ノイズがある量子振幅推定法の解析と実験

2021 年度

田中智樹

学位論文　博士（工学）

ノイズがある量子振幅推定法の解析と実験

2021 年度

慶應義塾大学大学院理工学研究科

田中智樹

要旨

本論文は, 全 7 章により構成される.

第 1 章では, 本論文の研究背景と構成について説明する. 特に, 本研究で用いる量子振幅推定法の応用面での重要性, そして現在の量子コンピュータで実行しようとした場合の課題について大まかに述べる.

第 2 章では, 本研究で用いる量子情報理論における基本事項について説明を行う.

第 3 章では, まず Brassard らによって提案された量子振幅推定法を説明する. これは, 現在の古典的なコンピュータを用いる場合より計算量において二乗の優位性があることが知られており, 量子コンピュータを用いることの優位性が保証されている手法の一つである. そして, この手法は振幅増幅法と位相推定法という二つの方法から構成されており, 幾つもの量子ビットや多数のゲート操作の数が必要となることが知られている. 他方, 現在の量子コンピュータは, NISQ と呼ばれる限られたハードウェアしかなく, 振幅推定法の実行が困難である. そこで, 振幅増幅法を並列的に実行し, 量子コンピュータから得た結果を最尤推定で処理することで, 位相推定法を用いない振幅推定法を提案する. この最尤推定法を用いた振幅推定法では, 量子ビットや ゲート操作の数等の必要なリソースを大幅に減らすことができ, 現在の量子コンピュータにおける振幅推定法の活用の第一歩を進めたものである. そして, この手法の有効性はシミュレーションにより検証する.

第 4 章では, 第 3 章で述べた最尤推定法を用いた振幅推定法を実際の量子コンピュータで実行する場合を述べる. 現在の NISQ と呼ばれる実機では, ノイズがあるため, そのノイズによる影響を考える必要があるが, 実際の量子コンピュータで起きているノイズを特定することは困難である. そこで, 本章では, 脱分極ノイズを仮定したモデルを構築し, たとえノイズがあったとしても量子コンピュータの優位性を実現できる可能性について述べる. そして, 実際の量子コンピュータを用いた結果からこのモデルが相応に妥当であることを述べる. また, 推定するパラメータだけでなく, ノイズの大きさも未知のパラメータとすることで, 複数のパラメータ推定問題となり, これにより生じる問題も取り扱う.

第 5 章では, 第 4 章で述べた脱分極ノイズがある場合の, 振幅推定法の推定精度を改善する手法を述べる. 具体的には, これまでの第 3 章および 第 4 章で用いた演算子とは別の基底で作用する振幅増幅演算子を導入する. すると, 従来の演算子では達成できない推定精度の下限を理論的に達成できる可能性があることを示す.

第 6 章では, 第 4 章で述べた脱分極ノイズモデルを拡張したモデルを導入し, より一般的なノイズを扱うことができるようにする. モデルの拡張に際し, パラメータを増やすことを行うが, 単に増やすだけでは興味のあるパラメータを推定することができなくなる. そこで, 元の量子回路と同程度のノイズを受けると想定される補助回路を用意することで, パラメータの推定を可能にする. しかしながら, 一般に, 複数パラメータの最尤推定, つまりは, 多次元の最適化問題を解くことは困難である. そこで, パラメータ直行化法という手法を用いることで, 複数パラメータの最尤推定問題を直接解かず, 一次元の最尤推定問題に帰着させることができる. 特に, パラメータ直交化法を用いる際には条件式となる微分方程式を解く必要があるが, 本問題ではこの方程式が解析的に解けることを示す. 最後に, 本手法をシミュレーションと量子コンピュータの実機を用いて検証する.

第 7 章では, 本論文のまとめと展望を行う.

目次

第 1 章　序章 1

1.1　背景 1

1.2　量子振幅推定法 2

1.3　本論文の構成 7

第 2 章　基本的な事項の説明 9

2.1　Dirac の表記法 9

2.2　量子ビット 9

2.3　量子ビットに対する基本演算 10

2.4　最尤推定および Fisher 情報量 11

第 3 章　最尤推定を用いた量子振幅推定 13

3.1　量子振幅推定について 13

3.2　最尤推定を用いた量子振幅推定アルゴリズムについて 16

3.3　数値実験 19

3.4　計算量に関わる議論 20

3.5　モンテカルロ積分への適用 20

3.5.1　振幅推定を用いたモンテカルロ積分 21

3.5.2　三角関数を用いたモンテカルロ積分の例 22

3.6　本章のまとめ 23

第 4 章　ノイズ下における量子振幅推定 27

4.1　脱分極ノイズのある量子振幅推定 28

4.2　脱分極ノイズ下での, 実現可能な推定誤差 31

4.3　実デバイスを用いた実験 34

4.3.1　２量子ビットを用いた実験 34

4.3.2　3 量子ビットを用いた実験 37

4.4　例外的な変数についての議論 38

4.4.1　例外的な変数について 38

4.4.2　例外的な変数の出現頻度 39

4.4.3　例外的な変数の回避方法 40

4.5　ハードウェアの見積もり 42

4.6　計算量に関する議論 45

4.7　本章のまとめ 45

第 5 章　修正した量子振幅増幅演算子 47
5.1　修正した量子振幅増幅演算子 47
5.2　脱分極ノイズがある場合 49
5.3　量子 Fisher 情報量について 51
5.4　数値実験 54
5.5　本章のまとめ 55

第 6 章　ノイズパラメータを推定しない量子振幅推定 59
6.1　パラメータ直交化法について 60
6.2　補助回路の導入 61
6.3　パラメータ直交化法を用いた振幅推定法 64
6.4　数値実験および実機を用いた検証 65
6.4.1　数値実験 65
6.4.2　量子コンピュータ実機を用いた実験 68
6.5　本章のまとめ 69

第 7 章　結論 71

謝辞 75

付録 77
A　別の 2 量子ビットを用いた実験結果 77
B　パラメータ直交化法の条件式の導出 79

参考文献 81

図目次

1.1 従来の振幅推定手法の量子回路 2
1.2 制御された振幅増幅演算子を用いない振幅増幅回路 3
1.3 実機を用いた振幅増幅の結果 4
1.4 修正した振幅増幅演算子の回転の概要 5

3.1 量子振幅増幅演算子の回転 14
3.2 従来の振幅推定手法の量子回路 (図 1.1 再掲) 15
3.3 最尤推定を用いた手法のイメージ図 17
3.4 制御された振幅増幅演算子を用いない振幅増幅回路 (再掲) 17
3.5 最尤推定を用いた振幅推定法のシミュレーション結果 19
3.6 モンテカルロ積分を, 振幅増幅のみで行う回路 22
3.7 モンテカルロ積分を, 従来の位相推定を用いる振幅推定で行う回路 22
3.8 演算子 $\mathcal{T}$ を実装した回路 23
3.9 制御演算子を用いない振幅増幅の回路 ($m_k = 1$ の時) の例 23
3.10 制御演算子を用いた振幅推定の回路 ($m_k = 1$) の例 23

