A Modified MWPM Decoding Algorithm for Quantum Surface Codes Over Depolarizing Channels

Yaping Yuan
Department of Electrical Engineering
National Tsing Hua University
Hsinchu 30013, Taiwan
yapingyuan@mx.nthu.edu.tw

Chung-Chin Lu
Department of Electrical Engineering
National Tsing Hua University
Hsinchu 30013, Taiwan
cclu@ee.nthu.edu.tw

Abstract—A quantum surface code is a quantum topological stabilizer code whose stabilizers and qubits are geometrically related. Due to their special structures, surface codes have great potential to be implemented in large-scale quantum computing systems. In the minimum weight perfect matching (MWPM) decoding of surface codes, bit-flip errors and phase-flip errors are assumed to be independent for simplicity. However, these two kinds of errors are likely to be correlated in the real world. In this paper, we propose a modification to the MWPM decoding for surface codes to deal with the noise in depolarizing channels where bit-flip errors and phase-flip errors are correlated. With this modification, we obtain thresholds of 17% and 15.3% for surface codes with mixed boundaries and surface codes with a hole, respectively.

I. INTRODUCTION

Quantum error-correcting codes play a very important role in the development of quantum computation because of the inherent sensitivity of quantum systems to noise. Stabilizer codes are a class of quantum error-correcting codes that have a strong connection with classical error-correcting codes. The code space of a stabilizer code is determined by the so-called stabilizers. Topological codes are a class of stabilizer codes whose stabilizers and data qubits are topologically related. It is believed that topological codes have great potential to be implemented on large scales due to their special structures. Therefore, topological codes have gained a lot of attention in recent years. Surface codes are a family of topological codes defined on a 2D lattice of qubits [1], [2].

Various decoders for surface codes have been developed in recent years, such as decoders based on belief-propagation (BP) [3], [4], union-find (UF) [5], or matrix product states (MPS) [6], [7]. The most standard decoder for surface codes is the minimum weight perfect matching (MWPM) decoder. When bit-flip errors and phase-flip errors are assumed to be uncorrelated, maximum likelihood decoding (MLD) of a surface code can be reduced to problems of finding a minimum weight perfect matching on a graph. However, the depolarizing noise model, where bit-flip errors and phase-flip errors are correlated, is actually met in the real world. In this paper, we propose a modification to the MWPM decoding of surface codes to deal with the noise in depolarizing channels.

Our method is based on iteratively reweighting the dual lattice and the primal lattice with the correction pattern on the other lattice. Similar methods were proposed in [9], [10], but it was not shown whether it is possible that the weight of the correction operation will increase along iterations. In this paper, besides showing how the iteratively reweighted MWPM decoding works, we will also prove that the weight of the correction operation will never increase along iterations.

This paper is organized as follows. In Section II, we review the structures of surface codes and the MWPM decoder. In Section III, we discuss our modification to the MWPM decoder. In Section IV, we provide the simulation results for the IRMWPM decoding. Section V concludes this paper.

II. BASICS OF SURFACE CODES

A. Structures of Surface Codes

In this paper, we describe a surface code in a similar way as [11] does. A surface code is defined on a square lattice and every edge on this lattice is associated with a qubit. There are two types of stabilizer generators: plaquette stabilizer generators and vertex stabilizer generators. Every plaquette is associated with a plaquette stabilizer generator. A plaquette stabilizer generator consists of a tensor product of Pauli Z operators acting on the qubits that lie on the plaquette’s boundary, as illustrated in Fig. 1(a). For every vertex, there is a vertex stabilizer generator which consists of a tensor product of Pauli X operators acting on the qubits adjacent to the vertex, as shown in Fig. 1(b).

There are two main types of surface codes, one is built on a lattice with mixed boundaries [1], and the other is built on a lattice with holes (or defects) [2]. Surface codes with mixed boundaries are constructed on a lattice surrounded by two pairs of different boundaries, as shown in Fig. 2(a). Surface codes with a hole are constructed on a lattice where a plaquette stabilizer generator in the middle of the lattice is removed, as shown in Fig. 2(b). Note that the size of a hole is not necessarily $1 \times 1$. In Fig. 2, the original lattices are called the...
A Pauli $Z$ error and a $Z$ distance of Fig. 2(a) is two pieces of boundary of the same type. Therefore, the code distance is the length of a shortest path which connects the stabilizer itself. For a surface code with mixed boundaries, its code distance is the length of a shortest path which connects two plaquette generators, but both $s_2$ and $s_3$ anti-commute with only one plaquette operator.

