

# バーストエラーに耐性のある表面符号の通信方式の設計

一宮 有希<sup>1,2,a)</sup> 鈴木 泰成<sup>2,3</sup> 徳永 裕己<sup>2</sup>

**概要:** 量子計算は現在急速に研究開発が進んでいるが、量子ビットのエラー確率の高さと拡大の難しさが実用化への課題となっている。表面符号を用いた量子誤り訂正で量子計算の誤り率を小さくし、分散量子計算で計算機を拡大する方法は現在最も注目されているアプローチの1つである。分散量子計算で表面符号で符号化された論理量子ビットをノード間で通信する際には、送付する物理量子ビットに連続したエラーが生じるバーストエラーが発生してしまう恐れがある。この時、表面符号を構成する物理量子ビットを送る順番を無計画に決めてしまうと、訂正に失敗する確率が大きくなってしまふ。一方で、誤り訂正の失敗確率を最小化する順番を全探索で求めようとするのは計算論的に困難である。この論文では、2-opt法を用いて量子ビットを送る順番を決めることで、バーストエラー発生時に誤り訂正が失敗する確率を抑える方法を提案した。提案した手法のバーストエラーに対する耐性を数値的に評価し、閾値付近で論理誤り率を約0.5倍まで小さくするほか、小さなバーストエラーの区間に対しても高い論理誤り率削減効果があることを示した。

## Design of a burst-error tolerant communication scheme with surface code

**Abstract:** The research and development of quantum computing have been intensively conducted, but high error rates and limited scalability of qubits pose a significant challenge to its practical implementation. Surface codes emerged as one of the most widely recognized quantum error-correcting codes that enable quantum computation with low error probabilities. To scale quantum computing up to a practical size, we need a technique of distributed quantum computation, where logical qubits encoded with surface codes are transmitted between nodes. During the communication, burst errors, i.e., successive errors on the transmitted physical qubits, would like to occur as in classical communication. When the transmission order of the physical qubits comprising the surface code is determined haphazardly, the failure probability of error correction increases. On the other hand, the computational cost of finding the optimal transmission order that minimizes the failure probability would increase exponentially and becomes not practical. Thus, we need a method to determine the near-optimal ordering of physical qubits for the efficient transmission of logical qubits in a scalable manner. In this paper, we propose a novel method to reduce the failure probability of logical-qubit transmission under burst errors by optimizing the order of qubit transmission using the 2-opt method. We numerically evaluated the performance of the proposed method and showed that the proposed method halves the logical error rates around the threshold value. The proposed method also significantly reduces the logical error rates even when the region of burst errors is small.

### 1. 概要

従来の古典計算機よりも計算量理論的に高速に計算を行える期待がある計算機として、物質の量子的な性質を用いた量子計算機 [1] が挙げられ、研究開発が現在急速に加速している。例えば素因数分解アルゴリズム [2] や Grover のア

ルゴリズム [3] や、原子・分子のエネルギー差を計算が可能で、量子化学計算に有用な量子位相差推定 [4] が挙げられる。量子計算機は古典計算機よりもエラーが発生しやすいため、正しく解が求められる保証がある計算を行うことが出来ない。実際、現在の量子コンピュータは NISQ (Noisy Intermediate-Scale Quantum device) と言われており、エラー確率は現在 1%とされている [5]。これを仮定すると、例えば、100 量子ビットに対して 100 回演算を行うとエラーが起きずに正しく演算できる確率は  $0.99^{10000} \approx 2.25 \times 10^{-44}$

<sup>1</sup> 大阪大学大学院 基礎工学研究科

<sup>2</sup> NTT コンピュータ&データサイエンス研究所

<sup>3</sup> JST さきがけ

a) u141486f@ecs.osaka-u.ac.jp

となって、ほとんど演算に成功しない。そのため、量子計算機の誤り率を小さくするのは、量子計算機を実用的な規模の問題に適用するうえで必要不可欠となる。量子計算機のエラー確率を削減する最も有望な手法の一つが量子誤り訂正符号で、現在は誤りを訂正しながら計算を行う量子コンピュータ、FTQC(Fault-Tolerant Quantum Computer)[1]の実現に向けて研究開発がなされている [1], [6]。

量子誤り訂正符号を用いてエラーを訂正するには、エラーが作用した瞬間と次にエラーが作用する間に、シンドローム測定によってエラー検出を行い、得られた測定値からエラーを推定する必要がある。特に、表面符号 [7], [8] は量子ビットを二次元的に配列し、3つや4つの量子ビットに対する測定をすることで容易に実装でき、また訂正の性能が高く、符号化された量子状態に直接演算を行うことができるので、最も期待されている量子誤り訂正符号の1つとなっている [9], [10]。

量子誤り訂正は誤り率を任意の値まで小さくできるが、そのためには多くの量子ビットが必要となる。超伝導回路、イオントラップ、光量子回路など代表的な量子デバイスにおいて一つのチップに集積化可能な量子ビット数には技術的な上限があると予想されるため、大規模な量子計算を行うには小規模なノードでネットワークを構築する分散量子計算が必要となる [11]。分散量子計算とは、古典計算機における分散処理の計算を量子計算に置き換えたものである。つまり、複数の小・中規模の量子計算機で分散して計算を行い、ネットワークでノード間を結合した分散処理システムにおいて、割り当てられたタスクを実行する仕組みである。ノード間の量子通信を行う場合、誤りに耐性がある形でノード間で論理量子ビットの情報をやり取りする必要がある。この通信においては計算と同様にエラーが生じるが、このエラーも量子誤り訂正や、量子誤り検出とポストセレクションを組み合わせて訂正できる。

ノード間で量子通信を行う場合、送信される量子ビットには理論的に解析しやすさから独立な脱分極ノイズを仮定するのが一般的である。しかし、光子やマイクロ波を使って通信をすると、通信する途中で光ファイバの特性が揺らぐなどの理由から実際には連続した量子ビットに時間的な相関を持ったバーストエラーが生じると予想される。従って、高い効率でノード間で誤り耐性のある量子通信を行うには、バーストエラーの影響が緩和されるように量子通信を行う際の物理量子ビットを送るプロトコルを最適化すべきである。しかし、こうしたバーストエラーに耐性のある量子ビットの送付方法はこれまで検討されていなかった。

本研究では、バーストエラーの影響を軽減する one-way の論理量子ビット送付方法を提案する。本プロトコルでバーストエラーの影響を軽減するうえで着目したのは、チャンネルを通して物理量子ビットを送付する順序の並び替えである。長いバーストエラーが生じたとしても出来るだけ

