\mathbf{u} = [\mathbf{u}[1], \cdots, \mathbf{u}[k]]^\top \in \mathcal{U}$ is encoded to the codeword $\mathbf{x} = [\mathbf{x}[1], \cdots, \mathbf{x}[n]]^\top \in \mathcal{X}$. The channel is assumed to be memoryless, described by $\Pr(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{n} \Pr(\mathbf{y}[i]|\mathbf{x}[i])$. The receiver receives the channel output $\mathbf{y} = [\mathbf{y}[1], \cdots, \mathbf{y}[n]]^\top \in \mathcal{Y}$; which is the codeword $\mathbf{x}$ modulated, and corrupted with independently and identically distributed (i.i.d.) additive noise $\mathbf{z} = [\mathbf{z}[1], \cdots, \mathbf{z}[n]]^\top \in \mathcal{Z}$. The goal of the decoder is to recover the message $\mathbf{u}$ from the channel output $\mathbf{y}$. The input to the decoder is the log-likelihood ratio (LLR) of the posterior probabilities denoted by $\boldsymbol{\lambda} \in \mathbb{R}^{n\times 1}$ and is given as $\boldsymbol{\lambda}[i] = \log \frac{\Pr(\mathbf{x}[i]=0|\mathbf{y}[i])}{\Pr(\mathbf{x}[i]=1|\mathbf{y}[i])}$, for $1 \le i \le n$. Denote the output of the NBP decoder with T decoding iterations as $\hat{\mathbf{x}} = f(\boldsymbol{\lambda})$, where $f(\cdot)$ denotes the decoding function.

The architecture of the NBP decoder is derived from the trellis representation of $\mathcal{G}$ and illustrated in Fig. 1(b). Each decoding iteration t (where, $1 \le t \le T$) corresponds to two hidden layers each of width $|\mathcal{E}| = nd_v$, namely: (1) variable layer $\mathbf{v_t}$, (2) parity check layer $\mathbf{p_t}$. The hidden nodes in layers $\mathbf{v_t}$ and $\mathbf{p_t}$ correspond to the messages passed along the edges of the Tanner graph $\mathcal{G}$. For instance, the output of the node $\mathbf{v_t}[\{l,m\}]$ in the NBP decoder corresponds to the message passed from variable node v_l to parity check node p_m in the t-th iteration, and is given as,

$$\mathbf{v_t}[\{l,m\}] = \mathbf{W}_1^{(t)}[\{l,m\},l]\boldsymbol{\lambda}[l] + \sum_{m' \in \mathcal{V}(l)\setminus m} \mathbf{W}_2^{(t)}[\{l,m\},\{l,m'\}]\mathbf{p_{t-1}}[\{l,m'\}], \quad (1)$$

where, $\mathbf{p_{t-1}}[\{l,m'\}]$ corresponds to the message passed from the parity check node $p_{m'}$ to the variable node v_l in the $(t-1)$-th iteration. For $t = 1$, we have $\mathbf{p_0} = [0, \cdots, 0]^\top$. $\mathbf{W}_1^{(t)} \in \mathbb{R}^{nd_v \times n}$, and $\mathbf{W}_2^{(t)} \in \mathbb{R}^{nd_v \times nd_v}$ are sparse weight matrices trained using backpropagation in the t-th decoding iteration. $\mathbf{W}_1^{(t)}$ is strictly a lower triangular matrix with exactly d_v non-zero entries in every column, and one non-zero entry in every row. $\mathbf{W}_2^{(t)}$ has exactly $d_v - 1$ non-zero entries in every row, and $d_v - 1$ non-zero entries in every column. We consider that the t-th decoding iteration is characterized by weight matrices $\mathbf{W}_1^{(t)}$, and $\mathbf{W}_2^{(t)}$, where t can take integer values $t \in \{1, \cdots, T\}$. The output of the parity check node layer in the t-th decoding iteration for the NBP decoder is,

$$\mathbf{p_t}[\{l,m\}] = 2\tanh^{-1}\left(\prod_{l' \in \mathcal{P}(m)\setminus l} \tanh\left(\frac{\mathbf{v_t}[\{l',m\}]}{2}\right)\right) \quad (2)$$

Implementing (2) is computationally expensive in hardware due to the multiplicative operations and hyperbolic functions. For practical implementation, (2) can be made computationally feasible by using the min-sum operation, which is described as follows:

$$\mathbf{p_t}[\{l,m\}] = \prod_{l' \in \mathcal{P}(m)\setminus l} sign(\mathbf{v_t}[\{l',m\}]) \min_{l' \in \mathcal{P}(m)\setminus l} |\mathbf{v_t}[\{l',m\}]|. \quad (3)$$

We note that learnable parameters can be incorporated into the min-sum operations. Specifically, the output of the parity check node layer can be scaled with weights as follows:

$$\mathbf{p_t}[\{l,m\}] = \boldsymbol{\beta_t}[\{l,m\}] \prod_{l' \in \mathcal{P}(m)\setminus l} sign(\mathbf{v_t}[\{l',m\}])\ \tilde{\mathbf{p}}_\mathbf{t}[\{l,m\}]. \quad (4)$$

The parameter vector $\boldsymbol{\beta_t}$ is learned in a data-driven manner. Alternatively, the output of the parity check node layer can be offset using the parameter $\boldsymbol{\beta_t}$ as follows:

$$\mathbf{p_t}[\{l,m\}] = \prod_{l' \in \mathcal{P}(m)\setminus l} sign(\mathbf{v_t}[\{l',m\}]) \times ReLu\left(\tilde{\mathbf{p}}_\mathbf{t}[\{l,m\}] - \boldsymbol{\beta_t}[\{l,m\}]\right). \quad (5)$$

In this paper, we concentrate on the scenario where the learnable parameters are used solely for computing the output of the variable node layer. The estimated codeword after T decoding iterations in the NBP decoder is given as,

$$\hat{\mathbf{x}}[l] = s(\mathbf{W}_4^{(T)}[l,l]\boldsymbol{\lambda}[l] + \sum_{m' \in \mathcal{V}(l)} \mathbf{W}_3^{(T)}[l,\{l,m'\}]\mathbf{p_T}[\{l,m'\}]) \quad (6)$$

where, $\mathbf{W}_3 \in \mathbb{R}^{n \times nd_v}$, $\mathbf{W}_4 \in \mathbb{R}^{n \times n}$, and $s(\cdot)$ is the sigmoid activation. $\mathbf{W}_3$ is strictly an upper triangular matrix with exactly d_v non-zero entries in every row, while $\mathbf{W}_4$ is a diagonal matrix.

$\mathbf{y}[i] \rightarrow i$-th index in $\mathbf{y}$	$\mathbf{H}[i,j] \rightarrow i$-th row and j-th column entry in $\mathbf{H}$
$n \rightarrow$ Blocklength	$\mathcal{C} \rightarrow$ Linear block code of length n and dimension k
$k \rightarrow$ Dimension of the code	$f(\boldsymbol{\lambda})[j] \rightarrow j$-th output bit of NBP decoder for input $\boldsymbol{\lambda}$
$d_v \rightarrow$ Variable node degree	$S = \{(\boldsymbol{\lambda}_j, \mathbf{x}_j)\}_{j=1}^m \rightarrow$ Dataset to train NBP decoder f
$\mathbf{k} \rightarrow$ Message vector	$\mathbf{x}$, $\mathbf{y}$, $\mathbf{z} \rightarrow$ Channel input, output, and noise, respectively
$\kappa \rightarrow$ Code rate	$\boldsymbol{\lambda}$, $\hat{\mathbf{x}} \rightarrow$ Decoder input, decoder output, respectively
$T \rightarrow$ Decoding iterations	$\mathcal{F}_T \rightarrow$ Function class of NBP decoders with T iterations
$l_{\text{BER}}(\cdot) \rightarrow$ BER loss	$\mathcal{N}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) \rightarrow$ Covering number of $\mathcal{F}_T$ with respect to k^{th} norm
$\mathcal{F}_{L,T} \rightarrow$ Hypothesis class	$\mathcal{M}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) \rightarrow$ Packing number of $\mathcal{F}_T$ with respect to k^{th} norm
$\mathbf{H} \rightarrow$ Parity check matrix	$\mathcal{P}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) \rightarrow$ Packing of $\mathcal{F}_T$ with respect to the k^{th} norm
$\mathcal{G} \rightarrow$ Tanner graph	$\mathcal{R}_{\text{BER}}(f) \rightarrow$ True risk of NBP decoder f
$\mathcal{V} \rightarrow$ Variable nodes set	$\hat{\mathcal{R}}_{\text{BER}}(f) \rightarrow$ Empirical risk of NBP decoder f
$\mathcal{P} \rightarrow$ Parity check nodes set	$R_m(\mathcal{F}_{L,T}) \rightarrow$ Empirical Rademacher complexity
$\mathcal{E} \rightarrow$ Edge set in graph $\mathcal{G}$	$R_m(\mathcal{F}_T[j]) \rightarrow$ Bit-wise Rademacher complexity
$\mathbf{W}_i \rightarrow$ Weight matrices	$\{v_i, p_j\} \rightarrow$ Edge connecting variable node v_i, parity check node p_j
$B \rightarrow$ Norm bounds	$\mathbf{v_t} \rightarrow$ Output of variable node hidden layer in t-th iteration
$b \rightarrow$ Absolute value bound	$\mathbf{p_t} \rightarrow$ Output of parity-check node hidden layer in t-th iteration
$b_\lambda \rightarrow$ Bit-wise bound on $\boldsymbol{\lambda}$	$\mathbf{v_t}[\{l,m\}] \rightarrow$ Message from variable node v_l to parity node p_m
$w \rightarrow$ Bound on weights	$\mathbf{p_t}[\{l,m\}] \rightarrow$ Message from parity node p_m to variable node v_l
$\|\cdot\|_2 \rightarrow$ Spectral norm	$\mathbf{W}_1[\{l,m\}, l] \rightarrow$ Weight between l-th input $\boldsymbol{\lambda}[l]$ and node $\mathbf{v_t}[\{l,m\}]$
$\|\cdot\|_F \rightarrow$ Frobenius norm	$\mathbf{W}_2[\{l,m\}, \{l,m'\}] \rightarrow$ Weight between nodes $\mathbf{v_t}[\{l,m\}]$,$\mathbf{p_t}[\{l,m'\}]$
$\|\cdot\|_1 \rightarrow$ Max. column sum	$\mathbf{W}_3[l, \{l,m\}] \rightarrow$ Weight between $\mathbf{v_T}[\{l,m\}]$ and l-th output $\hat{\mathbf{x}}[l]$
$\|\cdot\|_\infty \rightarrow$ Max. row sum	$\mathbf{W}_4[l,l] \rightarrow$ Weight between $\boldsymbol{\lambda}[l]$ and $\hat{\mathbf{x}}[l]$
$B_{W_i} \rightarrow$ Bound on $\|\mathbf{W}_i\|_2$	$B_{w_i} \rightarrow L_2$ norm bounds of rows in $\mathbf{W}_i$, where $i \in \{1,2,3,4\}$

TABLE I: Notations used in the paper.

The NBP decoder (denoted by $f(\cdot)$) is characterized by the following four sparse weight matrices: (a) $\mathbf{W}_1^{(t)}$, where $t = 1, \cdots, T$, (b) $\mathbf{W}_2^{(t)}$, where $t = 1, \cdots, T$, (c) $\mathbf{W}_3$, and (d) $\mathbf{W}_4$. The weight matrices are learnt by training the NBP decoder to minimize the bit error rate (BER) loss that is defined as,

$$l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) = \frac{d_H(f(\boldsymbol{\lambda}), \mathbf{x})}{n} = \frac{\sum_{j=1}^n \mathbb{1}\left(f(\boldsymbol{\lambda})[j] \neq \mathbf{x}[j]\right)}{n}. \tag{7}$$

Here, $d_H(\cdot,\cdot)$ denotes the Hamming distance, and $\mathbb{1}(\cdot)$ denotes the indicator function. In practice, we train the NBP decoder to minimize the BER loss over the dataset $S = \{(\boldsymbol{\lambda}_j, \mathbf{x}_j)\}_{j=1}^m$ comprising of pairs of log-likelihood ratio and its corresponding codeword. Then, we define the empirical risk of f as $\hat{\mathcal{R}}_{\text{BER}}(f) = \frac{1}{m}\sum_{j=1}^m l_{\text{BER}}(f(\boldsymbol{\lambda}_j), \mathbf{x}_j)$. The true risk of f is defined as $\mathcal{R}_{\text{BER}}(f) = \mathbb{E}_{\boldsymbol{\lambda},\mathbf{x}}[l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x})]$.

Problem Statement. The generalization gap is defined as the difference $\mathcal{R}_{\text{BER}}(f) - \hat{\mathcal{R}}_{\text{BER}}(f)$. The main goal of this paper is to understand the behavior of the generalization gap (specifically upper bounds) as a function of a) training dataset size, m, b) the *complexity* of the NBP decoder, in terms of the number of decoding iterations T and c) code parameters, such as message length k, blocklength n, variable node degree d_v, parity check node degree d_c.

III. Main Results

In this section, we present our main results on the generalization gap for NBP decoders. Let $S = \{(\boldsymbol{\lambda}_j, \mathbf{x}_j)\}_{j=1}^m$ be the training dataset, and we assume that the dataset is i.i.d from a fixed distribution. Let $\mathcal{F}_T$ be a class of NBP decoders with T decoding iterations. For the scope of this paper, we focus on the family of NBP decoders whose non-zero weight entries are bounded by a constant w. Specifically, we assume that for every (i,j) and $1 \leq t \leq T$, $|\mathbf{W}_1^{(t)}[i,j]| \leq w$, $|\mathbf{W}_2^{(t)}[i,j]| \leq w$, $|\mathbf{W}_3[i,j]| \leq w$ and $|\mathbf{W}_4[i,j]| \leq w$, i.e., the maximum absolute value of the (i,j) coordinates for all the weight matrices are bounded by a non-negative constant w. In addition, we also assume that input log-likelihood ratio $|\boldsymbol{\lambda}[i]| \leq b_\lambda$ for all $i = 1, \ldots, n$.

