

二値化ニューラルネットワークを用いた量子誤り訂正の検討

水野 太資¹ 上野 洋典² 近藤 正章^{2,3}

概要：量子コンピュータの誤り訂正符号として Surface code が知られており、その復号をニューラルネットワークで行う手法が提案されている。また、超伝導素子による量子コンピュータの誤り訂正は高速・低消費電力でなされる必要がある。本稿ではニューラルネットワークによる Surface code の復号を高速・低消費電力化するために、二値化ニューラルネットワーク（BNN）を用いる手法について検討する。Surface code のシンドロームを入力してエラー訂正に必要な操作を出力する BNN を構築し学習することで、BNN による Surface code の復号器を構成した。数値シミュレーションにより物理エラー率に対する論理エラー率を評価し、その復号器が復号手法として有効であることを確認した。

1. はじめに

ポストムーア時代の新たな計算機として、量子コンピュータが注目されている。量子コンピュータとは、量子状態 $|0\rangle$ と量子状態 $|1\rangle$ の確率的な重ね合わせで表現される「量子ビット」により情報を表現し、量子ビットの量子状態を操作することで計算を実行するコンピュータである。量子ビットの物理的な実現方法は光子 [1]、量子ドット [2]、超伝導素子 [3]、シリコン量子ビット [4] など様々である。中でも有望な実装方式の一つと考えられているのが超伝導素子によるものであり、2019 年には Google の研究チームが超伝導量子ビットを用いた量子コンピュータの量子超越性、つまり古典コンピュータでは莫大な実行時間がかかり現実的に解くことが不可能な特定の問題が、量子コンピュータにより現実的な実行時間で解けることが示されたと報告した [5]。本稿は超伝導量子ビットからなる量子コンピュータを対象にしている。

量子コンピュータの実現には様々な障壁があるが、そのうちの大きな一つに量子ビットの高いエラー率が挙げられる。エラー訂正機能を持つ量子コンピュータを誤り耐性量子計算機あるいは FTQC (Fault Tolerant Quantum Computer) と呼び、一般的には複数の物理量子ビットを組み合わせて 1 つの論理量子ビットを表すことで、物理量子ビットに発生するエラーを検出・訂正する。この際の論理ビットの構築には量子誤り訂正符号が用いられ、量子誤

り訂正符号の復号プロセスは古典コンピュータにより実行できることが知られている。つまり、古典コンピュータを量子コンピュータのアシストとして用いることで、量子コンピュータに発生するエラーを訂正・検知することができる。本稿ではこのエラー訂正を行う古典コンピュータを復号器、あるいはデコーダと呼ぶ。

超伝導量子ビットは超伝導現象を利用しているため、20mK 程度の極低温環境でのみ動作するという物理的な制約がある。そのため、超伝導量子コンピュータは希釈冷凍機の中で動作させるのが一般的である。量子ビットが極低温環境で動作するのに対し、デコーダは一般的に室温環境で動作させるため、量子ビットとデコーダを繋ぐ異なる温度環境間の配線数の多さがボトルネックとなり、量子コンピュータのスケラビリティが制限されてしまうという課題がある。

このボトルネックを解消する方法の 1 つとして、デコーダを極低温環境で動作させることが考えられる。しかし、極低温を保つために希釈冷凍機内で許容される消費電力には厳しい制約があり、デコーダの消費電力が制約以上であるとデコーダの発熱により極低温環境が破壊されてしまい、量子コンピュータが正常に動作しなくなってしまう。現在の希釈冷凍機において、量子ビットの動作する 20mK 程度の層では $10\sim 100\mu\text{W}$ 程度、その 1 つ上の 4K 程度の層では 1W 程度の消費電力しか許容されない [6]。半導体を低温環境で動作させる CryoCMOS 技術もあるが、それでも消費電力は 20mK および 4K 層の許容消費電力に比べると大きい [7], [8]。また、1 回の復号プロセスにかかる時間が長くなると、その間に量子ビットにより多くのエラーが発生してしまい、結果としてエラー訂正に失敗する可能性も高くなる。そこで、スケラブルな誤り耐性量子コンピュータシステムの実現には、高速かつ超低消費電力な量

¹ 東京大学工学部
Faculty of Engineering, The University of Tokyo

² 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology,
The University of Tokyo

³ 理化学研究所計算科学研究センター
RIKEN Center for Computational Science

子誤り訂正符号向けのデコーダが必要となる。

本稿では、スケーラブルな量子コンピュータシステムの実現に向け、量子誤り訂正符号の復号を高速かつ低電力に行うために、二値化ニューラルネットワークを用いたエラー訂正手法について検討する。従来から、量子誤り訂正符号において、その復号を畳み込みニューラルネットワーク (CNN) で行えることが知られており [9]、本稿ではそれを二値化ニューラルネットワークで実装することを考える。量子誤り訂正評価用のミュレータを用い、二値化ニューラルネットワークによるデコーダの量子誤り訂正性能、および計算量を評価する。