高い確率で誤り訂正に成功するようにプロトコルを設計するには、論理量子ビットにエラーが生じる物理量子ビットの並びが送付する並びの中に出来るだけ現れないようにすれば良い。これを実現する素朴な方法はシーケンスをランダムに並び替える方法だが、都度ランダムな並び替えを行うのは実験的なコストが大きい。そこで、本研究では巡回セールスマン問題などで用いられる 2-opt 法を用いてバーストエラーの影響を軽減する手法を提案した。提案手法ではまず、表面符号に連続した  $r$  量子ビットにバーストエラーが生じると仮定し、2-opt 法による並び方の候補を複数提示する。得られた候補で生じる論理エラー率を変更前と比較し、論理エラー率が小さくなる並べ方を選択する。これを逐次的に行うことで、近似的に最適な並べ方を効率的に計算する。本研究では実際に提案手法を用いて並び順の最適化を行い、得られた並びのバーストエラーに対する耐性を数値的に評価した結果、ランダムな並び替えに比べて閾値付近で論理エラー確率をおよそ 0.39 倍に低く抑えることができた。

## 2. 背景

### 2.1 スタビライザー符号

スタビライザー符号 (stabilizer code)[12] とは、量子誤り訂正符号の代表的なクラスの1つである。スタビライザー符号を導入することで、符号化された量子状態や誤り訂正のプロセスを簡潔に記述することができる。

スタビライザー符号を定義するには、誤り訂正するのに必要なスタビライザー演算子と論理演算子の定義を用いてエラー検出する枠組みを用意する必要がある。まず、 $2^n$  次元 Hilbert 空間である  $\mathbb{C}^{2^n}$  に作用する演算子に関して、スタビライザー群を定義して、誤り訂正のプロセスを導入する。

**Definition 2.1.**  $\mathcal{G}_n$  を

- Pauli 群 (Pauli group)

$$\mathcal{P}_n := \{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n} \quad (2.1)$$

の部分集合

- 独立な集合
- $\mathcal{G}_n$  の元は互いに可換
- $\mathcal{G}_n$  が生成する群  $G = \langle \mathcal{G}_n \rangle$  に  $-I$  を含まない

を満たすとする。このとき  $\mathcal{G}_n$  の元をスタビライザー生成元 (stabilizer generator),  $\langle \mathcal{G}_n \rangle$  をスタビライザー群 (stabilizer group), その元をスタビライザー演算子 (stabilizer operator) という。

以上で定義したスタビライザー生成元を用いると、符号空間が定義でき、ある量子状態に対して符号化を施すこと

ができる。

**Definition 2.2.** 符号空間 (code space) とは

$$\mathcal{C} := \{|\psi\rangle \in \mathbb{C}^{2^n} \mid \forall g \in \mathcal{G}_n, g|\psi\rangle = |\psi\rangle\} \quad (2.2)$$

を満たす集合のことをいう。また、 $\mathcal{C}$  の元をスタビライザー状態 (stabilizer state) という。

次に上で定義した符号空間を用いて、どのようにエラーを検出すればよいかを述べる。スタビライザー状態にエラー  $E \in \mathcal{P}_n$  が作用したとする。  $E$  とスタビライザー演算子  $g \in \mathcal{G}_n$  は Pauli 演算子で記述されるので、可換または反可換の関係を持つ。可換の場合

$$E|\psi\rangle = Eg|\psi\rangle = gE|\psi\rangle \quad (2.3)$$

より  $g$  の固有値は  $+1$ 、反可換の場合

$$E|\psi\rangle = Eg|\psi\rangle = -gE|\psi\rangle \quad (2.4)$$

より  $g$  の固有値は  $-1$  となる。この固有値をシンδροーム (syndrome) という。スタビライザー状態の定義よりシンδροームが  $-1$  の場合、エラーを検出することができる。

以上がスタビライザー符号の定義となる。しかし、このエラーを検出する枠組みでも、検出できないエラーが存在し、訂正に失敗してしまう場合がある。以下では、訂正できないエラーは何かを分析するため、論理演算子を導入する。そして、エラーがスタビライザー状態に作用したとき、そのエラーは検出できるかどうか、スタビライザー状態を変えるかどうかによって、そのエラーを分類する。これを知るため、次の定理 [13] を用いる。

**Theorem 2.3.** 商群

$$Q := \mathcal{N}(\mathcal{G}_n) / (\{\pm 1, \pm i\} \times \langle \mathcal{G}_n \rangle) \quad (2.5)$$

を考える。  $\mathcal{N}(\mathcal{G}_n)$  は  $\mathcal{G}_n$  の正規化群で

$$\mathcal{N}(\mathcal{G}_n) := \{h \in \mathcal{P}_n \mid h\mathcal{G}_n = \mathcal{G}_nh\} \quad (2.6)$$

で定義される。商群  $Q$  の生成元集合を考え、生成元集合の各剰余類の代表元の集合を  $\mathcal{L}_n$  とする。このとき、  $L \in \langle \mathcal{L}_n \rangle$  を論理演算子 (logical operator) という。また、全ての  $g \in \mathcal{G}_n$  に対して、固有状態  $E|\psi\rangle$  に対応する  $|\mathcal{G}_n|$  個のシンδροームを  $c \in \{1, -1\}^{|\mathcal{G}_n|}$  とする。このとき

$$T(c)(E|\psi\rangle) \in \mathcal{C} \quad (2.7)$$

を満たす演算子、つまり  $E|\psi\rangle$  を符号空間に戻す演算子を1つ選んで  $T(c)$  とすると、任意のエラー  $E \in \mathcal{P}_n$  に対し、位相因子を無視して

$$E = LGT(c) \quad (2.8)$$

を満たす  $L \in \langle \mathcal{L}_n \rangle, G \in \langle \mathcal{G}_n \rangle$  が一意に存在する。

式 (2.8) より、  $L = I$  と  $L \neq I$  に場合分けして調べることでエラーは表 1 のように分類できる。

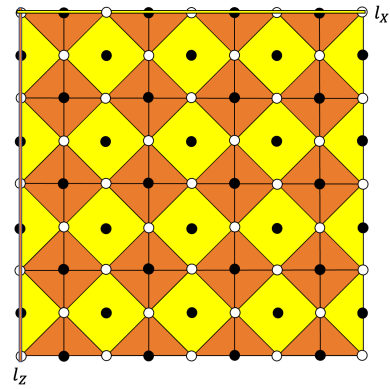


図 1 一辺が  $N = 8$  の二次元格子で定義した表面符号

## 2.2 表面符号

表面符号 [7], [8], [9] はスタビライザー符号に分類される量子誤り訂正符号である。表面符号による誤り訂正は性能が高く、超伝導量子ビットや光子などで実装しやすいため、実現に近い量子誤り訂正符号として注目を集めている。

