

量子計算回数決定法の QAOA 適用に向けた検討

松尾 脩平^{1,a)} 谷本 輝夫^{2,b)} 川上 哲志^{4,c)} 田淵 豊^{3,d)} 廣川 真男^{4,e)} 井上 弘士^{4,f)}

概要: 現在我々は、NISQ コンピュータの出力結果の信頼性担保を目的とし、統計的手法を用いた量子計算回数 (ショット数) 決定法に関する研究を進めている。本稿では、実用レベルに近い複雑な量子アプリケーションとして Quantum Approximate Optimization Algorithm (QAOA) に着目し、量子計算回数決定法を適用するための初期検討を行う。具体的には、量子ビットのエラー特性が QAOA の出力結果に与える影響、ならびに、量子計算回数が最適化反復回数へ与える影響を調査する。そして、「量子ビットエラー特性 (忠実度に悪影響を与える)」と「最適化アルゴリズムに内在するロバスト性 (最終解の質に好影響を与える)」のトレードオフを考察し、QAOA 向け量子計算回数決定法の構築に向けた指針を示す。

1. はじめに

ムーアの法則 [1] の終焉により、現代コンピュータ (以下、古典コンピュータと呼称) の性能向上は限界に達すると言われている [2]。この問題を解決する有望な一つの方向性として、量子ゲート方式コンピュータが挙げられる。特にその中でも、数十~数百量子ビットの搭載を前提に、ノイズの影響に起因するエラーの訂正を行わない NISQ (Noisy Intermediate-Scale Quantum) コンピュータが世界中で注目を集めている。

NISQ コンピュータの出力は実行する量子計算の確率分布に従うため、同一の量子回路で同じ量子計算を繰り返し実行する必要がある。しかしながら、現状、解の導出には最尤推定が用いられており、決め打ちされた実行回数から得られたデータの出現確率をそのまま確率分布に置き換える。この場合、必要十分な確率分布推定が行われていない可能性があり、NISQ コンピュータによる計算結果の信頼性を判断することが難しい状況となる。この問題を解決すべく、現在我々は、NISQ コンピュータの出力結果の信頼性担保を目的とし、統計的手法を用いた量子計算回数決定法に関する研究を進めている [3]。しかしながら、現状、量

子フーリエ変換などの単純な応用を対象としたケーススタディに留まっており、より実用レベルに近い複雑な量子アプリケーションへの適用を検討する必要がある。

そこで本稿では、NISQ コンピュータの応用として近年注目を集めている Quantum Approximate Optimization Algorithm (QAOA) を対象とし、量子計算回数決定法を適用するための初期検討を行う。具体的には、これまでに我々が構築した NISQ シミュレータ (デコヒーレンス、初期化エラー、読み出しエラーを反映可能) [4] を利用し、量子ビットエラー特性が QAOA の出力結果に与える影響、ならびに、量子計算回数が最適化反復回数へ与える影響を調査する。ここでのポイントは、「量子ビットエラーが、最適化アルゴリズムの実行挙動に影響を与える可能性がある」という点である。すなわち、「量子ビットエラー特性 (忠実度に悪影響を与える)」と「最適化アルゴリズムに内在するロバスト性 (最終解の質に好影響を与える)」のトレードオフが存在する可能性があり、これらの相関を適切に考慮することが重要となる。本稿では、これらの分析に基づき、今後の QAOA 向け量子計算回数決定法の設計に向けた指針を示す。

本稿の構成は次の通りである。まず、第 2 章で量子ゲート方式コンピュータの実行モデルとエラーについて述べる。次に第 3 章で、実アプリケーションとして設定した QAOA の詳細について解説する。最後に第 4 章で実験及びその結果と考察を示した後、第 5 章で本研究のまとめを行う。

2. 量子ゲート方式コンピュータ

本章では、量子ゲート方式コンピュータの基本デバイス

¹ 九州大学大学院システム情報科学府情報知能工学専攻

² 九州大学情報基盤研究開発センター

³ 東京大学先端科学技術研究センター

⁴ 九州大学大学院システム情報科学研究科

a) shuhei.matuo@cpc.ait.kyushu-u.ac.jp

b) tteruo@kyudai.jp

c) satoshi.kawakami@cpc.ait.kyushu-u.ac.jp

d) tabuchi@qc.rcast.u-tokyo.ac.jp

e) hirokawa@inf.kyushu-u.ac.jp

f) inoue@ait.kyushu-u.ac.jp

である量子ビットの特徴，ならびに，実行モデルとエラーの詳細について説明する。

2.1 量子ビットの特徴

量子ゲート方式コンピュータが古典コンピュータに対して潜在的優位性を有するのは，量子ビットと古典ビットの性質の違いに起因する．一般的に，量子ビットでは物理的な状態に2進数を割り当てる．たとえば，粒子のスピンの上下や超伝導電極に蓄えられた電荷の正負に0と1を割り当てる．これらすべての量子ビットは，0と1の状態が重ね合わさった状態（量子重ね合わせ）になることが可能である．量子ビットを $|\psi\rangle$ とした場合，状態は式(1)で表現できる．なお， $|\rangle$ はケット記号と呼び，量子状態を表す．