2. 量子誤り訂正の理論的背景と関連研究

2.1 量子ゲート方式の量子コンピュータの原理

量子コンピュータとは、量子ビットによって情報が表現されるコンピュータである。量子ビットは測定するとある確率で量子状態 $|0\rangle$ を、またある確率で量子状態 $|1\rangle$ が観測される量子状態をとることができ、 $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ のように表現される。ここで α, β は複素数で $\alpha^2 + \beta^2 = 1$ であり、 $|\Psi\rangle$ は確率 α^2 で量子状態 $|0\rangle$ を、確率 β^2 で量子状態 $|1\rangle$ をとる。 α, β の値が 0 あるいは 1 しか取らない場合が古典ビットに相当するため、量子ビットの表現力は古典ビットを大きく上回る。

量子ゲート方式をとる量子コンピュータでの計算は、量子ビットの状態 (α, β) を量子ゲートを用いて操作することで実行される。1 量子ビットに対する量子ゲート操作は 2×2 複素行列で表すことができ、規格化条件からユニタリ演算である必要がある。この量子ゲート中で基本的なものはパウリゲートと呼ばれ、量子状態を変化させない I ゲート、ビット反転させる X ゲート、位相反転させる Z ゲート、ビットと位相を反転させる Y ゲートがある。I は単位行列であり、X, Y, Z はそれぞれ

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

である。

2.2 量子コンピュータにおけるエラー

量子ビットはノイズに弱く、その量子状態 $|\Psi\rangle$ は非常に短い時間で崩壊し量子状態 $|0\rangle$ になる [10], [11]。量子コンピュータでは、量子ビットにどのようなエラーが発生しても、射影測定により重ね合わせ状態を壊すと、結果としてビット反転エラー、位相反転エラー、またはその両方に帰着できる。これらは、パウリゲート X, Z, Y で量子状態を操作した結果と同じであるため、それぞれ X エラー、Z エラー、Y エラーと呼ばれる。

量子誤り訂正アルゴリズムを検討・評価する上で、量子ビットに発生するエラーを忠実に再現しようとする、量子状態の直接的なシミュレーションが必要となり、必要な

計算量やメモリ量が膨大となる。そのため、エラーを検討する際には単純なエラーモデルを仮定することが多い。エラーモデルの代表例には、ビット反転と位相反転が独立に起きる、つまり Y エラーを考慮せず X エラーと Z エラーだけが発生するという Capacity モデルと、ビット反転と位相反転に相関がある、つまり X, Y, Z エラーがそれぞれ発生するという Depolarize モデルの 2 つがある。本稿では後述する理由から Depolarize モデルを用いてエラー訂正手法の検討とその性能評価を行う。

2.3 量子誤り訂正

量子ビットにおける誤り訂正の基本は古典コンピュータと同じく冗長化であり、同じ情報を複数ビットで保持することで、いくつかのビットでエラーが発生しても正しい状態へと修正するものである。しかしながら、量子ビットは測定すると量子状態が壊れ $|0\rangle$ か $|1\rangle$ の状態になってしまうため、情報を保持している量子ビット (データ量子ビット) を直接測定してエラーを検出・訂正することはできない。また、量子ビットは複製不可能定理から古典コンピュータの場合のように状態を複製するような冗長化はできないほか、取り得る状態も無数にあるためエラーのパターンも無数に考えられる。これらが古典コンピュータに比べて量子コンピュータのエラー訂正が困難な点である。

そこで、量子ビットの測定には目的の量子状態を保持するデータ量子ビットに加えて観測用の補助量子ビットを設け、両者を量子もつれ状態にする方法が用いられる。データ量子ビットと量子もつれ状態にある補助量子ビットを測定することで間接的にデータ量子ビットの量子状態についての情報が得られる。これにより、データ量子ビットの量子状態を壊さずに測定が可能となる。

補助量子ビットを用いた一般的な量子誤り訂正符号の構成方法としてスタビライザ符号 [12] がある。これはスタビライザと呼ばれる演算子とスタビライザ状態と呼ばれる量子状態によって量子誤り訂正符号を構成する方法である。スタビライザとはスタビライザ群に含まれる演算子であり、スタビライザ群とは、パウリ演算子 $\{I, X, Y, Z\}$ と $\{\pm 1, \pm i\}$ の積であるパウリ群 $\{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n}$ のうち $-I$ を含まない可換部分群である。また、スタビライザ状態とはスタビライザの固有値が +1 となる量子状態を指す。スタビライザ群の指定によって量子状態を指定する方法をスタビライザ形式という。スタビライザ符号の誤り訂正は、各スタビライザの固有値を測定する測定プロセスと、そこから得られた情報を元に誤りを特定し訂正する復号プロセスの 2 つのプロセスに分けられる。論理ビットが符号状態、すなわちスタビライザ状態である場合には、測定プロセスで得られる固有値の測定値はすべて +1 となる。一方、物理量子ビットにエラーが生じ、論理ビットがスタビライザ状態でなくなると測定プロセスでいくつかのスタビライザから固有値 -1 が測定される。固有値 -1 が測定された

