Vol.2019-ARC-236 No.13
2019/6/12

NISQにおけるfidelityが処理時間へ与える影響の解析 ～古典-量子ハイブリッドアーキテクチャの実現に向けて～

松尾 脩平[1,a)]　谷本 輝夫[2,b)]　川上 智志[3,c)]　田渕 豊[4]　廣川 真男[3]　井上 弘士[5,d)]

概要：近年，MOSFET で実装された一般的なコンピュータ (古典コンピュータ) の性能限界が指摘されている．この問題を解決する方法の一つとして量子コンピュータが挙げられる．量子コンピュータでは，基本デバイスである量子ビットによる超並列処理が可能となる．量子コンピュータは同一の処理を繰り返し行い，出力されたデータをもとに確率分布の推定を行い解を導出する．しかし，今現在量子コンピュータの解を導出するための推定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ている．すなわち，必要十分な確率分布推定が行われていない可能性がある．そのため，量子コンピュータの解の確率分布に対して最適な推定方法を導入する必要がある．本研究では NISQ 向けの性能評価環境を構築し，確率分布の推定に必要となる最小の実行回数と fidelity の関係を解析した．

1. はじめに

我々の生活を支えているコンピュータ（本論文では古典コンピュータと呼ぶ）は，様々な角度から進化を遂げてきた．しかし，MOSFET の面積，動作速度と消費電力の関係を示したデナードスケーリング [1] やトランジスタ集積度を予測したムーアの法則 [2] の破綻から，古典コンピュータの性能限界が指摘されている．この問題を解決すべく，古典コンピュータに取って代わる新たなコンピュータの実現法に関する研究開発が進められている．

この新たなコンピュータの候補の一つに量子コンピュータが挙げられる．量子コンピュータは量子アニーリング方式と量子ゲート方式の 2 種類に大別される．量子アニーリング方式とは，量子ゆらぎを利用してアニーリングを実行することで最適化問題を解く手法である [3]．一方，量子ゲート方式とは，量子状態に理論値「0」や「1」を割り当てた量子ビットを用いて，量子ゲート演算と呼ばれる処理を実行することで計算を行う方式である [4]．本論文では後者の量子ゲート方式に着目する．以後本論文で登場する量子コンピュータは，全て量子ゲート方式コンピュータを指す．

この量子ゲート方式コンピュータは現在研究段階にあり，日々進歩している．多くの研究者が量子ビットのデバイス実験を重ね，使用可能な量子ビット数や量子ビットを用いて計算可能な時間に関係する緩和時間は増加傾向にある．しかしながら，量子コンピュータをシステムとしてとらえ，計算機性能を解析・評価した事例は少ない．量子コンピュータは，①量子ビットの初期化，②量子ゲート演算，③測定，といった一連の処理を複数回行い，確率分布に基づき最終的な計算結果を出力する．すなわち，複数回の計算結果から確率分布を推定する必要がある．特に，初期化時及び測定時におけるノイズの影響により，計算結果の分布は分散が大きくなる．そのため，確率分布の推定が困難になると予想される．したがって，量子コンピュータの設計においては，エラーを考慮した確率分布の適切な推定が極めて重要である．

量子コンピュータの実行時間は先に述べた①～③までの実行に要する時間と，確率分布推定に必要となる繰り返し実行回数の積である．現在は決められた回数実行し解の出現頻度を数え上げ，出現確率を計算する（最尤推定値を採用する）ことで確率分布推定を行うことが多い．例えば，IBM Q [5] では 1，1024，4096，8192 の 4 つの実行回数から選択することができる．しかしながら，確率分布推定に必要十分な実行回数は非自明である．不足している場合は解の信頼性が損なわれ，過剰な場合は実行時間に無駄が生

[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] 九州大学大学院システム情報科学府 I&E ビジョナリー特別部門
[4] 東京大学先端科学技術研究センター
[5] 九州大学大学院システム情報科学研究院情報知能工学部門
[a)] shuhei.matuo@cpc.ait.kyushu-u.ac.jp
[b)] tteruo@kyudai.jp
[c)] satoshi.kawakami@cpc.ait.kyushu-u.ac.jp
[d)] inoue@ait.kyushu-u.ac.jp

© 2019 Information Processing Society of Japan

じる．したがって，複数回の実行過程で確率分布推定を行い，十分な試行回数に達した時点で実行終了するのが望ましい．

ここで，fidelity という概念を導入する．本研究では，初期化や測定時のエラーによる量子ビット状態の変化度合いを fidelity と定義した．量子コンピュータの系全体での fidelity，初期化エラーや読み出しエラー等の個別の fidelity が定義可能である．例として系全体の fidelity を数式で表すと，　fidelity=1-(解のエラー率)　となる．

本研究では統計的アプローチによる解の確率分布推定を行い，fidelity と量子コンピュータの実行時間への影響を明らかにする．これを実現するために，オープンソースシミュレータ “Intel-QS” [6] をベースとし，初期化エラー，デコヒーレンス，読み出しエラー，ならびに，バックアクションの影響を考慮可能なシミュレータへと拡張した．なお，量子ビットの実現には様々な方法が存在するが，本シミュレータでは超電導トランズモン量子ビットを対象としている．本シミュレータを用いて初期化の fidelity が確率分布の推定に与える影響に関する解析を行った結果，初期化の fidelity は解の正答率のみならず，解分布推定に要する実行時間にも影響することが分かった．現在標準的な 1024 回実行時に比べ，実行時間を 35%削減できる可能性があることが分かった．

本稿の構成は次の通りである．まず，第 2 章では量子ビットの特徴，ならびに量子ゲート方式コンピュータの概要とそのエラーについて述べる. 次に第 3 章では，量子コンピュータ向けシミュレータの問題点と本研究でベースとして使用した “Intel-QS” について解説する. 続いて，第 4 章では，初期化の fidelity が確率分布の推定に与える影響を解析し，最後に第 5 章でまとめる．

