

Soft Syndrome Decoding of Quantum LDPC Codes for Joint Correction of Data and Syndrome Errors

Nithin Raveendran, Narayanan Rengaswamy, Asit Kumar Pradhan, and Bane Vasić

Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ, 85721 USA

Email: { nithin , narayananr , asitpradhan }@arizona.edu, vasic@ece.arizona.edu

Abstract—Quantum errors are primarily detected and corrected using the measurement of syndrome information which itself is an unreliable step in practical error correction implementations. Typically, such faulty or noisy syndrome measurements are modeled as a binary measurement outcome flipped with some probability. However, the measured syndrome is in fact a discretized value of the continuous voltage or current values obtained in the physical implementation of the syndrome extraction. In this paper, we use this “soft” or analog information without the conventional discretization step to benefit the iterative decoders for decoding quantum low-density parity-check (QLDPC) codes. Syndrome-based iterative belief propagation decoders are modified to utilize the syndrome-soft information to successfully correct both data and syndrome errors simultaneously, without repeated measurements. We demonstrate the advantages of extracting the soft information from the syndrome in our improved decoders, not only in terms of comparison of thresholds and logical error rates for quasi-cyclic lifted-product QLDPC code families, but also for faster convergence of iterative decoders. In particular, the new BP decoder with noisy syndrome performs as good as the standard BP decoder under ideal syndrome.

I. INTRODUCTION

Quantum error correction (QEC) is a necessary step in realizing scalable and reliable quantum computing. For fault-tolerant QEC, quantum low-density parity-check (QLDPC) codes are advantageous over surface codes, the current leading candidate [1]–[5], in terms of the scaling of qubit overhead and minimum distance. Recent breakthroughs related to various product constructions of QLDPC codes provided constructions of “good” QLDPC code families, i.e., codes with finite asymptotic rate and relative minimum distance [4], [6]–[9]. Similarly, there has been significant progress in improving the iterative decoding performance of finite-length QLDPC codes using post-processing and heuristic techniques [10]–[14]. However, the QLDPC decoding problem still has unanswered questions and, in particular, faster decoders for QLDPC codes are needed to meet the stringent timing constraints in hardware. The problem is even more challenging than classical LDPC code decoding since we have to overcome syndrome measurement errors in addition to the errors on physical qubits. To address these errors without repeating the measurements, data-syndrome codes and “single-shot” error correction codes have been proposed [15]–[18]. However, noisy syndrome measurement has not been analyzed under

This work is generously supported by the National Science Foundation (NSF) under grants CCF-2100013, CCF-2106189, CCSS-2027844, CCSS-2052751, ERC-1941583, CIF-1855879, and NASA under the SURP Program.

iterative decoding, which is of great interest while decoding these good QLDPC codes.

The first, and primary, challenge is that when a decoder uses a noisy syndrome as input, it significantly and non-trivially alters the iterative decoding dynamics. Secondly, in the absence of a referent “true” syndrome, it is difficult to formulate the halting condition, and an iterative decoder can, in principle, run indefinitely without finding the error pattern. A naïve binary quantization of the syndrome would lead to an estimated error that is possibly matched to a wrong syndrome.

In the literature, noisy measurements are typically modeled by a simple phenomenological error model in which binary measurement outcomes are flipped with some probability. However, the measurement is typically more informative since it corresponds to continuous voltages, currents, or discrete photon counts based on the specific physical implementation. Instead of quantizing this analog information to a binary outcome and losing crucial information, one can leverage the soft syndrome in the iterative decoder and modify the update rules accordingly. In a recent work by Pattinson *et al.* [19], syndrome measurement outcomes beyond simple binary values were considered. The decoders used for surface codes, such as minimum weight perfect matching and union-find, were modified accordingly to obtain higher decoding thresholds.

In this work, we model a realistic syndrome measurement by a perfect measurement with an ideal bipolar (± 1) outcome followed by a noisy channel that leads to a continuous soft outcome. We restrict our attention to the simple setting of a symmetric Gaussian noise on each of the ideal syndrome measurements. In the context of iterative decoding, we *modify* the min-sum algorithm (MSA) based decoder (a hardware friendly variant of the belief propagation (BP) decoder) to use these soft syndrome values as additional information to drive the message passing algorithm towards convergence. This iterative decoder for QLDPC codes that employs *soft syndrome information* is novel and is one of the main contributions of this paper. It is also unprecedented in the setting of iterative decoding of classical LDPC codes, although faulty components in decoders have been studied [20]–[22]. Furthermore, even if the code does not have the single-shot property, our strategy allows one to perform effective decoding *without repeating syndrome measurements*, which is attractive practically.

We consider three scenarios for decoding:

- 1) **Perfect Syndrome:** In this case, there is no noise added to the syndrome and we use the standard MSA decoder.

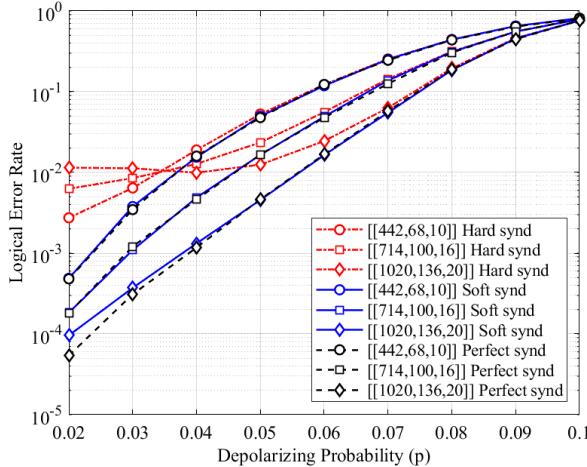


Fig. 1: Two-threshold phenomenon observed for ‘Hard Syndrome’ (which uses standard MSA), but the ‘Soft Syndrome’ setting (which uses modified MSA) is devoid of this issue.