We define the hypothesis class $\mathcal{F}_{L,T}$, derived from the class $\mathcal{F}_T$ of NBP decoders as follows:

$$\mathcal{F}_{L,T} = \left\{(\boldsymbol{\lambda}, \mathbf{x}) \mapsto l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) : f \in \mathcal{F}_T\right\}. \tag{8}$$

Intuitively, for each $f \in \mathcal{F}_T$, the output of the corresponding function in $\mathcal{F}_{L,T}$ is the BER loss of the decoder f. We next define the empirical Rademacher complexity of $\mathcal{F}_{L,T}$.

Definition 1. *(Rademacher complexity of $\mathcal{F}_{L,T}$) The empirical Rademacher complexity of $\mathcal{F}_{L,T}$ is defined as*

$$R_m(\mathcal{F}_{L,T}) \triangleq \mathop{\mathbb{E}}_{\sigma}\left[\sup_{f \in \mathcal{F}_T} \frac{1}{m}\sum_{i=1}^m \sigma_i l_{BER}(f(\boldsymbol{\lambda}_i), \mathbf{x}_i)\right], \tag{9}$$

where σ_i's are i.i.d. Rademacher random variables, i.e., $\Pr(\sigma_i = 1) = \Pr(\sigma_i = -1) = \frac{1}{2}$.

We note that the loss function l_{BER} takes the values between $[0,1]$; and consequently using a standard result from PAC learning literature (Theorem 3.3 in [32]), one can bound the

generalization gap in terms of $R_m(\mathcal{F}_{L,T})$. Specifically, for any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap for any $f \in F_T$ is bounded as follows:

$$\mathcal{R}_{\text{BER}}(f) - \hat{\mathcal{R}}_{\text{BER}}(f) \leq 2R_m(\mathcal{F}_{L,T}) + \sqrt{\frac{\log(1/\delta)}{2m}}. \quad (10)$$

To proceed further, we introduce bit-wise Rademacher complexity of $\mathcal{F}_T$; which is a new notion and captures the correlation between j-th channel output of the NBP decoder and a random decision (Rademacher random variable).

Definition 2. *(Bit-wise Rademacher complexity of $\mathcal{F}_T$) For a NBP decoder class $\mathcal{F}_T$, the empirical bit-wise Rademacher complexity corresponding to its j-th output bit is defined as:*

$$R_m(\mathcal{F}_T[j]) \triangleq \mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{m}\sum_{i=1}^{m}\sigma_i \cdot f(\boldsymbol{\lambda}_i)[j]\right]. \quad (11)$$

We next present Proposition 1 in which we upper bound the generalization gap as a function of the empirical bit-wise Rademacher complexity $R_m(\mathcal{F}_T[j])$.

Proposition 1. *For any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap for any NBP decoder $f \in \mathcal{F}_T$ can be upper bounded as follows,*

$$\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \frac{1}{n}\sum_{j=1}^{n} R_m(\mathcal{F}_T[j]) + \sqrt{\frac{\log(1/\delta)}{2m}}, \quad (12)$$

where $R_m(\mathcal{F}_T[j])$ denotes the bit-wise Rademacher complexity for the jth output bit.

The proof of Proposition 1 is presented in Appendix A. We now present Theorem 1 which is the main result of this paper. The main technical challenge is to bound the bit-wise Rademacher complexity $R_m(\mathcal{F}_T[j])$ as a function of the number of decoding iterations T, training dataset size m and code parameters (blocklength n and variable node degree d_v).

Theorem 1. *For any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap for any NBP decoder $f \in \mathcal{F}_T$ can be upper bounded as follows,*

$$\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}} + 12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}\log\left(8\sqrt{mn}wd_vb_\lambda\right)}, \quad (13)$$

where n denotes the blocklength, d_v is the variable node degree, T is the number of decoding iterations (number of layers in NBP), m is the training dataset size; w and b_λ are upper bounds on the weights in the NBP decoder and input log-likelihood ratio, respectively.

Proof-sketch of Theorem 1: The detailed proof of Theorem 1 is presented in Appendix B and here we briefly describe the main ideas. We first upper bound the bit-wise Rademacher complexity in terms of Dudley entropy integral (specifically, leveraging Massart's Lemma in [39] and adapting it to our problem). The resulting bound is expressed in terms of the covering number of the NBP decoder class, i.e., the *smallest* cardinality of the set of functions in $\mathcal{F}_T$ that can closely approximate the NBP decoding function f. To further bound the covering number, we first show that the NBP decoder is Lipschitz in its weight matrices which is proved in Lemma 1 (see Appendix B). In other words, for a given input, the output of the NBP decoder remains invariant to small perturbations in its weight matrices. Using this fact, we obtain a bound on the covering number of the NBP decoder class in terms of a product of covering numbers (each corresponding to a weight matrix). We then observe that the weight matrices for the NBP decoder are sparse, where the structure and number of non-zero entries is determined by the parity check matrix and the code parameters (such as blocklength n, variable node degree d_v etc.). We then use the fact that the covering number of a sparse weight matrix is always smaller than that of a non-sparse vector (of the same size as the total non-zero entries in the original sparse matrix). Using our result in Lemma 3, we can finally upper bound the bit-wise Rademacher complexity as a function of the code parameters to deduce the result in Theorem 1.

Remark 1 (Representation in Terms of Code-rate and Parity Check Node Degree). *The result in Theorem 1 can also be expressed as follows:*

$$\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}} + 12\sqrt{\frac{(nd_c^2(1-\kappa)^2T+1)(T+1)}{m}\log\left(8\sqrt{mn}wd_vb_\lambda\right)}. \quad (14)$$

We use the fact that the blocklength, message length, variable node degree, and parity check node degree are related as $nd_v = (n-k)d_c$. Using this relation in Theorem 1 we obtain (14)*. From the result in* (14) *we note that the generalization gap reduces for codes with a high code-rate κ.*

Remark 2 (Impact of the Code-parameters). *We plot the generalization gap bound obtained in Theorem 1 in Fig. 2 for blocklength $n = 100$, variable node degree $d_v = 10$, decoding iterations $T = 10$, and dataset size $m = 10^6$. To understand the dependence of the generalization gap on a parameter, we vary that parameter while keeping the values of the remaining parameters fixed. Smaller training dataset size results in overfitting, and therefore corresponds to a larger generalization gap. We observe this in Fig. 2(a), wherein the generalization gap decays as $\mathcal{O}(\frac{1}{\sqrt{m}})$. While more decoding iterations (i.e., more hidden layers) are expected to improve decoding performance, it can also overfit the training data. Therefore, we expect the generalization gap to increase with the number of decoding iterations. As seen from Fig. 2(b), we note that the generalization gap of the NBP decoder scales linearly as $\mathcal{O}(T)$. Our theoretical result in Theorem 1 tells us that the generalization gap scales with the blocklength as $\mathcal{O}(\sqrt{n})$ as shown in Fig. 2(c). However, the generalization gap scales linearly with the variable node degree as $\mathcal{O}(d_v)$ as shown in Fig. 2(d).*

Remark 3 (Comparison with Other Approaches for Bounding the Generalization Gap). *Vapnik-Chervonenkis (VC)*

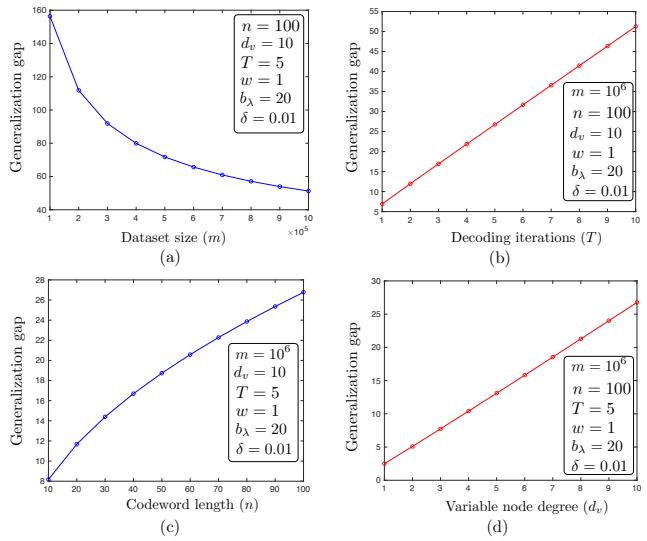

Fig. 2: (a) RHS in Theorem 1 vs Dataset size (m), (b) RHS in Theorem 1 vs Decoding iterations (T), (c) RHS in Theorem 1 vs Blocklength (n), (d) RHS in Theorem 1 vs Variable node degree (d_v).

dimension bounds [33], [34], PAC-Bayes analysis [35], [36], [49] are other techniques to upper bound the generalization gap. While VC-dimension approach yields a bound independent of the data distribution, it is found that these bounds are vacuous [35], [53] and scales exponentially with the number of parameters of the neural network. To obtain tighter and non vacuous generalization bounds, prior works [35], [54], [55] have proposed the use of PAC-Bayes analysis. For any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap using PAC-Bayes analysis is upper bounded as, $\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \sqrt{\frac{KL(\zeta||\Gamma)+\log\sqrt{m}+\log(2/\delta)}{2m}}$. The PAC-Bayes prior on the space of neural network decoders ζ is chosen independent of the training data [54], [56]. The KL divergence term between the PAC-Bayes prior ζ and posterior Γ is typically the dominant term in the bound for the generalization gap. While the posterior Γ achieves minimal empirical risk, and is data-dependent; the KL divergence term can be large as the data-independent priors are chosen arbitrarily causing the bound to be vacuous [56]. In [49], the PAC-Bayes framework is utilized to establish an upper bound on the generalization gap as a function of the network's sharpness, where sharpness is defined as the change in network output relative to the perturbation of the weight matrices. The resulting bound is a function of the spectral and Frobenius norms of the weight matrices, assuming that both the perturbation and the prior are from a Gaussian distribution. Exploring PAC-Bayes analysis for the NBP (Neural Belief Propagation) decoder paradigm, and understanding how the bound scales with different priors, is an interesting future direction. PAC-Learning approach used in this paper leads to a cleaner analysis (inspired by recent results on generalization bounds for graph neural networks and recurrent neural networks [57], [58]), and the bound obtained has a closed-form expression with explicit dependence on code parameters, decoding iterations, and the training dataset size.

We next show that Theorem 1 can be readily generalized to irregular parity check matrices. Specifically, consider an irregular parity check matrix $\mathbf{H} \in \{0,1\}^{(n-k)\times n}$ where d_{v_i} is the variable node degree of the i-th bit in the codeword, and d_{c_j} is the parity check node degree of the j-th parity check equation. The NBP decoder corresponding to such this irregular parity check matrix is characterized by the weight matrices $\{\mathbf{W}_1^{(t)}|1 \leq t \leq T\}$, $\{\mathbf{W}_2^{(t)}|1 \leq t \leq T\}$, $\mathbf{W}_3$, $\mathbf{W}_4$. Here, for every $1 \leq t \leq T$, and $\theta = \sum_{i=1}^{n} d_{v_i}$, we have that $\mathbf{W}_1^{(t)} \in \mathbb{R}^{\theta\times n}$, $\mathbf{W}_2^{(t)} \in \mathbb{R}^{\theta\times\theta}$, $\mathbf{W}_3 \in \mathbb{R}^{n\times\theta}$, and $\mathbf{W}_4 \in \mathbb{R}^{n\times n}$. For any value of t, the weight matrix $\mathbf{W}_1^{(t)}$ has one non-zero entry in every row, and d_{v_i} non-zero entries in the i-th column. In the weight matrix $\mathbf{W}_2^{(t)}$, the i-th bit in the codeword with variable node degree d_{v_i} corresponds to d_{v_i} rows and d_{v_i} columns, and these rows and columns each have exactly $d_{v_i}-1$ non-zero entries.

Theorem 2. *For any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap for any NBP decoder $f \in \mathcal{F}_T$ corresponding to irregular parity check matrix can be upper bounded as follows,*

$$\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}} + 12\sqrt{\frac{\sum_{j=1}^{n} d_{v_j}^2 (T+1)^2}{m} \log\left(8\sqrt{mn}w \max_i d_{v_i} b_\lambda\right)}. \quad (15)$$

The proof of Theorem 2 follows similar steps used to prove Theorem 1, and is presented in Appendix E.

In Theorem 1, we assumed that the log-likelihood ratios are bounded and this result does not take the channel SNR into account. We study the impact of the bound on input log-likelihood ratios and the channel SNR in Theorem 3 which is presented next.

Theorem 3. *For any $\delta \in (0,1)$, with probability at least $1-\delta$, the generalization gap for any NBP decoder $f \in \mathcal{F}_T$ with unbounded log-likelihood ratios is upper bounded as follows,*

$$\mathcal{R}_{BER}(f) - \hat{\mathcal{R}}_{BER}(f) \leq \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}} + \min_{b_\lambda} \phi(n, d_v, T, m, w, b_\lambda). \quad (16)$$

where, $\phi(n, d_v, T, m, w, b_\lambda) = \Pr(\exists i \in [n], s.t. |\boldsymbol{\lambda}[i]| > b_\lambda) + 12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m} \log(8\sqrt{mn}wd_v b_\lambda)}$. Suppose the symbols are modulated using binary phase shift keying (BPSK) modulation, and the channel is AWGN with variance β^2, then $\Pr(\exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda) = \left(1 - Q\left(\frac{\beta^2 b_\lambda + 2}{2\beta}\right) - Q\left(\frac{\beta^2 b_\lambda - 2}{2\beta}\right)\right)^n$.