表面符号の定義を説明する。まず、  $N \times N$  の二次元格子を構成する頂点集合  $V$  と辺集合  $E$  を用いて、グラフ  $G = (V, E)$  を考える。そして  $F$  を  $V$  と  $E$  から構成される面の集合とし、図 1 のように各辺  $e \in E$  に量子ビットを配置する。このとき、頂点  $v \in V$  と面  $f \in F$  を用いて

$$A_v := \prod_{e \in \delta v} Z_e, B_f := \prod_{e \in \partial f} X_e \quad (2.9)$$

とする。ここで  $\delta v$  は頂点  $v$  に接続する辺の集合、  $\partial f$  は面  $f$  を構成する辺の集合を表し、  $e$  を添字とする Pauli 演算子は辺  $e$  に位置する量子ビットに作用する Pauli 演算子である。図 1 で、  $A_v$  を茶色の図形の頂点にある量子ビットに作用する演算子、  $B_f$  を黄色の図形の頂点にある量子ビットに作用する演算子として図示した。  $A_v$  を頂点演算子 (site operator)、  $B_p$  を面演算子 (face operator) とすると、  $A_v$  と  $B_p$  をスタビライザー生成元として選んで

$$\mathcal{G}_n := \{A_s \mid \forall s \in V\} \cup \{B_p \mid \forall p \in F\} \quad (2.10)$$

のようにスタビライザー符号を定義できる。また、図 1 に示した二次元格子の境界  $l_X, l_Z$  を用いて、

$$L_X := \prod_{e \in l_X} X_e, L_Z := \prod_{e \in l_Z} Z_e \quad (2.11)$$

とし、  $\langle L_X, L_Z \rangle$  の元を論理演算子とする。ここで  $e \in l_X, e \in l_Z$  とは境界  $l_X, l_Z$  に存在する辺  $e$  を表す。

## 2.3 最小重み完全マッチング

2.2 節ではスタビライザー符号の特別な例である表面符号を紹介した。誤り耐性量子計算では、エラーを検出するだけではなく、エラーを訂正する必要がある。エラーが作用した後に、シンδροームからエラーを推定し、推定結果を論理量子ビットに作用させることを訂正という。訂正手

エラー	エラー検出	スタビライザー状態
$E \in \{\pm 1, \pm i\} \times \langle \mathcal{G}_n \rangle$	不可	不変
$E \in \mathcal{N}(\mathcal{G}_n) \setminus (\{\pm 1, \pm i\} \times \langle \mathcal{G}_n \rangle)$	不可	変化
otherwise	可能	変化

表 1 エラーの分類

法はどのようにエラーシンドロームからエラーを推定するかには依存するので、複数の方法が考えられる。

まず、具体的なアルゴリズムに入る前に、理論的に訂正できるエラーは何かを紹介するため、符号距離 (code distance) を導入する。

**Definition 2.4.**  $\forall i, \sigma_i \in \{I, X, Y, Z\}$  に対し

$$\text{wt}(\sigma_1 \otimes \sigma_2 \otimes \cdots \otimes \sigma_n) := \sum_{i=1}^n \chi_{\{X, Y, Z\}}(\sigma_i) \quad (2.12)$$

$$\chi_{\{X, Y, Z\}}(\sigma) := \begin{cases} 1 & \text{if } \sigma \in \{X, Y, Z\} \\ 0 & \text{if } \sigma \notin \{X, Y, Z\} \end{cases} \quad (2.13)$$

とするとき、

$$d := \min_{L \in \mathcal{L}_n, G \in \langle \mathcal{G}_n \rangle} \text{wt}(LG) \quad (2.14)$$

を符号距離という。

この  $d$  は訂正に失敗するエラーのうち、含まれる  $X, Y, Z$  の最小の個数を表す。つまり  $\lfloor (d-1)/2 \rfloor$  量子ビットに対する任意のエラーは尤もらしいエラーを推定すれば常に訂正できる。

いくつかある訂正手法に関して、性能を評価する指標として閾値 (threshold) がよく用いられる。閾値  $p_{\text{th}}$  とは、物理エラー確率  $p$  と符号距離  $d$  に対して、 $p < p_{\text{th}}$  より小さければ  $d$  を大きくすることでエラー訂正に失敗する確率である論理エラー確率  $p_{\text{err}}(p, d)$  を任意の小さい値にできるような  $p_{\text{th}}$  の値である。一方で、現実的なノイズモデルではこのような  $p_{\text{th}}$  を求めることは解析的にも数値的にも難しいことから、符号距離が小さい  $d = 3, 5$  などの領域で  $p_{\text{err}}(d, p) < p_{\text{err}}(d+2, p)$  となるような最大の  $p$  を疑似的な閾値と呼ぶ。

ここでは、一様なノイズモデルに対して閾値が 10.3% [14] と最適値ではないが、時間計算量が物理量子ビット数  $n$  の多項式オーダー  $\mathcal{O}(n^3)$  で高速に訂正できる最小重み完全マッチング (Minimum weight perfect matching) で訂正する手法を紹介する。まず、最小重み完全マッチング問題について説明する。一般に最小重み完全マッチング問題とは、グラフ  $G = (V, E, w)$  に対して ( $V$  は頂点集合、 $E$  は辺集合、 $w$  は重み関数)、各辺  $e \in E$  の重み  $w(e)$  が与えられたとき、コスト  $\sum_{e \in M} w(e)$  が最小となる完全マッチング  $M$  を見つける問題である。ここで完全マッチング  $M$  とは、部分集合  $M \subset E$  であって、 $M$  のどの 2 辺も頂点を共有せず、任意の頂点  $v \in V$  は接続する  $M$  の辺  $m \in M$  が存在することをいう。最小重み完全マッチング問題を解