- 2) **Hard Syndrome:** Here, there is Gaussian noise added to the syndrome, but this continuous value is thresholded to input a hard syndrome to the standard MSA decoder.
- 3) **Soft Syndrome:** Now, as in the hard syndrome case, the syndrome is noisy and it is thresholded to produce a sign (± 1), but the magnitude is converted into a log-likelihood ratio (LLR) to also provide soft information. The decoder in this case is our new modified MSA, where the check node update rules are modified and, specifically, the LLR corresponding to the syndrome is utilized and also updated during the check node update.

As one might expect, the threshold for the ‘Hard Syndrome’ case is worse than ‘Perfect Syndrome’, since the decoder is unchanged. But, more interestingly, for the ‘Hard Syndrome’, when we fix the variance of the Gaussian noise affecting the syndrome and sweep the qubit depolarizing rate, below what we identify as the usual decoding threshold, there arises a “second (inverted) threshold”; below this (lower) threshold, the logical error rate creeps up again because the syndrome error dominates the qubit error much more strongly (see Fig. 1). This suggests that, under realistic settings, it is insufficient to only optimize the threshold, and one needs to avoid such problematic behavior below the threshold.

Next, we propose the modified MSA decoder with justification for our new rules for updating nodes as well as the syndrome LLR. We show that the resulting decoder does not suffer from this spurious “second threshold” phenomenon. Besides, the decoding threshold is now almost as good as that for the ‘Perfect Syndrome’ setting. Hence, our work paves the way for efficient and effective decoding of good QLDPC codes under a physically motivated phenomenological noise model, without the use of repetitive syndrome measurement rounds.

The paper is organized as follows. In Section II we introduce the necessary notations and explain the syndrome-based MSA and its variants. In Section III we motivate the soft syndrome decoders and explain our modification to correct the noisy

syndromes. Section IV presents simulation results showing the benefits of our modified MSA. Finally, Section V concludes this paper with discussion on future work.

II. PRELIMINARIES

A. Pauli Operators and CSS Codes

The single-qubit Pauli operators are $I, X, Z, Y = iXZ$ and the n -qubit Pauli group \mathcal{P}_n consists of tensor products of single-qubit Paulis up to a quaternary global phase $1, i, -1$, or $-i$, where $i = \sqrt{-1}$. There is a useful binary representation $[a, b] \in \{0, 1\}^{2n}$ of Pauli operators, e.g., $X \otimes Z \otimes Y \mapsto [a, b] = [101, 011]$, so that entries of a (resp. b) indicate which qubits are acted upon by X (resp. Z), and $a_i = b_i = 1$ (resp. 0) indicates a Y (resp. I). An $[[n, k, d]]$ CSS code [23] encodes k logical qubits into n physical qubits, and d is the minimum distance of the code that indicates the minimum number of qubits on which the error channel must act to disturb the logical information. The CSS code is defined by $(n - k)$ commuting Pauli operators, called *stabilizers*, some of which are purely X -type, $[a, 0]$, and the others are purely Z -type, $[0, b]$. Hence, the stabilizers can be represented using two binary parity-check matrices H_X (whose rows map to $[a, 0]$) and H_Z (whose rows map to $[0, b]$) that must satisfy $H_X H_Z^T = 0$ so that these Pauli operators commute.

B. Noise Model

The single-qubit depolarizing channel, also known as the memoryless Pauli channel, is a widely studied error model characterized by the depolarizing probability p : the channel randomly introduces a Pauli error according to the probabilities $P_X = P_Y = P_Z = p/3$ and $P_I = 1 - p$. For the physical data qubits, we assume an i.i.d. depolarizing error model, i.e., each physical qubit is affected by an independent depolarizing error. Let $\mathbf{e} = [e_X, e_Z]$ be the binary representation of a Pauli error acting on the n qubits. The corresponding syndrome is computed as $[\mathbf{s}_X, \mathbf{s}_Z] = [H_Z \mathbf{e}_X^T, H_X \mathbf{e}_Z^T]^T \pmod{2}$. Since we consider decoding of H_X and H_Z separately, we avoid using subscripts X and Z in syndrome \mathbf{s} .

C. Syndrome Noise Model

As mentioned in Section I, we model a realistic syndrome measurement by a perfect measurement with an ideal bipolar (± 1) outcome followed by i.i.d. symmetric Gaussian noise on each component of the syndrome. After introducing symmetric Gaussian noise $n_i \sim \mathcal{N}(0, \sigma^2)$ to the ideal syndrome (component) $s_i \in \{\pm 1\}$, we have a noisy soft syndrome $\tilde{r}_i = s_i + n_i$ which when conditioned on s_i is distributed normally: $[\tilde{r}_i \mid s_i = 1] \sim N(+1, \sigma^2)$ and $[\tilde{r}_i \mid s_i = -1] \sim N(-1, \sigma^2)$. Other relevant asymmetric syndrome measurement models are discussed in [19, Section 1.4].

Based on the normally distributed soft syndromes, we can compute the syndrome log-likelihood ratios (LLRs) as

$$\gamma_i := \log \frac{\Pr(\tilde{r}_i \mid s_i = +1)}{\Pr(\tilde{r}_i \mid s_i = -1)} = \frac{2\tilde{r}_i}{\sigma^2}. \quad (1)$$

Since the noise is zero-mean, the sign of the soft syndrome, $\text{sgn}(\gamma_i)$, indicates if the parity-check i is satisfied ($\text{sgn}(\gamma_i) = +1$) or unsatisfied ($\text{sgn}(\gamma_i) = -1$). This is exactly same as the measured syndrome s_i used in conventional min-sum decoder. However, its magnitude, $|\gamma_i|$, provides valuable soft information about the reliability of the measured syndrome.