The proof of Theorem 3 is presented in Appendix F. To take the unbounded input log-likelihood ratio into account for analysis, we use the law of total expectations, and condition the true risk with the event that the input log-likelihood ratio is not bounded, i.e., $|\boldsymbol{\lambda}[i]| > b_\lambda$ for any $i \in [n]$. We bound the probability that log-likelihood ratio is un-

bounded assuming BPSK modulation, and AWGN channel. In addition, we have the true risk conditioned on the event that the input log-likelihood ratio is bounded which directly follows from Theorem 1 in this paper. In (16), the term $12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}}\log\left(8\sqrt{mn}wd_vb_\lambda\right)$ is an increasing function of b_λ, and $\Pr\left(\exists i\in[n],\text{s.t.}|\boldsymbol{\lambda}[i]|>b_\lambda\right)$ is a decreasing function of b_λ. Therefore, minimizing the two terms over b_λ provides the upper bound on the generalization gap.

Remark 4 (Minimizing the generalization gap by selecting the bound on LLR (b_λ) based on Channel SNR). *The generalization gap in Theorem 3 comprises of two terms: (a) $12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}}\log\left(8\sqrt{mn}wd_vb_\lambda\right)$, which increases with b_λ; (b) $\Pr\left(\exists i\in[n],s.t.|\boldsymbol{\lambda}[i]|>b_\lambda\right)$, which is a decreasing function of b_λ, β. The choice of b_λ to minimize the total generalization gap is a function of β. To illustrate this, we plot the generalization gap bound obtained in Theorem 3, the generalization gap bound obtained in Theorem 1 for bounded input LLR, and the probability that the input LLR is unbounded in Fig. 3(a). We set blocklength $n=100$, variable node degree $d_v=10$, decoding iterations $T=10$, and dataset size $m=10^6$. As seen in Fig. 3(a), the generalization gap terms obtained in Theorem 1 are minimized in region R1 that corresponds to smaller values of β (or large channel SNR). This behavior is attributed to lower values of b_λ (i.e., $b_\lambda=10$) as observed in Fig. 3(b). As the β is increased (or reducing the channel SNR) in region R2 in Fig. 3(a), the minimum generalization gap is obtained for larger values of b_λ. The term $\Pr\left(\exists i\in[n],s.t.|\boldsymbol{\lambda}[i]|>b_\lambda\right)$ also decreases with increase in β (or lower channel SNR values). In other words, there is a trade-off between the terms $12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}}\log\left(8\sqrt{mn}wd_vb_\lambda\right)$, and $\Pr\left(\exists i\in[n],s.t.|\boldsymbol{\lambda}[i]|>b_\lambda\right)$ based on the channel SNR. We adopt this approach to establish a dependency on the channel SNR, considering that our method for bounding the generalization gap was previously data-independent. In contrast, data-dependent bounds create a direct link between the generalization gap and the mutual information involving the input dataset and the algorithm output [42], [43]. This methodology implicitly accounts for various factors, including the dataset, hypothesis set, learning algorithm, and the loss function employed. Consequently, the integration of mutual information-based approaches could be crucial in demonstrating a direct correlation between the generalization gap and the channel SNR.*

IV. Experimental Results

In this section, we present some numerical results to complement our theoretical bounds. We consider binary phase shift keying (BPSK) modulation and AWGN channel, and the received channel output for $1\le i\le n$ is given as $\mathbf{y}[i]=(-1)^{\mathbf{x}[i]}+\mathbf{z}[i]$. We focus on Tanner codes with: (i) $n=155$, $k=64$, $d_v=3$, $d_c=5$; (ii) $n=310$, $k=128$, $d_v=3$, $d_c=5$ and study the empirical generalization performance of NBP decoders whose architecture was proposed in [5], and also described in Section II of this paper. We adopt the

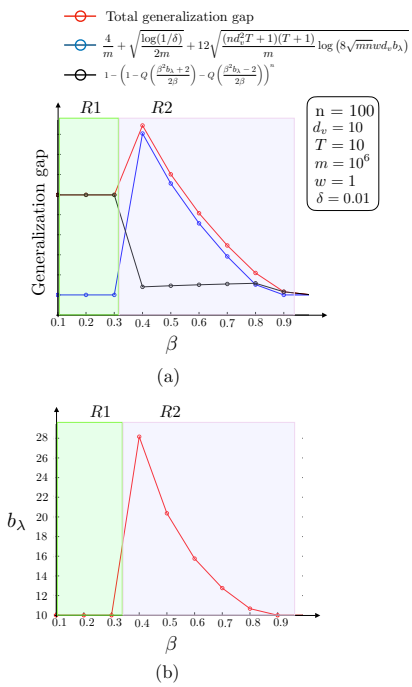

Fig. 3: (a) The total generalization gap from Theorem 3, generalization gap from Theorem 1, and the generalization gap due to unbounded log-likelihood ratio as a function of the channel SNR, (b) Selecting the bound on LLR (b_λ) to minimize the generalization gap.

software provided with the papers [6], [7] for our experiments. We train the weights of the NBP decoder until convergence by minimizing the cross-entropy loss between the true and the predicted codeword. We use ADAM optimizer for training with a learning rate of 0.01. We evaluate the NBP decoder by measuring the generalization gap (difference between average BER attained on the test and training datasets). We perform each experiment for 10 trials, and the distribution of the generalization gap over these 10 randomized runs are plotted on a boxplot. We next discuss the impact of the dataset size (m), and the decoding iterations (T) on the generalization gap.

a. Impact of training dataset size (m): We consider the NBP decoder with $T=3$ decoding iterations (equivalently, 6 layers) trained for channel SNR of 2 dB; we vary the training data set size from $m=10^3$ to $m=10^4$ in steps of 1000. From the results in Fig. 4(a), (b), we observe that the generalization gap is the largest for $m=1000$, and generally decays with m. For a smaller dataset size, the overfitting on the training samples is severe. Therefore, the NBP decoder fails to generalize on unseen samples in the test data. We also repeated the above experiment for various values of T as well as by changing SNR. We found the inverse monotonic dependence on m to be consistent across different values of T and SNR (plots are omitted due to lack of space).

b. Impact of decoding iterations (T): In this experiment, we study the impact of decoding iterations (which is proportional to the number of hidden layers) in the NBP decoder on the generalization gap. Here, we fixed channel SNR of 2 dB, training dataset size $m=10^4$ and varied T from $\{2,3,\ldots,10\}$. As seen in Fig. 5 the generalization gap grows linearly with T,

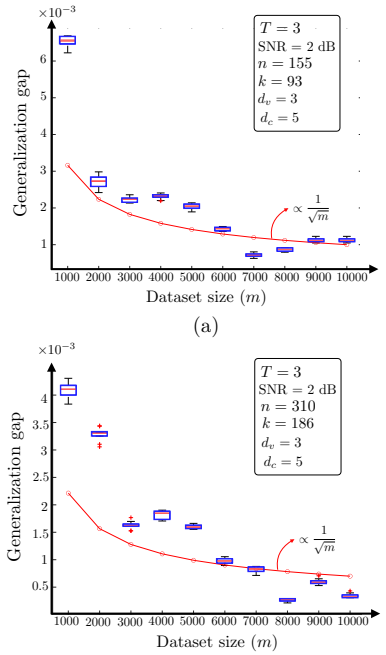

Fig. 4: Generalization gap as a function of the dataset size m at channel SNR = 2 dB for (a) Tanner code with $n = 155$, and $k = 93$, (b) Tanner code with $n = 310$, and $k = 186$.

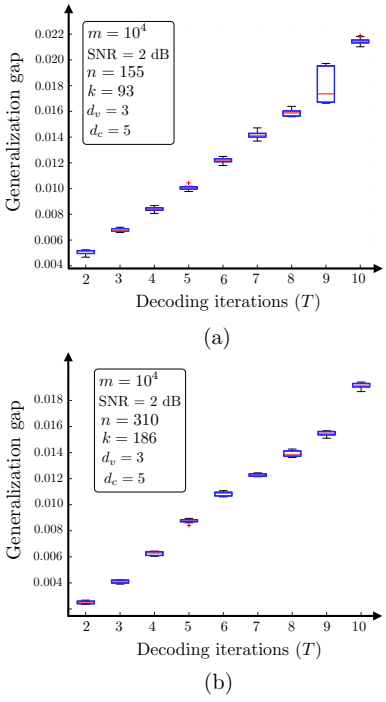

Fig. 5: Generalization gap as a function of the decoding iterations T ($\propto$ number of layers) at channel SNR = 2 dB for (a) Tanner code with $n = 155$, and $k = 93$, (b) Tanner code with $n = 310$, and $k = 186$.

which is consistent with Theorem 1 (which behaves as $\mathcal{O}(T)$). Increasing the number of parameters will cause overfitting of the NBP decoder resulting in a larger generalization gap. We note that this observation (i.e., linear dependence on T) was consistent for different dataset sizes, and channel SNR values.

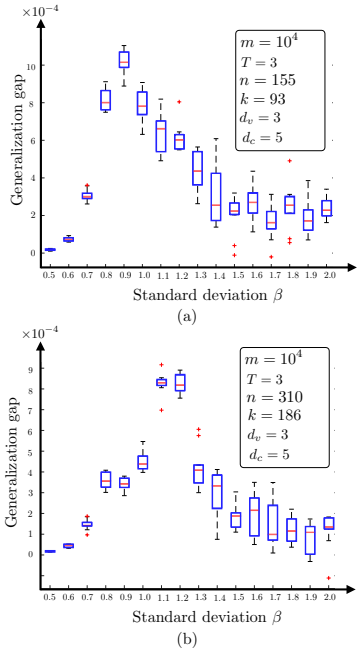

Fig. 6: Generalization gap as a function of the standard deviation of the Gaussian noise β for (a) Tanner code with $n = 155$, and $k = 93$, (b) Tanner code with $n = 310$, and $k = 186$.

c. Impact of standard deviation of the Gaussian noise (β): In this experiment, we examine the effects of varying the standard deviation of Gaussian noise, represented by β, on the generalization gap of NBP decoders. We consider NBP decoders with $T = 3$ iterations and trained on a dataset of size $m = 10^4$. According to the results depicted in Fig. 6(a) and (b), we observe that the generalization gap exhibits non-monotonic behavior in relation to β. Specifically, for $\beta \leq 1$, there is an increase in the generalization gap. Conversely, for $\beta \geq 1$, the generalization gap begins to decrease. When β is substantially large, the channel output becomes statistically independent of the input due to the increased channel noise. Consequently, this results in higher training and test BER, which in turn leads to a reduced generalization gap.

d. Impact of blocklength (n): To study the impact of blocklength keeping the variable node degree fixed, we use Progressive Edge-Growth (PEG) algorithm [59], [60] to construct two quasi-cyclic LDPC (QC-LDPC) parity check matrices with: (i) $n = 680$, $k = 340$, $d_v = 3$, $d_c = 6$; (ii) $n = 1000$, $k = 500$, $d_v = 3$, $d_c = 7$. We use the two codes as the parent code, and derive the parity check matrices for varying blocklengths keeping the variable node degree fixed by masking the columns of the parity-check matrix of the parent codes. Specifically, in Fig. 7 (a) we consider codes with blocklengths $n = 425, 510, 595$ derived from QC-LDPC parity check matrix with $n = 680$, $k = 340$; and in Fig. 7 (b) we consider codes with blocklengths $n = 600, 700, 800, 900$ derived from QC-LDPC parity check matrix with $n = 1000$, $k = 500$. We note that in addition to the blocklength, the descendent codes have different code-rates than the parent QC-LDPC code. However, this method allows us to generate descendent codes that have the same structure (or same nature) as that

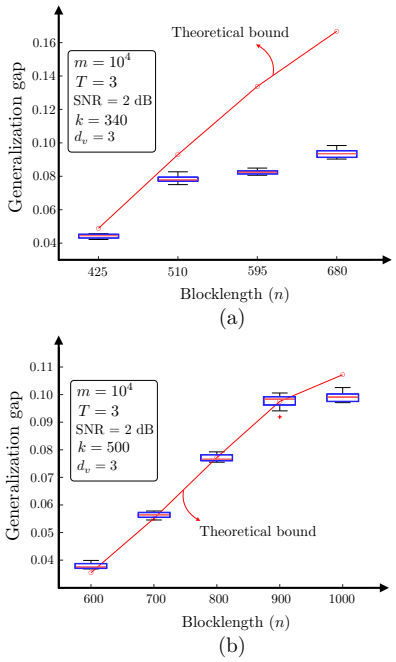

Fig. 7: Generalization gap as a function of the blocklength n at channel SNR = 2 dB for parent QC-LDPC codes with (a) $n = 1000$, and $k = 500$, (b) $n = 680$, and $k = 340$. The descendent codes with shorter blocklengths are derived by masking columns of parity check matrix in parent code.

of the parent code, thereby, eliminating the impact of varying code structures on the generalization gap. To account for the different training bit-errors for codes with varying code rates, we normalize the generalization gap with the training bit-error-rate (or the empirical risk). We also plot the theoretical bound derived in Theorem 1 normalized with the training bit-error-rate (i.e., $12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}\log\left(8\sqrt{mn}wd_vb_\lambda\right)/\hat{R}_{\text{BER}}(f)}$). We consider NBP decoder with $T = 3$ decoding iterations trained for channel SNR of 2 dB using dataset whose size $m = 10^4$. As seen in Fig. 7 the generalization gap grows with n, which is consistent with Theorem 1. The implication of this result is that for a fixed set of parity check equations, the decoding complexity increases with the blocklength; and the generalization gap is expected to increase with the blocklength for the codes with same structure.