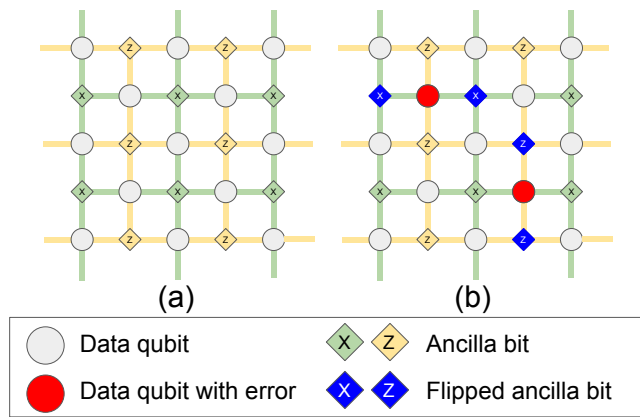


図 1 (a): 符号距離 3 の Surface Code (b): データ量子ビットにエラーが発生した場合

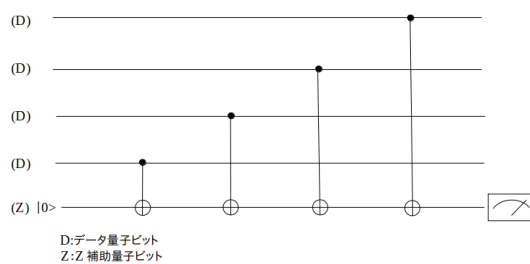


図 2 Z スタビライザ

スタビライザの箇所からエラーの箇所を特定し、訂正するのが復号プロセスである。

2.4 Surface Code

Surface Code[13] はスタビライザ符号の 1 つで、図 1(a) のようにデータ量子ビットと補助量子ビット（アンシラビット）を格子状に並べた論理量子ビット上で動作する誤り訂正符号である。符号理論においてある符号語（スタビライザ符号ではスタビライザ状態）を別の符号語に変えるために必要な操作の数を符号距離というが、Surface Code の符号距離は 1 辺のデータ量子ビットの数と一致する。一般に、誤り訂正符号は符号距離 d の半分未満のエラーのみを正しく訂正可能である。

補助量子ビットは X 補助量子ビットと Z 補助量子ビットの 2 種類あり、図 1 (b) のようにそれぞれが X エラー、Z エラーを検出する。X 補助量子ビットと Z 補助量子ビットの構造は、それぞれ 4 つのデータ量子ビットのパリティチェックをすることでエラーを検出する仕組みであり、この機構をそれぞれ Z スタビライザ、X スタビライザという。図 2 は Z スタビライザを量子回路で表した図である図 2 において、D はある補助量子ビットの前後左右に隣接した 4 つのデータ量子ビットであり、これらデータ量子ビットに制御された 4 つの CNOT ゲートが補助量子ビットに作用している。データ量子ビットのうち奇数個がビット反転している場合、補助量子ビットの出力が 1 になる。また、補助量子ビットの出力の集合のことをシンドロームと呼ぶ。

2.4.1 Surface Code の復号プロセス

図 1(b) に Surface Code においてデータ量子ビットにエラーが発生した様子を示す。前述のようにデータ量子ビットを直接観測することはできないので赤丸で表したエラー発生箇所を直接特定することはできない。そこで、得られたシンドロームからデータ量子ビットに発生したエラーの位置と種類を特定するのが Surface Code の復号プロセスである。

Surface Code の復号において、論理誤り率を最も小さくするような回復操作を行う最適復号、あるいはシンドローム値を再現する最も確率の高いエラーパターンを選ぶ最尤推定を行うことが望ましい。しかし、どちらの処理も一般に NP 困難であるため、最適復号や最尤推定よりも性能は劣るがより計算量の小さい復号手法が使われることが多い。以下にその代表例を 2 つ述べる。

2.4.2 最小重み完全マッチングによる復号

Surface Code の代表的な復号方法の 1 つとして、完全グラフの最小重み完全マッチング (MWPM: Minimum Weight Perfect Matching) 問題へ帰着する方法が知られている。この手順は以下のように説明できる。まず、固有値-1 を観測したアンシラビットをグラフの頂点とし、重みをそれらの間のマンハッタン距離として完全グラフ、すなわち任意の頂点間に枝を持つグラフを構築する。そのグラフ上の完全マッチングのうち、マッチングに含まれる辺の重みの和を最小にするものをとる。グラフ理論においてマッチングとは、グラフの枝の集合で互いに端点を共有しないものを指す。そのうち完全マッチングとは、グラフ上の全ての頂点がマッチング中のいずれかの枝の端点になっているものを指す。この操作を行うことで、シンドロームからなるべく距離の短いエラー鎖の集合を見つけることができる。