2. 量子ゲート方式コンピュータ

本章では，量子ビットの特徴，ならびに量子ゲート方式コンピュータの概要とそのエラーについて述べる．

2.1 量子ビットの特性

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多くの優位性を持っている要因の一つに，量子ビットの存在が関連している．

一般的に，量子ビットとは物理的な状態に 2 進数を割り当てたものである．例えば粒子のスピンの上下に 0 と 1 を割り当てたものや，超伝導電極に蓄えられた電荷の正負に当てはめているものが存在する．そ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）になることができるという特徴を持つ．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．

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{1}$$

右辺の係数 α と β は複素数であり，$|\alpha|^2 + |\beta|^2 = 1$ を満たしている．この重ね合わさった状態が，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している．

また，量子ビット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一つの例である．

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

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．

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

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 になってしまう．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を表現することが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 [7] が存在する．

2.2 超伝導トランズモン量子ビット

量子ビットには数多くの種類があり，それぞれのデバイ

© 2019 Information Processing Society of Japan

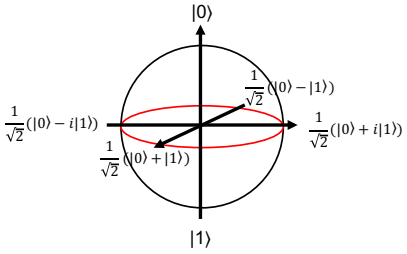


図 1 ブロッホ球

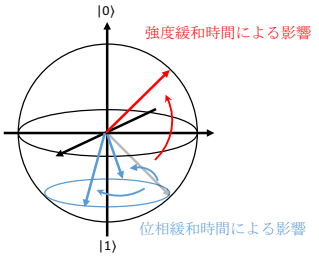


図 2 緩和時間による状態の遷移

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に注目した．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 [8]．便宜上，本論文では基底状態を $|0\rangle$，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，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大きな特徴として緩和時間の存在が挙げられる．緩和時間は強度緩和時間 (T_1)，位相緩和時間 (T_2) の 2 種類が存在する．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，ブロッホ球上の点で視覚的に表現可能である．図 1 はブロッホ球を表しており，この球は式 (1) のように表される状態 (これを純粋状態と呼ぶ) を単位球面上で表現したものである．図 2 では，実際にブロッホ球上で緩和時間を表現している．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．

2.3 量子ゲート方式コンピュータの概要

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 (ブロッホ球上の点を移動させる) 操作を繰り返し計算を行う．この量子ビットの状態変化の工程 (量子ゲート演算) がいわゆる古典コンピュータの論理ゲート（物理ゲート）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物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，単に量子ビット同士を繋げた配置 (トポロジー) で構成される．

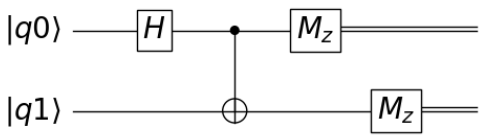


図 3 量子回路の例

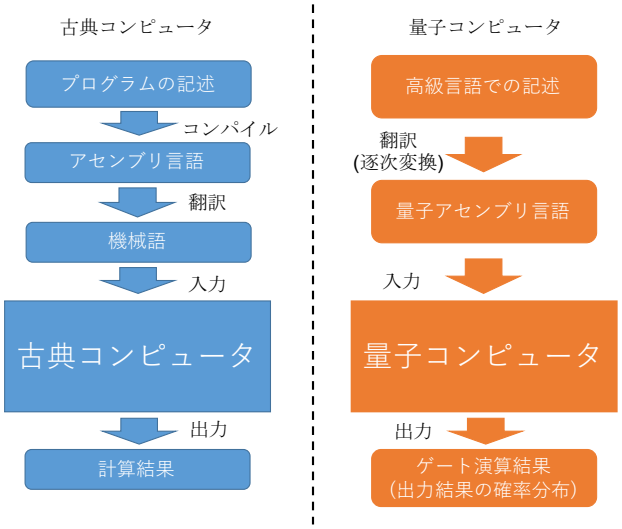


図 4 古典コンピュータと量子コンピュータの入出力比較

しかし，古典コンピュータの回路のように，量子コンピュータも量子ゲート演算を実行する対象や順番を決定する必要がある．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．この量子回路も，実際には存在しておらず，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（$q0$，$q1$）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．

上述した内容をふまえて，古典コンピュータと量子コンピュータでの実行フローの違いを図 4 で説明する．古典コンピュータの処理の流れは，まず指定された高級言語で記述された処理内容をアセンブリ言語にコンパイル，そし

© 2019 Information Processing Society of Japan



図 5　量子コンピュータの内部処理

て機械語に翻訳する．このとき高級言語で記述される内容は，計算内容そのものである (足し算や引き算)．その後，機械語を入力として古典コンピュータ内部で実際に計算が行われ，電気的なビットとして出力が現れる．

量子コピュータは，基本的に OpenQSAM(量子アセンブリ言語)[9] もしくは別の高級言語を用いて量子アセンブリと同抽象度の形式で処理内容を記述する．最初に量子ビットの状態を保持するための量子ビットレジスタを宣言し，適用する量子ゲート演算を順次書き並べ，最後に測定する量子ビットを指定することが基本となる．この量子アセンブリ言語を入力として，量子コンピュータ内で量子ゲート演算を実際に行う．その後，超伝導量子ビットの場合は測定した際に生じる電圧から 0/1 を判定し，出力結果とする．この出力結果は，量子ゲート演算により変化した $|0\rangle$ や $|1\rangle$ の係数に従った確率で現れるため，必ずしも欲している結果とは限らない．そのため量子コンピュータは同じ処理を複数回試行し，出力された結果の累積値を確率分布として読み取ることが基本となる．このとき確率分布の推定は統計的な処理を行っているのではなく，一定回数実行した後尤度推定（解の出現頻度をそのまま確率分布にする）によって推定するのが一般的である．