4.1 実機を用いた振幅増幅の結果（再掲） 27
4.2 脱分極ノイズがある場合の A の呼び出し回数と推定精度の関係 32
4.3 脱分極ノイズがある場合のノイズの大きさ κ と推定精度の関係 33
4.4 2 量子ビットを用いた場合の回路 ($m_k = 1$ の場合) 34
4.5 実機を用いた振幅増幅の結果（再掲） 35
4.6 2 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係 36
4.7 3 量子ビットを用いた場合の回路 ($m_k = 1$ の場合) 37
4.8 3 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係 37
4.9 例外的な場合のノイズの大きさ κ と推定誤差の関係 39
4.10 例外的な場合の, 目標の値と, ノイズの大きさ κ と推定誤差の関係 40
4.11 例外的な場合の推定誤差と, 例外度合いの関係 41
4.12 例外的な場合のノイズの大きさ κ と推定誤差の関係 (m_k を変えた場合) 42

5.1 修正した振幅増幅演算子の回転の概要 (再掲) 48
5.2 演算子 A の呼び出し回数と Fisher 情報量の関係 50
5.3 演算子 A の呼び出し回数と量子フィッシャー情報量の関係 53
5.4 修正した振幅推定法の数値シミュレーションの結果 56

6.1 振幅増幅回路と補助回路に対する振幅増幅回数を増やした時の $|1\rangle$ の出る割合 61

6.2 振幅増幅されない演算子 R および振幅増幅演算子 G の回路図 62
6.3 演算子 R を用いた補助的な振幅増幅回路 $RG^{m_k-1}\left|\Psi\right\rangle_{n+1}$ および振幅増幅回路 $G^{m_k}\left|\Psi\right\rangle_{n+1}$. 63
6.4 パラメータ直交化法を用いた後の対数尤度関数 66
6.5 パラメータ直交化法を用いた後の対数尤度関数 (拡大図) 67
6.6 パラメータ直交化法を用いた数値シミュレーション結果 67
6.7 パラメータ直交化法を用いた実機での検証の結果 69

A.1 2 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係 78

表目次

1.1 主な量子アルゴリズムの一覧 2
1.2 振幅推定法の比較の概要 3

3.1 振幅増幅の回数 m_k の種類と標準偏差の一覧 18
3.2 数値シミュレーションにおけるパラメータの一覧 20
3.3 m_k に応じた計算量の一覧 21
3.4 既存手法と提案手法における CNOT ゲートの数と量子ビットの数の比較 24
3.5 振幅推定法の比較の概要 (再掲) 24

4.1 2 量子ビット (実機) を用いた際のパラメータ一覧 35
4.2 例外的な場合の発生割合 42
4.3 ハードウェアの見積もりの各数値の一覧 44

5.1 修正した振幅推定法の数値シミュレーションのパラメータ一覧 55

6.1 2 量子ビット (実機) を用いた際のパラメータ一覧 68

A.1 追加実験におけるパラメータ一覧 77

第 1 章

序章

1.1 背景

次世代のコンピュータの一つとして, 昨今, 量子コンピュータに注目が集まっている. 歴史的には, 1982 年に Richard Phillips Feynman が量子コンピュータの可能性を指摘し [1], 1985 年に David Deutsch が量子コンピュータを提案した [2]. そして, 1992 年には David Deutsch と Richard Jozsa によって, Deutsch–Jozsa のアルゴリズム [3] が提案され, 既存の決定論的古典アルゴリズムより早い量子アルゴリズムが発見された. そして, 1994 年には Peter Williston Shor が Shor のアルゴリズムを提案した [4]. これは素因数分解や離散対数問題を多項式時間で解けるアルゴリズムの提案である. 素因数分解の困難性は, 現在, RSA 暗号 [5] や楕円曲線 [6, 7] を始めとした様々な暗号方式の計算量的安全性の根拠になっているものであり, 非常に大きな影響があることから, 量子コンピューティングに注目が集まるきっかけの一つとなっている. その後, Andrew Steane[8] や A. R. Calderbank ら [9] によって量子誤り訂正が, Grover によってデータベース探索を効率よく行う Grover のアルゴリズム [10] が, Brassard らによって量子状態に埋め込まれた振幅を効率よく推定する振幅推定手法 [11] など, 様々な量子アルゴリズムが提案されている [12, 13, 14].

他方, ハードウェアの開発も進み, 1999 年には, 中村らによって超電導の量子ビットの実現がなされた [15]. そして, 直近 2016 年には, IBM 社がクラウドベースで超電導量子コンピュータの提供 [16, 17] を始め, 並行して, Google 社 [18], rigetti 社 [19] も超電導量子コンピュータを開発を進めており, Google 社からは量子超越 [18] に関する発表がされている. 更に, 超電導だけでなく, Honeywell 社 [20] や IonQ 社 [21] 等からはイオントラップ方式など, 様々な方式での実装が行われている.

しかしながら, これらのデバイスは, まだノイズがあり, 量子ビットの数も限られている量子コンピュータであり, Preskill は NISQ（Noisy Intermediate-Scale Quantum)[22] と名付けている. 前述の量子コンピュータは, まだ数十量子ビット程度であったが, 2021 年 11 月には IBM 社から 100 を超える量子ビットを持つ量子コンピュータの発表がされ, 日進月歩で発展している. 今後, 大規模化が進み, FTQC（Fault Tolerant Quantum Computer）[23, 24, 25, 26, 27, 28, 29, 30], つまり誤り訂正 [8, 9, 31, 32] された量子コンピュータの実現を目指し開発が進んでいくことが期待されるが, 一方で, 現状のノイズがある量子コンピュータを使うことも重要なテーマの一つである. 将来, 誤り訂正ができるようになり, 論理量子ビットを扱う状況であったとしても, 物理量子ビットではノイズがあり, ノイズに対しての解析が必要なことは明らかである.

前述の通り, 量子コンピュータは, 現状のコンピュータより優位なアルゴリズムがあることが知られいており, 主なものを表 1.1 にまとめる. Shor のアルゴリズム [4] や Kitaev による位相推定アルゴリズム [33], 連立方程式を高速に解く HHL(Harrow-Hassidim-Lloyd) アルゴリズム [34] は指数加速が, Grover のアルゴリズム [10] や振幅推定アルゴリズム [11] は二乗加速することが知られている. これらは, 現状知られている古典的な計算に対しての計算量での優位性が保証されている一方で, これらのアルゴリズムを使うためには, 相応のハー

表 1.1　主な量子アルゴリズムの一覧

#	アルゴリズム名	何を計算するか	古典計算に対しての優位性
1	Shor のアルゴリズム	素因数分解	指数加速
2	位相推定アルゴリズム	固有値計算	指数加速
3	HHL アルゴリズム	連立方程式	指数加速
4	Grover のアルゴリズム	探索問題	二乗加速
5	振幅推定アルゴリズム	振幅推定	二乗加速

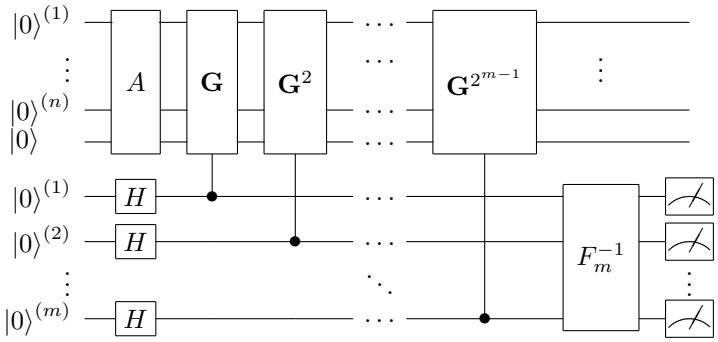


図 1.1　従来の振幅推定手法の量子回路

制御された振幅増幅演算子と, 量子フーリエ変換で構成されており, $n+1+m$ 量子ビットを用いる.