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

右辺の係数 α と β は複素数であり， $|\alpha|^2 + |\beta|^2 = 1$ を満たす．この量子重ね合わせが，量子ゲート方式コンピュータの最大の特徴である超並列処理を可能にする．

また，量子ビットにおけるもう1つの大きな特性として量子もつれ（量子エンタングルメント）が存在する．これは，複数の量子ビットで量子状態を表すことは可能であるが，一つ一つの量子ビットの量子状態の単純な積では表すことができない状態を指す．式(2)は2量子ビットの量子もつれの例である．

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad (2)$$

この状態が量子もつれであることを示す．各ケット記号内部の左インデックスが第一量子ビット，右インデックスが第二量子ビットである．第一量子ビットの状態を $|\psi_1\rangle = \alpha|0\rangle + \beta|1\rangle$ ，第二量子ビットの状態を $|\psi_2\rangle = \gamma|0\rangle + \delta|1\rangle$ で表すとき，この2量子ビットの直積状態 $|\psi_1\psi_2\rangle$ は，式(3)で表される．

$$\begin{aligned} |\psi_1\psi_2\rangle &= (\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle) \\ &= \alpha\gamma|00\rangle + \alpha\delta|01\rangle + \beta\gamma|10\rangle + \beta\delta|11\rangle \end{aligned} \quad (3)$$

ここで，式(3)を式(2)と同じ状態にするには， $|01\rangle$ と $|10\rangle$ の係数を0にしなければならない．しかし， $|01\rangle$ と $|10\rangle$ の係数を0にすると，必ず $|00\rangle$ もしくは $|11\rangle$ の係数も0となる．この結果は，第一量子ビットと第二量子ビットの表現が間違っていることを意味しており，量子もつれ状態の量子ビットは，個々の量子状態の直積で表現できないことを示している．

さらに，量子ビットの特性として測定による量子状態の崩壊が挙げられる．これは量子ビットの状態がどのようなものであっても，測定すると必ず $|0\rangle$ または $|1\rangle$ の状態に収束することを指している．例えば，量子ビットの状態が

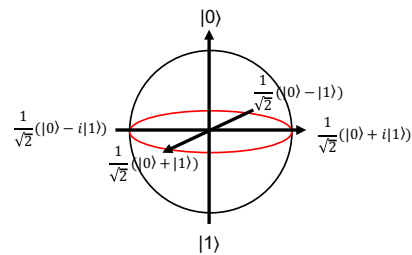


図1 ブロツホ球

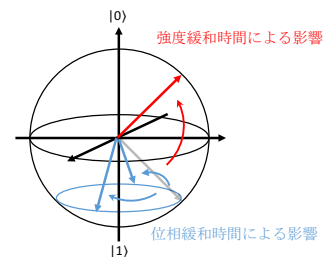


図2 緩和時間による状態の遷移

式(1)のとき，測定を行うと量子ビットの状態は， $|\alpha|^2$ の確率で $|0\rangle$ に， $|\beta|^2$ の確率で $|1\rangle$ に確定する．すなわち，量子コンピュータの計算結果は確率的な分布を持つことになる．また，量子もつれが発生している場合は特殊な現象が発生する．式(2)の量子状態の場合に第一量子ビットを測定すると，50%の確率で $|0\rangle$ または $|1\rangle$ に確定する．このとき，第一量子ビットの状態が確定すると第二量子ビットの状態も同時に確定するのは式(2)より明らかである．これは，量子力学の非局所性と呼ばれており，この特性を利用したアイデアに量子テレポーテーション[5]が存在する．

さらに量子ビットには数多くの種類があり，それぞれのデバイスによって特性が異なる．本研究では，超伝導電極を用いる超伝導量子ビット目した．超伝導量子ビットとは，超伝導電極の間にジョセフソン接合を挟むことで低エネルギー状態と高エネルギー状態の2準位を作り出し，それぞれに $|0\rangle$ と $|1\rangle$ を割り当てたものである[6]．本稿では，基底状態を $|0\rangle$ ，励起状態を $|1\rangle$ に割り当てたものを考える．超伝導量子ビットには，電荷型，位相型，磁束型，ならびに，電荷型の亜種としてトランズモン型，の計4種類が存在する．本研究では，4種類の中でも比較的ノイズが少ないとされる超伝導トランズモン型量子ビットを対象とした．