V. Conclusions

In this work, we presented results on the generalization gap of NBP decoders as a function of training dataset size, decoding iterations and code parameters (such as blocklength, message length, variable node degree, and parity check node degree). We utilize the PAC-learning theory to express the generalization gap as a function of the Rademacher complexity of class of NBP decoders. The sparse connections in the NBP decoder architecture plays a critical role in further upper bounding the Rademacher complexity term as a function of the code parameters. Our bounds exhibit mild polynomial dependence on the blocklength n and the decoding iterations (layers) T. To the best of our knowledge, our work is the first to provide theoretical guarantees for NBP decoders corresponding to both regular and irregular parity check matrices. We also present generalizations of our theoretical result to account for different channel characteristics. We also presented comprehensive set of experiments using Tanner codes and Quasi-cyclic LDPC codes to evaluate our theoretical bounds in this paper. In our empirical results, we observe that the generalization gap increases linearly with the decoding iterations, and grows with the blocklength. In addition, we observe that the generalization gap decays with the training dataset size, thereby supporting the theoretical results in this paper. There are several interesting directions for future work, including a) obtaining generalization gap bounds for NBP decoders when the decoder is trained and tested over a range of SNR values; b) obtaining generalization bounds for ML based decoders with practical constraints (such as quantized weights); c) extending the ideas for other type of ML based codes/decoders (i.e., beyond BP type decoder architectures); d) a framework to select the code parameters (i.e., blocklength and variable node degree pairs) that minimize the generalization gap for a given channel SNR is also an important research direction.

Appendix A
Proof of Proposition 1

The proof of Proposition 1 directly follows from the Definition 1 of empirical Rademacher complexity.

$$
\begin{aligned}
R_m(\mathcal{F}_{L,T}) &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{m}\sum_{i=1}^{m}\sigma_i l_{\text{BER}}(f(\boldsymbol{\lambda}_i),\mathbf{x}_i)\right]\\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\frac{d_H(f(\boldsymbol{\lambda}_i),\mathbf{x}_i)}{n}\right]\\
&\overset{(a)}{=} \mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{mn}\sum_{i=1}^{m}\sigma_i\sum_{j=1}^{n}\left(\frac{1-f(\boldsymbol{\lambda}_i)[j]}{2}-\frac{1-\mathbf{x}_i[j]}{2}\right)^2\right]\\
&\overset{(b)}{=} \frac{1}{2}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{mn}\sum_{i=1}^{m}\sigma_i\sum_{j=1}^{n}(1-f(\boldsymbol{\lambda}_i)[j]\cdot\mathbf{x}_i[j])\right]\\
&= \frac{1}{2}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(-f(\boldsymbol{\lambda}_i)[j]\cdot(\sigma_i\cdot\mathbf{x}_i[j]))\right]. \qquad (17)
\end{aligned}
$$

where, in step (a), we consider the mapping such that $\mathbf{x}_i[j] \in \{-1,1\}$, and $f(\boldsymbol{\lambda}_i[j]) \in \{-1,1\}$; and in step (b), we express the Hamming distance $d_H(f(\boldsymbol{\lambda}_i),\mathbf{x}_i) = \frac{1}{n}\sum_{j=1}^{n}(1-f(\boldsymbol{\lambda}_i)[j]\times\mathbf{x}_i[j])$. In what follows, we take the supremum over $\mathcal{F}_T$ inside the summation. We note that the distribution of $-\sigma_i\mathbf{x}_i[j]$ and σ_i is the same, and we obtain:

$$
\begin{aligned}
R_m(\mathcal{F}_{L,T}) &\le \frac{1}{2}\mathop{\mathbb{E}}_{\sigma}\left[\sum_{j=1}^{n}\sup_{f\in\mathcal{F}_T}\frac{1}{mn}\sum_{i=1}^{m}(-f(\boldsymbol{\lambda}_i)[j]\cdot(\sigma_i\cdot\mathbf{x}_i[j]))\right]\\
&= \frac{1}{2n}\sum_{j=1}^{n}\mathop{\mathbb{E}}_{\sigma}\left[\sup_{f\in\mathcal{F}_T}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\cdot f(\boldsymbol{\lambda}_i)[j]\right]\\
&= \frac{1}{2n}\sum_{j=1}^{n}R_m(\mathcal{F}_T[j]). \qquad (18)
\end{aligned}
$$

In (18), the last step follows from the Definition 2.

APPENDIX B
PROOF OF THEOREM 1

In this appendix, we present the proof of Theorem 1. We first define the covering number and packing number of $\mathcal{F}_T$ which is the set of all NBP decoders with T decoding iterations.

Definition 3. ***(Covering Number)*** *The covering number $\mathcal{N}(\mathcal{F}_T, \epsilon, \|\cdot\|_k)$ of the set $\mathcal{F}_T$ with respect to the k-th norm for $\epsilon > 0$ is defined as*

$$\mathcal{N}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) = \min_n |\{g_1, \cdots, g_n\}|, \tag{19}$$

$$\text{s.t. } \min_{1\le i\le n} \|f(\boldsymbol{\lambda}) - g_i(\boldsymbol{\lambda})\|_k \le \epsilon. \tag{20}$$

(20) *must be satisfied for any $f \in \mathcal{F}_T$ and input log likelihood ratio $\boldsymbol{\lambda}$; then the set $\{g_1, \cdots, g_n\} \subseteq \mathcal{F}_T$ is the ϵ-cover of $\mathcal{F}_T$.*

Definition 4. ***(Packing number)*** *For a set of NBP decoders $\mathcal{F}_T$, its packing number $\mathcal{M}(\mathcal{F}_T, \epsilon, \|\cdot\|_k)$ with respect to the k-th norm for $\epsilon > 0$ is defined as*

$$\mathcal{M}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) = \max_n |\{g_1, \cdots, g_n\}|, \tag{21}$$

$$\text{s.t. } \|g_i(\boldsymbol{\lambda}) - g_j(\boldsymbol{\lambda})\|_k > \epsilon. \tag{22}$$

(22) *must be satisfied for any $g_i, g_j \in \{g_1, \cdots, g_n\}$; and the set $\mathcal{P}(\mathcal{F}_T, \epsilon, \|\cdot\|_k) = \{g_1, \cdots, g_n\} \subset \mathcal{F}_T$ that satisfies* (21)*,* (22) *is called the ϵ-packing of $\mathcal{F}_T$.*

From Proposition 1, we have that, $\mathcal{R}_{\text{BER}}(f) \le \hat{\mathcal{R}}_{\text{BER}}(f) + \frac{1}{2n}\sum_{j=1}^{n} R_m(\mathcal{F}_T[j]) + \sqrt{\frac{\log(1/\delta)}{2m}}$. Next, we use a PAC-Learning approach to bound the bit-wise Rademacher complexity term $R_m(\mathcal{F}_T[j])$ as a function of m, and spectral norm of the weight matrices of the NBP decoder. We use similar reasoning to that used in generalization bound results for graph neural networks and recurrent neural networks in [57], [58]; and we can adopt Lemma A.5. in [39] to bound the bit-wise Rademacher complexity as:

$$R_m(\mathcal{F}_T[j]) \le \inf_{\alpha>0}\left(\frac{4\alpha}{\sqrt{m}} + \frac{12}{m}\int_{\alpha}^{\sqrt{m}} \sqrt{\log \mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2)}\, d\epsilon\right). \tag{23}$$

To further upper bound the bit-wise Rademacher complexity, we make use of the upper bounds on the spectral norm of the weight matrices. Recall our assumption made in Section III that the maximum absolute value of the non-zero entries in the weight matrices are bounded by a non-negative constant w. In addition, we also use the result in [61] that the spectral norm of any matrix $\mathbf{A}$ can be upper bounded as a function of its maximum absolute column sum norm $\|\mathbf{A}\|_1$, and maximum absolute row sum norm $\|\mathbf{A}\|_\infty$ as $\|\mathbf{A}\|_2 \le \sqrt{\|\mathbf{A}\|_\infty \|\mathbf{A}\|_1}$. Using the assumption and the result in [61], the spectral norm of the weight matrices $\mathbf{W}_1^{(t)}$, $\mathbf{W}_2^{(t)}$ for any $1 \le t \le T$ can be upper bounded as:

$$B_{W_1} = \|\mathbf{W}_1^{(t)}\|_2 \le w\sqrt{d_v},$$
$$B_{W_2} = \|\mathbf{W}_2^{(t)}\|_2 \le w(d_v - 1). \tag{24}$$

Similarly, we obtain the L_2 norm bounds on for any $j-$th row vector in matrices $\mathbf{W}_3$ and $\mathbf{W}_4$ as:

$$B_{w_3} = \|\mathbf{W}_3[j,:]\|_2 \le w\sqrt{d_v},$$
$$B_{w_4} = \|\mathbf{W}_4[j,:]\|_2 \le w. \tag{25}$$

We use these spectral norm bounds in Lemma 1 in which we show that the NBP decoder is Lipschitz continuous with respect to its weight matrices. That is, for the j-th bit we have,

$$\begin{aligned}\|f(\boldsymbol{\lambda})[j] - f'(\boldsymbol{\lambda})[j]\|_2 \le& \sum_{i=1}^{T} \rho_{W_1^{(i)}} \left\|\mathbf{W}_1^{(i)} - \mathbf{W}_1^{\prime(i)}\right\|_2 \\ &+ \sum_{i=2}^{T} \rho_{W_2^{(i)}} \left\|\mathbf{W}_2^{(i)} - \mathbf{W}_2^{\prime(i)}\right\|_2 + \rho_{w_3}\|\mathbf{W}_3[j,:] - \mathbf{W}_3'[j,:]\|_2 \\ &+ \rho_{w_4}\|\mathbf{W}_4[j,:] - \mathbf{W}_4'[j,:]\|_2.\end{aligned} \tag{26}$$

In (26), ρ_{W_1}, ρ_{W_2}, ρ_{w_3}, ρ_{w_4} are Lipschitz parameters that is a function of the spectral norm bounds of the weight matrices, the L_2 norm bound of input $\boldsymbol{\lambda}$, and are given as,

$$\begin{aligned}\rho_{W_1^{(i)}} &= nb_\lambda B_{w_3}\left(\sqrt{n}B_{W_2}\right)^{T-i}, \\ \rho_{W_2^{(i)}} &= nTb_\lambda B_{W_1} B_{w_3} \frac{\left(\sqrt{n}B_{W_2}\right)^{T-1} - 1}{\sqrt{n}B_{W_2} - 1} \\ &\quad + nb_\lambda B_{W_1} B_{w_3} \left(B_{W_2}\right)^{T-2} \frac{(\sqrt{n})^{T-1} - 1}{\sqrt{n} - 1}, \\ \rho_{w_3} &= \sqrt{n}B_{W_1} b_\lambda \left(\frac{\left(B_{W_2}\right)^{T-1} - 1}{B_{W_2} - 1} + \left(B_{W_2}\right)^{T-1}\right), \\ \rho_{w_4} &= b_\lambda.\end{aligned} \tag{27}$$

As a result of (26), the covering number $\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2)$ in (23) can be upper bounded using the covering number of all distinct weight matrices as follows,

$$\begin{aligned}\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2) \le& \prod_{i=1}^{T} \mathcal{N}(\mathbf{W}_1^{(i)}, \frac{\epsilon}{(2T+1)\rho_{W_1^{(i)}}}, \|\cdot\|_F) \\ &\times \prod_{i=2}^{T} \mathcal{N}(\mathbf{W}_2^{(i)}, \frac{\epsilon}{(2T+1)\rho_{W_2^{(i)}}}, \|\cdot\|_F) \\ &\times \mathcal{N}(\mathbf{W}_3[j,:], \frac{\epsilon}{(2T+1)\rho_{w_3}}, \|\cdot\|_F) \\ &\times \mathcal{N}(\mathbf{W}_4[j,:], \frac{\epsilon}{(2T+1)\rho_{w_4}}, \|\cdot\|_F)\end{aligned} \tag{28}$$

We now upper bound the covering number of the set of decoders $\mathcal{F}_T$ in (28) using Lemma 3, in which we derive an upper bound on the covering number of sparse matrices as a function of the number of rows, columns, number of non-zero entries in the sparse matrix, and the spectral norm bound.

Bounding $\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2)$: We know that the matrices $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{W}_3$, and $\mathbf{W}_4$ have 1, $d_v - 1$, d_v, and 1 non-zero entries in each row, respectively. Using Lemma 3, the covering number of the weight matrices of the NBP decoder f for $1 \le i \le T$

can be bounded as follows:

$$\begin{aligned}
&\mathcal{N}(\mathbf{W}_1^{(i)}, \frac{\epsilon}{(2T+1)\rho_{_{W_1^{(i)}}}}, \|\cdot\|_F) \leq \left(1+\frac{(4T+2)\sqrt{n}B_{W_1}\rho_{_{W_1^{(i)}}}}{\epsilon}\right)^{nd_v}\\
&\mathcal{N}(\mathbf{W}_2^{(i)}, \frac{\epsilon}{(2T+1)\rho_{_{W_2^{(i)}}}}, \|\cdot\|_F) \leq \left(1+\frac{(4T+2)\sqrt{nd_v}B_{W_2}\rho_{_{W_2^{(i)}}}}{\epsilon}\right)^{(d_v-1)nd_v}\\
&\mathcal{N}(\mathbf{W}_3[j,:], \frac{\epsilon}{(2T+1)\rho_{_{w_3}}}, \|\cdot\|_2) \leq \left(1+\frac{(4T+2)B_{w_3}\rho_{_{w_3}}}{\epsilon}\right)^{d_v}\\
&\mathcal{N}(\mathbf{W}_4[j,:], \frac{\epsilon}{(2T+1)\rho_{_{w_4}}}, \|\cdot\|_2) \leq \left(1+\frac{(4T+2)B_{w_4}\rho_{_{w_4}}}{\epsilon}\right)
\end{aligned} \tag{29}$$

Substituting (29) in (28), we obtain,

$$\begin{aligned}
\mathcal{N}(\mathcal{F}_T[j],\epsilon,\|\cdot\|_2) \leq \prod_{i=1}^{T}\left(1+\frac{(4T+2)\sqrt{n}B_{W_1}\rho_{_{W_1^{(i)}}}}{\epsilon}\right)^{nd_v}\\
\times\prod_{i=2}^{T}\left(1+\frac{(4T+2)\sqrt{nd_v}B_{W_2}\rho_{_{W_2^{(i)}}}}{\epsilon}\right)^{(d_v-1)nd_v}\\
\times\left(1+\frac{(4T+2)B_{w_3}\rho_{_{w_3}}}{\epsilon}\right)^{d_v}\times\left(1+\frac{(4T+2)B_{w_4}\rho_{_{w_4}}}{\epsilon}\right).
\end{aligned} \tag{30}$$

Substituting the values of Lipschitz parameters obtained in (27), and assuming $nd_v > (n+1)$, the term $\mathcal{N}(\mathcal{F}_T[j],\epsilon,\|\cdot\|_2)$ can be bounded as:

$$\begin{aligned}
\mathcal{N}(\mathcal{F}_T[j],&\epsilon,\|\cdot\|_2) \leq \left(1+\frac{(4T+2)\sqrt{nd_v}\,(c_1+c_2)}{\epsilon}\right)^{(nd_v^2T+1)}.
\end{aligned} \tag{31}$$

where, $c_1 = nTB_{W_1}B_{W_2}B_{w_3}b_\lambda\frac{(\sqrt{n}B_{W_2})^{T-1}-1}{\sqrt{n}B_{W_2}-1}$, $c_2 = nB_{W_1}B_{w_3}b_\lambda\left(\sqrt{n}B_{W_2}\right)^{T-1}$. This value of c_1, and c_2 are obtained for large enough n, T, and d_v, which is an assumption that we make.