次に，量子コンピュータ内部で実行されている処理について解説する．図 5 は量子コンピュータが入力を受け取ってから出力するまでの処理の全体図を示す．まずはじめに，使用する量子ビットの初期化を行う．初期化についての詳細は第 2.5 節で説明する．量子ビットの初期化が完了すると，入力された量子アセンブリ言語にしたがって量子ゲート演算処理を行う．超伝導量子ビットでは，マイクロ波を照射することで量子ビットのエネルギー状態を変化させている [8]．最後に量子ビットの測定を行うが，前述した通り量子コンピュータは複数回処理を繰り返して出力結果の確率分布を導くため，測定後は再び量子ビットの初期化に戻る．複数回処理を実行した後，累積した結果を出力することで，量子ゲート演算による量子状態の確率分布が得られる．以上が，量子コンピュータの基本的な処理の流れである．

2.4　本研究で対象とするエラー

本節では，量子ゲート方式コンピュータにおける各エラーについて説明する．

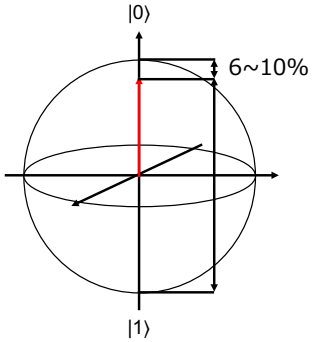


図 6　初期化エラーのイメージ

2.4.1　初期化エラー

本論文の第 2.5.1 節でも後述するが，量子ビットの初期化方法に一定時間待つことで状態を $|0\rangle$ に戻すパッシブリセットが存在する．このとき，量子ビットは完全に $|0\rangle$ の状態にはならず，確率的に 6〜10%ほど $|1\rangle$ である状態に収束する．このように，パッシブリセットによって 100%$|0\rangle$ の状態にならないことを初期化エラーと定義する．

このようなブロッホ球上の点では表現できず，複数の純粋状態を併せ持った状態のことを混合状態と呼ぶ．イメージとして視覚的に 1 量子ビットの初期化エラーを表現するのならば，ブロッホ球内の $|0\rangle$ 方向の矢印が直径に対して 6〜10%短くなっているようなイメージである（図 6）．

2.4.2　デコヒーレンス

第 2.2 節で超伝導量子ビットは強度緩和時間と位相緩和時間の影響により，状態が時定数 T_1，T_2 で基底状態に戻ることを述べた．本論文では，この両緩和時間の影響により，量子ゲート演算の際に時間とともに状態が崩壊していくことをデコヒーレンスと定義する．これは量子ビットの情報が時間とともに失われていくことを意味しており，量子ビットが扱いづらい原因の一つとなっている．

T_1，T_2 は時定数であるため，状態は $e^{(-t/T)}$(t : 時間，T : 時定数) に従う曲線で基底状態に近づいていく．そのため，量子ゲート演算の時間に応じてデコヒーレンスによるノイズが増減する．

2.4.3　読み出しエラー

量子ゲート演算を実行した後に量子ビットの状態を測定するが，その測定の際に生じるエラーを読み出しエラーと定義する．読み出しエラーには状態 $|0\rangle$ を 1 と読み間違える場合と状態 $|1\rangle$ を 0 と読み間違える場合がある．前者が起こる確率は 1%未満であるが，後者は時定数 T_1 の影響により確率が変化する．測定する際に量子ビットは状態 $|0\rangle$ または $|1\rangle$ に確定するが，$|1\rangle$ の場合は $e^{(-t/T_1)}$ に従って状態が基底状態に近づいてしまう．そのため，測定時間に応じて読み間違える確率が上昇する．

© 2019 Information Processing Society of Japan

2.4.4 バックアクション

初期化～測定を実行したあとに発生するエラーとしてバックアクションが存在する．これは量子力学的な現象で，測定後に量子ビットの状態が反転（$|0\rangle$ ならば $|1\rangle$，$|1\rangle$ ならば $|0\rangle$）することを指している．

先行研究 [10] によると，このエラーは測定マイクロ波の強度に応じて発生確率が上がり，マイクロ波強度が低い場合は，式 (4) に従った確率で反転することがわかっている．このとき，τ は測定マイクロ波の周波数である．

$$P = 1 - \mathrm{e}^{(-\tau/T_1)} \tag{4}$$

2.5 量子ビットの初期化方法

超伝導量子ビットは緩和時間によって刻一刻と状態が変化していくため，ある一定の状態へ遷移させることは難しい．しかし，量子ビットを使用して計算を行うためには，量子ビットの初期値は極めて重要な要素である．そのため，デバイスレベルの実験では様々な初期化方法が発案，実証されている．なお，ここでの初期化とは量子ビットの状態を $|0\rangle$ にする行為に相当する．

量子ビットは，測定を終えると状態が $|0\rangle$ もしくは $|1\rangle$ の 2 通りに確定する．そのため，状態が $|1\rangle$ である場合に $|0\rangle$ へと遷移させる行為を初期化と定義する．本研究では超伝導トランズモン量子ビットを初期化するための方法を 3 つ取り上げる．

2.5.1 パッシブリセット

超伝導量子ビットには緩和時間が存在し，時間とともに基底状態へと戻っていくことは第 2.2 節でも述べた．パッシブリセットは，この性質を利用して状態を初期化する方法である．

測定後の状態が $|1\rangle$ であるとき，その状態は時定数 T_1 で $|0\rangle$ に近づいていく．図 7 は測定後の状態 $|1\rangle$ の変化を表す減衰曲線 $\mathrm{e}^{(-t/T_1)}$ である．縦軸は量子ビットの状態 (重ね合わせの重み，縦軸の中央であれば $|0\rangle : |1\rangle = 1 : 1$)，横軸は時間を表す．時間を T_1 刻みにすると，5 T_1 を超えた時点で量子ビットの状態は 99%以上 $|0\rangle$ になる ($\mathrm{e}^{-5} = 0.006737...$)．そこで，測定後に 5 T_1 待機することで状態を $|0\rangle$ に初期化する．