D. QLDPC Codes and Syndrome Min-Sum Algorithm (MSA)

A CSS-QLDPC code has sparse stabilizer generators and is represented by a pair of bipartite (Tanner) graphs whose biadjacency matrices are the stabilizer matrices H_X and H_Z . The Tanner graphs have n variable nodes, denoted by $j \in \{1, 2, \dots, n\}$, represented by circles in Fig. 2, and m check nodes, denoted by $i \in \{1, 2, \dots, m\}$, represented by squares in Fig. 2. If the (i, j) -th entry of the binary parity-check matrix is not zero, then there is an edge between variable node i and check node j . The sets of neighbors of variable node j and check node i are denoted by $\mathcal{N}(j)$ and $\mathcal{N}(i)$, respectively.

Like classical LDPC codes, QLDPC codes also employ decoders based on iterative BP or message-passing algorithms. The fundamental difference between the decoders of QLDPC codes and LDPC codes is that, the decoders for the former take as input the syndrome obtained after the stabilizer measurements, whereas decoders for the latter take as input a noisy version of the codeword. The QLDPC decoder's goal is to output an error pattern whose syndrome matches the measured syndrome. Let us first consider the case when these stabilizer measurements are perfect, meaning there is no noise in the syndrome measurement process.

Suppose that the coded qubits are corrupted by an X-type or Z-type error, corresponding to the binary error vector $\mathbf{e} = [e_1, e_2, \dots, e_n]$ whose entries are realizations of independent Bernoulli random variables with parameter p . Given binary syndrome \mathbf{s} vector of length m , the decoder performs a finite number of message-passing iterations over the Tanner graph to compute *a posterior probabilities* $\Pr(e_j | \mathbf{s})$, for $j \in \{1, \dots, n\}$, corresponding to error bit e_j conditioned on the measured syndrome \mathbf{s} .

Next, we briefly describe the min-sum algorithm (MSA) based decoder. For this purpose, we first introduce the required notations. Denote by $\mu_{i,j}$ the message sent from check node i to variable node j . Similarly, the message sent from variable node j to check node i is denoted by $\nu_{i,j}$. The measured syndrome value is $s_i \in \{\pm 1\}$, $i \in \{1, 2, \dots, m\}$. Since the error e_j , $j \in \{1, 2, \dots, n\}$, at variable node j is a realization of Bernoulli random variable with parameter p , the log-likelihood ratio (LLR), denoted by λ_j , corresponding to the j th variable node is given by $\lambda_j = \ln\left(\frac{1-p}{p}\right)$.

In the first decoding iteration, all the outgoing messages from variable node j , for all j , are initialized to λ_j . In subsequent iterations, for $i \in \mathcal{N}(j)$, message $\nu_{i,j}$ is computed using the update rule

$$\nu_{i,j} = \lambda_j + \sum_{i' \in \mathcal{N}(j) \setminus \{i\}} \mu_{i',j}. \quad (2)$$

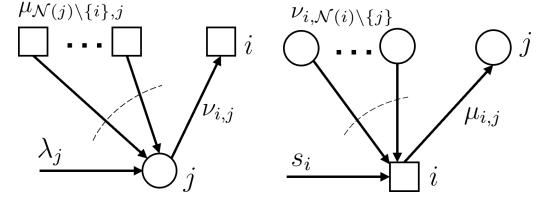


Fig. 2: Node updates for MSA based on Eqs. (2) and (3).

For $j \in \mathcal{N}(i)$, the message $\mu_{i,j}$ is computed using the rule

$$\mu_{i,j} = \left(s_i \cdot \prod_{j' \in \mathcal{N}(i) \setminus \{j\}} \text{sgn}(\nu_{i,j'}) \right) \cdot \left(\min_{j' \in \mathcal{N}(i) \setminus \{j\}} |\nu_{i,j'}| \right). \quad (3)$$

The sign function is defined as $\text{sgn}(A) = -1$ if $A < 0$, and $+1$ otherwise. The first and second parenthesis in Eq. (3) gives the sign and the magnitude of the computed check-to-variable message $\mu_{i,j}$, respectively. Iteration will be indicated as a superscript of $\mu_{i,j}$ and $\nu_{i,j}$ when required. Min-sum update rules are less complex than that of BP decoders (which are never used in practice), but as the min-sum approximation overshoots the BP messages, the MSA is typically modified using a normalized MSA [24], wherein check node messages are multiplied (in Eq. (3)) by a scalar β as a correction factor, where $0 < \beta < 1$.

A min-sum decoder employs the update functions in Eq. (2) and Eq. (3) in each iteration $\ell \leq \ell_{\max}$ to determine the error at each variable node. The error estimate \hat{x}_j at the variable node j is computed using

$$\hat{x}_j = \hat{\Phi}(\lambda_j, \mu_{i,j}) = \text{sgn} \left(\lambda_j + \sum_{i \in \mathcal{N}(j)} \mu_{i,j} \right), \quad (4)$$

for all $j \in \{1, 2, \dots, n\}$, i.e., it is based on the sign of the *decision update function* output, $\hat{\Phi}(\lambda_j, \mu_{i,j})$, that uses the LLR value and *all* incoming check node messages to the variable node j , where $i \in \mathcal{N}(j)$.