We now make use of the spectral norm bounds in (24) and (25) in (31); and further upper bound the covering number $\mathcal{N}(\mathcal{F}_T[j],\epsilon,\|\cdot\|_2)$ as,

$$\begin{aligned}
&\mathcal{N}(\mathcal{F}_T[j],\epsilon,\|\cdot\|_2) \leq \left(1+\frac{(4T+2)\sqrt{nd_v}}{\epsilon}b_\lambda(T+1)\left(\sqrt{n}wd_v\right)^{T+1}\right)^{(nd_v^2T+1)}
\end{aligned} \tag{32}$$

In (23), the integral can be further upper bounded using (32) and we have that,

$$\begin{aligned}
\int_{\alpha}^{\sqrt{m}}\sqrt{\log\mathcal{N}(\mathcal{F}_T[j],\epsilon,\|\cdot\|_2)}d\epsilon &\leq \int_{\alpha}^{\sqrt{m}}\sqrt{(nd_v^2T+1)\text{term}_1}d\epsilon\\
&\leq \sqrt{m(nd_v^2T+1)\text{term}_2}
\end{aligned} \tag{33}$$

where, $\text{term}_1 = \log\left(\frac{8\sqrt{nd_v}}{\epsilon}b_\lambda(T+1)^2\left(\sqrt{n}wd_v\right)^{T+1}\right)$; $\text{term}_2 = \log\left(8\sqrt{mnd_v}b_\lambda(T+1)^2\left(\sqrt{n}wd_v\right)^{T+1}\right)$. Substituting (33) in (23) and assuming that the term $\left(\sqrt{m}b_\lambda\right)^{T+1}$ is large enough to approximate the term inside the logarithm, we have that,

$$\begin{aligned}
\mathcal{R}_{\text{BER}}(f) - \hat{\mathcal{R}}_{\text{BER}}(f) \leq \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}}\\
+ 12\sqrt{\frac{(nd_v^2T+1)(T+1)}{m}\log\left(8\sqrt{mn}wd_vb_\lambda\right)}.
\end{aligned} \tag{34}$$

APPENDIX C
LIPSCHITZNESS IN NBP DECODERS

Lemma 1. *For n length codeword, the bit-wise output of the NBP decoder $f \in \mathcal{F}_T$ is Lipschitz in its weight matrices $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{W}_4$ such that,*

$$\begin{aligned}
&\|f(\boldsymbol{\lambda})[j] - f'(\boldsymbol{\lambda})[j]\|_2 \leq\\
&\sum_{i=1}^{T}\rho_{_{W_1^{(i)}}}\left\|\mathbf{W}_1^{(i)}-\mathbf{W}_1^{\prime(i)}\right\|_2 + \sum_{i=2}^{T}\rho_{_{W_2^{(i)}}}\left\|\mathbf{W}_2^{(i)}-\mathbf{W}_2^{\prime(i)}\right\|_2\\
&+\rho_{_{w_3}}\|\mathbf{W}_3[j,:]-\mathbf{W}_3'[j,:]\|_2 + \rho_{_{w_4}}\|\mathbf{W}_4[j,:]-\mathbf{W}_4'[j,:]\|_2.
\end{aligned}$$

The coefficients $\rho_{_{W_1}}$, $\rho_{_{W_2}}$, $\rho_{_{w_3}}$ and $\rho_{_{w_4}}$ are as follows:

$$\begin{aligned}
\rho_{_{W_1^{(i)}}} &= nb_\lambda B_{w_3}\left(\sqrt{n}B_{W_2}\right)^{T-i},\\
\rho_{_{W_2^{(i)}}} &= nTb_\lambda B_{W_1}B_{w_3}\frac{\left(\sqrt{n}B_{W_2}\right)^{T-1}-1}{\sqrt{n}B_{W_2}-1}\\
&\quad + nb_\lambda B_{W_1}B_{w_3}\left(B_{W_2}\right)^{T-2}\frac{(\sqrt{n})^{T-1}-1}{\sqrt{n}-1},\\
\rho_{_{w_3}} &= \sqrt{n}B_{W_1}b_\lambda\left(\frac{\left(B_{W_2}\right)^{T-1}-1}{B_{W_2}-1}+\left(B_{W_2}\right)^{T-1}\right),\\
\rho_{_{w_4}} &= b_\lambda.
\end{aligned} \tag{35}$$

Proof. For outputs $f(\boldsymbol{\lambda})$ and $f'(\boldsymbol{\lambda})$, respectively we consider the following parameter sets: (a) $\mathbf{W}_1^{(1)},\cdots,\mathbf{W}_1^{(T)},\mathbf{W}_2^{(1)},\cdots,\mathbf{W}_2^{(T)},\mathbf{W}_3,\mathbf{W}_4$, and (b) $\mathbf{W}_1^{\prime(1)},\cdots,\mathbf{W}_1^{\prime(T)},\mathbf{W}_2^{\prime(1)},\cdots,\mathbf{W}_2^{\prime(T)},\mathbf{W}_3',\mathbf{W}_4'$. For the j-th output in the NBP decoder, we have that,

$$\begin{aligned}
\|f(\boldsymbol{\lambda})[j] - f'(\boldsymbol{\lambda})[j]\|_2 &= \|s\left(\mathbf{W}_4[j,:]\boldsymbol{\lambda}[j]+\mathbf{W}_3[j,:]\mathbf{p_T}\right)\\
&\quad - s\left(\mathbf{W}_4'[j,:]\boldsymbol{\lambda}[j]+\mathbf{W}_3'[j,:]\mathbf{p_T'}\right)\|_2\\
&\leq \|\left(\mathbf{W}_4[j,:]-\mathbf{W}_4'[j,:]\right)\boldsymbol{\lambda}[j]+\mathbf{W}_3[j,:]\mathbf{p_T}-\mathbf{W}_3'[j,:]\mathbf{p_T}\\
&\quad + \mathbf{W}_3'[j,:]\mathbf{p_T}-\mathbf{W}_3'[j,:]\mathbf{p_T'}\|_2\\
&\leq \|\left(\mathbf{W}_4[j,:]-\mathbf{W}_4'[j,:]\right)\boldsymbol{\lambda}[j]\|_2\\
&+ \|\left(\mathbf{W}_3[j,:]-\mathbf{W}_3'[j,:]\right)\mathbf{p_T}\|_2 + \|\mathbf{W}_3'[j,:]\left(\mathbf{p_T}-\mathbf{p_T'}\right)\|_2\\
&\leq \|\mathbf{W}_4[j,:]-\mathbf{W}_4'[j,:]\|_2\, b_\lambda\\
&+ \|\mathbf{p_T}\|_2\|\mathbf{W}_3[j,:]-\mathbf{W}_3'[j,:]\|_2 + B_{w_3}\|\mathbf{p_T}-\mathbf{p_T'}\|_2.
\end{aligned} \tag{36}$$

where, $\|\mathbf{W}_3'[j,:]\|_2 \leq B_{w_3}$. We next find an upper bound $\|\mathbf{p_T}\|_2$ as a function of the number of iterations T, Lipschitz constants of the activation functions, and spectral norm bounds of the weight matrices of the NBP decoder. To further upper bound $\|\mathbf{p_T}\|_2$, we know from Lemma 2 that $\|\mathbf{p_T}\|_2 \leq \|\mathbf{v_T}\|_2$, and therefore we have

$$\begin{aligned}\|\mathbf{p_T}\|_2 \leq \|\mathbf{v_T}\|_2 &= \left\|\mathbf{W}_1^{(T)}\boldsymbol{\lambda} + \mathbf{W}_2^{(T)}\mathbf{p_{T-1}}\right\|_2 \\ &\leq \left\|\mathbf{W}_1^{(T)}\boldsymbol{\lambda}\right\|_2 + \left\|\mathbf{W}_2^{(T)}\mathbf{p_{T-1}}\right\|_2 \\ &\leq \sqrt{n}B_{W_1}b_\lambda + B_{W_2}\left\|\mathbf{p_{T-1}}\right\|_2. \end{aligned} \quad (37)$$

where, (a) follows from Talagrand's concentration lemma [62]. Applying (37) recursively across T decoding iterations we have,

$$\begin{aligned}\|\mathbf{p_T}\|_2 &\leq \|\mathbf{v_T}\|_2 \\ &\leq \sqrt{n}B_{W_1}b_\lambda \sum_{i=0}^{T-2}\left(B_{W_2}\right)^i + \left(B_{W_2}\right)^{T-1}\|\mathbf{v_1}\|_2. \end{aligned} \quad (38)$$

Also, we have $\mathbf{v_1} = \mathbf{W}_1^{(1)}\boldsymbol{\lambda}$. Therefore, $\|\mathbf{v_1}\|_2$ can be upper bounded as

$$\|\mathbf{v_1}\|_2 = \left\|\mathbf{W}_1^{(1)}\boldsymbol{\lambda}\right\|_2 \leq \sqrt{n}B_{W_1}b_\lambda. \quad (39)$$

Substituting (39) in (38), we have

$$\|\mathbf{p_T}\|_2 \leq \sqrt{n}B_{W_1}b_\lambda\left(\frac{\left(B_{W_2}\right)^{T-1}-1}{B_{W_2}-1} + \left(B_{W_2}\right)^{T-1}\right). \quad (40)$$

To upper bound $\|\mathbf{p_T} - \mathbf{p_T'}\|_2$, we express $\|\mathbf{p_T} - \mathbf{p_T'}\|_2$ in terms of $\|\mathbf{v_T} - \mathbf{v_T'}\|_2$ as follows:

$$\|\mathbf{p_T} - \mathbf{p_T'}\|_2 \leq \sqrt{n}\,\|\mathbf{v_T} - \mathbf{v_T'}\|_2, \quad (41)$$

where, $\|\mathbf{v_T} - \mathbf{v_T'}\|_2$ can be upper bounded as follows:

$$\begin{aligned}\|\mathbf{v_T} - \mathbf{v_T'}\|_2 &= \Big\|\mathbf{W}_1^{(T)}\boldsymbol{\lambda} + \mathbf{W}_2^{(T)}\mathbf{p_{T-1}} \\ &\quad - \left(\mathbf{W}_1'^{(T)}\boldsymbol{\lambda} + \mathbf{W}_2'^{(T)}\mathbf{p'_{T-1}}\right)\Big\|_2 \\ &\leq \left\|\left(\mathbf{W}_1^{(T)} - \mathbf{W}_1'^{(T)}\right)\boldsymbol{\lambda}\right\|_2 + \left\|\mathbf{W}_2^{(T)}\mathbf{p_{T-1}} - \mathbf{W}_2'^{(T)}\mathbf{p'_{T-1}}\right\|_2 \\ &\leq \sqrt{n}b_\lambda\left\|\mathbf{W}_1^{(T)} - \mathbf{W}_1'^{(T)}\right\|_2 \\ &+ \left\|\mathbf{W}_2^{(T)}\mathbf{p_{T-1}} - \mathbf{W}_2'^{(T)}\mathbf{p_{T-1}} + \mathbf{W}_2'^{(T)}\mathbf{p_{T-1}} - \mathbf{W}_2'^{(T)}\mathbf{p'_{T-1}}\right\|_2 \\ &\leq \sqrt{n}b_\lambda\left\|\mathbf{W}_1^{(T)} - \mathbf{W}_1'^{(T)}\right\|_2 + \left\|\mathbf{p_{T-1}}\right\|_2\left\|\mathbf{W}_2^{(T)} - \mathbf{W}_2'^{(T)}\right\|_2 \\ &\quad + B_{W_2}\left\|\mathbf{p_{T-1}} - \mathbf{p'_{T-1}}\right\|_2 \\ &\leq \sqrt{n}b_\lambda\left\|\mathbf{W}_1^{(T)} - \mathbf{W}_1'^{(T)}\right\|_2 + \left\|\mathbf{p_{T-1}}\right\|_2\left\|\mathbf{W}_2^{(T)} - \mathbf{W}_2'^{(T)}\right\|_2 \\ &\quad + \sqrt{n}B_{W_2}\left\|\mathbf{v_{T-1}} - \mathbf{v'_{T-1}}\right\|_2. \end{aligned} \quad (42)$$

Applying (42) recursively across T decoding iterations we have,

$$\begin{aligned}&\|\mathbf{v_T} - \mathbf{v_T'}\|_2 \\ &\leq \sqrt{n}b_\lambda\left(\sum_{i=0}^{T-2}\left(\sqrt{n}B_{W_2}\right)^i\left\|\mathbf{W}_1^{(T-i)} - \mathbf{W}_1'^{(T-i)}\right\|_2\right) \\ &+ \left(\sum_{i=1}^{T-1}\left(\sqrt{n}B_{W_2}\right)^{T-1-i}\|\mathbf{v_i}\|_2\left\|\mathbf{W}_2^{(i+1)} - \mathbf{W}_2'^{(i+1)}\right\|_2\right) \\ &+ \left(\sqrt{n}B_{W_2}\right)^{T-1}\|\mathbf{v_1} - \mathbf{v_1'}\|_2 \end{aligned} \quad (43)$$