ドウェアリソースつまり, 量子ビットや量子ゲート操作が必要となるため, 現在のノイズのある量子コンピュータで実行することは簡単ではないと考えられている. そこで, 本論文では, 特にノイズのある量子コンピュータで振幅推定アルゴリズムについて述べる.

1.2　量子振幅推定法

前節でもいくつかのアルゴリズムを紹介したが, その中の一つに量子振幅推定法 [11] と呼ばれる手法がある. これは Grover のアルゴリズム [10] を一般化したものであり, 二乗加速することが知られている. そして, 量子振幅推定法は, モンテカルロ積分 [35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47] や機械学習 [48, 49, 50, 51, 52, 53, 54] など様々な応用が提案されている非常に重要なアルゴリズムの一つである. 他方, 量子振幅推定法は, 図 1.1 で示す通り, 量子振幅増幅と量子位相推定の二つのパートが必要とされていた [11]. この手法は, 複数の振幅増幅された結果を量子位相推定を用いて, 一度に取り出すことができる. ただし, それらの回路を同時に実行する必要があるため, 非常に多くのゲートが必要になる. 例えば, [37] では, 金融分野における重要なリスク計算の指標の一つである VaR(Value at Risk) に対する例が幾つか示されており, 高々 10 を超える量子ビット数を利用する場合で, 量子ビット同士が全結合を仮定したとしても, 数千から数万個の CNOT ゲートが必要であり, 量子ビット間の結合が制限される場合には更に数倍のオーバーヘッドがかかることを述べている. これは現在の量子コンピュータでは実行が困難である. [38] では, 実機を用いた場合の結果を示しているが, 振幅増幅の回数は 1 回にとどまっており, 量子コンピュータを用いたメリットを得ることができていない.

そこでアルゴリズム自体を改良していく必要がある. [55] では位相推定を用いないアプローチを取っているが, 繰り返し毎に演算子を変更する必要があり, 現実的には非常に難しいものである. また, [56] では, ベイズ推定を用いてランダムな Haar 測度に従うユニタリ演算子を適用することで, 制御された演算子を用いずに高速化を提案している. ただし, この方式はランダムなユニタリ演算子の実装が難しいという課題があり, こちらも

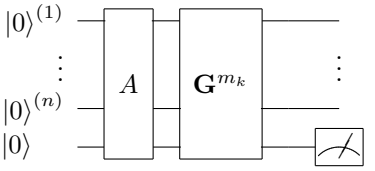


図 1.2　制御された振幅増幅演算子を用いない振幅増幅回路

制御された振幅増幅演算子ではなく, 単なる振幅増幅演算子を用いることで, 大幅な CNOT ゲートを減らすことができる. さらに, 量子フーリエ変換も用いない. m 回の量子回路自体の実行が必要となるものの, 量子ビット自体は $n+1$ 個のみで, 測定する量子ビットも 1 個のみとなる.

表 1.2　振幅推定法の比較の概要

項目	Brassard らの手法	著者らの手法
実行回数	1 回	m 回
演算子 A の呼び出し回数	$N_{\mathrm{shot}} \sum_k m_k$ 回	$N_{\mathrm{shot}} \sum_k m_k$ 回
必要な量子ビット	$n+1+m$ 個	$n+1$ 個
測定する量子ビット	m 個	1 個
必要なゲート数	N_{gate} 個	$0.15 * N_{gate}$ 個

現実的には非常に難しい点がある.

そこで, 量子振幅推定法において更なる改良が, つまり, 量子位相推定を用いないアルゴリズムの開発が行われている [57, 58, 59, 60, 61]. 著者らによる [57] では, 後述するとおり, 最尤推定を用いることで量子位相推定を用いずに量子振幅推定法を行う方法を提案した. 続く, [58] では, 厳密性のある量子フーリエ変換を用いない手法を提案し, [59],[60] では, 更に厳密性のオーバーヘッドが改良されている. 改めて, 本稿でも述べる通り, 著者らによって, 異なる回数の量子振幅増幅を並列的に実行し, その結果を最尤推定法を用いて, 振幅を推定できることを [57] で述べている. これらは量子回路の実行を, 図 1.2 で示す振幅増幅演算子毎に実行する必要がある一方で, ” 制御された”振幅増幅演算子および, 量子フーリエ変換が不要となる. 結果として, 大幅なゲート数の削減になり, [57] で示しているモンテカルロ積分の一例では, 約 85% の CNOT ゲートを削減することができる. 更に, 従来の量子位相推定を用いる手法では, より複雑な計算を行いたい場合, 振幅増幅の数の種類を増やす必要があり, それに応じて量子ビット数を増やす必要があるが, 最尤推定を用いる量子振幅推定では, 並列的に実行することで, 振幅増幅の数の種類を増やしても量子ビットを増やす必要がなくなる. これは非常に有益なことであり, 振幅増幅の数の種類を増やす際に, 量子ビットの数に律速されないことを示しており, 必要な量子ビットの数が非常に少なくて済むことを示している. 簡単にこの比較を表 3.5 にまとめる. ここで, 実行回数に関しては, 提案手法の方が多いが, 一方で計算量を評価する上で用いる演算子 A の呼び出し回数は変わらない, つまり計算量を考える上では全くデメリットにならないことを述べておく.

また, 最尤推定を用いることで, 漸近有効性という性質があるため, 十分に大きな測定回数を行えば, Fisher 情報量から計算される, Cramér–Rao の不等式から導かれる下限, つまり Heisenberg 限界を達成できることが理論的に示される [62]. 一方で, 実際に計算を行う場合, 量子コンピュータを用いるか用いないかによらず, できる限り測定回数を減らしたい, できる限り計算リソースは使わずに計算を行いたいというモチベーションがある. そこで, 最尤推定を用いた振幅推定法を数値シミュレーションを行うことで, 高々 100 回程度で十分に, Cramér–Rao の下限を達成できることを示す.