超伝導量子ビットの特徴として緩和時間が存在する．緩和時間は強度緩和時間(T_1)，位相緩和時間(T_2)の2種類に分けられる．この緩和時間は，量子ビットの励起された状態が基底状態に戻るまでの時間を定めた時定数になっており，ブロツホ球上の点で視覚的に表現可能である．図1はブロツホ球を表しており，この球は式(1)のように表される状態（純粋状態）を単位球面上で表現したものである．図2では，実際にブロツホ球上で緩和時間を表現して

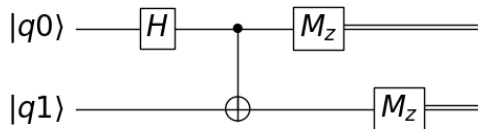


図 3 量子回路の例

いる。強度緩和時間 T_1 は、励起された状態がブロッホ球の Z 軸方向に基底状態へと戻る時定数であり、位相緩和時間 T_2 は、励起された状態がブロッホ球の XY 平面に並行な面で一様に振れるまでの時定数である。

2.2 実行モデル

量子ゲート方式コンピュータは、量子ビットに対して状態を変化させる操作を繰り返し計算を行う。これは、ブロッホ球上の点を移動させることに相当する。量子ビットの状態変化の工程（量子ゲート演算）が古典コンピュータの論理ゲートを電気信号が通過する行為に相当することから、量子ゲート方式と呼ばれている。そのため、量子ゲート方式コンピュータには、古典コンピュータを構成する論理ゲートを使用した回路は必要ない。量子ゲート方式コンピュータは、量子ビット同士を繋げた配置（トポロジー）で構成される。

しかしながら、量子コンピュータも量子ゲート演算を実行する対象ビットやその順番を規定する必要がある。そのため、量子ゲート方式コンピュータには量子ゲート演算の対象や順番を示す量子回路が存在する。この量子回路は実回路を示すものではなく、量子ゲート演算の順番を決める上で人間がわかりやすいように記述した一種のフローチャートである。図 3 に 2 量子ビットの量子回路の例を示す。上下 2 本の横線がそれぞれ量子ビット (q_0 , q_1) を表し、四角で囲まれたものが 1 入力量子ゲート、2 つの横線をつなげるものが 2 入力量子ゲートである。 H と書かれた量子ゲートは $|0\rangle$ と $|1\rangle$ を同じ重みで重ね合わせ状態にする“アダマールゲート”，2 つの量子ビットをつなげている量子ゲートは制御側のビットが 1 のときにのみ操作側を反転させる“制御 NOT ゲート”， M_Z と書かれた量子ゲートは測定するためのゲートに対応している。

量子ゲート方式コンピュータは①量子ビットの初期化、②量子ゲート演算、③測定、により実行を完了する。しかしながら、前述の通り量子ゲート方式コンピュータの出力は確率分布に従って値が出力される。量子ビットが保持する確率分布を直接確認することは不可能であるため、複数回同じ演算を繰り返して出力値の分布を推定する必要がある。図 4 に量子ゲート方式コンピュータの実行モデル概要図を示す。演算の対象ビットや順番を表す量子回路を入力した後、基本処理である①～③を複数回行い、出力値の確

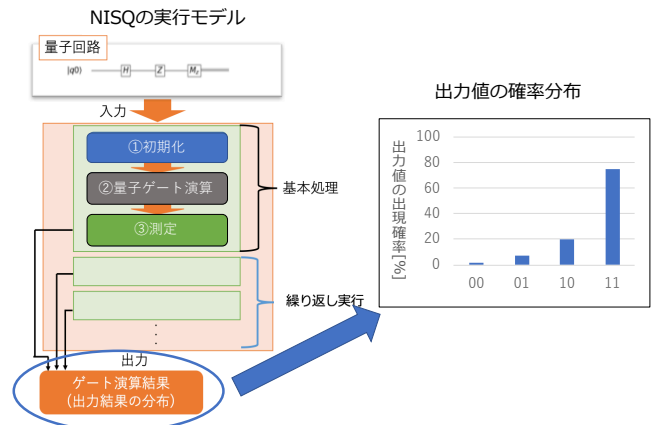


図 4 量子ゲート方式コンピュータの実行モデル

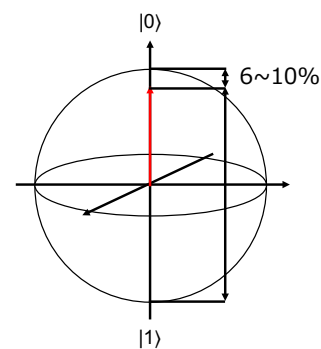


図 5 初期化エラーのイメージ

率分布を測定する。

2.3 量子ビットのエラー

量子ゲート方式コンピュータは量子状態を情報として利用しているため、量子的なノイズに起因するエラーが存在する。本研究では、初期化エラー、デコヒーレンス緩和、ならびに、読み出しエラーを考慮する。