この復号方法は最も標準的な方法であり、 $O(n^3)$ で動作する Blossom アルゴリズムが知られている [14]。ノイズモデルとして Capacity モデルを仮定した場合、すなわち量子ビットに X と Z のどちらかのエラーが完全に独立に発生する場合、この復号方法の結果は最尤推定と一致する。エラーモデルとして Depolarize モデルを仮定した場合、Y エラーの存在により X と Z エラーの存在位置には相関がある。しかし、この手法では X と Z エラーが独立に発生するという仮定を暗に置いている、すなわち相関の情報を使わずに復号を行うため、その結果は最尤推定よりも悪くなる。

2.4.3 ニューラルネットワークによる復号

別の復号方法として、ニューラルネットワークを用いる方法がある [9]。シンドローム値を入力とし、エラー訂正に必要な回復操作を出力するニューラルネットワークを学習することで復号を行う。ニューラルネットワークの計算量はモデル次第ではあるが NP 困難ではないため、最適復号や最尤推定よりも低い計算コストで実行可能である。ま

た、適切に学習を行うことができれば、Y エラーの存在による X と Z の相関についても考慮した復号を行うことができる。そのため、仮定するエラーモデルが Depolarize モデルの場合において、ニューラルネットワークによる復号は MWPM よりも良い性能となることが期待される。

2.5 関連研究

デコーダを量子コンピュータと同じ環境で動作させるために超伝導単一磁束量子回路 (Single Flux Quantum) を用いた研究がある。Holmes ら [15] は、SFQ 回路で実装したデコーダを用いて復号を量子コンピュータに近接した場所で行うことで、復号時間を大きく短縮できる結果を示している。

ニューラルネットワークを用いて Surface Code の復号を行った研究として、Davaasuren ら [9] が行った畳み込みニューラルネットワーク (CNN) によるものがある。この研究では 3 層の畳み込み層と 3 層の全結合層を持った 32 ビットの CNN モデルを用いて誤り訂正性能を検証し、CNN の量子誤り訂正における有効性を示している。

3. 提案手法

本章では二値化ニューラルネットワークを用いて Surface code の復号を行う手法を提案し、その詳細について述べる。

3.1 二値化ニューラルネットワーク

ニューラルネットワーク (NN) の学習・推論は、通常単精度浮動小数点演算で行うことが多いが、ある程度ビット数を削減しても精度には大きな影響がないことが知られており、これを量子化という。NN は計算量が大きく、それゆえ実行時間・消費電力も大きくなることから、NN の精度を保ちつつ計算を高速・低消費電力化する方法として量子化 [16] が広く用いられている。

量子化の最も極端な例が、NN のパラメータを -1 と 1 の 2 値で表現する二値化ニューラルネットワーク (BNN: Binarized Neural Network) である [17], [18]。BNN の計算は整数や浮動小数の数値計算ではなく、シンプルなビット演算のみで実行可能なことから、3 ビット以上の量子化に比べて計算の飛躍的な高速・低消費電力化が期待される。本稿では、Courbariaux らによる先行研究 [17] を参考に BNN を利用する。

3.2 NN の入出力

本稿では、Surface Code のシンドロームを入力、回復に必要な論理操作を表すデータを出力とする NN を構築する。図 3 は符号距離 3 の場合の CNN モデルの入出力の具体例を示しており、データ量子ビットに図 1(b) と同様のエラーが発生しているとする。図 3 に示すように、入力データは Surface Code の論理ビットと同様の平面構造を持った 2 次元配列である。2 次元配列において、データ量子ビットの

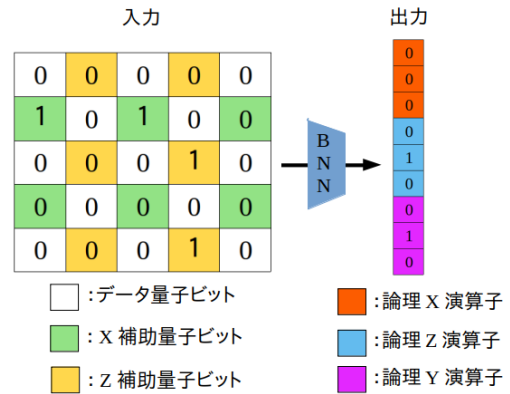


図 3 符号距離 3 の場合の CNN モデルの入出力

位置に対応する要素は常に 0 であり、X と Z の各補助量子ビットの位置の要素は、対応する補助量子ビットが反転していれば 1、そうでなければ 0 である。すなわち、Surface Code の符号距離 d の場合、入力データは $[2d-1, 2d-1]$ の次元を持つ各要素が 0 または 1 の 2 次元配列である。ただし、入力層が全結合層のモデルを考える場合、入力データとして 2 次元配列を取ることができない。その場合は X 補助量子ビットと Z 補助量子ビットの出力のみを並べた 1 次元配列を入力データとし、その配列の要素数は符号距離 d に対して $2d(d-1)$ となる。