更に, 現在の量子コンピュータは NISQ と呼ばれている通り, 量子ビットの数が限られており, 量子回路の深さ, つまりは実行できるゲートの数も制限があり, 更にノイズがある. 前述の最尤推定を用いた振幅推定法によ

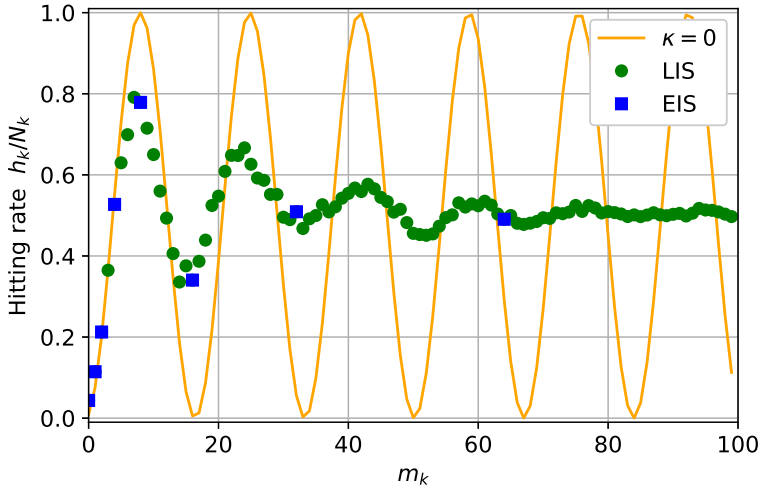


図 1.3　実機を用いた振幅増幅の結果

超電導量子コンピュータ IBM Quantum *ibm_valencia* を用いて振幅増幅を行なった結果. 横軸には振幅増幅の回数 m_k, 縦軸には最後の量子ビットが 1 を観測した確率であり, 橙色がシミュレーションによる結果, 緑色の丸点及び青色の四角点が実機から得られた結果である. 実験の詳細は, 4.3 節で説明を行う.

り, 量子ビットは振幅増幅の数の種類によらず定数化することができ, 量子回路の深さ, つまりゲートの数も大幅に改善することができる. ここで, もう一つの課題として, ノイズの影響を考える必要がある. 例えば, [63] では, Grover のアルゴリズムに対して, 1 量子ビット, 2 量子ビットに対する反転エラーを仮定した研究が行われており, [64] では, 振幅推定手法に対して, いくつかのノイズモデルを仮定して議論を行なっているが, 実機での検証などは行われていない. ただし, これらの研究は, 現実の量子コンピュータを用いておらず, 何かしらのノイズモデルを仮定した上で行われているものである. 一方で, 実際のハードウェアから生じるノイズについて, 完璧なモデリングを行うことは困難である. これはノイズがある量子コンピュータを使う上では考えていかなければならない課題であり, 実際の量子コンピュータを用いて, 検証することが非常に重要なテーマの一つである.

そこで, 著者らによる [65] では, 実際の量子コンピュータ IBM Quantum *ibm_valencia* [16] 及び Qiskit[66] を用いて, 最尤推定を用いた量子振幅推定を行なっており, これを, 本稿で説明する. まず, 非常に一般的かつ, 数学的に扱いやすい脱分極ノイズモデル [12] を導入し, ノイズがある振幅推定法を構成する. この脱分極ノイズは, ある一定の確率で量子ビットが完全混合状態になってしまうものであり, 最小あるいは最悪の場合のノイズモデルとして選択され [67, 68, 69], また, ランダマイズドベンチマーク [70, 71, 72] でも用いられるノイズモデルである. そして, 実機を用いて, 実験を行なった結果を実際の実験結果の図 1.3 に示しており, 横軸には振幅増幅の回数, 縦軸には最後の量子ビットが 1 を観測した確率であり, 橙色がシミュレーションによる結果, 緑色の丸点及び青色の四角点が実機から得られた結果である. ここで実機から得られた結果は, 何かしらのノイズの影響を受けていることがわかると共に, 振幅増幅の回数を増やすほど, ほぼ 1/2 に収束する減衰振動となっていることがわかる. これは, 実際において, 脱分極ノイズが発生しているかどうかはわからないものの, 実験の結果を見る限り, 脱分極ノイズモデルを仮定することが自然であることを示唆している.

なお, 量子コンピュータの状態は, 時刻とともに変化するため, 特に, 実機のデータの解析を行うためには, ノイズの大きさを事前に調べられることを仮定せずに, つまり, ノイズの大きさを示すパラメータも未知のパラメータとして, 推定したい振幅とのマルチパラメータ推定問題として扱う. この問題設定において, つまり脱分

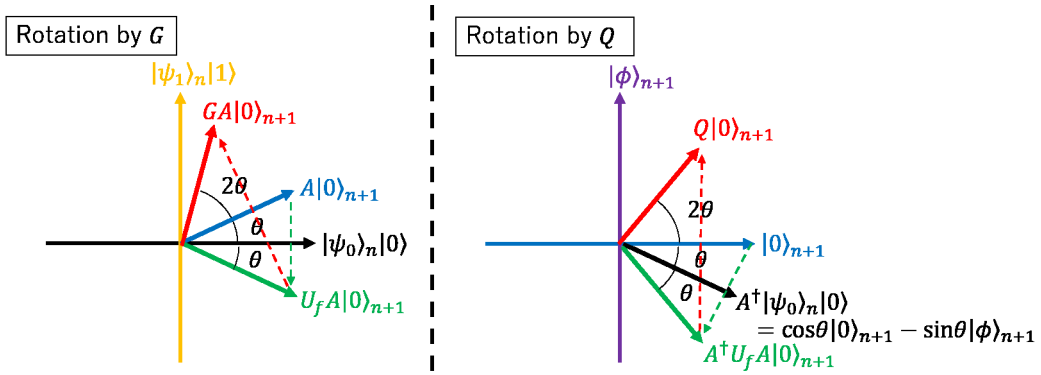


図 1.4　修正した振幅増幅演算子の回転の概要

従来の振幅増幅演算子 G（左側）と修正さいた振幅増幅演算子 Q（右側）の回転の概要図．どちらも初期状態を青色, 演算後を赤色で表し, 2θ だけ回転していることを示している．詳細は第 5 章で説明する．

極ノイズモデルを仮定するものの, ノイズの大きさはわからない状況での, 最尤推定を用いた振幅推定法を考えると, 脱分極性ノイズの数学的に扱いやすい特性により, 推定の精度を議論するためのフィッシャー情報行列の明示的な形式が得ることができ, Heisenberg 限界が達成できる領域等の議論を行うことができる．そして, 実際の量子コンピュータの結果を解析することで, 本モデルが最良ではないかもしれないが, 実験結果を相応に説明することができる, 悪くないモデルであることを示す．また, 推定したい精度が与えられた時に, ほぼ二次的な加速を達成するためのノイズの大きさを決める関係を導くことができる．ただし, 複数のパラメータが未知である問題は一般的に解くことが難しく, 本件では, 例外的なパラメータたちが存在することを示す．例外的なパラメータは, ノイズの大きさが小さくなっても, 振幅の推定精度を改善することができないものであるが, これは高々 1 ～ 2% 程度しか発生せず, また例外的な場合を回避する方法も提案する．

改めて, ここまでのところで, 量子振幅推定に量子位相推定を用いない方法を提案し, 更に脱分極ノイズがある時の議論を行なった．この脱分極ノイズモデルを仮定した振幅推定法は, ノイズが十分に小さい場合は, Heisenberg 限界に近い値を取ることができるものの, 取りうる範囲は限られている．更に, （脱分極ノイズの大きさから導かれる）ある一定回数より大きな回数の振幅増幅を行なったとしても, Fisher 情報量がほとんど増えなくなる, つまり, 振幅増幅が意味をなさなくなってしまう領域がある．当然, ノイズの大きさが小さくなれば, 推定精度を改善できるものの, 基本的にノイズの大きさはハードウェアの性質によって決まっているものであり, ユーザー側からの改善は簡単ではない．そこで, 手法の改善を考える必要があり, 一つの改善手法としては, 振幅増幅の数列 m_k を工夫することである．Brown ら [64] は, $m_k = k$ という等差数列を用いることで, $m_k = a^k$ の場合に比べ推定精度の下限を改善できることを示しており, また, Giurgica ら [73, 74] では, $m_k = k^b$ とし, $m_k = a^k$ の場合に比べ推定精度の下限を改善できることを示すとともに, 更に b を推定精度に応じて決める手法を提案している．どちらの場合も, 古典的な手法に比べては優位性があるものの, Heisenberg 限界である二乗加速を諦めているものである．