2.3.1 初期化エラー

量子ビットの初期化方法は複数存在するが、本研究では passive reset と post-selection [7] を対象とする。passive reset は、量子ビットの状態が時間とともに基底状態に近づいていく特性を利用して初期化を行う。post-selection では、passive reset の後に、非破壊的な読み出しを行う。これらの方法では完全に $|0\rangle$ の状態にはならず、図 5 のような状態に収束する。このように、初期化後に完全に $|0\rangle$ の状態にならないことを初期化エラーと呼ぶ。文献 [7] において十分な時間 passive reset による初期化をした場合、93.3% まで $|0\rangle$ になると報告されており、post-selection では 99.8% まで改善される。

2.3.2 デコヒーレンス緩和

前述の通り、量子ビットは時間とともにその状態が基底状態に近づいていく特性を持つ。デコヒーレンス緩和とは、この特性に起因するエラーであり、演算時に量子ビットの

状態が時間とともに崩壊していくことを指す。これは、量子ビットの状態が時間とともに失われていくことを意味しており、量子ビットが扱わずら原因の一つとなっている。

2.3.3 読み出しエラー

量子ゲート方式コンピュータにおいて、測定の際に生じるエラーを読み出しエラーと定義する。これは量子ビットを測定する際に、状態 $|0\rangle$ を $|1\rangle$ に、状態 $|1\rangle$ を $|0\rangle$ に読み間違えるエラーである。このエラーの生起確率は非対称であることが文献 [7] で報告されている。

3. QAOA

本章では、実アプリケーションとして設定した Quantum Approximate Optimization Algorithm (QAOA) [8] について述べる。

QAOA は組み合わせ最適化計算を行う量子・古典ハイブリッドアルゴリズムである。前述したように、NISQ コンピュータにはエラー訂正機能が備わっていないため、量子ゲート演算が多くなると計算結果におけるエラーの影響が大きくなる。そこで、時間発展計算を分割して NISQ コンピュータでも計算できるよう提案されたアルゴリズムが QAOA である。

QAOA は、求めたい解を基底状態に持つコストハミルトニアン H_c と、自明な基底状態を持つリファレンスハミルトニアン H_r を用いて、式 11 のような時間とともに変化するハミルトニアンを考える。ここで、 t_f は時間 t を規格化するパラメータである。

$$H(t) = (1 - \frac{t}{t_f})H_r + \frac{t}{t_f}H_c \quad (4)$$

この時、量子状態 $|\psi\rangle$ は以下の微分方程式に従う。

$$-i\hbar \frac{\partial}{\partial t} |\psi\rangle = H(t) |\psi\rangle \quad (5)$$

その解は式 6 のようになる。

$$|\psi\rangle = e^{-i \int_0^t H(t) dt / \hbar} |\psi\rangle \quad (6)$$

ここで、式 6 を p ステップに離散化することで積分を行う。 p ステップに分割すると、

$$\begin{aligned} |\psi(t_p)\rangle &\approx e^{-i \frac{H(t_p)}{\hbar} \Delta t_p} e^{-i \frac{H(t_{p-1})}{\hbar} \Delta t_{p-1}} \dots e^{-i \frac{H(t_1)}{\hbar} \Delta t_1} |\psi(0)\rangle \\ &\rightarrow U(t_p) |\psi(0)\rangle \end{aligned} \quad (7)$$

となる。ただし、 $\Delta t_p = t_p - t_{p-1}$ とする。さらに、Lie-Trotter 分解を式 (8) のように適用する。

$$\begin{aligned} e^{-i \frac{H(t_j)}{\hbar} \Delta t_j} &\approx e^{-i \frac{1}{2} (1 - \frac{t_j}{t_f}) H_r \Delta t_j} e^{-i \frac{t_j}{t_f} H_c \Delta t_j} \\ &= e^{-i \beta_j H_r} e^{-i \gamma_j H_c} \end{aligned} \quad (8)$$

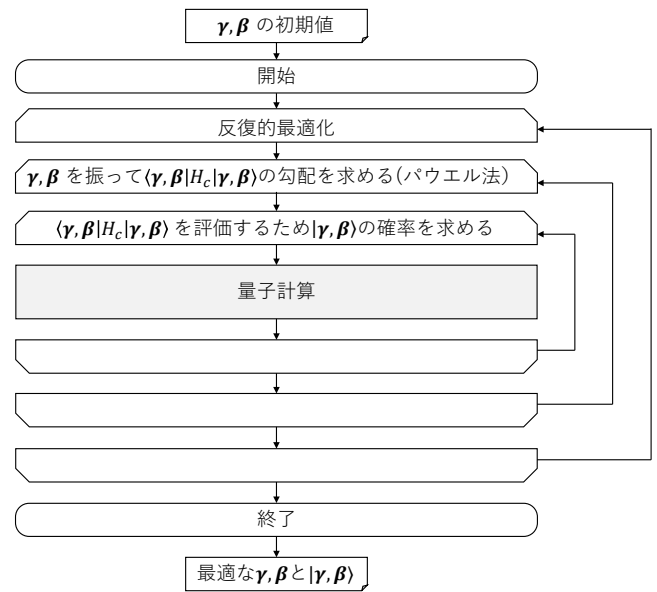


図 6 QAOA の実行フロー

ただし、このとき $\hbar = 1$ と近似しており、 $\beta_j = \frac{1}{2} (1 - \frac{t_j}{t_f}) \Delta t_j$ 、 $\gamma_j = \frac{t_j}{t_f} \Delta t_j$ である (ここで、 $\beta = \{\beta_1, \beta_2 \dots \beta_j \dots\}$ 、 $\gamma = \{\gamma_1, \gamma_2 \dots \gamma_j \dots\}$ である)。