また，本研究ではエラーにより量子ビットの状態が $|0\rangle$ になっている割合を fidelity(忠実度) と定義する．パッシブリセット作用後の量子ビットの fidelity は，$fidelity = 1 - \mathrm{e}^{(-t/T_1)}$ である．

2.5.2 反転リセット

初期化をしなければならない条件が，測定後の状態が $|1\rangle$ である場合，初期化は状態 $|1\rangle$ を $|0\rangle$ に遷移させる処理に置き換えることができる．すなわち，量子ビットの状態をブロッホ球の XY 平面で面対称に反転させる量子ゲート演算を行い，$|1\rangle$ を $|0\rangle$ に遷移させる．この初期化方法を反転

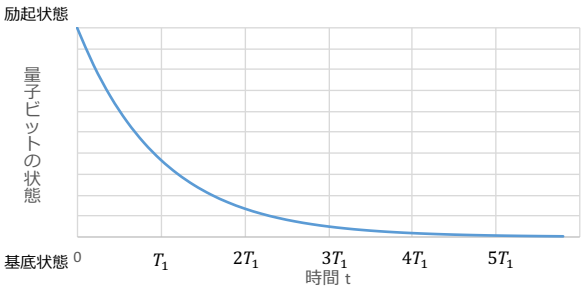


図 7 T_1 による減衰曲線

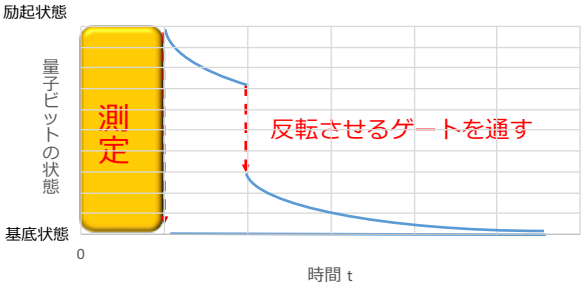


図 8 反転リセット

リセットと定義する．

この方法はパッシブリセットのように 5 T_1 待つ必要がない．しかし，大きな問題が 2 つ存在する．

第一に，測定後から反転させるまでに時間がかかると，$|1\rangle$ の状態が $|0\rangle$ に近づいてしまうため，反転後に $|0\rangle$ に近い状態にならない場合がある (図 8)．すなわち，測定にかかる時間を ob，反転にかかる時間を in とした時の fidelity は式 (5) の様になる．

$$fidelity = exp(-\frac{ob + in}{T_1}) \tag{5}$$

第二に，第 2.4.3 節で述べた読み出しエラーの影響を受ける点である．読み出しエラーが生じてしまうと $|1\rangle$ を $|0\rangle$ とみなしてしまうため，状態が $|1\rangle$ であるにもかかわらず初期化を実行しない場合がある．そしてその逆の場合もまた存在する．

2.5.3 アクティブリセット

従来，量子ビットの初期化は第 2.5.1 節のパッシブリセットを使用していた．しかし，その初期化方法ではレイテンシが 5 T_1 かかる上に，初期化エラーが発生する問題がある．そのため，エラーが少なくレイテンシも短い初期化方法が，多くの研究者から発案されている．受動的に初期化するパッシブリセットとは違い，能動的に初期化することから，それを総称してアクティブリセットと呼ぶ．すなわち，反転リセットも一種のアクティブリセットと呼ぶことが可能である．アクティブリセットには，マイクロ波の側波帯冷却 [11] やパーセル効果を用いた初期化 [12] が報告されている．本研究では，2 種類のマイクロ波を用いて初

© 2019 Information Processing Society of Japan

期化をする DDROP[13] を使用した初期化をアクティブリセットとして採用した．この初期化方法は，レイテンシが $3\mu s$ 以下，fidelity が 99.5%以上であると報告されている．

しかし，アクティブリセットはそのどれもが初期化を実行するために新たなハードウェアが必要になる．多大なコストやハードウェア実装の難しさが，アクティブリセットの問題点である．

3. 実環境を想定した量子ゲート方式コンピュータの評価環境

3.1 量子コンピュータ向け評価環境の現状と問題点

アーキテクチャレベルでの研究において実機を用いた各種実験は容易でなく，多くの場合でシミュレータが用いられている．そこで本節では，本研究で必要となるシミュレーション環境の構築に向けた基本方針を議論する．

本研究で使用するシミュレータに求める条件として 2 つが挙げられる．第一に実行速度の速さである．量子ビットは 1 つで同時に 0 と 1 の情報を持つことが可能であるため，その情報量も量子ビット数に対して指数的に増加する．そのため，通常のシミュレータでは十数量子ビットの計算を行うだけで，膨大な時間を費やしてしまう．エラーの影響まで計算を実行する場合は，さらにそれは顕著になる．また，情報量が指数的に増加していくため，メモリ容量の問題も発生する．メモリの上限を超える情報量は扱えないため，そこが一つの閾値となってしまう．

第二に，エラーの考慮である．エラー環境下での評価をするためには，エラーの影響への考慮が不可欠である．しかし，多くのシミュレータは純粋な量子ビットを想定しており，エラーの影響を考慮していない．そこで，上記の条件を最も満たしているシミュレータである Intel-QS [6] をベースに，必要なエラーのモデルを内部に実装する手法を採用した．第 3.2 節では，今回使用した Intel 社が提供する Intel-QS について解説し，第 3.3 節で評価環境の構築について説明する．

3.2 Intel-QS 概要

本節では，実験で使用した評価環境の元となった Intel-QS について紹介する．Intel-QS は Intel 社が提供する，1 入力量子ゲートと 2 入力の制御ゲートをシミュレート可能な C++と MPI で実装された量子シミュレータである [6]．最大の特徴は，並列分散処理によりシミュレーションの高速化を可能にしている点にある．入力は量子ビット数と量子回路が基本であり，デコヒーレンスの影響を計算する場合には T_1 と T_2 を入力することも可能である．なお，量子回路は GUI での操作はできず，量子ゲートを作用させる関数を書き並べるようにして入力する．図 9 は，入力に図 3 の量子回路を設定した場合の出力である．このときデコヒーレンスの影響は無視している．Intel-QS の出力は測定時に

```
permutation:  0 1
qubits=[
        0.70710678    + i * 0.00000000      % |00> p=0.500000
        0.00000000    + i * 0.00000000      % |10> p=0.000000
        0.00000000    + i * 0.00000000      % |01> p=0.000000
        -0.70710678   + i * 0.00000000      % |11> p=0.500000
]; % pcum = 1.000000
```