符号距離 d の Surface Code は X, Y, Z についての論理演算子をそれぞれ d 個持っている。よって、合計 $3d$ 個の論理演算子のうち、エラーを回復するのに必要な論理操作を 1、そうでないものを 0 で表した $3d$ 次元の配列をニューラルネットワークの出力とする。また、シンドロームを表す入力データの形状は、モデルの構造が全結合 NN と畳み込み NN の場合で異なるので、次節でそれぞれの場合について述べる。

3.3 モデル構造

3.3.1 全結合 NN

全結合 NN のモデルとして、各層 2800 ノードの 2 層の隠れ層を持つモデルを構築する。これをモデル 0 と呼称する。

3.3.2 CNN

CNN のモデル構造は図 4 のように 6 層の畳み込み層、3 層のプーリング層、2 層の全結合層を持つ。また、各畳み込み層の後にはバッチ正規化を行なっている。この構造の各畳み込み層のフィルタサイズ及びフィルタ数を表 1 の通り設定し、それぞれモデル 1~4 とした。ただし d は符号距離、 σ はガウス記号である。

また、各モデル、各符号距離における演算回数は表 2 の通りである。

4. 評価

本章では、前章で構築した BNN モデルによる Surface

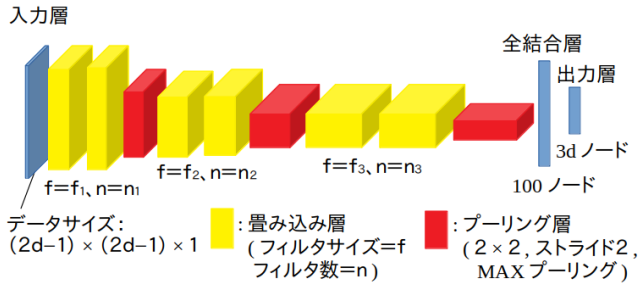


図 4 CNN のモデル構造

表 1 CNN モデルのフィルタサイズとフィルタ数

モデル	フィルタサイズ			フィルタ数		
	f_1	f_2	f_3	n_1	n_2	n_3
モデル 1	$[d/2] + 1$	$[f_1/2] + 1$	$[f_2/2] + 1$	512	512	512
モデル 2	d	$[f_1/2] + 1$	$[f_2/2] + 1$	512	512	512
モデル 3	d	$[f_1/2] + 1$	$[f_2/2] + 1$	128	256	512
モデル 4	d	$[f_1/2] + 1$	$[f_2/2] + 1$	256	512	1024

表 2 各モデル, 各符号距離における演算回数

符号距離	5	7	9	11
モデル 1	2.33×10^8	8.99×10^8	2.23×10^9	5.13×10^9
モデル 2	6.16×10^8	2.52×10^9	7.06×10^9	1.61×10^{10}
モデル 3	5.39×10^7	2.25×10^8	6.01×10^8	1.39×10^9
モデル 4	2.15×10^8	8.99×10^8	2.40×10^9	5.56×10^9

code のデコーダの性能を数値実験により評価する。4.1 節ではデコーダのエラー訂正性能を評価する指標について述べ、4.2 節では各モデルの学習方法について述べる。

4.1 評価指標

デコーダの性能を評価する指標として、しきい値を用いる。しきい値 p_{th} とは、符号化距離 d に依存せずに各デコーダ (本研究の例では各モデル) ごとに定義される値で、以下の条件を満たす物理エラー率 p を指す。すなわち、そのデコーダで誤り訂正を行なった場合の論理エラー率を p_L とし、 $p > p_{th}$ であれば符号距離 d が大きくなるにつれて p_L が大きくなり、 $p < p_{th}$ であれば符号距離 d が大きくなるに連れて p_L が小さくなるような p の値である。言い換えると、あるデコーダにしきい値 p_{th} が存在する場合、物理エラー率 p がしきい値よりも小さければ符号化距離 d を大きくして論理ビットの冗長性を高めれば、論理エラー率 p_L をいくらでも小さくできるということである。この値が大きいくほど、デコーダのエラー訂正性能は高いと言える。しきい値を確認できない場合、そのデコーダは符号距離に対するスケラビリティを持たず、誤り訂正手法として用いることはできない。