Fig. 1. (a) A plaquette stabilizer generator. (b) A vertex stabilizer generator.

Fig. 2. (a) A surface code with mixed boundaries. (b) A surface code with a hole.

Fig. 3. (a) A tensor product of $Z$ errors is depicted in blue lines and the corresponding syndrome nodes are depicted in blue filled circles. (b) A tensor product of $X$ errors is depicted in red lines and the corresponding syndrome nodes are depicted in red filled circles. The string $s_1$ anti-commutes with two plaquette generators, but both $s_2$ and $s_3$ anti-commute with only one plaquette operator.

B. Syndromes of Surface Codes

For a stabilizer code, each stabilizer generator corresponds to an element of the syndrome vector. For an error $E$, the stabilizer generators that anti-commute with $E$ will give a 1 in the syndrome vector, otherwise 0. For simplicity, we call the stabilizer generators that give nonzero syndrome elements in a surface codes “syndrome nodes”.

Suppose $E_Z$ is a tensor product of Pauli $Z$ errors. Since a Pauli $X$ anti-commutes with a Pauli $Z$, if we express $E_Z$ as strings on the primal lattice, then the syndrome nodes corresponding to $E_Z$ are the endpoints of those strings, as shown in Fig. 3(a). Similarly, we can express $X$-type errors as strings on the dual lattice, and the corresponding syndrome nodes are the endpoints of those strings, as shown in Fig. 3(b). Since a Pauli $Y$ anti-commutes with both a Pauli $X$ and a Pauli $Z$, we can treat a $Y$ error as a combination of an $X$ error and a $Z$ error.

The code distance of a stabilizer code is the minimal size of an error which commutes with all of the stabilizers but is not a stabilizer itself. For a surface code with mixed boundaries, its code distance is the length of a shortest path which connects two pieces of boundary of the same type. Therefore, the code distance of Fig. 2(a) is 4. For a surface code with a hole, operators that commute with all stabilizers but not stabilizers themselves are either loops of $Z$ operators that wind around the hole or strings of $X$ operators that connect the inner and outer boundaries. Let the distance of a shortest path between the inner boundary and the outer boundary be $d_h$ and the number of qubits around the hole be $d_b$. The code distance of a surface code with a hole is $d = \min(d_h, d_b)$. Therefore, the code distance of Fig. 2(b) is 2.

C. MWPM decoding of Surface Codes

Since the syndromes of a surface code can be viewed as nodes on the primal lattice and the dual lattice, the maximum likelihood decoding can be reduced to the problem that finds the most likely string patterns with the same syndrome nodes. How to choose the most likely correction strings depends on the noise model.

Suppose that $X$ errors and $Z$ errors are independent and a $Y$ error is considered as a combination of an $X$ error and a $Z$ error. Then we can decode $X$ errors and $Z$ errors separately. To decode $Z$ errors only, we just need to find the strings on the primal lattice with the minimum weight which connect all syndrome nodes on the primal lattice. It is similar for the decoding of $X$ type errors, but the lattice we are working on is the dual lattice instead. Therefore, the decoding of a surface code can be viewed as two minimum weight perfect matching problems. Although the number of syndrome nodes may be odd, with some modifications, the decoding can still be reduced to MWPM problems. The noise model where $X$ errors and $Z$ errors are independent to each other is called the uncorrelated noise model. To solve an MWPM problem, we can use the blossom algorithm developed by Jack Edmonds [8]. The time complexity of the blossom algorithm for a graph $G = (V, E)$ is $O(|V|^3)$. Therefore, the time complexity of the MWPM decoder is $O(n^3)$.

III. ITERATIVELY REWEIGHTED MWPM DECODING OF SURFACE CODES

The depolarizing noise model is the most considered noise model in quantum error-correction. In a depolarizing channel, each qubit has the probability of $1 - \epsilon$ to remain untainted and the probability of $\frac{\epsilon}{2}$ to be affected by $X$, $Y$, or $Z$, respectively. Therefore, if we view a $Y$ error as a combination of $X$ and $Z$, the conditional probability $P(\text{an } X \text{ error on a qubit } i | \text{a } Z \text{ error on a qubit } i) = 0.5$, so $X$ errors and $Z$ errors are not independent to each other.