そこで, 本稿では, 著者らの [75] でも述べているとおり, 修正した振幅増幅演算子を導入することで, この Heisenberg 限界, つまり二乗加速を前提として, 推定精度の改善することができることを説明する．まず, 修正した振幅増幅演算子 Q を導入する．回転の概要を図 1.4 でも示す通りだが, これまで議論に用いた振幅増幅演算子 $G = -AU_0A^\dagger U_f$ を 少しずらした演算子 $Q = -U_0A^\dagger U_fA$ であり, 同じ空間上での回転ながら, 別の基底で張られる空間を回転する演算子となる．言い換えると, 演算子 G においては, 最後の量子ビットが 0 か 1 かを判断する測定演算子を用いるが, 演算子 Q においては, すべての量子ビットが 0 か否かを判断する演算子を用いて計算を行う．ここで, ノイズがない場合を考えると, 従来の振幅増幅演算子 G を用いた場合も, 修正した

振幅増幅演算子 Q を用いた場合も, 古典 Fisher 情報量は一致する. 更に, 量子 Fisher 情報量も, これに一致する. ここでは, 量子 Fisher 情報量は最適な基底での測定を行なった場合の理論的な限界と捉えると, 従来の振幅増幅演算子 G においても, 修正した振幅増幅演算子 Q においても, 最適な基底での測定ができていることを示している.

一方で, 脱分極ノイズモデルを仮定した場合（この場合はノイズの大きさは既知として扱う）, 従来の振幅増幅演算子 G を用いた場合には, 古典 Fisher 情報量と量子 Fisher 情報量では乖離があるが, 修正した振幅増幅演算子 Q を用いた場合には, 古典 Fisher 情報量が原理的には量子 Fisher 情報量を達成できる. つまり, 従来の手法を原理的に上回ることができることを示している. ただし, 一般には, この量子 Fisher 情報量を得るためには, 適応型の測定が必要であり, 達成できるかはわからないものの, 今回は数値シミュレーションにより, 適用型の測定を用いない場合でも, 修正した振幅増幅演算子 Q を用いた方が, 量子 Fihsr 情報量に近づくことを示す.

また, 前述の通り, 量子コンピュータのノイズは脱分極ノイズだけではなく, 未知のノイズがある. そこで, 著者らの [76] では, 脱分極ノイズモデルを拡張したモデルを導入し, より一般化することを述べる. このノイズがあるシステムにおける問題の難しさの一つは, 完璧なモデルを与えることが不可能であるという事実に基づく. そこで, できる限りモデルを拡張することにより, モデルの表現力を高くし, 実際の量子コンピュータから得られる未知の真の分布と, 仮定するモデルとのギャップを小さくすることが重要である. ここで, 複数の未知のノイズを特徴付けるために, より複雑なモデル, つまり振幅増幅演算子の回数に対してノイズパラメータを導入する. するとパラメータが増えるため, 多次元の最尤推定を行おうとすると, 多次元の最適化問題を解く必要が生じてしまい, 特にノイズパラメータの数が大きいと, これは一般には簡単ではない.

そこでパラメータ直交化法 [77, 78] という手法を用いることで, 多次元の最適化問題を解くことを避ける, つまり 1 次元の最尤推定問題に帰着できる. これは, この問題設定においては, 振幅 θ のみが推定したいパラメータであり, ノイズから起因するパラメータたちは局外パラメータとして, 直接の推定は行わない. なお, 局外パラメータの数について, 無限大の極限を考えると, パラメトリックモデルからセミパラメトリックモデルになり, 無限次元のノイズの効果を暗に捉えることができる. セミパラメトリック推定の文脈でも, 無限次元の関数を除去し, 推定したいパラメータを推定できる可能性がある方法が知られている [79]. さらに, 直近では, パラメータ直行化法の量子版 [80, 78] やセミパラメトリック理論 [81, 82] の研究も行われているが, 本論文では古典的な推定論の活用を行う. その理由としては, 量子コンピュータを考える上でも, 計算結果は, 測定結果として現れるため, つまり, 重ね合わせの状態ではなく, 0 か 1 の古典データとして観測されるためであり, 古典的な統計的な推定法を用いることが重要な切り口の一つである.

なお, パラメータ直交化法では, 推定したいパラメータが一つの時には, 条件となる微分方程式が解ければ, 必ず直交化できることが知られている. ただし, 現在の拡張したモデルを用いた振幅推定の問題に, パラメータ直交化法を用いることを考えるが, Fisher 情報行列の行列式がゼロとなってしまうため, このままではうまく機能しない. そこで, 振幅増幅しない演算子 R を導入することで本状況を回避することを提案する. この演算子 R は, 振幅増幅演算子 $G = -AU_0A^\dagger U_f$ の U_f を恒等演算子に変えた演算子 $R = -AU_0A^\dagger$ である. 実際の Z 基底測定を行う量子コンピュータでは, U_f は Z ゲート一つで実装でき, これはフレームチェンジ [83, 84, 70] で実装されるため, 実際の実機ではパルスを当てることを行わない. つまり, G と R は, ほぼ同様のパルスが当たっていることから, ノイズの大きさは同じと考えられる. そして, これらの合わせた確率を用いて, Fisher 情報行列の行列式がゼロとなる問題を回避し, パラメータ直交化法を用いることができる. これにより多次元の最適化問題を回避し, 1 次元の最尤推定問題に帰着できる. これは非常に重要なことで, 仮に量子コンピュータを用いることで, 古典コンピュータより優位な計算量でできたとしても, 事後処理の部分でオーバーヘッドがかかってしまうと, 量子コンピュータを用いる優位性が消えてしまう可能性がある. この問題をパラメータ直交

化法を用い, 1 次元の最適化問題にすることは, 事後処理でのオーバーヘッドの問題を避けられる. 最後に, 数値シミュレーションおよび量子コンピュータの実機 IBM Quantum *ibm_kawasaki* を用いて, 検証することで, 本手法の有用性を述べる.

1.3 本論文の構成

本論文は全 7 章で構成されており, 本 1 章で導入を行う.

続いて, 第 2 章では, 本論文で必要な基本的な事項を説明する.