Halting Criterion: The output of the decoder at the ℓ -th iteration, denoted by $\hat{\mathbf{x}}^{(\ell)} = (\hat{x}_1^{(\ell)}, \hat{x}_2^{(\ell)}, \dots, \hat{x}_n^{(\ell)})$, is used to check whether all parity-check equations are matched, i.e., whether the syndrome at ℓ -th iteration $\hat{\mathbf{s}}^{(\ell)} := \hat{\mathbf{x}}^{(\ell)} \cdot \mathbf{H}^T$ is equal to the measured syndrome \mathbf{s} , in which case iterative decoding terminates and outputs $\hat{\mathbf{x}}^{(\ell)}$ as the error vector. Otherwise, the iterative decoding steps continue until a predefined maximum number of iterations, denoted by ℓ_{\max} , is reached. The decoding is deemed successful if the true error pattern is found, i.e., $\hat{\mathbf{x}}^{(\ell)} = \mathbf{e}$. Otherwise, the decoding is said to have failed. Miss-correction as referred to in the literature of classical coding theory occurs when post-correction step with the estimated error pattern results in a logical error. This is also classified as an error correction failure.

Note that the MSA decoder is exactly the “Perfect Syndrome” decoder, described in Section I, when the syndrome measurement process is noiseless. When we have a noisy

observation of the syndrome, recall that an estimate of the syndrome is obtained by passing the observations to a thresholding function. This estimated syndrome is fed to the MSA decoder to get the ‘Hard Syndrome’ decoder defined earlier.

III. SOFT SYNDROME DECODER

As discussed above, under noisy syndrome, the MSA decoder does not use the soft information available in the measured syndrome and treats the thresholded syndrome as the correct syndrome. Due to this reason, the decoder typically obtains error pattern estimates only close to the true error pattern. Also, note that the halting condition is never met in this case, so the decoder runs until the maximum number of iterations and declares a decoding failure. In particular, using a noisy syndrome has two disadvantages: it makes the iterative decoder’s halting condition ambiguous, even after the decoder converges to the true error estimate, and drives the decoder to produce error estimates matching the wrong syndrome. In this section, we propose a modified MSA decoder that uses the soft information available in the measured syndrome to address these issues. As mentioned in Section I, we refer to the proposed decoder as ‘Soft Syndrome’ (MSA).

To use the soft information from the observed syndrome, we modify the Tanner graph by adding another type of nodes, called ‘virtual variable nodes’, in addition to the existing variable and check nodes. A virtual variable node is added corresponding to each of the m check nodes. There is an edge between a virtual variable node and the corresponding check node. The connections between the variable and check nodes are not altered. Hence, the standard MSA decoder still holds.

The initialization and update rules at the variable nodes for the ‘Soft Syndrome’ decoder is same as that of the MSA decoder described in Section II. We now describe the update rules at the check nodes and virtual variable nodes for the ‘Soft Syndrome’ decoder. In addition to the measured syndrome s , a binary vector of length m , the ‘Soft Syndrome’ decoder receives the respective reliability (i.e., syndrome LLR) values γ_i as defined in Eq. (1). The goal of this ‘Soft Syndrome’ iterative decoder in the noisy syndrome setting is to find an error pattern that matches the *potentially revised* syndrome \tilde{s} using the syndrome reliability γ and the qubit LLR λ .

The magnitude of the soft syndrome LLR value determines the reliability of the check information. To deploy the ‘Soft Syndrome’ decoder, we determine a *cutoff* above which the soft syndrome value is deemed reliable and below which it is not. If the syndrome is reliable, i.e., if the $|\gamma_i|$ corresponding to check node i is greater than the predefined cutoff, then we use the conventional check node update rule given in Eq. (3). Choosing this cutoff value denoted by Γ is important and it can be optimized to improve the decoding performance. There are two main changes to the minimum rule used at the check nodes. With the soft syndrome γ_i available from the virtual variable node, first, we compute the check to variable message as the minimum of magnitude of the extrinsic variable node messages as well as the soft syndrome magnitude $|\gamma_i|$. For

$j \in \mathcal{N}(i)$, the magnitude and sign of message $\mu_{i,j}$ are given by

$$|\mu_{i,j}| = \begin{cases} \left(\min_{j' \in \mathcal{N}(i) \setminus \{j\}} |\nu_{i,j'}| \right) & \text{for } |\gamma_i| > \Gamma, \\ \left(\min_{j' \in \mathcal{N}(i) \setminus \{j\}} (|\nu_{i,j'}|, |\gamma_i|) \right) & \text{otherwise;} \end{cases} \quad (5)$$

$$\text{sgn}(\mu_{i,j}) = \left(s_i \cdot \prod_{j' \in \mathcal{N}(i) \setminus \{j\}} \text{sgn}(\nu_{i,j'}) \right). \quad (6)$$

Second, we update the soft syndrome, both magnitude and reliability, based on the magnitude and sign of all the incoming messages to the check node from its neighboring variable nodes, thus effectively pushing the estimated syndrome towards the correct syndrome. In the ℓ -th iteration, the estimated sign and reliability of syndrome corresponding to check node i is denoted by $\tilde{\gamma}_i^{(\ell)}$ and $\tilde{s}_i^{(\ell)}$, respectively. In the first iteration, we initialize $\tilde{s}_i^{(1)} = s_i$ and $\tilde{\gamma}_i^{(1)} = |\gamma_i|$. In the subsequent iterations, the reliability $\tilde{\gamma}_i^{(\ell)}$ of the estimated syndrome corresponding to the check node i is updated using

$$\tilde{\gamma}_i^{(\ell)} = \min_{j' \in \mathcal{N}(i)} |\nu_{i,j'}| \quad (7)$$

if $\min_{j' \in \mathcal{N}(i)} |\nu_{i,j'}| > |\tilde{\gamma}_i^{(\ell-1)}|$ and $\tilde{s}_i^{(\ell-1)} = \prod_{j' \in \mathcal{N}(i)} \text{sgn}(\mu_{i,j'})$, otherwise $\tilde{\gamma}_i^{(\ell)} = \tilde{\gamma}_i^{(\ell-1)}$.