As shown in Fig. 4, if we use the MWPM decoding, we will get a decoding result as in Fig. 4(a). However, if the noise model we are considering is the depolarizing noise model, the decoding result of MLD should be as in Fig. 4(b).

In Fig. 4(a), we have 4 $X$ operators and 4 $Z$ operators. Thus the total weight of this correction is 8. In Fig. 4(b), although there are 6 $X$ operators and 4 $Z$ operators, we have 4 $Y$ and 2 $X$ under the view point of the depolarizing noise model. Since the total weight in Fig. 4(b) is only 6, it is better than that in Fig. 4(a) when the noise model is the depolarizing noise model.
Suppose that the correction strings on the primal lattice are fixed. We can find that if a string on the dual lattice touches a string on the primal lattice, the intersection does not increase the total weight. This is because it just turns a single $Z$ correction into a single $Y$ correction. Therefore, a shortest path between two syndrome nodes on the dual lattice is not necessarily corresponding to a string that can minimize the total weight. However, if we reweight the edges on the dual lattice that touch the correction strings on the primal lattice to 0, the shortest path between the two syndrome nodes on the reweighted dual lattice is the correction string that causes the least extra total weight. As shown in Fig. 5, the shortest path between the two syndrome nodes on the reweighted dual lattice is now $P_2$ instead of $P_1$. Therefore, when the correction of $Z$-type error is fixed, finding the MWPM on the reweighted dual lattice can give us the error pattern that minimizes the total weight.

![Image of the reweighted dual lattice](image)

Fig. 5. The reweighted dual lattice.

Since the $Z$ correction in the decoding result of MLD may not be an MWPM on the primal lattice, we can repeat this process more than one time to give us a better decoding result. We can use an MPWM on the reweighted dual lattice to reweight the primal lattice in a similar way, and then use the new MWPM on the reweighted primal lattice to reweight the original dual lattice again. In this paper, we will prove that no matter how many times we repeat this reweighting process, the total weight will only become smaller and smaller or remain the same.

Let $P_0$ be the original primal lattice and $D_0$ the original dual lattice. Let $B_0$ be an MWPM on $P_0$ and $R_0$ an MWPM on $D_0$. We reweight $D_0$ with $B_0$ and call it the first reweighted dual lattice $D_1$. Let $R_1$ be an MWPM on $D_1$. We reweight $P_0$ with $R_1$ and call it the first reweighted primal lattice $P_1$. We can use similar way to construct $B_k$ and $R_k$, $k \in \mathbb{N}$. Note that for $i > 0$, $D_i$ is constructed by reweighting $D_0$ with $B_{i-1}$ and $P_i$ is constructed by reweighting $P_0$ with $R_i$.

When we calculate the total weights of $B_i$ and $R_i$, we can not just calculate their weights separately and sum them up. One of them must be calculated on the original lattice and the other is calculated on the lattice reweighted with the first matching. Since $P_i$ is constructed based on $R_i$, to calculate the total weights of $B_i$ and $R_i$, we can calculate the weight of $R_i$ on $D_0$ first, and calculate that of $B_i$ on $P_i$, and then sum them up.

Let the weight of a matching $M$ on the $i$th reweighted lattice as $W_i(M)$. We can define the $i$th total weight as

$$T_i = W_i(B_i) + W_i(R_i), \quad i \geq 1.$$  

For the case $i = 0$, we need a different definition, since $W_0(B_0)$ is clearly not the weight of $B_0$ on the lattice reweighted with $R_0$. But since $D_1$ is the lattice reweighted with $B_0$, we can sum the weight of $B_0$ on $P_0$ and that of $R_0$ on $D_1$. Thus, the total weight of $B_0$ and $R_0$ is

$$T_0 = W_0(B_0) + W_1(R_0).$$

An example of the modified decoding process can be seen in Fig. 6.

![Image of the modified decoding process](image)

Fig. 6. An example of the modified decoding process.