くアルゴリズムとして、Edmond の花アルゴリズム [15] が知られており、 $|E|, |V|$  の多項式時間で求解できる。

この最小重み完全マッチング問題を用いて、表面符号を用いた誤り訂正の手法は以下の通りである。

- (1) 式 (2.10) で定義したスタビライザー生成元で測定する。これをシンドローム測定 (syndrome measurement) という。そして以下の手順でグラフ  $G = (V, E, w)$  と  $G' = (V', E', w')$  を構成する。
  - (a) 測定して  $-1$  を得た  $A_v(B_p)$  の測定用量子ビットを頂点と  $x-y$  座標で表し、頂点集合  $V(V')$  に全て入れる。
  - (b)  $|V|(|V'|)$  が奇数の場合、頂点集合に新たに頂点  $v$  を追加する。
  - (c) 頂点集合  $V(V')$  から完全グラフを構成し、辺の重みを 2 頂点間のマンハッタン距離、境界を通る迂回経路のうち小さい法とする。ただし、二次元格子の境界を沿う経路の重みは 0 とし、 $v$  に接続する辺  $(v, u)$  の重みは  $u$  の  $y$  座標 ( $x$  座標) とする。
- (2) グラフ  $G = (V, E, w)$  と  $G' = (V', E', w')$  で最小重み完全マッチング問題を解く。
- (3) 完全マッチングの辺  $e \in M$  に存在する量子ビットに  $Z(X)$  を作用させる。

この結果、スタビライザー状態にスタビライザー演算子が作用している結果ならば訂正に成功し、そうでないなら訂正に失敗する。例えば図 2 では、赤丸で示したデータ量子ビットに  $Z$  エラーが生じ、青丸を中心とするスタビライザー生成元で測定するとシンドローム  $-1$  を得る。ここで、格子の上下の両端に  $Z$  測定を行ったが、 $|V|$  が奇数の場合の頂点の追加に対応する。そして図 3 のように最小重み完全マッチング問題を解く。この場合、図 4 のように訂正に成功する。

## 2.4 誤り耐性のあるノード間量子通信

大規模な誤り耐性量子計算を実施するには、大きな量子回路を複数のノードに分散して実行する分散量子計算が必要となる。分散量子計算を行うには、ノード間で誤りに耐性がある形で量子通信を行う仕組みが必要となる。ノード間で表面符号で符号化された論理量子ビットの送受信を行う方法は大きく二つに分けられる。以降では送信者を Alice、受信者を Bob として、それぞれが全体の計算を行

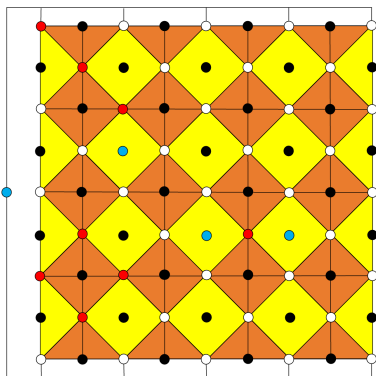


図 2 符号距離 5 の表面符号にエラーを作用してシンドローム測定

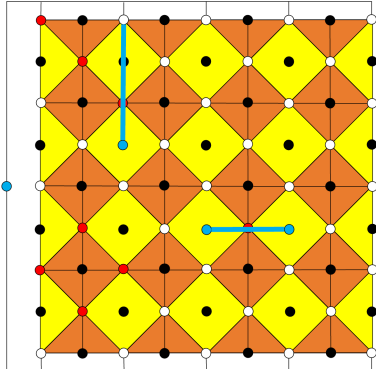


図 3 最小重み完全マッチング問題の求解

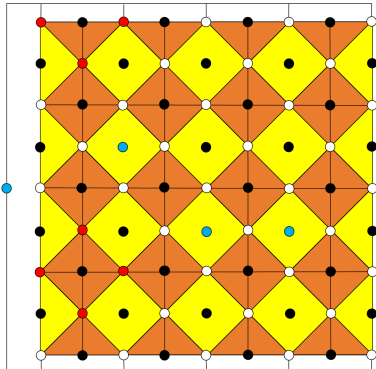


図 4 誤り訂正後の表面符号

うには十分ではないが量子中継や通信を行うには十分な数の量子ビットを持っている状況を想定し、論理量子ビットを送付するプロトコルの類型を俯瞰する。

最も素朴なアプローチは量子通信や量子中継で用いられているエンタングルメント蒸留 (entanglement distillation) を用いる方法である。この方法ではまず、Alice と Bob の間で忠実度の低いエンタングルメント (entanglement) を複数共有した後、これを用いてエンタングルメント蒸留を実施する。そして、得られた忠実度の高いエンタングルメントを表面符号に埋め込むことで表面符号で符号化されたエンタングルメントを共有できる [16]。ひとたび表面符号で符号化されたエンタングルメントが共有されれば、これを消費することで量子テレポーテーションなどによる論理量子ビットを送付することができる。この方法は素朴だが蒸留の過程で誤りに耐性のないメモリに一時的に量子ビッ

トが保管されることから、実用的な論理エラーレートを得るには表面符号に埋め込んだ後にも蒸留を行わねばならないなどの問題がある。

最終的に用いる符号が表面符号だと定まっている場合は、これを前提としてより効率的に論理量子ビットの通信を行うことも可能である。例えば文献 [17], [18], [19], [20] では表面符号などのスタビライザー符号による符号化を直接的に用いて論理量子ビットの共有を行う方法を提案している。通信ノードが十分近くに配置されており、通信中に生じるエラーが小さい場合、最も素朴なアプローチの例は表面符号で符号化されたエンタングル状態にある論理量子ビットのペアを Alice が作成し、一方を Bob に送付する方法である。通信中に生じるエラーは、その確率が十分小さければ受信した Bob が表面符号の誤り訂正を行うことでエラーを訂正することができる。このプロトコルはこの方法は Alice から Bob への一方的な情報送信のみで行える one-way な手法であるから、Alice と Bob の距離に依存せずエンタングル状態の論理量子ビットの生成や通信の帯域に応じて高速な繰り返しが可能であり、また Bob が受信した後に受け取ったエンタングルメントの片割れを表面符号に埋め込むなどの手続きも不要である。この場合のプロトコルは以下ようになる。

**Protocol 2.5.** (論理量子ビットの通信プロトコル)

- (1) 表面符号で符号化された論理量子ビットのエンタングルメントを Alice が作成する。
- (2) Alice はエンタングルメントの片方の表面符号を構成する  $n$  個の data qubit を Bob に送信する。
- (3) 送信中に各量子ビットに対して独立にエラー発生する。
- (4) Bob は受け取った量子ビットにより表面符号を再構成し、エラー検出に基づき通信中のエラーを訂正する。エラーの推定には最小重み完全マッチングを用いる。