しきい値は、あるデコーダに関して横軸に物理エラー率 p 、縦軸に論理エラー率 p_L をとり各符号化距離 d の結果をプロットした場合に、各プロットが 1 点で交差する点の p として観察できる。例として MWPM によるエラー訂正の結果を図 5 に示す。 $p=0.1$ 付近で各符号化距離 d についてのプロットが交差しており、MWPM によるデコーダのし

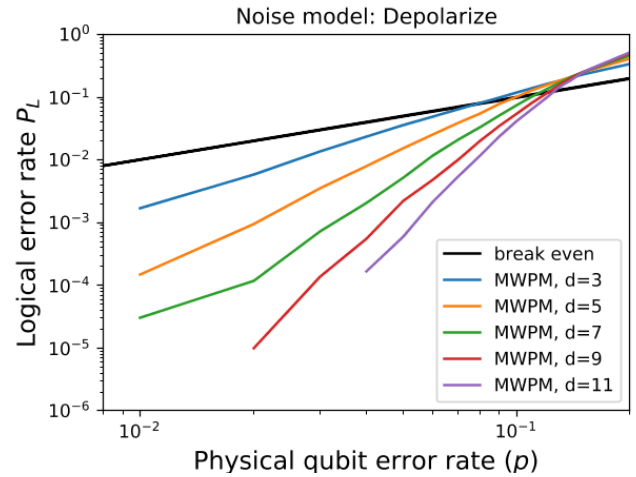


図 5 最小重み完全マッチングの場合の符号距離 5, 7, 9, 11 における物理エラー率と論理エラー率の関係

きい値は 0.1 程度であることがわかる。

4.2 学習方法

学習に用いるデータは、以下のように生成する。エラーモデルとして Depolarize モデルを仮定し、データ量子ビットにある確率 p に対して $[\frac{p}{3}, \frac{p}{3}, \frac{p}{3}]$ の確率で $[X, Z, Y]$ エラーを発生させる。データ量子ビットのエラーを元に補助量子ビットから出力されるシンδροームを算出する、すなわち復号と逆の操作を行うことでシンδροームを得る。また、エラーを発生させたデータ量子ビットの位置から、エラーを回復するのに必要な論理操作の組を決定する。このようにして、入力データであるシンδροームと、それに対する教師データであるエラー回復に必要な論理操作の組を得られる。

全結合 NN の学習は、符号距離 3, 5, 7, 9 の各場合に物理エラー率 0.01, 0.05, 0.10 の各場合について 1000000 の訓練データを生成して行なった。その後、学習済みモデルを 100000 の検証用データで論理エラー率を調べる、という方法で行った。

CNN の学習は以下の 2 種類の方法で行った。

- (1) 符号距離 5, 7, 9, 11 の各場合に物理エラー率 0.02, 0.04, 0.06, 0.08, 0.10 の各場合について訓練データ 100000 で学習し、検証用データ 100000 で論理エラー率を調べる。
- (2) 符号距離 5, 7, 9, 11 の各場合に物理エラー率 0.06 の場合について訓練データ 1000000 で学習し、検証用データ 100000 で物理エラー率 0.02, 0.03, ..., 0.14, 0.15 の場合の論理エラー率を調べる。

その他の学習条件は表 3 にまとめた。

学習方法 1 のように各物理エラー率ごとに学習する場合、各符号化距離 d および物理エラー率 p 、すなわちプロットの各点毎に学習を行うので、モデルの学習状況がプロットの各点で異なるということがあり得る。よって同一モデルについて学習方法 1 で学習した結果を各符号化距離 d につ

表 3 NN モデルの学習条件

Parameter	Description
エポック数	20
Optimizer	確率的勾配降下法 (SGD)
学習率	0-3 エポック: 0.01, 4-8 エポック: 0.001, それ以降: 0.0001

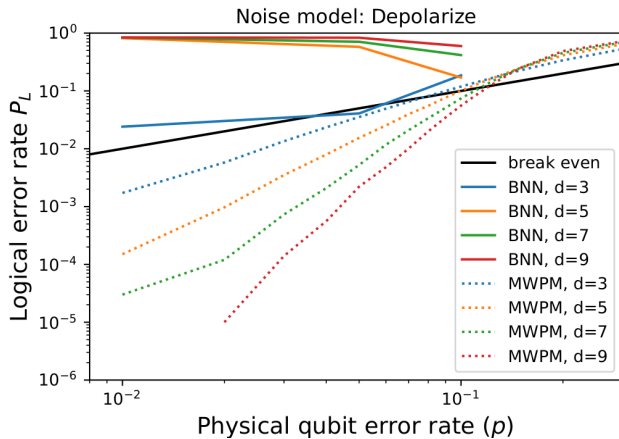


図 6 モデル 0 における、符号距離 3, 5, 7, 9 の物理エラー率と論理エラー率の関係。

いてプロットしても、プロットが 1 点で交わらない、すなわちしきい値が確認できない可能性がある。その場合は、各符号距離、物理エラー率において、物理エラー率 p に対して論理エラー率 p_L がどの程度小さくなったかでデコーダの性能を評価する。