次に第 3 章では, [57] にある, 量子位相推定を用いない量子振幅推定法について述べる. まず, 3.1 節で, Brassard らによって提案された量子位相推定を用いる量子振幅推定手法 [11] を説明する. 前述の通り, これは古典的な手法に比べ, 二乗の計算量の優位性があり, モンテカルロ積分や機械学習のエリアでいくつもの応用が提案されている重要な手法である. この従来の手法は, 一度の実行で複数の振幅増幅された結果を量子位相推定を用いることで, 取り出すことができる一方で, 複数の振幅増幅された回路を同時に実行し, 量子位相推定を用いることから, 非常に深い回路が必要となる. 更により複雑な計算を行いたい場合, 振幅増幅の数の種類を増やす必要があり, それに応じて量子ビット数を増やす必要がある. これは, 現在の NISQ[22] と呼ばれる, ノイズがあり, 量子ビットの数も限られている量子コンピュータでは実行が困難である. 当然, ノイズの影響から計算できる回路の深さも限られている. そこで, 振幅増幅を並列的に実行し, 量子コンピュータから結果を別々に取り出した後, それらの測定結果を合わせて最尤推定を行うことで, 量子位相推定を行わない量子振幅推定手法を著者らは [57] で提案し, これを 3.2 節で説明する. そして, 3.3 節では, 数値シミュレーションを行う. 本提案手法は最尤推定を用いることから, 漸近一致性, すなわち試行回数が多い時に Cramér–Rao を達成できることが知られているが, 一方で現実的には有限の試行回数である必要があり, 高々 100 回程度で良いことを示す. 3.4 節では, 計算量に関する議論を行い, 続く 3.5 節では, 実際にモンテカルロ積分の一例を用いて, どの程度, 量子ビット数や量子回路の深さを減らせるかを議論する. そして, 3.6 節では, 本章のまとめを行う.

第 4 章では, 前 3 章で述べた最尤推定を用いる量子振幅推定法を実際の量子コンピュータで実行する場合について考える. これは著者らによって [65] で述べられているものである. まず, 脱分極ノイズモデルを仮定した量子振幅推定法を 4.1 節で構築する. ノイズモデルを導入し, 尤度関数, Fisher 情報行列を計算し, Cramér–Rao の不等式から導かれる下限を数式として与える. 前述の通り, 脱分極ノイズの大きさを表すパラメータ κ は未知として扱い, 複数パラメータの推定問題として定式化を行う. 次に 4.2 節で, 本モデルにおける実現可能な推定誤差について議論する. ここで, ノイズがある場合では, ある振幅増幅の数 m_k より大きい範囲では推定精度をよくできないこと, つまり m_k を大きくしても Fisher 情報行列の逆行列の (1,1) 成分がほとんど大きくならない状況が起こることを示し, 一方で, 限られた範囲であれば, ほぼ Heisenberg 限界を達成できること, ノイズの値が小さくなれば推定精度をよくできることを示す. これは朗報であり, たとえ脱分極ノイズがあったとしても, 量子コンピュータを使うことで, 古典的な手法より良い結果, つまり, 同じ計算量であったとしてもより推定精度が良い結果を得られる可能性を示しているものである. そして, 超電導デバイス実機 IBM Quantum *ibm_valencia*[16] 及び Qiskit[66] を用いて検証を 4.3 節で行う. ここでは, 2 量子ビットと, 3 量子ビットの場合の実験を行う. 2 量子ビットを用いた実験では, 二乗加速には至らないものの, 僅かながら古典的な手法を用いた場合を上回る結果を確認することができた. 一方で 3 量子ビットの場合は, 古典的な手法を上回ることはできなかったが, これは現在のノイズの大きさを鑑みれば自然な結果である. 続く 4.4 節では, ノイズパラメータが未知である時に生じる例外的な場合を議論する. 前 4.2 節では, ノイズの値が小さくなれば推定精度をよくできると述べたものの, ノイズの値が小さくなっても推定精度がよくできない場合, つまり Fisher 情報量が増えない例外的な場合があることを示す. ただし, 例外的なケースは約 $1 \sim 2\%$ 程度しか発生せず, 発生した場合にも回避方法を提供する. ここで, 複数パラメータの推定問題は一般的にかつ本質的に難し

い問題であり, このような派生する問題を議論することは, 非常に価値のあることであると考える. 4.5 節では 4.2 節での議論に基づき, より複雑な計算を行う際の, 必要とされるハードウェアの性能の見積もりを行う. 具体的には, 目標となる推定精度が与えられた際の, 1 量子ビット, 2 量子ビットの忠実度と, 必要な計算時間を与える. これは重要な話題であり, 一つは実際に量子コンピュータで計算を行いたい場合に, どの程度のハードウェアが必要となるかがわかることである. つまり, 今後のハードウェアの進化と共にどのようなことができるようになっていくかを考える上で重要な要素を与える. もう一つは, 時間の観点である. 例え, 量子コンピュータの方が古典的な場合に比べ, 計算量で優位にあったとしても, 例えば 1 ステップあたりの計算時間に大幅な差がある場合は, 問題の大きさによっては, 古典的な手法の方が短い計算時間でできる可能性がある. このように実際に量子コンピュータを使う上では, 重要な議論であると考える. そして, 4.6 節では, 計算量に関わる議論を行う. 本章では, 複数のパラメータ推定問題として定式化しているため, 最尤推定を行う際にオーバーヘッドがかかってしまうが, 逐次的に推定区間を狭めながらやることでオーバーヘッドがかかる問題を解消することができる. 最後に, 4.7 節で本章のまとめを行う. なお, 本コードは, Qiskit community[85] に公開している.

続いて, 第 5 章では, 著者らにより [75] で議論している修正した振幅増幅演算子を用いた, 量子振幅推定法について述べる. まず, 5.1 節で修正した振幅増幅演算子 Q を定義し, ノイズがない場合（この場合はノイズの大きさは既知として扱う）には, 従来の振幅増幅演算子 G と同じ推定精度を出すことができることを示す. 5.2 節では, 脱分極ノイズを仮定した場合の振幅増幅演算子 G,Q の振る舞いを述べる. 特にそれぞれの演算子 G,Q を用いた場合の Fisher 情報量の原理的な限界を考えると, Q の方が大きくなることを示す. そして, 5.3 節では, 量子 Fisher 情報量に議論を拡大する. ノイズがない場合には, G,Q のどちらの振幅増幅演算子を用いた場合でも古典 Fisher 情報量と量子 Fisher 情報量が一致する. つまり, 最適な基底での測定を行えていることを示している. 一方で脱分極ノイズを仮定した場合には, 従来の振幅増幅演算子 G を用いた場合には, 古典 Fisher 情報量と量子 Fisher 情報量では乖離があるが, 修正した振幅増幅演算子 Q を用いた場合には, 古典 Fisher 情報量が原理的には量子 Fisher 情報量を達成できることを示す. つまり, 従来の手法を原理的に上回ることができることを示している. ただし, 一般には, この量子 Fisher 情報量を得るためには, 適応型の測定が必要であり, 達成できるかはわからないものの, 今回は数値シミュレーションにより, 適用型の測定を用いない場合でも, 修正した振幅増幅演算子 Q を用いた方が, 量子 Fihsr 情報量に近づくことを 5.4 節で示す. そして, 本章のまとめを 5.5 節で述べる.