ここで、通信中に生じるエラーとしては以下のような各量子ビットに独立に生じる脱分極チャネル (depolarizing channel) を考えるのが一般的である。

$$\mathcal{E}_p(\rho) := (1 - p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z) \quad (2.15)$$

仮に上記のようなチャネルになっていなかったとしても、ノイズが独立であるならランダムな 1 量子ビットクリフォード操作とその逆操作を送受信の前後でそれぞれ作用させる Twirling のテクニックを用いて、ノイズを上記のような脱分極チャネルに変換することができる。

上記の手続きは、Alice が表面符号のデータ量子ビットの数と同じだけの物理的なエンタングルメントを Bob と共有し、Alice と Bob が同じスタビライザー測定をそれぞれで施し、Alice が Bob にスタビライザー測定の結果を送付す

ることで同様のプロトコルとなる [17]。本論文での議論は通信が one-way であれば簡単な変形でこうしたプロトコルにも適用可能である。なお、Alice と Bob の間の通信が必ずしも高い確率で成功せず表面符号の閾値を超えることができない場合は、最初のエンタングルメントの生成を成功するまでポストセレクトするなどのアプローチが必要がある。この場合はプロトコルが成功したかの通信を Alice と Bob の双方向で行う two-way のプロトコルとなり、また実際にどの量子ビットが利用されるのかも送信地点では決まらない。本論文ではまずは one-way のプロトコルのケースに焦点を当てて解析する。本論文の議論は工夫により two-way のプロトコルへも拡張できると期待されるが、具体的な定式化と拡張は今後の課題とする。

### 3. バーストエラーに耐性のある誤り耐性ノード間通信

#### 3.1 バーストエラーの影響

十分に遠く離れた二者間でエンタングルメントを生成するには、光やマイクロ波の光子を用いることが一般的である。この時、実験的な制約からノード間に多くのチャンネルを用意することは難しく、一般には一つないし少数の物理的なチャンネルを用いて周波数や時間領域で多重化された通信を行うことになる。こうした通信においては通信路の一時的な揺らぎや周波数の特性の変化により、誤りの確率は時間的な相関を持つと予想される。このように送信に対して時間的な相関をもって現れるエラーをバーストエラーと呼ぶ。

バーストエラーが発生したとき、そのエラーの影響度合いはデータ量子ビットをどのような順序で送信したかに強く依存する。従って、バーストエラー存在下での論理量子ビット通信のプロトコルは以下のように修正される。

**Protocol 3.1.** (バーストエラー存在下での論理量子ビットの通信プロトコル)

- (1) 表面符号に存在する  $n$  個のデータ量子ビットに番号  $0, \dots, n-1$  を割り当てる。
- (2) 送信元は表面符号に存在するデータ量子ビットを送信する順番  $p = (p_0, p_1, \dots, p_{n-1})$  を生成する。
- (3) 送信元は  $p_0$  番の量子ビット,  $p_1$  番の量子ビット,  $\dots$ ,  $p_{n-1}$  番の量子ビットの順に送信する。
- (4) 送信中に連続した  $r$  量子ビットにエラー (バーストエラー) が発生する。
- (5) 量子ビットを受信した受信元は  $p$  に従って表面符号を再構成する。(送信元と受信元はあらかじめ  $p$  を共有しているとする)
- (6) 受信元は最小重み完全マッチングで表面符号を訂正

し、エラーを訂正する。

本論文では Protocol 3.1 の (4) で生じるエラーは以下のようにモデリング出来るとする。通信するごとにランダムに決まる定数  $0 \leq a \leq n-r-1$  に対し、 $n$  量子ビットに対するチャンネルを

$$\mathcal{E}'(\rho) := \rho^{\otimes a-1} \otimes \mathcal{E}_{0.25}(\rho)^{\otimes r} \otimes \rho^{\otimes n-a-r+1} \quad (3.16)$$

とする。つまりバーストエラーの開始地点  $a$  はランダムに決まり、 $a$  番目の量子ビットから数えて連続した  $r$  量子ビットに完全混合状態となるようなノイズを与えるとする。

従来の論理量子ビットの送信手法ではこうした時間的なエラー相関は考慮されていないものの、以下の例から、バーストエラーの存在下では論理量子ビットに含まれる物理量子ビットを送る順番はプロトコルの成功確率に強く影響することが分かる。例えば表面符号のデータ量子ビットを蛇腹状に順番に送付する場合を考える。すると符号距離  $d$  の表面符号で二次元格子の境界に沿って  $\lfloor (d+1)/2 \rfloor$  量子ビットの区間にバーストエラーが生じた場合、エラーが独立だとみなして尤もらしいエラーを推定してもエラー訂正に失敗してしまう。一方、 $\lfloor (d+1)/2 \rfloor$  個の量子ビットに続けてエラーが生じる場合でもその配置が表面符号の全体に散らばっていれば、エラーが生じている量子ビットの割合はデータ量子ビットの数に対して漸近的に 0 に近づくため、符号距離を十分大きくしておけば、最小重み完全マッチングを用いた訂正が閾値を持つことから高い確率で誤り訂正は可能であることが分かる。このことからわかるように、より高い効率でノード間での論理量子ビットの通信を行うには、上記のプロトコルにおけるデータ量子ビットの送信順序は最適化されるべきである。

最も素朴なバーストエラーの緩和方法は、量子ビットの並びを毎回ランダムに並び替えることである。この方法は実効的にバーストエラーをランダムな位置に対するエラーに変換するため、バーストエラーの影響を独立なエラー確率の揺らぎとして扱うことができる。一方、毎回ランダムに配置を決定する方法は量子ビットの配置を都度切り替えるためのスイッチが必要となり実験への負荷が高いたくなく、一定の確率でバーストエラーに耐性のない配置も選ばれてしまう。このため、より高い効率での誤り耐性のあるノード間通信のためには最適化された固定配置を選択すべきである。本研究の目的は上記のような形でモデリングされたバーストエラーの影響を低減する送信順序を決定することである。