図 9 2 量子ビットの出力例 (量子回路は図 3 を参照)

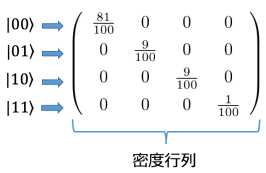


図 10 (1) における密度行列

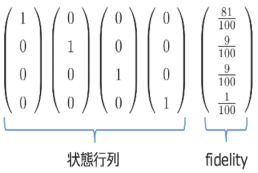


図 11 (2) における状態行列

各量子ビットがとり得る状態の確率を表しているため，図の結果は測定時に 00 もしくは 11 が 50%ずつの確率で現れることを示している．

3.3 評価環境の構築

本節では，各初期化方法とエラー，レイテンシの関係性を評価するため，Intel-QS に拡張案を解説する．

今回拡張する内容は，大きく分けて 3 つ存在する．初期化方法に関するパラメータの実装，初期化による fidelity や読み出しエラーなどによる影響の追加，そしてシミュレータの入出力追加である．

3.3.1 初期化方法パラメータの実装

第一に初期化方法によるエラーのモデルと，レイテンシを計算するモデルの実装について解説する．fidelity の実装においては (1) 量子 ビットの状態計算に密度行列を使用する方法，ならびに (2) 量子ビットの初期状態を全パターン用意しそれぞれに量子ゲート演算を実行する方法が考えられる．(1) における密度行列とは，対角成分に fidelity の情報を含むことが可能な量子状態を表現する行列である．これにより，量子ビット数 n に対して $2^n \times 2^n$ の行列演算を実行することで，エラーの影響を考慮した量子状態を計算可能である．一方，(2) は fidelity の情報を行列計算に組み込むことができないため，量子状態とは別に fidelity の情報を用意しなければならない．図 10，11 に各方法で必要な行列の初期状態の例を示す（図は 2 量子ビット，fidelity90%のとき）．図のように，(2) の方が (1) よりも 2^n 行の縦ベクトル分のメモリ領域を使用するデメリットが存在する．しかし，密度行列を使用すると量子ゲート演算の演算数が 2 倍に増加する欠点があり，また本研究で使用する Intel-QS の仕様上 (2) の方が実装が容易であるため，今回は (2) の方法を採用した．なお，本シミュレータでは，初期化前の量子ビットの状態は純粋な $|1\rangle$ であると仮定し，使用するすべての量子ビットを初期化するモデルとなっている．

© 2019 Information Processing Society of Japan

- 初期化方法の選択
  - パッシブリセット
  - 反転リセット
  - アクティブリセット
- 各初期化方法のパラメータ
  - パッシブリセット，反転リセットの実行時間
  - パッシブリセット，アクティブリセットのfidelity
- T_1, T_2
- 測定時間
- 読み出しエラー，バックアクションの有無
- バックアクションの確率
- 1入力，2入力量子ゲート演算の実行時間
- 反復回数

図 12 設定ファイルの内容

3.3.2 エラーモデルの実装

次に，各エラーによる影響の考慮について説明する．ここでは，初期化時に発生するエラーと，初期化以外で生じるエラーを個別に取り扱う．

- 読み出しエラー：測定時間に応じた確率で測定後に 1 を 0 に反転
- バックアクション：発生した場合は測定されたビットが反転
- 初期化エラー：測定される前の状態を初期化の fidelity に応じて選択
- デコヒーレンス：Intel-QS に実装済み

3.3.3 シミュレータ入出力

最後に，シミュレータの入出力について解説する．大きな追加点として，レイテンシとエラー用のパラメータ追加と最終的なレイテンシの出力である．ここでのレイテンシとは，量子コンピュータが最初の初期化を実行し始めてから最後の測定を終了するまでの時間と定義する．基本パラメータは量子ビット数と量子回路のみとし，その他のパラメータは設定ファイルにまとめることでプログラムの可読性の向上を図った．図 12 に，作成したシミュレータの設定ファイルに入力可能なパラメータを示す．また，パッシブリセットと反転リセットは入力パラメータによっては fidelity が変化する可能性があるため，出力に初期化による fidelity を追加している．今回のシミュレータでは，出力するレイテンシは初期化，量子ゲート演算，測定の 3 つの処理時間を合計し，それを反復回数倍している．このとき，各初期化方法や各種ゲート演算，測定の待機時間も設定ファイルから入力する．

4. エラー環境下における計算機性能の評価

4.1 実行環境

本研究では，計算機に本学情報基盤研究開発センターが保有するスーパーコンピュータ ITO [14] を使用した．また，使用するシミュレータは MPI ライブラリを用いた並列処理を実行するため，インテルの C++コンパイラ `mpiicpc` を使用した．

4.2 評価に用いたデバイスパラメータ概要

本研究で使用したシミュレータの設定パラメータは以下の通りである．

- T_1：50 μs，T_2：20 μs
- 測定時間：965 ns
- 読み出しエラー：あり
- バックアクション：あり
- バックアクションの確率：1%
- 1 入力量子ゲート演算の実行時間：25 ns
- 2 入力量子ゲート演算の実行時間：140 ns
- 反復回数：1024 回（第 4.6 節の実験では可変）

なお，T_1 と T_2 の値は IBM が公開している量子コンピュータの値を参考にした．また，測定時間は先行研究 [15] で使用された時間を採用した．

次に，初期化方法に関連するパラメータについて詳細を説明する．各初期化方法とそのパラメータは以下の様に設定した．

- パッシブリセット
  - 実行時間：実験によって可変（0.7 T_1〜5 T_1）
  - 最大 fidelity：90%
- 反転リセット
  - 実行時間：1 μs
- アクティブリセット
  - fidelity：99.5%