**Theorem 1.** $T_{i+1} \leq T_i$ for all $i \in \mathbb{N}_0$.

**Proof.** For an MWPM $M_P$ on the primal lattice and an MWPM $M_D$ on the dual lattice, there are two ways to
calculate the total weight. The first one is summing the weight of $M_P$ on the original primal lattice and the weight of $M_D$ on the dual lattice reweighted with $M_P$. The second one is the reverse, i.e., summing the weight of $M_D$ on the original dual lattice and the weight of $M_P$ on the primal lattice reweighted with $M_D$. Therefore, for $i \geq 1$, we have the following properties,

$$W_i(B_i) + W_0(R_i) = W_0(B_i) + W_{i+1}(R_i) \quad (1)$$

$$W_0(B_i) + W_{i+1}(R_i) = W_{i+1}(B_i) + W_0(R_{i+1}). \quad (2)$$

Since the definition of $T_i$ is different for $i = 0$ and for $i \geq 1$, we need to discuss two cases. For $i = 0$, since $R_1$ is an MWPM on $D_1$, we have $W_1(R_1) \leq W_1(R_0) = T_0$.

Since $W_0(B_0) + W_1(R_0) = W_1(B_0) + W_0(R_0)$, we have $W_1(B_0) + W_0(R_0) = W_1(B_0) + W_0(R_0) = T_0$.

Since $B_1$ is an MWPM on $P_1$, we have $W_1(B_1) \leq W_1(B_0)$. Therefore,

$$T_1 = W_1(B_1) + W_0(R_1) \leq W_0(B_0) + W_1(R_0) = T_0.$$

For $i \geq 1$, let us start from $T_i = W_i(B_i) + W_0(R_i)$. We have $T_i = W_i(B_i) + W_{i+1}(R_i)$ by Equation (1). Since $R_{i+1}$ is an MWPM on $D_{i+1}$, we have

$$W_0(B_i) + W_{i+1}(R_{i+1}) \leq W_0(B_i) + W_{i+1}(R_i) = T_i.$$

By Equation (2), we have

$$W_0(B_i) + W_{i+1}(R_{i+1}) = W_{i+1}(B_i) + W_0(R_{i+1}).$$

Similarly, since $B_{i+1}$ is an MWPM on $P_{i+1}$, we have $W_{i+1}(B_{i+1}) \leq W_{i+1}(B_i)$. Then, we have

$$T_{i+1} = W_{i+1}(B_{i+1}) + W_0(R_{i+1}) \leq W_{i+1}(B_i) + W_0(R_{i+1}) \leq T_i.$$

Here we need to indicate that this modification does not guarantee the minimum total weight result. Take Fig. 4 as an example. If the MWPM we find on the primal lattice is as Fig. 7, then this method will fail to give the correction pattern with minimum total weight.

![Fig. 7](image)

**Fig. 7.** For the syndrome in Fig. 4, if $Z$-type errors are decoded as shown here, then this algorithm will not give us the correction pattern with minimum total weight. This example shows that the IRMWPM decoder does not guarantee maximum likelihood decoding over a depolarizing channel.

In Section II, we did not discuss the time complexity of constructing a syndrome node graph, since the shortest path between any two syndrome nodes can be obtained in $O(1)$. However, the shortest path between two syndrome nodes on a reweighted lattice is not that clear. To find the shortest paths between all pairs of nodes in a graph, we can use Floyd-Warshall algorithm or use Dijkstra’s algorithm on each pair of nodes. For a graph with $n$ nodes, the time complexity is $O(n^3)$ for both methods. Therefore, the time complexity of constructing a syndrome node graph is $O(n^3)$.

We will see in Section IV that it is rare to need more than 5 iterations for lattices smaller than $30 \times 30$. Therefore, we can neglect how many iterations are used in the calculation of the total time complexity and the time complexity of the IRMWPM decoder is still $O(n^3)$.

**IV. Simulation Results**

Since we will repeat the same process more than one time, we need to set a stopping criterion. Stopping the iterations as soon as the error weight stops decreasing is not good enough because it is possible that the error weight stays at a particular value for a few iterations and then drops again. Suppose that the subroutine we use to find MWPMs will give us the same results for two same complete graphs. We can use whether there is a previous correction pattern which is the same as the current one as the stopping criterion.

Here, we show the decoding performance of three different cases in Fig. 8. In the first one, we only apply MWPM decoding without any reweighting. In the second one, we reweight the dual lattice only one time, i.e., using $B_0$ and $R_1$ as the correction. In the third one, iterating will not stop until the newest MWPM is the same as one of the previous MWPMs. We can see that the logical error rate decreases as more iterations are applied.