第 6 章では脱分極ノイズを更に拡張したモデルでの議論を行う. これは, 著者らによって [76] で述べられているものである. まず, 本章で用いるパラメータ直交化法 [77, 78] について 6.1 節で説明を行う. パラメータ直交化法は, 局外パラメータつまり, 推定したいパラメータではないパラメータたちであり, 本稿での場合はノイズに由来するパラメータの推定を行わずに, 推定したいパラメータを推定できる手法である. 特に推定したいパラメータが一次元の場合, ある微分方程式を解くことができれば, 必ず直交化が可能であることが知られている. 続いて, 6.2 節で, モデルの拡張と補助回路の導入を行う. 前章までで扱っていた脱分極ノイズモデルを拡張する形で, 振幅増幅の回数に応じてノイズパラメータが変わるものとして導入を行う. ここで単純にパラメータ直交化法を用いると, 拡張したモデルを用いた Fisher 情報行列の行列式がゼロになってしまうため, うまく直交化法を用いることができない. そこで, 非振幅増幅演算子 R の導入および, R を用いた補助回路の構成を行う. これは実際の量子コンピュータを用いた結果を見ても大きな違和感のないものであることを示す. そして, 6.3 節では, 実際に振幅推定法にパラメータ直交化法を適用する. パラメータ直交化法の難しいところは, 多変数の微分方程式を解く必要があるが, 本件では, 解析的な解を示すことができる. その結果, 最尤推定を用いた θ の推定は, $M+1$ 次元から, 1 次元の最適化問題にすることができる. 続く 6.4 節では, シミュレーションと量子コンピュータの実機 IBM Quantum *ibm_kawasaki* を用いて, 本手法の有効性を示す. 本節のまとめを, 節で行う.

最後に, 第 7 章では, 本論文のまとめと展望を行う.

第 2 章

基本的な事項の説明

この章では, 量子情報における基本的な事項の説明を行う.

2.1 Dirac の表記法

本節ではまず, Dirac の表記法を説明する. まず, d 次元の複素ユークリッド空間 $\mathbb{C}^d$ において, 列ベクトルを, $|\psi_a\rangle$ という表記をし, 次で表す.

$$|\psi_a\rangle = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_d \end{pmatrix} \tag{2.1}$$

なお, $a_i \in \mathbb{C}$ である. そして, $\langle\psi_a|$ を, $|\psi_a\rangle$ の複素共役な行ベクトル, つまり

$$\langle\psi_a| = (\overline{a_1}, \overline{a_2}, \cdots, \overline{a_d}) \tag{2.2}$$

で定義する. そして, $|\psi_b\rangle = (b_1, \cdots, b_d)^T \in \mathbb{C}^d$ とするときに, 内積を $\langle\psi_a|\psi_b\rangle$, 行列を $|\psi_a\rangle\langle\psi_b|$,

$$\begin{aligned} \langle\psi_a|\psi_b\rangle &= \sum_{i=1}^{d} \overline{a_i} b_i \\ |\psi_a\rangle\langle\psi_b| &= \begin{pmatrix} a_1 \\ \vdots \\ a_d \end{pmatrix} (\overline{b_1}, \cdots, \overline{b_d}) = \begin{pmatrix} a_1\overline{b_1} & \cdots & a_1\overline{b_d} \\ \vdots & \vdots & \vdots \\ a_1\overline{b_d} & \cdots & a_d\overline{b_d} \end{pmatrix} \end{aligned} \tag{2.3}$$

と書く.

2.2 量子ビット

続いて, 量子ビット (qubit) の定義を行う. 量子ビット $|\psi\rangle$ は, 以下の形で書けるものと定義する.

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle \tag{2.4}$$

ここで, $\alpha_0, \alpha_1 \in \mathbb{C}$ であり, $|\alpha_0|^2 + |\alpha_1|^2 = 1$ を満たすものとする. 特に, $|0\rangle$, $|1\rangle$ は次で定義する.

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{2.5}$$

つまり, $|0\rangle$, $|1\rangle$ は正規直交基底である. そして, 式 (2.4) は, 次のようにも書ける.

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix} \in \mathbb{C}^2 \tag{2.6}$$

ここで, 式 (2.6) から, 量子ビットの状態は, 2 次元複素ベクトル空間のベクトルであることがわかる. そして, この状態が足し合わされた状態を重ね合わせの状態と呼ぶ.

ここで, 実数 θ, ϕ_0, ϕ_1 を用いて, $\alpha_0 = e^{i\phi_0}\cos\theta$, $\alpha_1 = e^{i\phi_1}\sin\theta$ とすることで,

$$\begin{aligned} |\psi\rangle &= e^{i\phi_0}\cos\theta\,|0\rangle + e^{i\phi_1}\sin\theta\,|1\rangle \\ &= e^{i\phi_0}\left(\cos\theta\,|0\rangle + e^{i(\phi_1-\phi_0)}\sin\theta\,|1\rangle\right) \\ &= e^{i\phi_0}\left(\cos\theta\,|0\rangle + e^{i\phi}\sin\theta\,|1\rangle\right) \end{aligned} \tag{2.7}$$

と表せる. ここで, $\phi_1 - \phi_0 = \phi$ と置いた. そして, $e^{i\phi_0}$ は, 測定に影響の与えない位相であることから, 任意の 1 量子ビットの状態を次の式でも書くことができる.

$$|\psi\rangle = \cos\theta\,|0\rangle + e^{i\phi}\sin\theta\,|1\rangle \tag{2.8}$$

ここで, 状態 $|\psi\rangle$ に対して, 正規直交基底 $\{|\varphi_0\rangle, |\varphi_1\rangle\}$ による基底測定を行うと, $|\,|\varphi_0\rangle\,|^2$ を観測する確率は $\langle\varphi_0|\psi\rangle$, $|\varphi_1\rangle$ を観測する確率は $|\,\langle\varphi_1|\psi\rangle\,|^2$ で与えられる. 特に, $|0\rangle$ を観測する確率は $|\alpha_0|^2 = \cos^2\theta$ で, $|1\rangle$ を観測する確率は, $|\alpha_1|^2 = \sin^2\theta$ である.

そして, 2 個の量子ビット系について考える. 1 量子ビット目, 2 量子ビット目の状態をそれぞれ, $|\psi_0\rangle = \begin{pmatrix} \alpha_{00} \\ \alpha_{01} \end{pmatrix}$, $|\psi_1\rangle = \begin{pmatrix} \alpha_{10} \\ \alpha_{11} \end{pmatrix}$ とすると, 合成系は次で与えられるベクトルとして考える.

$$|\psi_0\rangle \otimes |\psi_1\rangle := \begin{pmatrix} \alpha_{00}\,|\psi_1\rangle \\ \alpha_{01}\,|\psi_1\rangle \end{pmatrix} := \begin{pmatrix} \alpha_{00}\alpha_{10} \\ \alpha_{00}\alpha_{11} \\ \alpha_{01}\alpha_{10} \\ \alpha_{10}\alpha_{11} \end{pmatrix} \tag{2.9}$$

そして, n 量子ビット系にも同様に拡張, 測定も前述同様に拡張することで表すこととする.

2.3 量子ビットに対する基本演算

次に, 量子ビットに作用する基本演算を定義する. 1 量子ビットに対して, 作用する演算子は, 2×2 の複素ユニタリー行列で与えられる. なお, 行列 u がユニタリー行列であるとは, $U^T U = I$ であり, U^T は U の転置共役行列, I は単位行列である. ユニタリー行列であることは, 元々の量子状態と, 演算後の量子状態の双方が正規化条件を満たすことから自然に導かれる.