4.3 評価に用いた量子回路

評価の際に入力として与える量子回路は，全ビットを $|1\rangle$ に反転，その後量子フーリエ変換，最後に逆量子フーリエ変換を実行するものとした．量子フーリエ変換とは，量子ゲート演算によりフーリエ変換を実行すること指し，その量子回路は図 13 のように表される．この図における R_k ゲートは，線でつながった量子ビットを制御ビットとして，$|1\rangle$ の場合のときのみ式 (6) のゲート演算を実行する．また，逆量子フーリエ変換は量子フーリエ変換の入力と出力を入れ替え，各ゲートをユニタリ逆変換行列 (転置かつすべての要素を共役な複素数にする) に置換した量子回路で表す．

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{\pi}{2^k}} \end{pmatrix} \tag{6}$$

この量子回路を選択した理由は，第一に出力が量子フーリエ変換回路への入力と等しくなるため正答がわかりやすい点，第二に 1 入力量子ゲートと 2 入力量子ゲートを多数使用している点である．さらに量子ビット数は，第 4.5 節，第 4.6 節で述べる実験ともに 5 とした．そのため，そ

© 2019 Information Processing Society of Japan

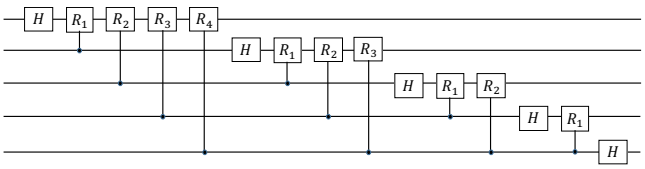


図 13 5 量子ビットの場合の量子フーリエ変換の量子回路

れぞれの実験で用いた量子回路の正答は 31（2 進数表記で 11111）である．

4.4 確率分布推定

本研究の目的は，量子コンピュータの出力である確率分布の推定である．ここでは，確率分布の推定方法として区間推定について解説する．

確率分布を推定するにあたって，量子コンピュータの出力分布におけるそれぞれの値を二項分布とみなした．二項分布とは式 (7) で与えられる分布で，ある事象（生起確率 p）を n 回試行した時の発生回数 x が従う分布を表したものである．今回は量子コンピュータの各出力データに対して，そのデータが出現するかしないかの二項分布とみなすことで確率分布の推定を行った．

$$f(x) = {}_nC_x p^x (1-p)^{(n-x)} \tag{7}$$

次に，今回の実験で採用した母比率の区間推定について解説する．区間推定とは，優位水準 α（$0 < \alpha < 1$）において，ある統計量の $(1-\alpha) \times 100\%$ 信頼区間を求める手法である [16]．二項分布の区間推定を実装するに当たって，F 分布を利用した Clopper&Pearson の信頼区間 [17] を用いた．各信頼区間の上限値と下限値は下記の数式で表される（F は F 分布の上側確率，下付き文字は自由度を表す，n は実行回数，x はデータの出現回数）．

$$p_{upper} = \frac{(x+1) \times F_{1-\alpha/2}(2(x+1), 2(n-x))}{(n-x) + (x+1) \times F_{1-\alpha/2}(2(x+1), 2(n-x))}$$

$$p_{lower} = \frac{x}{x + (n-x+1) \times F_{1-\alpha/2}(2(n-x+1), 2x)}$$

この区間推定によって，量子コンピュータの出力データの母比率（出現確率）を優位水準 α で計算し，その信頼区間が一定の範囲内（ここでは δ とする）に収まれば母比率が収束したとみなす．この処理を出力分布の各値ごとに二項分布とみなして行い，すべての母比率が収束したことを確認した段階で繰り返し実行を終了することで最小の実行回数を決定した．

4.5 各初期化方法に対する評価

量子コンピュータにおいて，初期化は量子ビットの fidelity を決める上で重要な要因となる．そこで本研究ではまず，1024 回実行における各初期化方法ごとのエラー率とレイテンシの関係を調査し，各初期化方法の特徴を調査した．この際，パッシブリセットの待機時間は 5 T_1 としている．また，反転リセットのみ読み出しエラーが発生した場合は，次の処理結果を破棄するようにシミュレートしている（そのため反転リセットの出力結果数は 1024 より少ない）．図 14 に各初期化方法のレイテンシとエラー率の比較を示している．ここでのエラー率とは，正答である 31（2 進数で 11111）以外の値が出力に現れる確率である．

上記の結果を見ると，デバイスの研究で多く採用されているパッシブリセットは 3 つの初期化方法の中で最もエラー率が高く，レイテンシも長い．エラー率のみを考慮するとアクティブリセットが最も低いが，レイテンシでは反転リセットが最も短い結果となった．すなわち，レイテンシを重要視するのであれば反転リセット，精度を重視するのであればアクティブリセットを選択すべきであることが明らかとなった．しかし，反転リセットとアクティブリセットは物理的な実装という観点で困難さが生じることも考慮しておかねばならない．

4.6 fidelity が計算機性能へ与える影響解析

次に本研究では，fidelity が可変であるパッシブリセットを用いて，fidelity が確率分布の推定に与える影響を解析した．

解析方法としては，第 4.4 節で解説した二項分布の母比率区間推定を用いて検証を行った．この際，区間推定のパラメータ α は $\alpha = 0.05$（5%）とした．また，δ のシンプルな設計として，①$\pm\alpha$ と設定する方法と，②$\pm\beta p$ と設定する方法の二種類が考えられる．ここで，α，β は δ を決めるパラメータであり，p はデータの出現確率である．①のパラメータでは，データの出現確率の大小にかかわらず一定の幅を取るため，確率が小さい場合は推定区間が大きくなりすぎてしまう恐れがあると判断し，②を考案した．本研究では上記①，②の両方に対応するため，$\delta = \pm(\alpha + \beta p)$ とし，α，β には 0.025 を設定した．これらのパラメータを用いて区間推定処理をすべての解に対して行い，最短の実行回数とその時の実行時間，正答率を調査した．また，初期化待機時間は 0.1〜2.4 T_1 の間で 0.1 T_1 刻みに実験を行った．初期化の fidelity が 90%で頭打ちするため，上限を 2.4 T_1 にした．