The sign of the estimated syndrome corresponding to the check node i is updated using

$$\tilde{s}_i^{(\ell)} \mapsto (-1) \cdot \tilde{s}_i^{(\ell-1)} \quad (8)$$

if $\min_{j' \in \mathcal{N}(i)} |\nu_{i,j'}| > |\tilde{\gamma}_i^{(\ell-1)}|$ and $\tilde{s}_i^{(\ell-1)} \neq \prod_{j' \in \mathcal{N}(i)} \text{sgn}(\mu_{i,j'})$, otherwise $\tilde{s}_i^{(\ell)} = \tilde{s}_i^{(\ell-1)}$.

The halting criterion for the ‘Soft Syndrome’ decoder is whether all parity-check equations are matched to these updated signs $\tilde{s}_i^{(\ell)} \forall i \in \{1, 2, \dots, m\}$, i.e., whether the syndrome at ℓ th iteration, $\hat{x}^{(\ell)} \cdot \mathbf{H}^T$ is equal to the \tilde{s} , in which case decoding terminates and outputs $\hat{x}^{(\ell)}$ as the error vector. The modified decoding rules are illustrated in Fig. 3.

Though in the simulation results in Section IV, we use the ‘Soft Syndrome’ discussed above, we also identify another variant where one do not update the reliability of the estimated syndrome given in Eq. (7), and only update the sign of the estimated syndrome. Upon selecting a suitable cutoff, this variant of ‘Soft Syndrome’ decoder also can correct both data and syndrome errors. We will analyze performance of these variants as well as the effect of the cutoff in future work.

Note that we are not using an explicit syndrome code for protecting the syndrome bits as in [16], [25], instead relying on the message passing algorithm to infer what the syndromes should be. Using this modified MSA decoder, we can avoid the overhead of repeated measurements as we can identify the measurement errors in the instances where this decoder converges.

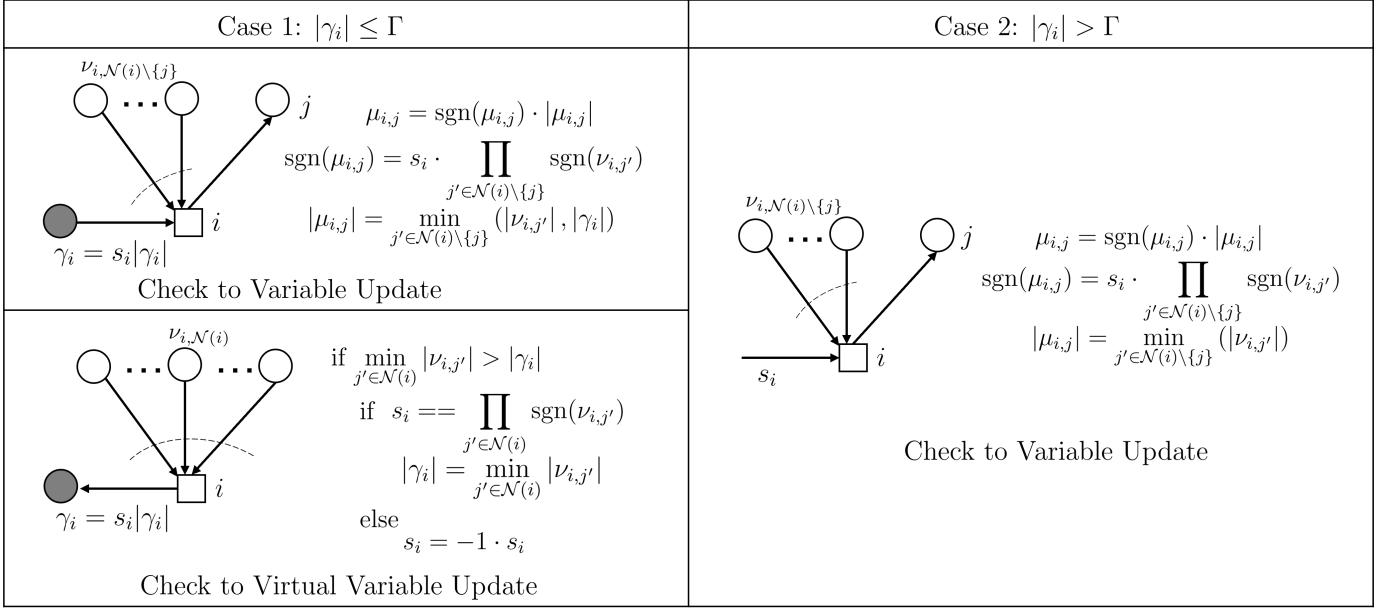


Fig. 3: Modification of the check node update for the ‘Soft Syndrome’ decoder based on a predefined cutoff, Γ . In addition to the ‘minimum’ update, check to virtual variable updates (shaded variable node) are updated conditionally.