4.3 評価結果

4.3.1 符号距離ごとの物理エラー率と論理エラー率の関係

モデル 0 及びモデル 1 の学習方法 2 における符号距離ごとの物理エラー率と論理エラー率の関係について、図 6, 図 7 に示す。比較対象として最小重みマッチングによるデコーの性能も点線で示している。図より、モデル 0 ではしきい値が存在しないが、モデル 1 では存在することが見て取れる。モデル 1 のしきい値はおおよそ 0.09 であり、モデル 2~4 の場合もしきい値は 0.09 付近である。なお、符号距離 11 の場合について、物理エラー率が低い場合に性能の悪化が見られるが、訓練データ数を増やすと性能が改善することを確認しているため、学習データ不足が要因と思われる。

4.3.2 CNN の計算量と論理エラー率の関係

学習方法 1 の物理エラー率 0.04 の場合における、モデル 1~4 の計算量と論理エラー率の関係をプロットした結果は図 8 の通りになった。このプロットではモデル 3, 4, 1, 2 の順に各点を結んでいる。

4.4 考察

本研究で用いた BNN モデルは、まず全結合 NN モデルと CNN モデルに大別される。このうち、全結合 NN モデルに関しては、符号距離 3 の物理エラー率 0.05 のとき以外

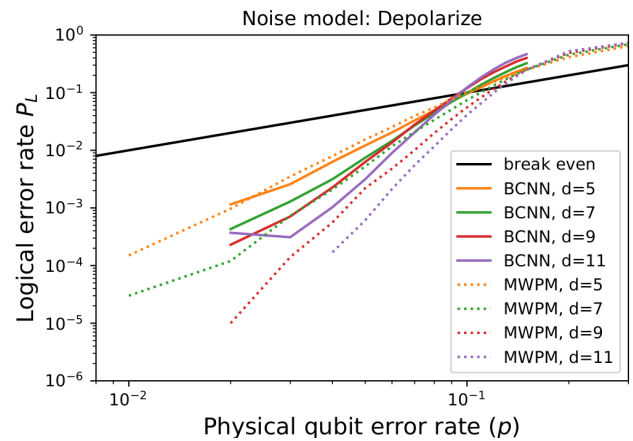


図 7 モデル 1, 学習方法 2 における、符号距離 5, 7, 9, 11 の物理エラー率と論理エラー率の関係。

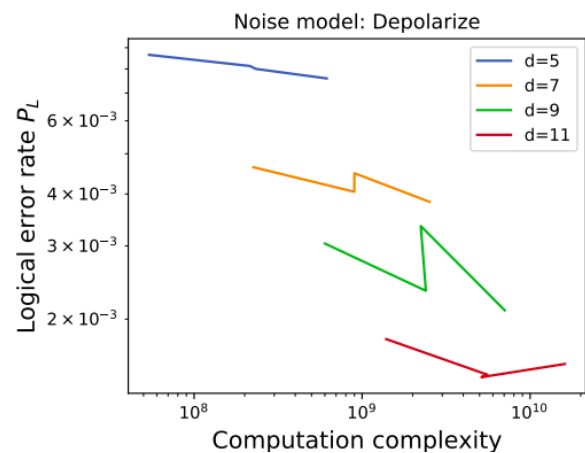


図 8 学習方法 1, 物理エラー率 0.04 の場合における、モデル 1~4 の演算回数と論理エラー率の関係

論理エラー率が物理エラー率を上回っている。また符号距離 5 以上ではいずれも物理エラー率 0.05 において 0.4 を上回り、物理エラー率 0.01 では約 0.9 になっている。これらのことから、このモデルは明らかに量子誤り訂正に用いることができない。

一方で、CNN モデルに関しては、いずれのモデルでも物理エラー率の低下、符号距離の増大に応じて論理エラー率が低下する傾向が確認でき、学習方法 2 の結果からしきい値が見て取れた。従って、量子誤り訂正に用いることが可能な性能であることが確認できる。学習方法 2 に関して、物理エラー率が低いとき、特に符号距離が大きい場合に性能の悪化が見られたが、モデル 1 の符号距離 11 の場合について訓練データ数を増やしたところ性能が改善したため、低物理エラー率のとき、物理エラー率 0.06 のエラーパターンから学習するには訓練データ数が不足していることが原因であり、しきい値の信頼性を損なうものではない。

CNN モデルの誤り訂正性能は、最小重み完全マッチングに本研究では劣っていたものの、誤り訂正の高速化・低消費電力化に寄与できるため、これが有効であると確認できたことは有意義であると言える。