To upper bound $\|\mathbf{v_1} - \mathbf{v_1'}\|_2$, we have that,

$$\begin{aligned}\|\mathbf{v_1} - \mathbf{v_1'}\|_2 &= \left\|\mathbf{W}_1^{(1)}\boldsymbol{\lambda} - \mathbf{W}_1'^{(1)}\boldsymbol{\lambda}\right\|_2 \\ &\leq \|\boldsymbol{\lambda}\|_2\left\|\mathbf{W}_1^{(1)} - \mathbf{W}_1'^{(1)}\right\|_2. \end{aligned} \quad (44)$$

In addition, the term $\sum_{i=1}^{T-1}\left(\sqrt{n}B_{W_2}\right)^{T-1-i}\|\mathbf{v_i}\|_2$ can be upper bounded as follows,

$$\begin{aligned}&\sum_{i=1}^{T-1}\left(\sqrt{n}B_{W_2}\right)^{T-1-i}\|\mathbf{v_i}\|_2 \\ &\leq \sum_{i=1}^{T-1}\left(\sqrt{n}B_{W_2}\right)^{T-1-i} \\ &\times \left((i-1)\sqrt{n}B_{W_1}b_\lambda + \sqrt{n}b_\lambda B_{W_1}\left(B_{W_2}\right)^{i-1}\right) \\ &\leq T\sqrt{n}B_{W_1}b_\lambda\frac{\left(\sqrt{n}B_{W_2}\right)^{T-1}-1}{\sqrt{n}B_{W_2}-1} \\ &+ \sqrt{n}b_\lambda B_{W_1}(B_{W_2})^{T-2}\sum_{i=1}^{T-1}(\sqrt{n})^{T-1-i} \\ &\leq T\sqrt{n}B_{W_1}b_\lambda\frac{\left(\sqrt{n}B_{W_2}\right)^{T-1}-1}{\sqrt{n}B_{W_2}-1} \\ &+ \sqrt{n}b_\lambda B_{W_1}(B_{W_2})^{T-2}\frac{(\sqrt{n})^{T-1}-1}{\sqrt{n}-1}. \end{aligned} \quad (45)$$

Substituting (44) and (45) in (43), we have

$$\begin{aligned}&\|\mathbf{v_T} - \mathbf{v_T'}\|_2 \\ &\leq \sqrt{n}b_\lambda\left(\sum_{i=0}^{T-1}\left(\sqrt{n}B_{W_2}\right)^i\left\|\mathbf{W}_1^{(T-i)} - \mathbf{W}_1'^{(T-i)}\right\|_2\right) \\ &+ \left(T\sqrt{n}b_\lambda B_{W_1}\frac{\left(\sqrt{n}B_{W_2}\right)^{T-1}-1}{\sqrt{n}B_{W_2}-1}\right. \\ &\left.+ \sqrt{n}b_\lambda B_{W_1}(B_{W_2})^{T-2}\frac{(\sqrt{n})^{T-1}-1}{\sqrt{n}-1}\right)\sum_{i=2}^{T}\left\|\mathbf{W}_2^{(i)} - \mathbf{W}_2'^{(i)}\right\|_2 \end{aligned} \quad (46)$$

Substituting (46) in (41) we have that,

$$\begin{aligned}
&\|\mathbf{p_T} - \mathbf{p'_T}\|_2 \\
&\leq nb_\lambda \left(\sum_{i=0}^{T-1} \left(\sqrt{n}B_{W_2}\right)^i \left\| \mathbf{W}_1^{(T-i)} - \mathbf{W}_1^{\prime(T-i)} \right\|_2 \right) \\
&+ \left(Tnb_\lambda B_{W_1} \frac{\left(\sqrt{n}B_{W_2}\right)^{T-1} - 1}{\sqrt{n}B_{W_2} - 1} \right. \\
&\left. + nb_\lambda B_{W_1} \left(B_{W_2}\right)^{T-2} \frac{(\sqrt{n})^{T-1} - 1}{\sqrt{n} - 1} \right) \sum_{i=2}^{T} \left\| \mathbf{W}_2^{(i)} - \mathbf{W}_2^{\prime(i)} \right\|_2
\end{aligned} \tag{47}$$

Using (40) and (47) in (36) we obtain

$$\begin{aligned}
&\|f(\boldsymbol{\lambda})[j] - f'(\boldsymbol{\lambda})[j]\|_2 \\
&\leq nb_\lambda B_{w_3} \left(\sum_{i=0}^{T-1} \left(\sqrt{n}B_{W_2}\right)^i \left\| \mathbf{W}_1^{(T-i)} - \mathbf{W}_1^{\prime(T-i)} \right\|_2 \right) \\
&+ nb_\lambda B_{W_1} B_{w_3} \left(T \frac{\left(\sqrt{n}B_{W_2}\right)^{T-1} - 1}{\sqrt{n}B_{W_2} - 1} \right. \\
&\left. + \left(B_{W_2}\right)^{T-2} \frac{(\sqrt{n})^{T-1} - 1}{\sqrt{n} - 1} \right) \sum_{i=2}^{T} \left\| \mathbf{W}_2^{(i)} - \mathbf{W}_2^{\prime(i)} \right\|_2 \\
&+ \sqrt{n} B_{W_1} b_\lambda \left(\frac{\left(B_{W_2}\right)^{T-1} - 1}{B_{W_2} - 1} + \left(B_{W_2}\right)^{T-1} \right) \\
&\qquad \times \|\mathbf{W}_3[j,:] - \mathbf{W}'_3[j,:]\|_2 \\
&+ b_\lambda \|\mathbf{W}_4[j,:] - \mathbf{W}'_4[j,:]\|_2 .
\end{aligned} \tag{48}$$

This completes the proof of Lemma 1. □

Next, we state and prove Lemma 2 in which we show that for any layer t, the value of $\|\mathbf{p_t}\|_2$ is less than $\|\mathbf{v_t}\|_2$. This inequality was used to obtain the result in Lemma 1.

Lemma 2. *Given the output of the parity check layer* $\mathbf{p_t}$ *defined by the min-sum operation as:*

$$\mathbf{p_t}[\{l,m\}] = \prod_{l' \in \mathcal{P}(m)\setminus l} sign(\mathbf{v_t}[\{l',m\}]) \min_{l' \in \mathcal{P}(m)\setminus l} |\mathbf{v_t}[\{l',m\}]| \tag{49}$$

Then, the norm $\|\mathbf{p_t}\|_2$ *is always bounded by the norm* $\|\mathbf{v_t}\|_2$*, i.e.,* $\|\mathbf{p_t}\|_2 \leq \|\mathbf{v_t}\|_2$.

Proof. It is straightforward to verify that for any decoding iteration the norm of output of parity check layer is always lower than the variable node layer. For n length code and k length message, let us consider parity check matrix $\mathbf{H} \in \{0,1\}^{n\times(n-k)}$ with variable node degree d_v, and parity check node degree d_c.

We denote the corresponding Tanner graph as $\mathcal{G} \in \{\mathcal{V}, \mathcal{P}, \mathcal{E}\}$, where $\mathcal{V} = \{v_1, \ldots, v_n\}$, $\mathcal{P} = \{p_1, \ldots, p_{n-k}\}$, and $\mathcal{E} = \{e_1, \ldots, e_{nd_v}\}$. Without loss of generality, let us consider parity check node p_1 in $\mathcal{G}$ such that $\mathcal{P}(p_1) = \{v_1, v_2, \ldots, v_{d_c}\}$, where $\{v_1, v_2, \ldots, v_{d_c}\} \subseteq \mathcal{V}$. Therefore, in the Tanner graph, d_c is the number of incoming messages to p_1; which translates to d_c hidden nodes in the variable check layer $\mathbf{v_t}$ for any $1 \leq t \leq T$ in the NBP decoder (whose architecture was described in Section III). Similarly, d_c is the number of outgoing messages from p_1; this translates to d_c hidden nodes in the parity check layer $\mathbf{p_t}$ in the NBP decoder.

The message passed $\mathbf{p_t}[\{1,1\}]$ from p_1 to v_1 in the NBP decoder is given as,

$$\begin{aligned}
\mathbf{p_t}[\{1,1\}] &= \prod_{l' \in \mathcal{P}(p_1)\setminus v_1} sign(\mathbf{v_t}[\{l',1\}]) \min_{l' \in \mathcal{P}(p_1)\setminus v_1} |\mathbf{v_t}[\{l',1\}]| \\
&= sign(\mathbf{v_t}[\{2,1\}]) \cdot sign(\mathbf{v_t}[\{3,1\}]) \cdots sign(\mathbf{v_t}[\{d_c,1\}]) \\
&\quad \times \min\left(|\mathbf{v_t}[\{2,1\}]|, |\mathbf{v_t}[\{3,1\}]|, \cdots, |\mathbf{v_t}[\{d_c,1\}]|\right) \\
&\overset{(a)}{=} sign(\mathbf{v_t}[\{2,1\}]) \cdot sign(\mathbf{v_t}[\{3,1\}]) \cdots sign(\mathbf{v_t}[\{d_c,1\}]) \\
&\quad \times |\mathbf{v_t}[\{2,1\}]|.
\end{aligned} \tag{50}$$

where, step (a) follows from our assumption that $|\mathbf{v_t}[\{1,1\}]| < |\mathbf{v_t}[\{2,1\}]| \cdots < |\mathbf{v_t}[\{d_c,1\}]|$, without loss of generality. Following similar steps as in (50), for the set $\{v_i | 2 \leq i \leq d_c\}$, we have that,

$$\mathbf{p_t}[\{i,1\}] = \prod_{l' \in \mathcal{P}(p_1)\setminus v_i} sign(\mathbf{v_t}[\{l',1\}]) \times |\mathbf{v_t}[\{1,1\}]| \tag{51}$$

This implies that outputs of $d_c - 1$ nodes in $\mathbf{p_t}$ correspond to the minimum absolute value of the d_c incoming messages to p_1. Therefore, for the parity check equation p_1, we have,

$$\begin{aligned}
&\left(\sum_{i=1}^{d_c} |\mathbf{p_t}[\{i,1\}]|^2 \right)^{\frac{1}{2}} \\
&\qquad \leq \left((d_c - 1)|\mathbf{v_t}[\{1,1\}]|^2 + |\mathbf{v_t}[\{2,1\}]|^2 \right)^{\frac{1}{2}}.
\end{aligned} \tag{52}$$

Following similar steps for parity checks $p_2, \cdots p_{n-k}$, we can conclude that,

$$\|\mathbf{p_t}\|_2 \leq \|\mathbf{v_t}\|_2, \tag{53}$$

where, equality in (53) is satisfied when the output of the hidden nodes in $\mathbf{v_t}$ have same absolute values. This completes the proof of Lemma 2. □

APPENDIX D
BOUND ON COVERING NUMBER OF SPARSE MATRICES

In this section, we derive an upper bound on the covering number of sparse matrices as a function of its rows, columns, and spectral norm bound.

Lemma 3. *Let* $\mathcal{W} = \{\mathbf{W} \in \mathbb{R}^{r\times c} : \|\mathbf{W}\|_2 \leq B_W$ *and* $\|\mathbf{W}[i,:]\|_0 = q,\ 1 \leq i \leq r\}$ *be the space of matrices with its spectral norm bounded by a constant* B_W*, and exactly* q *non-zero entries in each row. Then, its covering number* $\mathcal{N}(\mathbf{W}, \epsilon, \|\cdot\|_F)$ *with respect to the Frobenius norm can be upper bounded as follows:*

$$\mathcal{N}(\mathcal{W}, \epsilon, \|\cdot\|_F) \leq \left(1 + \frac{2\min\left(\sqrt{r}, \sqrt{c}\right) B_W}{\epsilon} \right)^{qr}. \tag{54}$$

where, $\epsilon > 0$ *is a constant.*

Proof. We consider $\Psi : \mathbb{R}^{r\times c} \to \mathbb{R}^{qr\times 1}$, a bijective mapping such that $\Psi(\mathbf{W}) \in \mathbb{R}^{qr\times 1}$ is a vector of all non-zero entries in $\mathbf{W} \in \mathcal{W}$. The vector space induced by the mapping Ψ is defined as ψ such that $\psi(\mathcal{W}) = \{\Psi(\mathbf{W}) : \mathbf{W} \in \mathcal{W}\}$. Therefore, we also have $\|\mathbf{W}\|_F = \|\Psi(\mathbf{W})\|_2$.