区間推定に基づく実行回数決定の結果を図 15 に示す．図 15 において重要なのは正答率が比較的高い，待機時間が 1.6 T_1 から 2.4 T_1 の領域である．特に，2.3 T_1 での実行回数は極小であり，670 である．この領域では区間推定に要する実行回数は右肩下がりの傾向にあり，より待機時間が長い，つまり初期化の fidelity が高い時により少ない実行回数で量子コンピュータの出力が従う確率分布を推定可能である．このことから，初期化の fidelity 低下は解の正答率低下のみならず，解分布推定に要する実行回数を増

© 2019 Information Processing Society of Japan

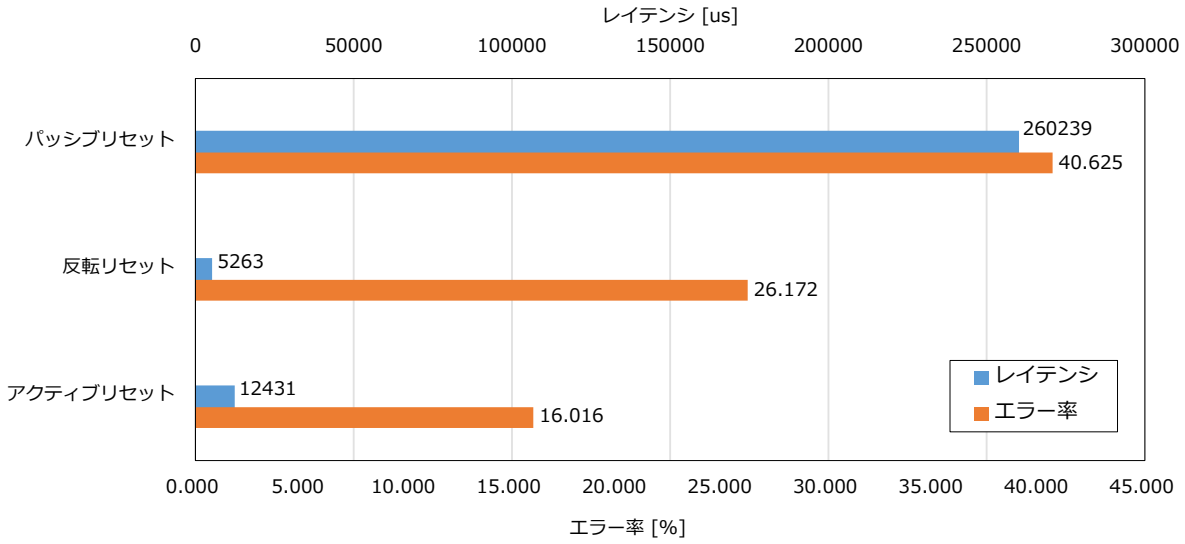


図 14 エラー率とレイテンシ

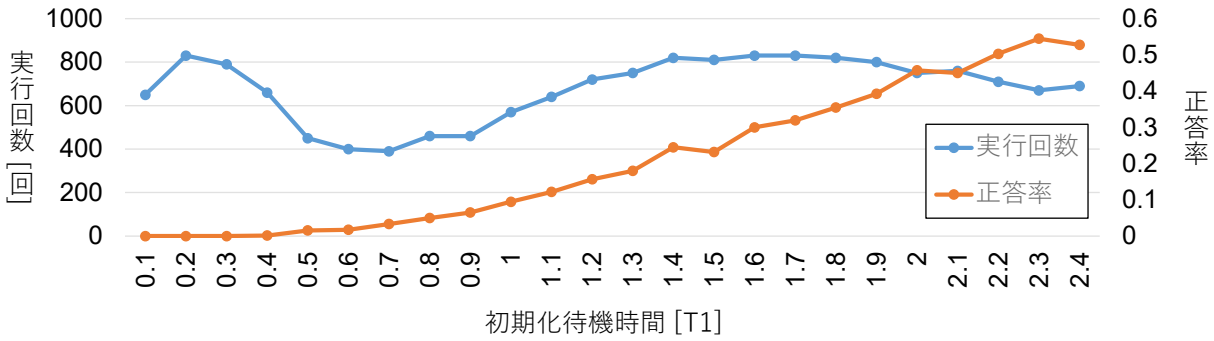


図 15 区間推定に要する実行回数と正答率

加させることが分かる．IBM Q などで標準となっている 1024 回実行と比べると，待機時間 2.3 T_1 では実行回数を 35%削減可能である．

一方，待機時間が 0.1〜1.5 T_1 の領域では区間推定に要する実行回数がより少ない場合があり，0.7 T_1 で最小となる．このとき，正答率はわずか 3%程度であり，これは 1/32 と近いことから解分布がほぼ一様となっていることが分かる．このような場合，発生確率 3%に対し，$\delta = \pm 2.5\%$ であり，δ が相対的に大きな値となることが実行回数減少に寄与していると考えられる．

また，待機時間が 0.1〜0.6 T_1 の領域では初期化後の量子ビットが高確率で想定と異なる状態になっていると考えられる．したがって，出力結果を信用することはできない．

区間推定に要する実行回数は δ を決定するパラメータ α, β に大きく依存する．そのため，デバイスパラメータや回路規模，計算する量子回路などとのさらなる関連調査が必要である．

また，実行回数と 1 回あたりの実行時間の積により得られる区間推定に要する実行時間を図 16 に示す．量子コンピュータ設計時には，実行時間を小さくすることも重要である．許容可能な正答率の中で最も実行時間の短い待機時間の選択が重要である．