CNN モデル同士の比較は図 8 からすることができる。おおまかな傾向としては演算回数の増加に応じて論理エラー率が低下することが見て取れるが、符号距離 7, 9 の場合合同程度の演算回数でもモデル 1 はモデル 4 に大きく性能で劣っており、モデルの形状によって性能が変化するさまが見て取れる。ただし符号距離 11 ではモデル 1 とモデル 4 の性能は同程度であり一概にモデル 4 が優れているとは言えない。

また、しきい値で比較すると、モデルによる大きな差異は認められない。

5. 結論

本稿では、量子誤り訂正符号 Surface Code の復号を高速・低消費電力で実行できる可能性のある復号手法として二値化ニューラルネットワークに着目した。二値化ニューラルネットワークの誤り訂正性能を符号距離と物理エラー率、論理エラー率の関係から検証した結果、二値化畳み込みニューラルネットワークがしきい値以下の物理エラー率で符号距離に応じて論理エラー率が下がる挙動が確認された。よって、二値化畳み込みニューラルネットワークは量子誤り訂正手法として有効性があると考えられる。今後の課題として、二値化畳み込みニューラルネットワークを SFQ 回路で実装した場合の速度と消費電力を評価することなどがあげられる。

謝辞

本研究の一部は、JST 未来社会創造事業 課題番号 JP-MJMI18E1, および JST CREST 課題番号 JPMJCR18K1 によるものである。

参考文献

- [1] Chao-Yang Lu, Daniel E. Browne, Tao Yang, and Jian-Wei Pan. Demonstration of shor's quantum factoring algorithm using photonic qubits. *Phys. Rev. Lett.* 99, 250504, 2007.
- [2] A. Imamoglu, D. D. Awschalom, G. Burkard, D. P. DiVincenzo, D. Loss, M. Sherwin, and A. Small. Quantum information processing using quantum dot spins and cavity qed. *Phys. Rev. Lett.* 83, 4204, 1999.
- [3] Y. Nakamura, Yu. A. Pashkin, and J. S. Tsai. Coherent control of macroscopic quantum states in a single-cooper-pair box. *Nature*, 398.
- [4] B. E. Kane. A silicon-based nuclear spin quantum computer. *Nature*, 393.
- [5] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor.
- [6] J. M. Hornibrook, J. I. Colless, I. D. Conway Lamb, S. J. Pauka, H. Lu, A. C. Gossard, J. D. Watson, G. C. Gardner, S. Fallahi, M. J. Manfra, and D. J. Reilly. Cryogenic control architecture for large-scale quantum computing. *Phys. Rev. Applied* 3, 024010, 2015.
- [7] E. Charbon, F. Sebastiano, M. Babaie, A. Vladimirescu, M. Shahmohammadi, R. B. Staszewski, H. A. R. Homulle, B. Patra, J. P. G. van Dijk, R. M. Incandela, L. Song, and B. Valizadehpasha. 15.5 cryo-cmos circuits and systems for scalable quantum computing. In *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, pages 264–265, 2017.
- [8] U. Kleine, J. Bieger, and H. Seifert. A low noise cmos preamplifier operating at 4.2 kelvin. In *ESSCIRC '93: Nineteenth European Solid-State Circuits Conference*, volume 1, pages 134–137, 1993.
- [9] Amarsanaa Davaasuren, Yasunari Suzuki, Keisuke Fujii, and Masato Koashi. General framework for constructing fast and near-optimal machine-learning-based decoder of the topological stabilizer codes. *Phys. Rev. Research* 2, 033399, 2020.
- [10] S.S.Tannu, Z.A.Myers, P.J.Nair, D.M.Carmean, and M.K.Qureshi. Taming the instruction bandwidth of quantum computers via hardware-managed error correction. *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture*, 2017.
- [11] Y.Tomita and K.M.Svore. Low-distance surface codes under realistic quantum noise. *Physical Review A*, 2014.
- [12] D. Gottesman. Stabilizer codes and quantum error correction. *arXiv:quant-ph/9705052*, 1997.
- [13] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A*, 86:032324, Sep 2012.
- [14] Jack Edmonds. Maximum matching and a polyhedron with 0,1-vertices. *JOURNAL OF RESEARCH of the National Bureau of Standards-B. Mathematics and Mathematical Physics*, 1965.
- [15] Adam Holmes, Mohammad Reza Jokar, Ghasem Pasandi, Yongshan Ding, Massoud Pedram, and Frederic T. Chong. *NISQ+: Boosting Quantum Computing Power by Approximating Quantum Error Correction*, page 556–569. IEEE Press, 2020.
- [16] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *J. Mach. Learn. Res.*, 18(1):6869–6898, January 2017.
- [17] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. In *NIPS*, 2016.
- [18] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. *arXiv:1603.05279 [cs.CV]*, 2016.