IV. SIMULATION RESULTS

A. QLDPC Codes and Simulation Setup

We use *lifted product* (LP) QLDPC codes proposed by Panteleev and Kalachev [26] which provide flexibility in constructing finite length QLDPC codes from good classical (and quantum) *quasi-cyclic* (QC) [27] LDPC codes. We choose QC-LDPC codes as constituent classical LDPC codes to construct LP code families for demonstrating our simulation results. For example, from the $[155, 64, 20]$ Tanner code [28] with quasi-cyclic base matrix of size 3×5 and circulant size $L = 31$, we obtain the $[[1054, 140, 20]]$ LP Tanner code [29]. To show the decoding threshold plots, we also chose the LP code family described in [29, Table II]. LP-QLDPC codes with increasing code length and minimum distance ($d = 10, 16, 20, 24$) are chosen to demonstrate the thresholds. The syndrome-based MSA is chosen with a normalization factor of β set to 0.75 empirically, and has a preset maximum of $\ell_{\max} = 100$ iterations. For the Monte-Carlo simulations, specifically for the threshold plots, we collect at least 10,000 logical errors (sufficient to avoid statistical errors). The Cutoff Γ is set to 10 for Fig. 1 and to 5 for the rest of the simulation plots.

Different scenarios of simulation setups — ‘Perfect’, ‘Hard’, and ‘Soft’ syndrome decoding as described in Section I — are considered, with depolarizing noise as the underlying noise for the physical qubits. We also denote the respective MSA decoders as ‘perfect synd’, ‘hard synd’, and ‘soft synd’.

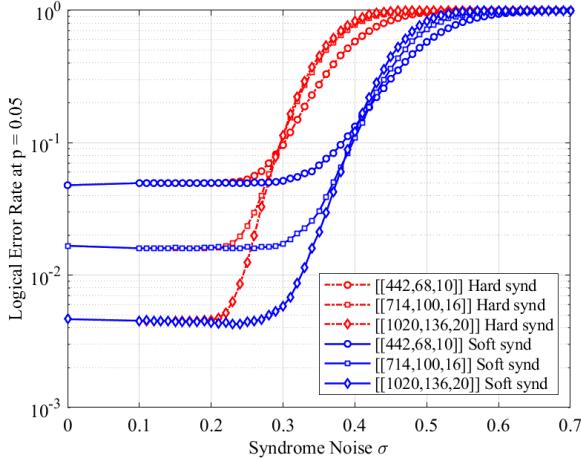
B. Threshold Plots

Threshold on Syndrome Noise: To observe the syndrome noise threshold, we plot in Fig. 4a the logical error rate for a fixed depolarizing probability of $p = 0.05$ against different values of the syndrome noise standard deviation σ . In this setting, the ‘Hard Syndrome’ decoder produces an early

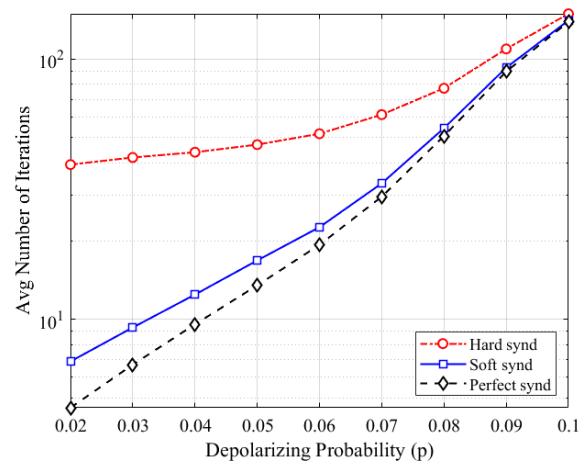
threshold at syndrome noise $\sigma \approx 0.25$. Above this syndrome noise level, a hard decoder cannot improve the decoding performance even by increasing the code length arbitrarily. However, with the ‘Soft Syndrome’ decoder we see that the threshold point is at a higher syndrome noise value of $\sigma \approx 0.4$. This indicates that the modified MSA decoder is more robust and can tolerate noisier syndromes.

This experiment also gives us the range of syndrome noise where ‘Soft Syndrome’ decoder performs close to the ‘Perfect Syndrome’ decoder at $p = 0.05$ (from Fig. 1). Interestingly, at low levels of syndrome noise ($\sigma < 0.25$), the ‘Soft Syndrome’ decoder is able to perform at least as good as the ‘Perfect Syndrome’ decoder. This is a particularly encouraging noise regime where the soft information about noisy syndromes can be dealt with the modified MSA intrinsically, without repeating measurements or requiring single-shot property in the code. For our subsequent simulations, we use the syndrome noise $\sigma = 0.3$ to observe the decoding threshold and the logical error rate suppression with respect to qubit depolarizing probability.

Threshold on Depolarizing Noise: Using the same QLDPC code family and a fixed syndrome noise standard deviation $\sigma = 0.3$, we plot the logical error rate against depolarizing probability for the decoders of interest in Fig. 5. One can see the transition from error suppression to error enhancement with increasing p which signifies the existence of the error threshold. As expected, since we use the same standard MSA decoder, we see in Fig. 5a that the threshold for the ‘Hard Syndrome’ case is worse than ‘Perfect Syndrome’. More interestingly, when we look at lower p in Fig. 1, below what we identify as the usual decoding threshold, there exists another crossover point for the ‘Hard Syndrome’ case where the logical error rate rises up again because the syndrome error starts to dominate the qubit error much more strongly.

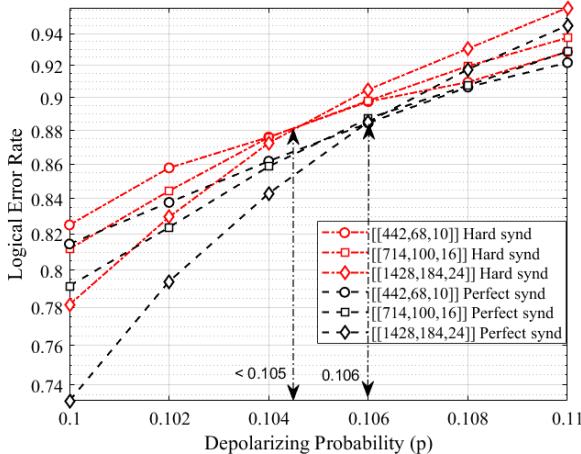


(a) Logical Error Rate for a fixed depolarizing probability of $p = 0.05$ is plotted against the syndrome noise standard deviation σ for the QLDPC code family with increasing code length and distance. We can observe the increase in σ threshold from ≈ 0.25 for ‘Hard Syndrome’ to ≈ 0.4 for ‘Soft Syndrome’.