$$U(t_p) = e^{-i \beta_p H_r} e^{-i \gamma_p H_c} e^{-i \beta_{p-1} H_r} e^{-i \gamma_{p-1} H_c} \dots e^{-i \beta_1 H_r} e^{-i \gamma_1 H_c} \quad (9)$$

この時、 H_c の期待値である $\langle \beta, \gamma | H_c | \beta, \gamma \rangle$ を最小化することを考える。量子状態 $|\beta, \gamma\rangle$ は以下の式で表される。

$$\begin{aligned} |\beta, \gamma\rangle &= |\psi(t_p)\rangle \\ &= U_X(\beta_p) U_C(\gamma_p) U_X(\beta_{p-1}) U_C(\gamma_{p-1}) \dots U_X(\beta_1) U_C(\gamma_1) |s\rangle \end{aligned} \quad (10)$$

ただし、 $U_X(\beta_j) = e^{-i \beta_j H_r}$ 、 $U_C(\gamma_j) = e^{-i \gamma_j H_c}$ であり、 $|s\rangle = |+\rangle^{\otimes n}$ ($|0\rangle$ の全量子ビットに H ゲートをかけたもの) である。この $|\beta, \gamma\rangle$ を用いて $\langle \beta, \gamma | H_c | \beta, \gamma \rangle$ を測定し、古典コンピュータでその値がより小さくなるようにパラメータ β 、 γ を更新する。それを繰り返して、最適な β 、 γ を求めることで解を導く。

QAOA の実行フローを図 6 に示す。QAOA の実行は 3 重ループ構造を持つことが分かる。それらは反復的最適化、コスト関数の勾配測定、コスト関数評価に用いる量子状態の確率測定である。最内ループで量子計算を行うため、その実行回数は非常に大きくなることが想定される。したがって、それぞれの反復数と最終解の関係や、忠実度などの量子ビット特性の最終解への影響を知ることは重要である。

4. 量子計算回数と QAOA の関係解析

本章では、使用したシミュレータの概要とその実装方法、及び実験方法とその結果について紹介する。

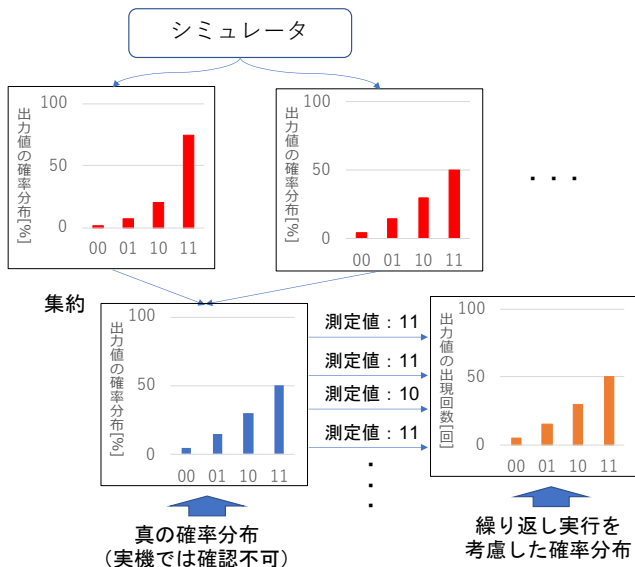


図 7 繰り返し実行を再現するためのモデル

4.1 シミュレータ

本研究では、この問題を対処可能なシミュレータである Intel-QS を採用している。本シミュレータはデコヒーレンス緩和を再現するために、複数回のシミュレーションを行い、各シミュレーションの確率分布を集約して混合状態の確率分布を求める。

4.2 シミュレータの拡張

Intel-QS でサポートされているエラーはデコヒーレンス緩和のみであり、初期化エラーと読み出しエラーは考慮されていない。また、Intel-QS は出力値の確率分布をそのまま出力しており、これは実機における無限の実行回数に相当する。そこで本研究では、大きく 2 点の拡張を行った。