5. おわりに

本研究では，量子ゲート方式コンピュータのエラー特性を調査し，既存のシミュレータである Intel-QS を拡張する形で各エラーのモデルを取り入れた実験環境を構築した．その後，量子コンピュータの出力確率分布を推定するまでに必要な実行回数に対する fidelity の影響を解析した．

本研究における量子コンピュータの fidelity と実行回数の解析により，初期化の fidelity は解の正答率のみならず，解分布推定に要する実行時間にも影響することが分かった．現在標準的な 1024 回実行時に比べ，実行時間を 35%削減できる可能性があることが分かった．しかしながら，この結果は量子ビット数が 5，入力量子ゲート数が 120 程度という条件下での結果である．また，区間推定に必要な実行回数は推定に用いるパラメータ値にも大きく依存する．したがって，さらなる量子ビット数や入力量子回路に応じた待機時間と試行回数の評価が必要である．

今後は，上述の課題である量子ゲート方式コンピュータの仕様と解きたい問題から，待機時間と試行回数を短時間で出力できる環境の構築を目指す．また，本研究では量子

© 2019 Information Processing Society of Japan

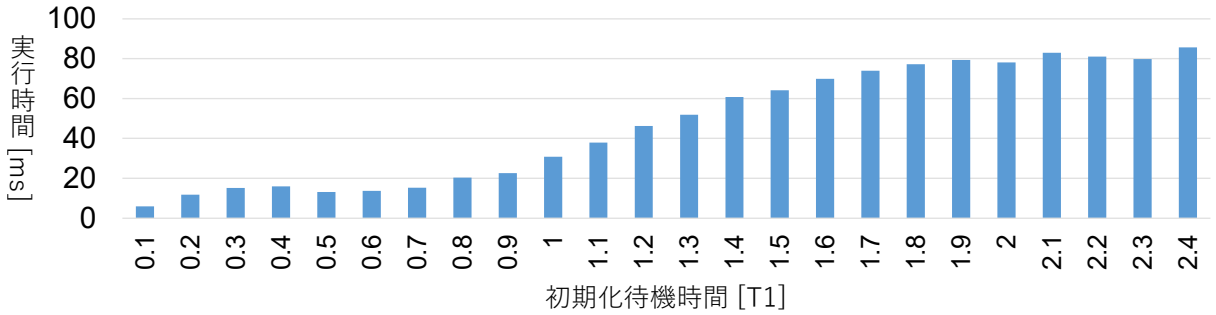


図 16 区間推定に要する実行時間

ゲート方式コンピュータの量子ビット間のトポロジー制約を考慮した評価環境の構築にも取り組む予定である．

謝辞 本研究は一部，JSPS 科研費 19H01105 の助成による．計算機の利用にあたり九州大学情報基盤研究開発センターの協力を得た．

参考文献

[1] Dennard, R. H., Gaensslen, F. H., Rideout, V. L., Bassous, E. and LeBlanc, A. R.: Design of ion-implanted MOSFET's with very small physical dimensions, *IEEE Journal of Solid-State Circuits*, Vol. 9, No. 5, pp. 256–268 (1974).
[2] Moore, G. E.: Cramming more components onto integrated circuits, *Electronics Magazine*, Vol. 38, No. 8, pp. 114–117 (1965).
[3] 大関真之，西森秀稔：量子アニーリング，日本物理学会誌， Vol. 64, No. 4, pp. 252–258 (2011).
[4] 宮野健次郎， 古澤明：量子コンピュータ入門，Vol. 2, 日本評論社 (2016).
[5] IBM: IBM Q, IBM（オンライン），入手先〈https://www.research.ibm.com/ibm-q/〉（参照 2018-12-15）.
[6] Smelyanskiy, M., Sawaya, N. P. D. and Aspuru-Guzik, A.: qHiPSTER: The Quantum High Performance Software Testing Environment (2016).
[7] 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).
[8] 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–万能ディ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).
[9] Cross, A. W., Bishop, L. S., Smolin, J. A. and Gambetta, J. M.: Open Quantum Assembly Language (2017).
[10] Hatridge, M., Shankar, S., Mirrahimi, M., Schacket, F., Geerlings, K., Brecht, T., Sliwa, K. M., Abdo, B., Frunzio, L., Girvin, S. M., Schoelkopf, R. J. and Devoret, M. H.: Quantum Back-Action of an Individual Variable-Strength Measurement, *Sience*, Vol. 339, No. 6116, pp. 178–181 (2013).
[11] Valenzuela1, S. O., Oliver, W. D., Berns, D. M., Berggren, K. K., Levitov, L. S. and Orlando, T. P.: Microwave-Induced Cooling of a Superconducting Qubit, *Sience*, Vol. 314, No. 5805, pp. 1589–1592 (2006).
[12] Reed, M. D., Johnson, B. R., Houck, A. A., DiCarlo, L., Chow, J. M., Schuster, D. I., Frunzio, L. and Schoelkopf, R. J.: Fast reset and suppressing spontaneous emission of a superconducting qubit, https://doi.org/10.1063/1.3435463 (2010).
[13] Geerlings, K., Leghtas, Z., Pop, I. M., Shankar, S., Frunzio, L., Schoelkopf, R. J., Mirrahimi, M. and Devoret, M. H.: Demonstrating a Driven Reset Protocol for a Superconducting Qubit (2013).
[14] 九州大学情報基盤開発研究センター：スーパーコンピュータシステム ITO，九州大学情報基盤開発研究センター（オンライン），入手先〈https://www.cc.kyushu-u.ac.jp/scp/system/ITO/〉（参照 2019-3-22）.
[15] Mallet, F., Ong, F. R., Palacios-Laloy, A., Nguyen, F., Bertet, P., Vion, D. and Esteve, D.: Single-shot qubit readout in circuit quantum electrodynamics (2009).
[16] 竹村彰通：現代数理統計学，創文社 (1991).
[17] Clopper, C. and Pearson, E.: The use of confidence or fiducial limits illustrated in the case of binomial, *Biometrika*, Vol. 26, pp. 404–413 (1934).

© 2019 Information Processing Society of Japan