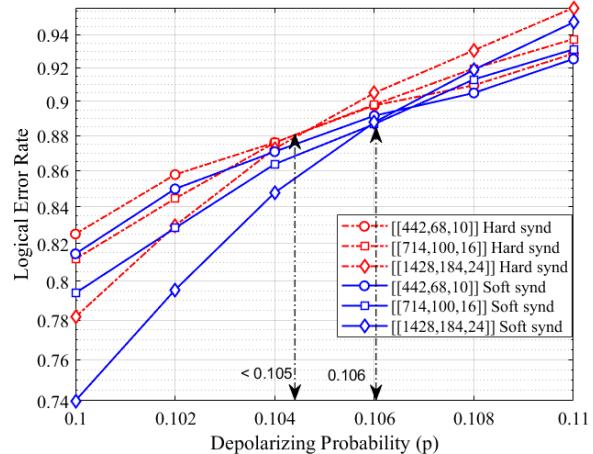


(b) For LP Tanner code with $\sigma = 0.3$, average number of iterations while considering decoding over both H_X and H_Z matrices shows that ‘Soft Syndrome’ decoder requires far less decoding iterations to correct both data and syndrome errors compared to the ‘Hard Syndrome’ decoder.

Fig. 4: Benefits of the ‘Soft Syndrome’ decoder: (a) syndrome noise threshold, (b) average number of iterations.



(a) Thresholds for the LP-QLDPC codes from LP118 code family in [29]: ‘Perfect Syndrome’ vs ‘Hard Syndrome’



(b) Thresholds for the LP-QLDPC codes from LP118 code family in [29]: ‘Soft Syndrome’ vs ‘Hard Syndrome’

Fig. 5: The set of curves in figures (a) and (b) correspond to use of different decoders in the perfect and noisy syndrome setting. The transition of the curves with increasing p signifies an error threshold.

However, the ‘Soft Syndrome’ curves in both Fig. 1 and Fig. 5b show that, besides achieving the same decoding threshold as ‘Perfect Syndrome’, the modified MSA is also devoid of the spurious “second threshold” phenomenon.

C. Faster Decoder Convergence

As we discussed in Section I and Section III, the stopping criterion of an iterative decoder in the presence of measurement errors is ambiguous. To understand the correct decoding performance of these decoders in the presence of syndrome errors, we do not automatically declare a logical error if the decoder exhausts the maximum number of iterations and fails to find an error matching the (evolved) syndrome. Instead, we declare a logical error if the true error ‘plus’ the error estimate produced by the decoder has a non-trivial syndrome or results

in a logical error. This gives a fair chance to the iterative decoders in the presence of syndrome errors.

Even in the presence of noisy syndrome, the ‘Soft Syndrome’ decoder is able to correct the syndrome and data errors quickly to converge to the true error pattern. The faster convergence of ‘Soft Syndrome’ decoder is shown in Fig. 4b, which clearly shows that it requires an average number of decoding iterations very close to that of the ‘Perfect Syndrome’ decoder. Hence, under a reasonable amount of syndrome noise, besides avoiding repeated syndrome measurements and single-shot property, the modified MSA also does not suffer from a penalty in decoding latency. These are very attractive features of the decoding procedure for practical implementations.

V. CONCLUSIONS AND FUTURE DIRECTIONS

In the past few months it has been proven that there is no fundamental obstacle to construct QLDPC code families with optimal parameters. However, one still needs to develop fast and effective decoders for these codes under practical settings where the syndrome measurement process can itself be noisy. In this work, we considered the min-sum algorithm (MSA) based iterative message passing decoder for QLDPC codes. We showed that standard MSA is affected severely by syndrome noise, and we proposed a modified MSA that is robust to a reasonable amount of syndrome noise even without repeated measurements. In future work, we will further investigate systematic ways to analyze iterative decoding under noisy syndromes, and also consider realistic circuit-level noise.