1 つ目はエラーの拡張である。具体的には、初期化エラーと読み出しエラーのエラーモデルを構築し、それをシミュレータに組み込んだ [4]。

初期化エラー：初期化後の量子ビットを 2 値 ($|0\rangle$ or $|1\rangle$) で考え、取りえる状態が従う生起確率をもとに初期化後の状態を決定する。passive reset での 2 量子ビットの場合の具体例を提示する。このとき、初期化後に取りえる状態は、 $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ の 4 状態である。初期化後にそれぞれの状態になる確率は、87.0%, 6.3%, 6.3%, 0.4% であるため、その確率に従って初期化後の状態を決定する。この処理を繰り返し実行のたびに行うことで、実機における初期化エラーを再現している。

読み出しエラー：次節で紹介する読み出しエラーの確率に従って、測定後の量子ビットの状態を反転させて再現する。

2 つ目は繰り返し実行モデルの構築である。前述した通り、Intel-QS は内部で計算した確率分布を出力するため、繰り返し実行を考慮していない。実機の NISQ コンピュー

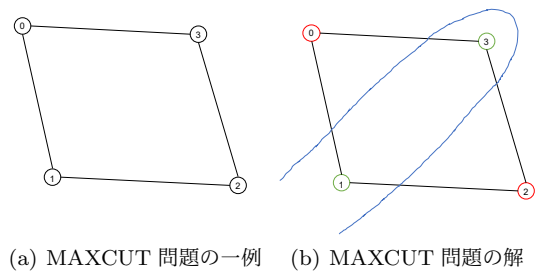


図 8 MAXCUT 問題

表 1 パラメータ設定

デコヒーレンス緩和	
T_1	48.7 μs
T_2	30.2 μs
1 量子ビットゲート遅延	50 ns
2 量子ビットゲート遅延	430 ns
基本ゲート	CNOT, R_X , R_Y , R_Z
初期化時の忠実度	
passive reset	93.3%
passive reset + post-selection	99.8%
読み出し時の忠実度	
	読み出し後の値
	$ 0\rangle$ $ 1\rangle$
読み出し前の値	$ 0\rangle$ 99.8% 0.2%
	$ 1\rangle$ 2.2% 97.8%

タでは内部で保持している出力値の確率分布を確認することが不可能なため、繰り返し実行を行うことで確率分布を推定する。本研究ではこの処理をシミュレータ上で再現する。その様子を図 7 に示す。前述したように、本シミュレータは複数回のシミュレーションを行い、その際の確率分布を集約して混合状態の確率分布を求める。そのようにして求めた確率分布に従う観測を疑似的に行い、確率推定を再現している。

4.3 実験

本研究では、量子計算回数が QAOA の実行結果に与える影響を調査することを目的としている。そのためまず、1 回の試行関数評価あたりの量子計算回数と確率分布の誤差の関係を調べ、量子計算回数の確率推定への影響を確認する。その後、QAOA の正答率とその時の最適化反復回数や総量子計算回数を調査し、それぞれの関係性を分析する。

4.3.1 実験方法

本実験での量子ビットの各パラメータは表 1 のように設定する。各パラメータは論文 [7], [9] を元に決定している。

本実験で採用したベンチマークは MAXCUT 問題である。MAXCUT 問題の例を図 8 に示す。図 8(a) のようなグラフ (頂点とその間の辺の集合) を考えたとき、グラフの 0, 1, 2, 3 の 4 つの頂点を 1 本の線で 2 つのグループに分ける。この時、2 グループの間を通る辺の重みの合計が最大になるような分け方を見つけるのが MAXCUT 問題

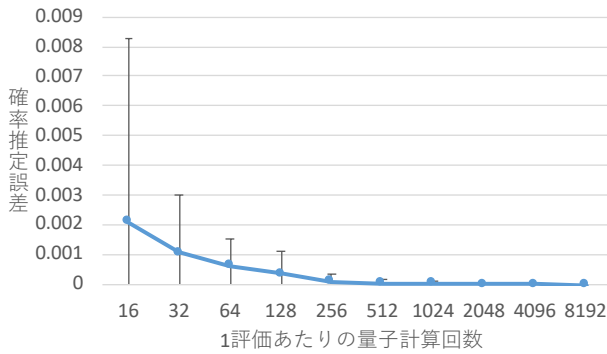


図 9 確率分布の推定誤差 (ideal)

である。辺の重みをすべて 1 とすると、この場合の解は図 8(b) のように頂点 0, 2 と 1, 3 が同グループになる時で、4 が最大値となる。本研究では図 8 のような 4 頂点で各頂点が他の 2 頂点と辺でつながっているグラフを対象とした。ただし、辺の重みはすべて 1 としている。この時、最小値を求めたいコストハミルトニアン H_C は、

$$H_C = -\frac{1}{2} \sum_{i,j} C_{ij}(1 - z_i z_j) \quad (11)$$

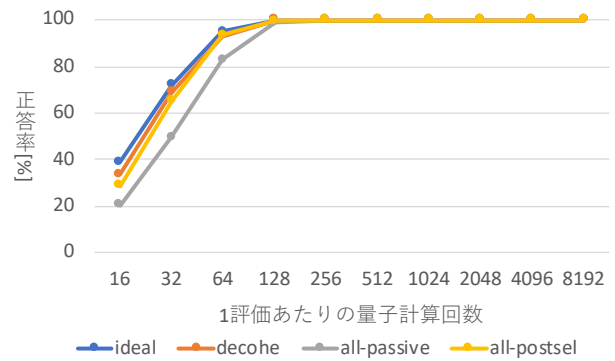
$$= \frac{1}{2}(z_0 z_1 + z_1 z_2 + z_2 z_3 + z_3 z_0)$$