We construct $\mathcal{C}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$, a ϵ-covering of $\psi(\mathcal{W})$ under the L_2 norm, and denote the corresponding covering number by $\mathcal{N}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$. Similarly, we construct $\mathcal{C}(\mathcal{W}, \epsilon, \|\cdot\|_F)$, a ϵ-covering of $\mathcal{W}$ under $\|\cdot\|_F$, and its covering number is denoted as $\mathcal{N}(\mathcal{W}, \epsilon, \|\cdot\|_F)$. In what follows, we have,

$$\mathcal{N}(\mathcal{W}, \epsilon, \|\cdot\|_F) \overset{(a)}{\leq} \mathcal{N}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2) \overset{(b)}{\leq} \mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2). \tag{55}$$

The inequality (a) in (55) follows from the fact that Ψ is a bijective map, and that $\Psi^{-1}\mathcal{C}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ is also a ϵ-cover of the matrices in $\mathcal{W}$. The inequality (b) in (55) follows from the definition of packing $\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ and the packing number $\mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$. In other words, let $\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ be the maximal packing. Suppose for some $\mathbf{W}_i \in \mathcal{W}$ there exists $\Psi(\mathbf{U}) \in \mathcal{W} \backslash \mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ such that,

$$\|\Psi(\mathbf{W}_i) - \Psi(\mathbf{U})\|_2 \geq \epsilon. \tag{56}$$

Then, $\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ being a maximal packing is a contradiction. Therefore, $\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ is also an ϵ-cover of $\psi(\mathcal{W})$ and it follows that its cardinality (i.e., packing number) denoted by $\mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ is greater than $\mathcal{N}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$. In the next step, we upper bound $\mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$. To this end, we make use of the definitions 3, 4 to determine this upper bound; and it follows from these definitions that the balls need not be disjoint for an ϵ-cover $\mathcal{C}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$, and it must be disjoint for ϵ-packing $\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$. Therefore, $\forall \mathbf{w}_i \in \mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ we have that,

$$\bigcup_{i=1}^{\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)} \mathcal{B}(\mathbf{w}_i, \frac{\epsilon}{2}) \subset \mathcal{B}(0, R + \frac{\epsilon}{2}). \tag{57}$$

where, the radius $R = \max_{\mathbf{W} \in \mathcal{W}} \|\Psi(\mathbf{W})\|_2$. Taking volume on both sides in (57), we have that,

$$\begin{aligned} &\text{vol}\left(\bigcup_{i=1}^{\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)} \mathcal{B}(\mathbf{w}_i, \frac{\epsilon}{2})\right) \leq \text{vol}\left(\mathcal{B}(0, R + \frac{\epsilon}{2})\right) \\ \implies &\sum_{i=1}^{\mathcal{P}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)} \text{vol}\left(\mathcal{B}(\mathbf{w}_i, \frac{\epsilon}{2})\right) \leq \text{vol}\left(\mathcal{B}(0, R + \frac{\epsilon}{2})\right). \end{aligned} \tag{58}$$

To find the value of the radius R, we bound the L_2 norm of $\Psi(\mathbf{W})$ in terms of the spectral norm of the sparse matrix $\mathbf{W}$, and we have that,

$$\begin{aligned} \|\Psi(\mathbf{W})\|_2 = \|\mathbf{W}\|_F &\leq \min(\sqrt{r}, \sqrt{c}) \|\mathbf{W}\|_2 \\ &\leq \min(\sqrt{r}, \sqrt{c}) B_W. \end{aligned} \tag{59}$$

By considering the L_2 ball $\mathcal{B}(0, R)$, where radius $R = \min(\sqrt{r}, \sqrt{c}) B_W$, and from (58) we obtain an upper bound on $\mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2)$ as follows,

$$\begin{aligned} \mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2) &\leq \frac{\left(R + \frac{\epsilon}{2}\right)^{qr}}{\left(\frac{\epsilon}{2}\right)^{qr}} \\ &\leq \left(1 + \frac{2 \min(\sqrt{r}, \sqrt{c}) B_W}{\epsilon}\right)^{qr}. \end{aligned} \tag{60}$$

Therefore, we have,

$$\begin{aligned} \mathcal{N}(\mathcal{W}, \epsilon, \|\cdot\|_F) &\leq \mathcal{M}(\psi(\mathcal{W}), \epsilon, \|\cdot\|_2) \\ &\leq \left(1 + \frac{2 \min(\sqrt{r}, \sqrt{c}) B_W}{\epsilon}\right)^{qr}. \end{aligned} \tag{61}$$

This completes the proof of Lemma 3. □

APPENDIX E
PROOF OF THEOREM 2

We consider an irregular parity check matrix $\mathbf{H} \in \mathbb{R}^{(n-k) \times n}$ where d_{v_i} is the variable node degree of the i-th bit in the codeword, and d_{c_j} is the parity check node degree of the j-th parity check equation. The NBP decoder corresponding to such this irregular parity check matrix is characterized by the weight matrices $\mathbf{W}_1^{(1)}$, $\mathbf{W}_1^{(2)}, \cdots, \mathbf{W}_1^{(T)}$, $\mathbf{W}_2^{(2)}$, $\mathbf{W}_2^{(2)}, \cdots, \mathbf{W}_2^{(T)}$, $\mathbf{W}_3$, $\mathbf{W}_4$.

Here, for $t \in \{1, \cdots, T\}$, and $\theta = \sum_{i=1}^{n} d_{v_i}$, we have that $\mathbf{W}_1^{(t)} \in \mathbb{R}^{\theta \times n}$, $\mathbf{W}_2^{(t)} \in \mathbb{R}^{\theta \times \theta}$, $\mathbf{W}_3 \in \mathbb{R}^{n \times \theta}$, and $\mathbf{W}_4 \in \mathbb{R}^{n \times n}$. For any value of t, the weight matrix $\mathbf{W}_1^{(t)}$ has one non-zero entry in every row, and d_{v_i} non-zero entries in the i-th column for integer values $i \in [n]$. In the weight matrix $\mathbf{W}_2^{(t)}$, the i-th bit in the codeword with variable node degree d_{v_i} corresponds to d_{v_i} rows and d_{v_i} columns, and these rows and columns each have exactly $d_{v_i} - 1$ non-zero entries.

The i-th row in the weight matrix $\mathbf{W}_3$ corresponds to the i-th bit in the codeword, and has exactly d_{v_i} non-zero entries. Lastly, the weight matrix $\mathbf{W}_4 \in \mathbb{R}^{n \times n}$ is a a diagonal matrix. If the weights in the NBP decoder are bounded in $[-w, w]$, then for any $t \in \{1, 2, \cdots, T\}$, the spectral norm of the weight matrices $\mathbf{W}_1^{(t)}$, and $\mathbf{W}_2^{(t)}$ can be bounded as follows:

$$\begin{aligned} B_{W_1} &= \|\mathbf{W}_1^{(t)}\|_2 \leq w \sqrt{\max_i d_{v_i}}, \\ B_{W_2} &= \|\mathbf{W}_2^{(t)}\|_2 \leq w \left(\max_i d_{v_i} - 1\right) \end{aligned} \tag{62}$$

The L_2 norm bounds on the row vector in matrices $\mathbf{W}_3$ and $\mathbf{W}_4$ are as follows:

$$\begin{aligned} B_{w_3} &= \|\mathbf{W}_3[j,:]\|_2 \leq w \sqrt{d_{v_j}} \leq w \sqrt{\max_i d_{v_i}}, \\ B_{w_4} &= \|\mathbf{W}_4[j,:]\|_2 \leq w \end{aligned} \tag{63}$$

For irregular parity check matrices, the upper bound (30) becomes,

$$\begin{aligned} &\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2) \\ &\leq \prod_{i=1}^{T} \left(1 + \frac{(4T+2)\sqrt{n} B_{W_1} \rho_{W_1^{(i)}}}{\epsilon}\right)^{\sum_{h=1}^{n} d_{v_h}} \\ &\times \prod_{i=2}^{T} \left(1 + \frac{(4T+2)\sqrt{n d_v} B_{W_2} \rho_{W_2^{(i)}}}{\epsilon}\right)^{\sum_{h=1}^{n} (d_{v_h} - 1) d_{v_h}} \\ &\times \left(1 + \frac{(4T+2) B_{w_3} \rho_{w_3}}{\epsilon}\right)^{\max_h d_{v_h}} \\ &\times \left(1 + \frac{(4T+2) B_{w_4} \rho_{w_4}}{\epsilon}\right). \end{aligned} \tag{64}$$

We can further upper bound $\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2)$ in (64) as follows.

$$\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2) \leq \left(1 + \frac{(4T+2)\sqrt{nd_v}\,(c_1+c_2)}{\epsilon}\right)^{(T+1)\sum_{h=1}^{n} d_{v_h}^2}. \quad (65)$$

where, $c_1 = nTB_{W_1}B_{W_2}B_{w_3}b_\lambda \frac{(\sqrt{n}B_{W_2})^{T-1}-1}{\sqrt{n}B_{W_2}-1}$, $c_2 = nB_{W_1}B_{w_3}b_\lambda\left(\sqrt{n}B_{W_2}\right)^{T-1}$. Substituting the values of the spectral norm bounds in (62) and (63) in (65) and assuming $n >> T$, we obtain,

$$\mathcal{N}(\mathcal{F}_T[j], \epsilon, \|\cdot\|_2) \leq \left(1 + \frac{(4T+2)\sqrt{n\max_i d_{v_i}}}{\epsilon} b_\lambda(T+1)\times \left(\sqrt{n}w\max_i d_{v_i}\right)^{T+1}\right)^{(T+1)\sum_{h=1}^{n} d_{v_h}^2} \quad (66)$$

For the NBP decoder corresponding to an irregular parity check matrices, the bit-wise Rademacher complexity is upper bounded as,

$$R_m(\mathcal{F}_T[j]) \leq \frac{4}{m} + \frac{12}{m}\sqrt{m(T+1)}\times \sqrt{\sum_{h=1}^{n} d_{v_h}^2 \log\left(8\sqrt{mn\max_i d_{v_i}}\, b_\lambda (T+1)^2 \left(\sqrt{n}w\max_i d_{v_i}\right)^{T+1}\right)} \quad (67)$$

Subsequently, we have the upper bound on the true risk $\mathcal{R}_{\text{BER}}(f)$ as,

$$\mathcal{R}_{\text{BER}}(f) \leq \hat{\mathcal{R}}_{\text{BER}}(f) + \frac{4}{m} + \sqrt{\frac{\log(1/\delta)}{2m}} + 12\sqrt{\frac{\sum_{h=1}^{n} d_{v_h}^2 (T+1)^2}{m} \log\left(8\sqrt{mn}w\max_i d_{v_i} b_\lambda\right)}. \quad (68)$$

APPENDIX F
PROOF OF THEOREM 3

From law of total expectations, we have that,

$$\begin{aligned}
&\Pr(l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) > 0) \\
&= \mathbb{E}\left[l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) \middle| \forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right] \\
&\times \underbrace{\Pr\left(\forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right)}_{\text{prob. that input is bounded}} \\
&+ \mathbb{E}\left[l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) \middle| \exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda\right] \\
&\times \underbrace{\Pr\left(\exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda\right)}_{\text{prob. that input is unbounded}}
\end{aligned} \quad (69)$$

We obtain the following inequality using the fact that any probability is upper bounded by 1 for the term $\Pr\left(\forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right)$, and that the true risk conditioned on the event that the log-likelihood ratios are unbounded is also upper bounded by 1.

$$\begin{aligned}
&\Pr(l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) > 0) \\
&\leq \mathbb{E}\left[l_{\text{BER}}(f(\boldsymbol{\lambda}), \mathbf{x}) \middle| \forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right] \\
&+ \Pr\left(\exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda\right)
\end{aligned} \quad (70)$$

Using the fact that the channel outputs are i.i.d to compute $\Pr\left(\forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right)$ as,

$$\Pr\left(\forall i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| \leq b_\lambda\right) = \prod_{i=1}^{n} \Pr\left(|\boldsymbol{\lambda}[i]| \leq b_\lambda\right) \quad (71)$$

We consider that the signal is modulated by binary phase shift keying (BPSK) modulation such that $\Pr(+1) = \Pr(-1) = \frac{1}{2}$. The channel is AWGN channel with noise variance β^2, then $\boldsymbol{\lambda}[i] = 2\mathbf{y}[i]/\beta^2$. We can upper bound the term $\Pr\left(|\mathbf{y}[i]| \leq \frac{\beta^2 b_\lambda}{2}\right)$ using Q-function as follows.

$$\Pr\left(|\boldsymbol{\lambda}[i]| \leq b_\lambda\right) = \left(1 - Q\left(\frac{\beta^2 b_\lambda + 2}{2\beta}\right) - Q\left(\frac{\beta^2 b_\lambda - 2}{2\beta}\right)\right), \quad (72)$$

Then, the term $\Pr\left(\exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda\right)$ is computed as,

$$\begin{aligned}
&\Pr\left(\exists i \in [n] \text{ s.t. } |\boldsymbol{\lambda}[i]| > b_\lambda\right) \\
&= 1 - \left(1 - Q\left(\frac{\beta^2 b_\lambda + 2}{2\beta}\right) - Q\left(\frac{\beta^2 b_\lambda - 2}{2\beta}\right)\right)^n
\end{aligned} \quad (73)$$

Substituting the values of (73) in (70) completes the proof of Theorem 3.

REFERENCES

[1] S. Adiga, X. Xiao, R. Tandon, B. Vasić, and T. Bose, "Generalization bounds for neural belief propagation decoders," in *2023 IEEE International Symposium on Information Theory (ISIT)*, pp. 596–601, IEEE, 2023.

[2] X. Li and A. Alkhateeb, "Deep Learning for Direct Hybrid Precoding in Millimeter Wave Massive MIMO Systems," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 800–805, IEEE, 2019.

[3] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised Learning-Based Fast Beamforming Design for Downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, 2018.

[4] T. Peken, S. Adiga, R. Tandon, and T. Bose, "Deep Learning for SVD and Hybrid Beamforming," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6621–6642, 2020.

[5] E. Nachmani, Y. Be'ery, and D. Burshtein, "Learning to Decode Linear Codes Using Deep Learning," in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 341–346, IEEE, 2016.

[6] E. Nachmani, E. Marciano, L. Lugosch, W. J. Gross, D. Burshtein, and Y. Be'ery, "Deep Learning Methods for Improved Decoding of Linear Codes," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 119–131, 2018.

[7] L. Lugosch and W. J. Gross, "Neural Offset Min-Sum Decoding," in *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 1361–1365, IEEE, 2017.

[8] B. Vasić, X. Xiao, and S. Lin, "Learning to Decode LDPC Codes with Finite-Alphabet Message Passing ," in *2018 Information Theory and Applications Workshop (ITA)*, pp. 1–9, IEEE, 2018.

[9] E. Nachmani and L. Wolf, "Hyper-Graph-Network Decoders for Block Codes," in *Advances in Neural Information Processing Systems*, pp. 2329–2339, 2019.

[10] N. Doan, S. A. Hashemi, E. N. Mambou, T. Tonnellier, and W. J. Gross, "Neural Belief Propagation Decoding of CRC-Polar Concatenated Codes," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2019.