REFERENCES

- [1] D. MacKay, G. Mitchison, and P. McFadden, “Sparse-graph codes for quantum error correction,” *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2315–2330, Oct. 2004.
- [2] Z. Babar, P. Botsinis, D. Alanis, S. X. Ng, and L. Hanzo, “Fifteen years of quantum LDPC coding and improved decoding strategies,” *IEEE Access*, vol. 3, pp. 2492–2519, 2015.
- [3] D. Gottesman, “Fault-tolerant quantum computation with constant overhead,” *Quantum Inf. and Computation*, vol. 14, no. 15–16, pp. 1338–1372, Nov. 2014.
- [4] A. Leverrier, J. Tillich, and G. Zémor, “Quantum expander codes,” in *Proc. IEEE 56th Ann. Symp. on Foundations of Computer Science*, Berkeley, CA, USA, Oct. 2015, pp. 810–824.
- [5] N. P. Breuckmann and J. N. Eberhardt, “Quantum low-density parity-check codes,” *PRX Quantum*, vol. 2, no. 4, p. 040101, Oct. 2021. [Online]. Available: <https://link.aps.org/doi/10.1103/PRXQuantum.2.040101>
- [6] M. B. Hastings, J. Haah, and R. O’Donnell, “Fiber bundle codes: Breaking the $n^{1/2} \text{polylog}(n)$ barrier for quantum LDPC codes,” in *Proc. of the 53rd Ann. ACM SIGACT Symp. on Theory of Computing*, New York, NY, USA, 2021, pp. 1276–1288.
- [7] N. P. Breuckmann and J. N. Eberhardt, “Balanced product quantum codes,” *IEEE Trans. Inf. Theory*, vol. 67, no. 10, pp. 6653–6674, Oct. 2021.
- [8] P. Panteleev and G. Kalachev, “Asymptotically good quantum and locally testable classical LDPC codes,” *arXiv preprint arXiv:2111.03654*, 2021. [Online]. Available: <https://arxiv.org/abs/2111.03654>
- [9] A. Leverrier and G. Zémor, “Quantum Tanner codes,” *arXiv preprint arXiv:2202.13641*, 2022. [Online]. Available: <https://arxiv.org/abs/2202.13641>
- [10] P. Panteleev and G. Kalachev, “Degenerate quantum LDPC codes with good finite length performance,” *Quantum*, vol. 5, p. 585, Nov. 2021.
- [11] K.-Y. Kuo and C.-Y. Lai, “Refined belief propagation decoding of sparse-graph quantum codes,” *IEEE Journal on Selected Areas in Inf. Theory*, vol. 1, no. 2, pp. 487–498, 2020. [Online]. Available: <http://dx.doi.org/10.1109/JSAIT.2020.3011758>
- [12] Y. H. Liu and D. Poulin, “Neural belief-propagation decoders for quantum error-correcting codes,” *Phys. Rev. Lett.*, vol. 122, p. 200501, May 2019.
- [13] A. Rigby, J. Olivier, and P. Jarvis, “Modified belief propagation decoders for quantum low-density parity-check codes,” *Phys. Rev. A*, vol. 100, p. 012330, Jul. 2019.
- [14] N. Raveendran and B. Vasić, “Trapping Sets of Quantum LDPC Codes,” *Quantum*, vol. 5, p. 562, Oct. 2021. [Online]. Available: <https://doi.org/10.22331/q-2021-10-14-562>
- [15] A. Ashikhmin, C. Y. Lai, and T. A. Brun, “Correction of data and syndrome errors by stabilizer codes,” in *Proc. IEEE Intl. Symp. Inf. Theory*, July 2016, pp. 2274–2278.
- [16] A. Ashikhmin, C.-Y. Lai, and T. A. Brun, “Quantum data-syndrome codes,” *IEEE Journal on Selected Areas in Commun.*, vol. 38, no. 3, pp. 449–462, 2020.
- [17] H. Bombín, “Single-shot fault-tolerant quantum error correction,” *Phys. Rev. X*, vol. 5, no. 3, p. 031043, 2015.
- [18] E. T. Campbell, “A theory of single-shot error correction for adversarial noise,” *Quantum Sci. Technol.*, vol. 4, no. 2, p. 025006, 2019.
- [19] C. A. Patterson, M. E. Beverland, M. P. da Silva, and N. Delfosse, “Improved quantum error correction using soft information,” *arXiv preprint arXiv:2107.13589*, 2021. [Online]. Available: <https://arxiv.org/abs/2107.13589>
- [20] B. Vasić and P. Ivanis, “Error Error Eicitur: A stochastic resonance paradigm for reliable storage of information on unreliable media,” *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3596–3608, 2016.
- [21] B. Vasić, P. Ivanis, and D. Declercq, “Approaching maximum likelihood performance of LDPC codes by stochastic resonance in noisy iterative decoders,” in *Inf. Theory and Applications Workshop*, San Diego, CA, Feb. 1–6, 2016.
- [22] N. Raveendran, P. J. Nadkarni, S. S. Garani, and B. Vasić, “Stochastic resonance decoding for quantum LDPC codes,” in *Proc. IEEE Intl. Conf. Commun.*, May 2017, pp. 1–6.
- [23] A. R. Calderbank and P. W. Shor, “Good quantum error-correcting codes exist,” *Phys. Rev. A*, vol. 54, no. 2, pp. 1098–1105, Aug. 1996.
- [24] J. Chen, A. Dholakia, E. Eleftheriou, M. Fossorier, and X.-Y. Hu, “Reduced-complexity decoding of LDPC codes,” *IEEE Trans. Commun.*, vol. 53, no. 8, pp. 1288–1299, Aug. 2005.
- [25] K.-Y. Kuo, I.-C. Chern, and C.-Y. Lai, “Decoding of quantum data-syndrome codes via belief propagation,” in *Proc. IEEE Intl. Symp. Inf. Theory*, IEEE, 2021, pp. 1552–1557.
- [26] P. Panteleev and G. Kalachev, “Quantum LDPC codes with almost linear minimum distance,” *arXiv preprint arXiv:2012.04068*, 2021. [Online]. Available: <https://arxiv.org/abs/2012.04068>
- [27] M. P. C. Fossorier, “Quasicyclic low-density parity-check codes from circulant permutation matrices,” *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1788–1793, Aug. 2004.
- [28] R. Tanner, D. Sridhara, A. Sridharan, T. Fuja, and J. Costello, D.J., “LDPC block and convolutional codes based on circulant matrices,” *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 2966–2984, Dec. 2004.
- [29] N. Raveendran, N. Rengaswamy, F. Rozpędek, A. Raina, L. Jiang, and B. Vasić, “Finite rate QLDPC-GKP coding scheme that surpasses the CSS Hamming bound,” *arXiv preprint arXiv:2111.07029*, 2021. [Online]. Available: <https://arxiv.org/abs/2111.07029>