となる。ここで、 C_{ij} は頂点 i, j を結ぶ辺の重みであり、 z_i は頂点 i が片方のグループに属していれば 1、属していなければ -1 とする (量子ビットの 1 と 0 を 1 と -1 に対応させる)。これを第 3 章で解説した QAOA で解く。なお、問題の分割数 p は 1 で固定しており、最適化パラメータ β, γ の初期値は 0.1 と設定する。さらに、QAOA における最適化には `scipy` の `minimize` 関数を用いており、パウエル法を採用する。

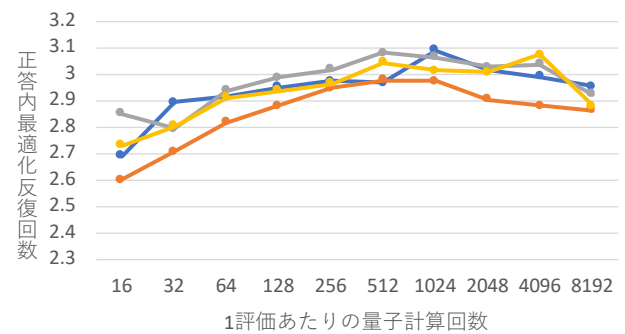
上記の条件下において、コスト関数の 1 評価あたりの量子計算回数を {16, 32, 64, 128, 256, 512, 1,024, 2,048, 4,096, 8,192} の 10 通りで変化させ、それぞれ理想状態 (ideal)、デコヒーレンス緩和考慮状態 (decohe)、passive reset の初期化エラー考慮状態 (all-passive)、post-selection の初期化エラー考慮状態 (all-postsel) の 4 状態で QAOA を実行する。なお、各状態において QAOA を 1,024 回実行し、それらから算出した統計量を実験結果とする。

4.3.2 実験結果と考察

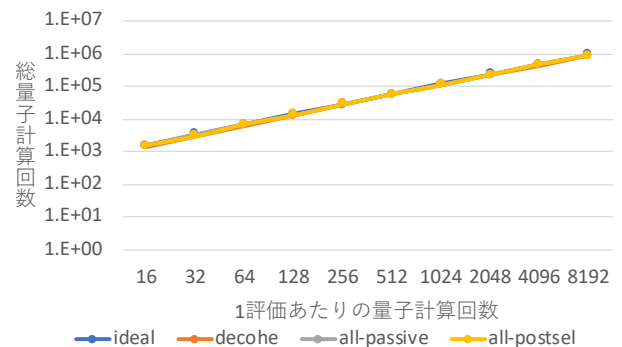
理想状態 (ideal) における 1 評価あたりの量子計算回数と確率分布の推定誤差の関係を図 9 に示す。ここでの誤差とは、理想状態における確率分布とベンチマークにおける真の確率分布とのユークリッド距離である。なお、当該グラフは 1,024 回実行した際のユークリッド距離の平均を表しており、エラーバーはその標準偏差である。この図から、1 評価あたりの量子計算回数が増加すると確率分布の誤差は小さくなる傾向にあることがわかり、量子計算回数は確率分布に大きな影響を及ぼすことが分かる。



(a) QAOA の正答率



(b) 正答内最適化反復回数



(c) 総量子計算回数

図 10 QAOA 実験の結果

次に、QAOA の正答率とその時の最適化反復回数及び総量子計算回数を図 10 に示す。図 10(a) は、それぞれのエラー条件下での QAOA の正答率を示している。ここでの正答とは、解となる 2 つの値 (1010 と 0101) の出力回数がどちらも、それ以外で最も多く出力されている値の出力回数より多い場合のことを指している。図を見ると常に理想状態の正答率が最も高く、passive reset を用いた初期化エラーが最も低くなっている。さらに、どのエラー条件下でも 1 評価あたりの量子計算回数を 128 回にした時点で正答率がほぼ 100% になる。図 10(b) は、1 回の QAOA 実行における最適化反復回数を示している。図より、最適化反復回数はそれぞれの量子計算回数において有意な差は無いという結果が読み取れる。また、誤った答えを出す場合も