[11] A. Buchberger, C. Häger, H. D. Pfister, L. Schmalen, and A. G. i Amat, "Pruning and Quantizing Neural Belief Propagation Decoders," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 1957–1966, 2020.

[12] V. G. Satorras and M. Welling, "Neural Enhanced Belief Propagation on Factor Graphs," in *International Conference on Artificial Intelligence and Statistics*, pp. 685–693, PMLR, 2021.

[13] E. Nachmani and Y. Be'ery, "Neural Decoding with Optimization of Node Activations," *IEEE Communications Letters*, 2022.

[14] T. Gruber, S. Cammerer, J. Hoydis, and S. ten Brink, "On Deep Learning-Based Channel Decoding," in *2017 51st Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, IEEE, 2017.

[15] N. Shlezinger, Y. C. Eldar, N. Farsad, and A. J. Goldsmith, "ViterbiNet: Symbol Detection Using a Deep Learning Based Viterbi Algorithm," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, 2019.

[16] J. Seo, J. Lee, and K. Kim, "Decoding of Polar Code by Using Deep Feed-Forward Neural Networks," in *2018 International Conference on Computing, Networking and Communications (ICNC)*, pp. 238–242, IEEE, 2018.

[17] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep Learning-Based Channel Estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652–655, 2019.

[18] L. Dai, R. Jiao, F. Adachi, H. V. Poor, and L. Hanzo, "Deep Learning for Wireless Communications: An Emerging Interdisciplinary Paradigm," *IEEE Wireless Communications*, vol. 27, no. 4, pp. 133–139, 2020.

[19] Y. C. Eldar, A. Goldsmith, D. Gündüz, and H. V. Poor, *Machine Learning and Wireless Communications*. Cambridge University Press, 2022.

[20] T. Erpek, T. J. O'Shea, Y. E. Sagduyu, Y. Shi, and T. C. Clancy, "Deep Learning for Wireless Communications," in *Development and Analysis of Deep Learning Architectures*, pp. 223–266, Springer, 2020.

[21] S. Peng, H. Jiang, H. Wang, H. Alwageed, Y. Zhou, M. M. Sebdani, and Y.-D. Yao, "Modulation Classification Based on Signal Constellation Diagrams and Deep Learning," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 3, pp. 718–727, 2018.

[22] X. Liu, D. Yang, and A. El Gamal, "Deep Neural Network Architectures for Modulation Classification," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pp. 915–919, IEEE, 2017.

[23] R. Fritschek, R. F. Schaefer, and G. Wunder, "Deep Learning for the Gaussian Wiretap Channel," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2019.

[24] T. O'shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[25] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep Learning for Wireless Physical Layer: Opportunities and Challenges," *China Communications*, vol. 14, no. 11, pp. 92–111, 2017.

[26] H. Kim, Y. Jiang, S. Kannan, S. Oh, and P. Viswanath, "Deepcode: Feedback Codes via Deep Learning," in *Advances in neural information processing systems*, pp. 9436–9446, 2018.

[27] Y. Jiang, H. Kim, H. Asnani, S. Kannan, S. Oh, and P. Viswanath, "Turbo Autoencoder: Deep learning based channel codes for point-to-point communication channels," in *Advances in Neural Information Processing Systems*, pp. 2758–2768, 2019.

[28] K. Choi, K. Tatwawadi, A. Grover, T. Weissman, and S. Ermon, "Neural Joint Source-Channel Coding," in *International Conference on Machine Learning*, pp. 1182–1192, 2019.

[29] H. Tang, J. Xu, S. Lin, and K. A. Abdel-Ghaffar, "Codes on Finite Geometries," *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 572–596, 2005.

[30] J. Liu and R. C. de Lamare, "Low-Latency Reweighted Belief Propagation Decoding for LDPC Codes," *IEEE Communications Letters*, vol. 16, no. 10, pp. 1660–1663, 2012.

[31] S. Zhang and C. Schlegel, "Causes and Dynamics of LDPC Error Floors on AWGN Channels," in *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1025–1032, IEEE, 2011.

[32] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT press, 2018.

[33] P. L. Bartlett, N. Harvey, C. Liaw, and A. Mehrabian, "Nearly-tight VC-dimension and Pseudodimension Bounds for Piecewise Linear Neural Networks," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 2285–2301, 2019.

[34] E. D. Sontag *et al.*, "VC Dimension of Neural Networks," *NATO ASI Series F Computer and Systems Sciences*, vol. 168, pp. 69–96, 1998.

[35] G. K. Dziugaite and D. M. Roy, "Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data," *arXiv preprint arXiv:1703.11008*, 2017.

[36] R. Liao, R. Urtasun, and R. Zemel, "A PAC-Bayesian Approach to Generalization Bounds for Graph Neural Networks," *arXiv preprint arXiv:2012.07690*, 2020.

[37] M. Mohri and A. Rostamizadeh, "Rademacher Complexity Bounds for Non-I.I.D. Processes," *Advances in Neural Information Processing Systems*, vol. 21, 2008.

[38] P. L. Bartlett and S. Mendelson, "Rademacher and Gaussian Complexities: Risk Bounds and Structural Results," *Journal of Machine Learning Research*, vol. 3, no. Nov, pp. 463–482, 2002.

[39] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, "Spectrally-normalized margin bounds for neural networks," *Advances in neural information processing systems*, vol. 30, 2017.

[40] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain Adaptation: Learning Bounds and Algorithms," *arXiv preprint arXiv:0902.3430*, 2009.

[41] B. Neyshabur, R. Tomioka, and N. Srebro, "Norm-Based Capacity Control in Neural Networks," in *Conference on Learning Theory*, pp. 1376–1401, 2015.

[42] A. Xu and M. Raginsky, "Information-theoretic analysis of generalization capability of learning algorithms," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[43] Y. Bu, S. Zou, and V. V. Veeravalli, "Tightening mutual information-based bounds on generalization error," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 121–130, 2020.

[44] M. J. Kearns and U. Vazirani, *An Introduction to Computational Learning Theory*. MIT press, 1994.

[45] L. G. Valiant, "A Theory of the Learnable," *Communications of the ACM*, vol. 27, no. 11, pp. 1134–1142, 1984.

[46] O. Bousquet, S. Hanneke, S. Moran, R. Van Handel, and A. Yehudayoff, "A Theory of Universal Learning," in *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 532–541, 2021.

[47] J. Langford and R. Caruana, "(Not) Bounding the True Error," *Advances in Neural Information Processing Systems*, vol. 14, 2001.

[48] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro, "Exploring Generalization in Deep Learning," *Advances in neural information processing systems*, vol. 30, 2017.

[49] B. Neyshabur, S. Bhojanapalli, and N. Srebro, "A Pac-Bayesian Approach to Spectrally-Normalized Margin Bounds for Neural Networks," *arXiv preprint arXiv:1707.09564*, 2017.

[50] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima," *arXiv preprint arXiv:1609.04836*, 2016.

[51] N. Weinberger, "Generalization bounds and algorithms for learning to communicate over additive noise channels," *IEEE Transactions on Information Theory*, vol. 68, no. 3, pp. 1886–1921, 2021.

[52] A. Tsvieli and N. Weinberger, "Learning maximum margin channel decoders," *IEEE Transactions on Information Theory*, 2023.

[53] G. K. Dziugaite, A. Drouin, B. Neal, N. Rajkumar, E. Caballero, L. Wang, I. Mitliagkas, and D. M. Roy, "In Search of Robust Measures of Generalization," *Advances in Neural Information Processing Systems*, vol. 33, pp. 11723–11733, 2020.

[54] F. Biggs and B. Guedj, "Non-Vacuous Generalisation Bounds for Shallow Neural Networks," in *International Conference on Machine Learning*, pp. 1963–1981, PMLR, 2022.

[55] P. Alquier, "User-friendly introduction to PAC-Bayes bounds," *arXiv preprint arXiv:2110.11216*, 2021.

[56] G. K. Dziugaite and D. M. Roy, "Data-dependent PAC-Bayes priors via differential privacy," *Advances in neural information processing systems*, vol. 31, 2018.

[57] V. Garg, S. Jegelka, and T. Jaakkola, "Generalization and Representational Limits of Graph Neural Networks," in *International Conference on Machine Learning*, pp. 3419–3430, PMLR, 2020.

[58] M. Chen, X. Li, and T. Zhao, "On Generalization Bounds of a Family of Recurrent Neural Networks," *arXiv preprint arXiv:1910.12947*, 2019.

[59] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Progressive Edge-Growth Tanner Graphs," in *GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No. 01CH37270)*, vol. 2, pp. 995–1001, IEEE, 2001.

[60] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Regular and Irregular Progressive Edge-Growth Tanner Graphs," *IEEE transactions on information theory*, vol. 51, no. 1, pp. 386–398, 2005.

[61] R. Mathias, "The Spectral Norm of a Nonnegative Matrix," *Linear Algebra and its Applications*, vol. 139, pp. 269–284, 1990.
[62] M. Ledoux and M. Talagrand, "Probability in Banach spaces. Classics in Mathematics," 2011.

Sudarshan Adiga received his B.E. degree in Telecommunication Engineering from Ramaiah Institute of Technology, Bangalore, in 2015. He then completed his M.S. degree in Electrical and Computer Engineering from the University of Arizona, Tucson, AZ, USA, in 2019. Currently, he is pursuing his Ph.D. at the Electrical and Computer Engineering Department at the University of Arizona, focusing on Machine Learning, Information Theory, and Wireless Communications. Prior to his doctoral studies, Sudarshan worked as a software engineer at Bosch Corporation from 2015 to 2017. He also served as a research intern at NTT Docomo in 2022 and at Marvell Technology in 2023.

Xin Xiao received the B.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2012, the M.E. degree in system LSI from the Graduate School of Information, Production and System, Waseda University, Kitakyushu, Fukuoka, Japan, in 2013, the M.S. degree in electrical engineering from Shanghai Jiao Tong University in 2015, and the Ph.D. degree in electrical and computer engineering from The University of Arizona, Tucson, AZ, USA. Since 2021, she has been a Research Scientist with Meta Platform Inc., Bellevue, WA, USA. Her current research interests include error correction coding for communication and storage systems and deep learning and optimization in channel coding.

Ravi Tandon (Senior Member, IEEE) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur (IIT Kanpur), in 2004, and the Ph.D. degree in electrical and computer engineering from the University of Maryland, College Park (UMCP), in 2010. He is currently the Litton Industries John M. Leonis Distinguished Associate Professor with the Department of ECE, The University of Arizona. Prior to joining The University of Arizona in Fall 2015, he was a Research Assistant Professor at Virginia Tech with positions at the Bradley Department of ECE, Hume Center for National Security and Technology; and the Discovery Analytics Center, Department of Computer Science. From 2010 to 2012, he was a Post-Doctoral Research Associate at Princeton University. His current research interests include information theory and its applications to wireless networks, signal processing, communications, security and privacy, machine learning, and data mining. He was a recipient of the 2018 Keysight Early Career Professor Award, the NSF CAREER Award in 2017, and the Best Paper Award at IEEE GLOBECOM 2011. He serves as an Editor for IEEE TRANSACTIONS ON INFORMATION THEORY, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and IEEE TRANSACTIONS ON COMMUNICATIONS.

Bane Vasić Dr. Bane Vasić is a Professor of Electrical and Computer Engineering and Mathematics at the University of Arizona and a Director of the Error Correction Laboratory. He is an inventor of the soft error-event decoding algorithm for intersymbol interference channels with correlated noise, and the key architect of a detector/decoder for Bell Labs data storage read channel chips which were regarded as the best in industry. His pioneering work on structured low-density parity check (LDPC) error correcting codes based on combinatorial designs has enabled low-complexity iterative decoder implementations. Structured LDPC codes are today adopted in a number of communications standards and data storage systems. Dr. Vasić's work on codes on graphs, trapping sets and error floor of iterative decoding algorithms has led to decoders for the binary symmetric channel with best error-floor performance known today. Dr. Vasić is a PI on a Department of Energy multi-university 115M-project led by Fermi National Laboratory to establish a Center for Superconducting Materials and Systems. He is a co-PI of the 52M NSF Center for Quantum Network hosted at the University of Arizona, and involved in University of Arizona Quantum Hub, a group of researchers leading effort to establish a graduate program in quantum information science and engineering at the UArizona. He is also funded by NASA- Jet Propulsion Laboratory through the Strategic University Partnership Program for the development of quantum codes and error correction algorithms for NASA space missions, and is a PI on seven research grants funded by the National Science Foundation. He is a founder of Codelucida, a company developing advanced error correction solutions for communications and data storage. Recently, Codelucida has received numerous innovation awards including 2017 Arizona Innovation Challenge Award from Arizona Commerce Authority, 2018 I-Squared Startup of the Year from Tech Launch Arizona, and Best of Show Award for the Most Innovative Flash Memory Technology at the 2019 Flash Memory Summit, the World largest exhibition of flash memory technologies. Codelucida is a Xilinx Partner providing LDPC Codec IP cores for flash memory controllers. He is an IEEE Fellow, Fulbright Scholar, da Vinci Fellow, and a past Chair of IEEE Data Storage Technical Committee.

Tamal Bose Dr. Tamal Bose is a co-founder and Vice President of Cognitive Sensors and RF of EpiSci, located in the San Diego area. His research interests are in the areas of signal classification, cognitive radios, wireless communications, and electronic warfare. Dr. Bose served as professor and head of ECE at the University of Arizona from 2012 - 2022. Previously, he was a tenured full professor of the Bradley Department of Electrical and Computer Engineering at Virginia Tech. He is author of the text Digital Signal and Image Processing, John Wiley, 2004, and co-author of Basic Simulation Models of Phase Tracking Devices Using MATLAB, Morgan and Claypool Publishers, 2010. He is also the author or co-author of over 70 journal papers and over 100 conference papers. Dr. Bose received his Ph.D. from Southern Illinois University.