### 4. バーストエラーの影響を最小化するプロトコルの提案

バーストエラーの影響を最小化するには、論理エラー確率を最小にする並べ方の最適解を求めればよい。しかし、

全部の並べ方のパターンは量子ビットの数  $n$  に対して階乗で増えていき、表面符号においては量子ビットの数  $n$  は  $\mathcal{O}(d^2)$  で増えていくため、この方法は  $d = 3, 5$  のような小さな符号でも現実的に解くことが難しい計算量となってしまふ。この問題をより効率的に解く方法として、最適な並べ方を求める問題を 0,1 整数計画問題に帰着するアプローチを考えることができる。  $n$  個の量子ビットの並べ方は、並びの  $i$  番目に  $j$  番目の量子ビットがいるときに  $x_{ij} = 1$  となるような  $\mathcal{O}(n^2)$  個の変数  $\{x_{ij}\}$  に適切な拘束条件を付けて表すことができる。バーストエラーの長さ  $r$  がそれほど長くない場合、論理エラーを起こしうる影響の大きい  $r$  個のエラーのパターンを列挙し、このパターンが  $\{x_{ij}\}$  が表す並び中に無いことを保証することでバーストエラーに耐性のある並びを考えることができると期待される。しかし、  $r$  が増えていくと論理エラーを起こしうる影響の大きいパターンは指数的に増大すること、また、長さ  $n$  のシーケンスの部分列として  $r$  個の集合と一致するものが無いかを調べるには  $r$  次の多項式が  $\mathcal{O}(r!(n-r))$  個必要となるため、やはり小さい符号距離でも非現実的なサイズの問題に増大してしまう。このことから、  $r$  が大きくなれば実用的な時間で解くことは難しくなってしまうことが分かる。そこで、本研究では近似的な手法より、最適に近い並び方を探索することにした。上記の整数計画問題による解法は適切な変形と高速なソルバにより小規模なサイズであれば現実的に達成できる可能性もあるが、この最適化は今後の課題とする。

本研究ではバーストエラーを最小化する近似手法として、巡回セールスマン問題などで順列の最適化に用いられる 2-opt 法を活用し、最適化するアルゴリズムを提案した（セールスマン巡回問題や 2-opt 法の詳細については Appendix. A を参照）。巡回セールスマン問題の解法である 2-opt をバーストエラーの影響を最小化する問題に適用する理由は、以下の解釈による問題の類似性によるものである。巡回セールスマン問題は 2 頂点を結ぶ辺に重みが与えられ、長さ  $n$  の経路中の  $n-1$  個の重みの合計を最小化する問題である。一方、バーストエラーの最小化問題は、  $n$  個の物理量子ビットを並べた経路に対し、連続した  $r$  量子ビットにエラーが作用したもとの論理エラー確率を最小化する問題である。つまり長さ  $n$  の経路中に含まれる  $r$  頂点を結ぶ経路に対して、バーストエラーによる論理エラー確率として重みが与えられると見なせ、長さ  $n$  の経路中の  $n-r+1$  個の重みを考慮して最小化することになる。この類似性から、巡回セールスマン問題のヒューリスティックな解法である 2-opt は、今回の問題の解探索にも有効であると期待できる。

今回提案するアルゴリズムの手続きは以下の通りである。

**Algorithm 4.1.** (2-opt 法)

- 入力：列  $p_0 = \{0, \dots, n-1\}$
  - 出力：列  $p = \{p'_0, \dots, p'_{n-1}\}$
- (1) 入力列の順番を擬似乱数を用いてランダムに量子ビットの順番を並べ替え、
 
$$p_0 = \{0, \dots, n-1\} \rightarrow p = \{p_0, \dots, p_{n-1}\} \quad (4.17)$$
 とする。
  - (2)  $0 \leq i < j \leq n-1$  に対して、次のように量子ビットの順番を並べ替える。
 
$$p = \{p_0, p_1, \dots, p_i, p_{i+1}, p_{i+2}, \dots, p_j, p_{j+1}, p_{j+2}, \dots, p_{n-1}\} \quad (4.18)$$

$$\rightarrow p' = \{p_0, p_1, \dots, p_i, p_j, p_{j-1}, \dots, p_{i+1}, p_{j+1}, p_{j+2}, \dots, p_{n-1}\} \quad (4.19)$$
  - (3) 並べ方  $p'$  での論理エラー確率  $p_e(p')$  を調べる。本研究では、最小重み完全マッチングを用いたモンテカルロシミュレーションにより、  $0 \leq r \leq \lceil 0.2n \rceil$  に対して式 (3.16) のノイズモデルでの論理エラー確率を全て求め、合計したものを  $p_e(p')$  とする。
    - (a)  $p_e(p) > p_e(p')$  ならば、並べ方  $p'$  を採用する。  $p' = p$  として (2) の最初に戻る。
    - (b) 任意の  $0 \leq i < j \leq n-1$  に対して、  $p_e(p) \leq p_e(p')$  なら (4) に進む。
  - (4) 並べ方  $p$  を出力し、アルゴリズムを停止する。
    - 時間計算量：  $\mathcal{O}(lf(n)n^2)$   
 ただし  $l$  は (3)(a)  $\rightarrow$  (2) に戻った回数、  $f(n)$  は (3) で論理エラー確率を調べる時間計算量である。

## 5. 性能評価

提案手法の性能を評価するため、数値計算による実験を行った。本研究では以下の 2 つの手法によって論理エラー確率を調べ性能を比較した。

- (1) ノイズモデル式 (3.16) のもとの、表面符号に存在する物理量子ビットの並べ方  $p$  をランダムに生成し、  $N$  個用意する。それぞれの並べ方に対して Protocol 3.1 を 1 回実行して、訂正に成功する回数  $c$  を求める。その後、論理エラー確率  $(N-c)/N$  を求める。
- (2) ノイズモデル式 (3.16) のもとの、 Algorithm 4.1 を実行して表面符号に存在する物理量子ビットの並べ方  $p$  を 1 回生成する。その並べ方に対して Protocol 3.1 を  $N$  回実行して、訂正に成功する回数  $c$  を求める。その後、論理エラー確率  $(N-c)/N$  を求める。