最適化は収束しており、かつ、それに必要な最適化ステップ数は正答する場合とほぼ同等である。これは間違っただけでも収束していることを示しており、本ベンチマークにおいて今回の最適化手法では量子計算回数に関わらず少ない最適化反復回数で収束すると言える。図 10(c) は 1 回の QAOA 実行における総量子計算回数を示しており、総量子計算回数はエラー条件に関わらず 1 評価あたりの量子計算回数が支配的であることがわかる。すなわち、正答率を維持できる範囲でできるだけ少ない量子計算回数を採用するのが望ましい（本実験においては 128 回）。

本実験のような 4 量子ビットかつ $p=1$ で十分解ける問題でも、1 評価あたりの量子計算回数を低くすると正答率が悪化し、多くしすぎると無駄な量子計算が増加する。さらに、考慮するエラーを変えることで正答率が変化するため、実機を用いた実行においても本実験における 128 回のような最適な量子計算回数を決める必要がある。また、本研究の実装においては答えが間違っている場合でも早い段階で最適化が収束する。このことに着目し、パウエル法において正答率が高くなるように設定すると、正答率や最適化反復回数が増えることが予想される。そのような設定における総量子計算回数を今後調査すべきであると考えられる。また、実機における 19 量子ビットでの QAOA 実行に関する文献 [10] では、最適化反復回数は 50 以上になると報告されている。したがって、今後、より大きな問題サイズにおける調査が必要である。このとき、問題によっては最適解に到達するためには p 値を大きくする必要があることが知られており [11]、 p 値をパラメータとみなした実験にも今後取り組む。

5. まとめ

NISQ コンピュータをシステムとして構築するためには、量子計算回数を適切に決定することは重要である。本論文では、特定条件下における QAOA の正答率と量子計算回数の関係を実験的に調査した。評価の結果、エラー考慮の有無によって正答率が変化すること、最適な量子計算回数が存在することが分かった。この結果は、適切な実行回数をエラーの影響を考慮しながら探索することの重要性を示している。量子計算の分割数や最適化手法のパラメータを変化させた際の正答率や最適化反復回数、量子計算回数の調査は今後の課題である。

謝辞 本研究は一部、JST、未来社会創造事業、JP-MJMI18E1、ならびに、文部科学省科学研究費補助金 JP19H01105 の支援を受けたものである。なお、必要な計算の一部は九州大学情報基盤研究センターの協力による。

参考文献

- [1] Moore, G. E.: Cramming more components onto integrated circuits, *Electronics Magazine*, Vol. 38, No. 8, pp. 114–117 (1965).
- [2] 井上弘士, 川上哲志, 田中雅光: ムアの法則の限界が見えた今, デバイス研究はどこに向かうのか: コンピュータアーキテクチャの視点から, 電子情報通信学会誌, Vol. 102, No. 10, pp. 957–962 (2019).
- [3] Tanimoto, T., Matsuo, S., Kawakami, S., Tabuchi, Y., Hirokawa, M. and Inoue, K.: Practical error modeling toward realistic NISQ simulation, *In Proceedings of the first International Workshop on Quantum Computing: Circuits Systems Automation and Applications (QC-CSAA) in conjunction with the ISVLSI* (2020).
- [4] Tanimoto, T., Matsuo, S., Kawakami, S., Tabuchi, Y., Hirokawa, M. and Inoue, K.: Practical error modeling toward realistic NISQ simulation, *In Proceedings of the first International Workshop on Quantum Computing: Circuits Systems Automation and Applications (QC-CSAA) in conjunction with the ISVLSI* (2020).
- [5] 古澤 明: 量子テレポーテーション, 表面科学, Vol. 32, No. 12, pp. 801–803 (2011).
- [6] 阿部英介, 伊藤公平: 固体量子情報デバイスの現状と将来展望-万能デジタル量子コンピュータの実現に向けて, 応用物理, Vol. 86, No. 6, pp. 453–466 (2017).
- [7] Kono, S., Koshino, K., Tabuchi, Y., Noguchi, A. and Nakamura, Y.: Quantum non-demolition detection of an itinerant microwave photon, *Nature Physics*, Vol. 14, No. 6, pp. 546–549 (2018).
- [8] Farhi, E., Goldstone, J. and Gutmann, S.: A Quantum Approximate Optimization Algorithm (2014).
- [9] Saki, A. A., Alam, M. and Ghosh, S.: QURE: Qubit Re-allocation in Noisy Intermediate-Scale Quantum Computers, *Proc. of the 56th ACM/IEEE Design Automation Conf.*, pp. 1–6 (2019).
- [10] Otterbach, J. S., Manenti, R. and et al., N. A.: Unsupervised Machine Learning on a Hybrid Quantum Computer (2017).
- [11] Heim, B., Rønnow, T. F., Isakov, S. V. and Troyer, M.: Quantum versus classical annealing of Ising spin glasses, *Science*, Vol. 348, No. 6231, pp. 215–217 (online), DOI: 10.1126/science.aaa4170 (2015).