![Fig. 8](image)

**Fig. 8.** Simulation of decoding surface codes with mixed boundaries.

Now we want to know how many iterations do we need. Empirically, it is rarely over 5 iterations when the lattice is smaller than $30 \times 30$. The counting of extra iterations
χ and the IRMWPM decoder are codes with a hole, and the thresholds of the MWPM decoder as shown in Fig. 10. Similar effects are observed for surface boundaries, the thresholds of the MWPM decoder is greater than the threshold. For surface codes with mixed boundaries need for different code distances.

For a surface code, we may want to increase the code size in order to lower the logical error rate, but the larger the lattice is, the more errors may be introduced into the system. Threshold is an index for a surface code decoder to evaluate this phenomenon. The logical error rate increases as the size of the code gets larger and larger when the qubit error rate is greater than the threshold. For surface codes with mixed boundaries, the thresholds of the MWPM decoder is 15.5% and that of the IRMWPM decoder is improved to 17.0%, as shown in Fig. 10. Similar effects are observed for surface codes with a hole, and the thresholds of the MWPM decoder and the IRMWPM decoder are 14.2% and 15.3%, respectively, as shown in Fig. 11. The thresholds and time complexity of various decoders are provided in Table I and \( \chi \) is a parameter that controls the approximation precision.

V. CONCLUSION

We propose a modification to the conventional MWPM decoding of quantum surface codes to deal with the noise in depolarizing channels where bit-flip errors and phase-flip errors are correlated. Our method is mainly based on repeatedly using an MWPM on one lattice to reweight the other lattice to get a correction pattern with possibly less total weight. In this paper, we prove that the total weight will never increase when we repeat this reweighting process, and we present simulation results of both surface codes with mixed boundaries and surface codes with a hole to show the improvement of the decoding performance.

REFERENCES

[1] S. B. Bravyi and A. Y. Kitaev, “Quantum codes on a lattice with boundary,” arXiv preprint quant-ph/9811052, 1998.
[2] M. H. Freedman and D. A. Meyer, “Projective plane and planar quantum codes,” Foundations of Computational Mathematics, vol. 1, no. 3, pp. 325-332, 2001.
[3] K.-Y. Kuo and C.-Y. Lai, “Exploiting degeneracy in belief propagation decoding of quantum codes,” arXiv preprint arXiv:2104.13659, 2021.
[4] S. Bravyi, M. Suchara, and A. Vargo, “Efficient algorithms for maximum likelihood decoding in the surface code,” arXiv preprint arXiv:1310.0863, 2013.
[5] B. Criger and I. Ashraf, “Multi-path Summation for Decoding 2D Topological Codes,” Quantum, vol. 2, pp. 102, 2018.
[6] N. Delfosse and N. H. Nickerson, “Almost-linear time decoding algorithm for topological codes,” arXiv preprint arXiv:1709.06218, 2017.
[7] M. H. Freedman and D. A. Meyer, “Projective plane and planar quantum codes,” Foundations of Computational Mathematics, vol. 1, no. 3, pp. 325-332, 2001.
[8] K.-Y. Kuo and C.-Y. Lai, “Exploiting degeneracy in belief propagation decoding of quantum codes,” arXiv preprint arXiv:2104.13659, 2021.
[9] B. Criger and I. Ashraf, “Multi-path Summation for Decoding 2D Topological Codes,” Quantum, vol. 2, pp. 102, 2018.
[10] N. Delfosse and H. Nickerson, “Almost-linear time decoding algorithm for topological codes,” arXiv preprint arXiv:1709.06218, 2017.
[11] S. Bravyi, M. Suchara, and A. Vargo, “Efficient algorithms for maximum likelihood decoding in the surface code,” Phys. Rev. A, vol. 90, no. 032326, 2014.
[12] C. T. Chubb, “General tensor network decoding of 2D Pauli codes,” arXiv preprint arXiv:2101.04125v3, 2021.
[13] J. Edmonds, “Paths, trees, and flowers,” Canadian Journal of mathematics, vol. 17, pp. 449-467, 1965.
[14] A. G. Fowler, “Optimal complexity correction of correlated errors in the surface code,” arXiv preprint arXiv:1310.0863, 2013.
[15] N. Delfosse and J. Tillich, “A decoding algorithm for CSS codes using the XZ correlations,” 2014 IEEE International Symposium on Information Theory, pp. 1071-1075, 2014.
[16] H. Bombin, “Topological codes,” in Quantum Error Correction, D. A. Lidar and T. A. Brun, Eds., 1st ed. Cambridge: Cambridge University Press, 2013.