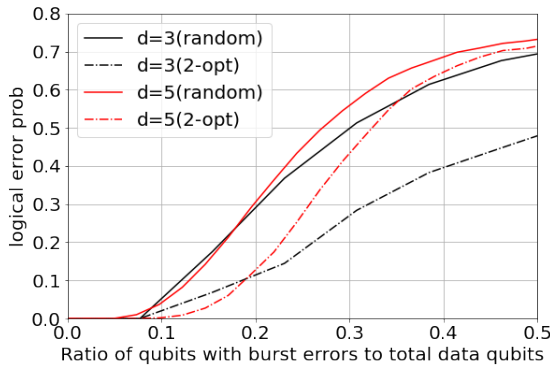


図 5 バーストエラー後の訂正

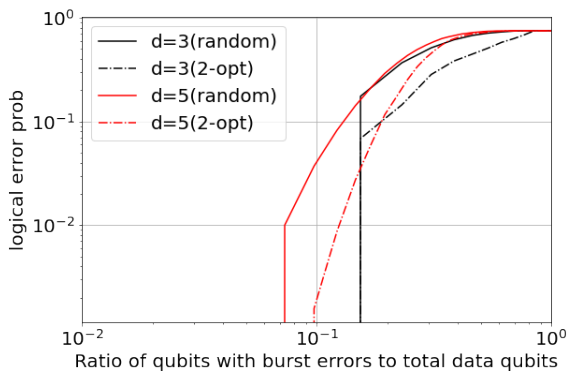


図 6 バーストエラー後の訂正 (両対数)

今回の計算では  $N = 10000$  とし、この 2 つの手法を符号距離  $d = 3, 5$  について行った。プログラミング言語 C++ を用いて行った。最小重み完全マッチングのソルバとして Kolmogorov による Blossom V を用いた [15]。

横軸に物理量子ビット数に対するバーストエラー長の比  $r/n$ 、縦軸に論理エラー確率をとったものを図 5, 6 に示した。(1) の手法は実線、(2) の手法は一点鎖線で示した。ランダムな配置を行う (1) の手法に比べて、今回提案した (2) の手法は小さな論理エラー確率を達成できることが分かる。両対数グラフのバーストエラーが小さい領域において論理エラー確率が 0 に飛んでいるのは、バーストエラーの区間が短くなると、ある地点でエラーが生じる個数が論理エラーが生じない領域に転じるためである。

グラフから本提案手法により固定した並び方を用いてバーストエラーを抑えられ、閾値付近である  $r/n = [0.2n]/n$  では (1) の手法に比べて (2) の手法は  $d = 3$  では 0.393 倍、 $d = 5$  では 0.480 倍に論理エラー確率を低くなったことが分かる。また、 $d = 5$  においては両対数グラフにおいて論理エラー率の減少幅が大きくなっており、大きな符号距離でバーストエラーの区間が小さい領域で比率として大きな改善が期待できる。

次に、提案手法である Algorithm 4.1 の実行時間を表 2 に示す。 $d = 3, 5$  に関しては今回の手法により実用的な時間で近似解を求めることができている。一方で、物理量子

ビットの増加によって実行時間は上昇しており、実用的な問題を解くのに必要な  $d = 21$  といった大きな符号距離では現実的な時間で最適化を行うことが難しいと予想される。この問題の解決策としては、数値計算を並列化したり、既に試した並びを保存してスキップすることで、最適化が現実的な時間で終わるように調整する方法などが考えられる。実用的な規模の符号距離まで適用可能にするためのこうした数値的な高速化は今後の課題とする。

## 6. まとめと展望

本研究では、分散量子計算などにおける表面符号で符号化された論理量子ビットの通信における、バーストエラーの影響を軽減する手法について検討を行った。少数のチャンネルで量子ビットを送付するとき、バーストエラーの影響を軽減するには物理量子ビットの並び方を最適化することがバーストエラーの影響を軽減するうえで有効である。本研究では物理量子ビットの並び方を 2-opt 法によって探索する手法を提案し、その性能を数値計算により評価した。図 5, 6 の結果から、通信する度に送信順序をランダムに並び方を変えるとより、提案手法は論理エラー確率を低く抑えられることが分かった。また、 $d = 3, 5$  などの符号距離において、現実的な時間で最適化を実施することが出来ている。

今後の課題として、より最適に近い順序のさらなる探索がある。今回提案した手法で用いた 2-opt 法は局所探索法なので、局所解に陥っている可能性があり、最適解から遠い解が得られている可能性がある。この問題を解決する方法の一つは、物理量子ビットの並び方を求める問題を整数計画問題などに変換し、最適解あるいはそれに近い解を得る方法である。しかし、3 章にも書いたように、素朴に考えると、符号距離  $d = 5, 7, \dots$  では、バーストエラーが生じるパターンが急激に増加して実用的な時間で解くのが難しい。問題の変換方法に検討を重ね効率的な問題帰着を行うことで、バーストエラーにより耐性のある通信プロトコルが提案できると期待される。

**謝辞** 本研究は JST さきがけ (助成番号: No. JPMJPR1916)、内閣府ムーンショット (助成番号: No. JPMJMS2061) の助成の元で行いました。

### A. 2-opt 法

2-opt 法 [21] はよく巡回セールスマン問題 [22] において近似解を求める手法としてよく用いられる。巡回セールスマン問題とは、 $n$  都市と各 2 都市間の移動コストが与えられたとき、出発した都市から他都市へそれぞれ 1 回だけ訪れ、最後に出発都市に戻る巡回路のうち、最小の総移動コストを求める問題である。また、NP 困難な問題として知られており、最適解を多項式時間で求めることはできない。そのため、近似解を得るアルゴリズムが数多く考案されて



符号距離 $d$	実行時間 [s]
3	38 s
5	7138 s

表 2 実行時間

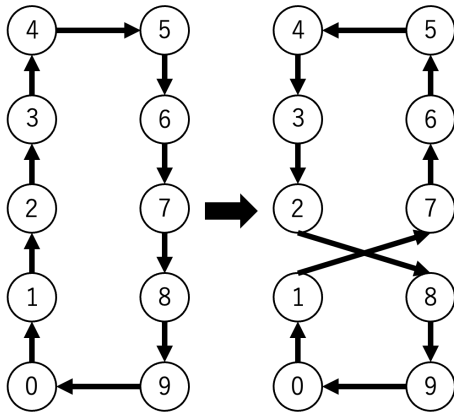


図 7  $x = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}, i = 1, j = 7$  のときの巡回路の変更

いる。

巡回路は長さ  $n$  の列  $\{x_0, x_1, \dots, x_{n-1}\}$  を用いて表現し、これは  $x_0$  番目の都市から出発し、 $x_1$  番目の都市を訪れ、最後に  $x_{n-1}$  番目の都市を訪れて  $x_0$  番目の都市に戻ることを意味する。巡回セールスマン問題を現実的な時間で解く手法はいくつかあるが、その1つに局所探索法の1つである 2-opt 法が挙げられる。例えば  $n = 10$  の場合、巡回路を最初

$$x = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\} \quad (1.20)$$

に選んでいたとする。2-opt 法は全ての辺から二辺を選択して繋ぎ替える手法である。例えば、図7のように1と2、7と8を結ぶ辺を1と7、2と8を結ぶ辺に繋ぎ替えて

$$x' = \{0, 1, 7, 6, 5, 4, 3, 2, 8, 9\} \quad (1.21)$$

にしたとする。このとき、 $x$  と  $x'$  の巡回路の総移動コストを比べ、小さくなる巡回路を採用し、 $x' = x$  とする。この処理を繰り返すのが 2-opt 法である。2-opt 法の利点として、時間計算量は  $O(n^2)$  なので、巡回セールスマン問題を高速に求解可能である点が挙げられる。

### 参考文献

[1] Nielsen, M. A. and Chuang, I.: Quantum computation and quantum information (2002).  
 [2] Shor, P. W.: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, *SIAM Journal on Computing*, Vol. 26, No. 5, p. 1484–1509 (1997).  
 [3] Grover, L. K.: A fast quantum mechanical algorithm for database search (1996).  
 [4] Sugisaki, K., Sakai, C., Toyota, K., Sato, K., Shiomi, D. and Takui, T.: Bayesian phase difference estimation:

a general quantum algorithm for the direct calculation of energy gaps, *Phys. Chem. Chem. Phys.*, Vol. 23, pp. 20152–20162 (online), DOI: 10.1039/D1CP03156B (2021).

[5] Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., Biswas, R., Boixo, S., Brandao, F. G. S. L., Buell, D. A., Burkett, B., Chen, Y., Chen, Z., Chiaro, B., Collins, R., Courtney, W., Dunsworth, A., Farhi, E., Foxen, B., Fowler, A., Gidney, C., Giustina, M., Graff, R., Guerin, K., Habegger, S., Harrigan, M. P., Hartmann, M. J., Ho, A., Hoffmann, M., Huang, T., Humble, T. S., Isakov, S. V., Jeffrey, E., Jiang, Z., Kafri, D., Kechedzhi, K., Kelly, J., Klimov, P. V., Knysh, S., Korotkov, A., Kostrița, F., Landhuis, D., Lindmark, M., Lucero, E., Lyakh, D., Mandrà, S., McClean, J. R., McEwen, M., Megrant, A., Mi, X., Michielsen, K., Mohseni, M., Mutus, J., Naaman, O., Neeley, M., Neill, C., Niu, M. Y., Ostby, E., Petukhov, A., Platt, J. C., Quintana, C., Rieffel, E. G., Roushan, P., Rubin, N. C., Sank, D., Satzinger, K. J., Smelyanskiy, V., Sung, K. J., Trevithick, M. D., Vainsencher, A., Villalonga, B., White, T., Yao, Z. J., Yeh, P., Zalcman, A., Neven, H. and Martinis, J. M.: Quantum supremacy using a programmable superconducting processor, *Nature*, Vol. 574, No. 7779, pp. 505–510 (online), DOI: 10.1038/s41586-019-1666-5 (2019).  
 [6] Lidar, D. A. and Brun, T. A.: *Quantum error correction*, Cambridge University Press (2013).  
 [7] Kitaev, A. Y.: Quantum computations: algorithms and error correction, *Russian Mathematical Surveys*, Vol. 52, No. 6, pp. 1191–1249 (1997).  
 [8] Bravyi, S. B. and Kitaev, A. Y.: Quantum codes on a lattice with boundary, *arXiv preprint quant-ph/9811052* (1998).  
 [9] Fowler, A. G., Whiteside, A. C. and Hollenberg, L. C.: Towards practical classical processing for the surface code, *Physical review letters*, Vol. 108, No. 18, p. 180501 (2012).  
 [10] Fowler, A. G. and Gidney, C.: Low overhead quantum computation using lattice surgery, *arXiv preprint arXiv:1808.06709* (2018).  
 [11] Denchev, V. S. and Pandurangan, G.: Distributed Quantum Computing: A New Frontier in Distributed Systems or Science Fiction?, *SIGACT News*, Vol. 39, No. 3, p. 77–95 (online), DOI: 10.1145/1412700.1412718 (2008).  
 [12] Gottesman, D.: Stabilizer Codes and Quantum Error Correction (1997).  
 [13] Poulin, D.: Optimal and efficient decoding of concatenated quantum block codes, *Physical Review A*, Vol. 74, No. 5 (online), DOI: 10.1103/physreva.74.052333 (2006).  
 [14] Wang, C., Harrington, J. and Preskill, J.: Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory, *Annals of Physics*, Vol. 303, No. 1, pp. 31–58 (online), DOI: 10.1016/s0003-4916(02)00019-2 (2003).  
 [15] Kolmogorov, V.: Blossom V: a new implementation of a minimum cost perfect matching algorithm, *Mathematical Programming Computation*, Vol. 1, No. 1, pp. 43–67 (2009).

- [16] Dür, W., Briegel, H.-J., Cirac, J. I. and Zoller, P.: Quantum repeaters based on entanglement purification, *Physical Review A*, Vol. 59, No. 1, p. 169 (1999).
- [17] 前田雄也, 鈴木泰成, 山本 俊, 徳永裕己, 藤井啓祐: 2次元格子に集積化された量子ビット間の表面符号を用いた効率的な論理ベル状態の蒸留プロトコルの提案, 第7回量子ソフトウェア研究会 (2022).
- [18] Jiang, L., Taylor, J. M., Nemoto, K., Munro, W. J., Mester, R. V. and Lukin, M. D.: Quantum repeater with encoding, *Physical Review A*, Vol. 79, No. 3 (online), DOI: 10.1103/physreva.79.032325 (2009).
- [19] Fowler, A. G., Wang, D. S., Hill, C. D., Ladd, T. D., Mester, R. V. and Hollenberg, L. C. L.: Surface Code Quantum Communication, *Physical Review Letters*, Vol. 104, No. 18 (online), DOI: 10.1103/physrevlett.104.180503 (2010).
- [20] Li, Y. and Benjamin, S. C.: Hierarchical surface code for network quantum computing with modules of arbitrary size, *Phys. Rev. A*, Vol. 94, p. 042303 (online), DOI: 10.1103/PhysRevA.94.042303 (2016).
- [21] G.A.Croes: A Method for Solving Traveling-Salesman Problems, *Operations Research*, Vol. 6, No. 6, pp. 791–812 (1958).
- [22] Biggs, N. L., Lloyd, E. K. and Wilson, R. J.: *Graph Theory 1736-1936*, Clarendon Press (1976).