そして, 代表的な行列（ゲート操作）を次に示す. まず, 単位行列 I とパウリ行列を以下で与える.

$$I := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\ X := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},\ Y := \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},\ Z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{2.10}$$

そして, アダマールゲート H を

$$H := \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{2.11}$$

で与える. そして, 制御 NOT(CNOT,CX) 演算を次の行列で与える.

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.12}$$

これは, 1 つ目の量子ビットの状態は変えないものの, 2 つ目の量子ビットは 1 つ目の量子ビットが $|1\rangle$ の時のみ, 反転させるものである.

2.4 最尤推定および Fisher 情報量

続いて, 次章以降で用いる最尤推定 (maximum likelihood estimation) と Fisher 情報量 (Fisher information) について説明する.

まず, 確率変数を x, パラメータを $\boldsymbol{\theta} = [\theta_1, \cdots, \theta_M]^\top$ とし, 確率密度関数を $p(x; \boldsymbol{\theta})$ とする. ここで, 訓練標本 $\mathbf{x} = \{x_j\}_{j=1}^n$ が生起する確率をパラメータ $\boldsymbol{\theta}$ の関数と見なしたものである尤度 (likelihood) を考える. これらの訓練標本が独立同一分布 (independent and identically deistributed; i.i.d.) と仮定すると, 尤度 $L(\boldsymbol{\theta}; \mathbf{x})$ は次の形で与えられる.

$$L(\boldsymbol{\theta}; \mathbf{x}) := P(\mathbf{x}; \boldsymbol{\theta}) = \prod_{j=1}^{n} p(x_j; \boldsymbol{\theta}) \tag{2.13}$$

そして, 最尤推定は, 尤度が最大になるようにパラメータの値を決定する方法である. つまり, 最尤推定によって得られるパラメータ $\hat{\boldsymbol{\theta}}_{\mathrm{ML}}$ は, 次で与えられる.

$$\hat{\boldsymbol{\theta}}_{\mathrm{ML}} := \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}; \mathbf{x}) = \arg\max_{\boldsymbol{\theta}} \ln L(\boldsymbol{\theta}; \mathbf{x}) \tag{2.14}$$

なお, 対数関数は単調増加であることから, 尤度の大小関係は変わらず, (2.14) 式が成り立ち, $\ln L(\boldsymbol{\theta}; \mathbf{x})$ を対数尤度 (log likelihood) と呼ぶ. 対数尤度は, 最大化する関数が足し算の形となり, 計算しやすいため, よく用いられるものである. そして, 最尤推定は, 一致性 (consistency), 漸近不偏性 (asymptotic unbiasedness), 漸近有効性 (asymptotically efficiency), 漸近正規性 (asymptotic normality) という性質を持っており, 扱いやすい推定法であることが知られており, 各性質は次の通りである. なお, 漸近性とは, 訓練標本 n が十分に大きい時, 達成する性質である.

- 一致性　　: 訓練標本が無限大の極限, つまり $n \to \infty$ の時に, 最適な推定量が得られる.
- 漸近不偏性: 訓練標本の数 n が十分に大きい時, バイアスがゼロに収束する.
- 漸近正規性: 訓練標本の数 n が十分に大きい時, 推定量が近似的に正規分布に従う.
- 漸近有効性: 訓練標本の数 n が十分に大きい時, Cramér–Rao の下限を達成する.

ここで, Cramér–Rao の下限について述べる. まず, パラメータが一次元の場合, 任意の不偏推定量 $\hat{\theta}$ の分散 $V(\hat{\theta})$ の下界は, 次の Cramér–Rao の不等式で与えられる.

$$V(\hat{\theta}) = \mathbb{E}\left[(\hat{\theta} - \theta)^2\right] \geq \frac{1}{nJ(\theta)} \tag{2.15}$$

ただし, $J(\theta)$ は Fisher 情報量であり, 次式で定義されるものである.

$$J(\theta) := \mathbb{E}\left[\left(\frac{\partial}{\partial \theta} \ln P(\mathbf{x}; \theta)\right)^2\right] \tag{2.16}$$

なお, $\mathbb{E}[\cdots]$ は, 期待値である.

次に, パラメータが M 次元の場合の Cramér–Rao の下限を述べるために, M 次元の Fisher 情報行列 $J(\boldsymbol{\theta})$ を次で定義する.

$$J(\boldsymbol{\theta}) := \mathbb{E}\left[\left(\frac{\partial}{\partial \boldsymbol{\theta}} \ln P(\mathbf{x}; \boldsymbol{\theta})\right)\left(\frac{\partial}{\partial \boldsymbol{\theta}} \ln P(\mathbf{x}; \boldsymbol{\theta})\right)^\top\right] \tag{2.17}$$

なお, Fisher 情報行列の (i, j) 成分を $[J(\boldsymbol{\theta})]_{i,j}$ と書くと以下で与えられる.

$$[J(\boldsymbol{\theta})]_{i,j} = \mathbb{E}\left[\left(\frac{\partial}{\partial \theta_i} \ln P(\mathbf{x}; \boldsymbol{\theta})\right)\left(\frac{\partial}{\partial \theta_j} \ln P(\mathbf{x}; \boldsymbol{\theta})\right)^\top\right] \tag{2.18}$$

そして, Cramér–Rao の不等式, つまりは, 任意の不偏推定量 $\hat{\boldsymbol{\theta}}$ の分散共分散行列 $\mathrm{Cov}(\hat{\boldsymbol{\theta}})$ の下界は

$$\mathrm{Cov}(\hat{\boldsymbol{\theta}}) = \mathbb{E}[(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T] \geq \frac{1}{nJ(\boldsymbol{\theta})} \tag{2.19}$$

となり, 最尤推定が漸近有効性を持つということは, この Cramér–Rao の下限を漸近的に達成できる, つまり次を満たすことである.

$$n\mathrm{Cov}(\hat{\boldsymbol{\theta}}) \to \frac{1}{J(\boldsymbol{\theta})} \tag{2.20}$$

なお, 正方行列 A, B に対する不等式 $A \geq B$ は, $C := A - B$ が半正定値行列であることである. つまり, 行列 C が半正定値であるとは, 任意のベクトル γ に対して次が成り立つことである.

$$\gamma^\top C \gamma \geq 0 \tag{2.21}$$

第 3 章

最尤推定を用いた量子振幅推定

第 3 章では, [57] にある, 量子位相推定を用いない量子振幅推定法について述べる. まず, 3.1 節で, Brassard らによって提案された量子位相推定を用いる量子振幅推定法 [11] を説明する. 前述の通り, これは古典的な手法に比べ, 二乗の計算量の優位性があり, モンテカルロ積分 [35, 36, 37, 38, 39, 40, 41, 42, 43, 45] や機械学習 [48, 49, 50, 51, 52, 53, 54] のエリアでいくつもの応用が提案されている重要な手法である. この従来の手法は, 一度の実行で複数の振幅増幅された結果を量子位相推定を用いることで, 取り出すことができる一方で, 複数の振幅増幅された回路を同時に実行し, 量子位相推定を用いることから, 非常に深い回路が必要となる. 更により複雑な計算を行いたい場合, 振幅増幅の数の種類を増やす必要があり, それに応じて量子ビット数を増やす必要がある. これは, 現在の NISQ[22] と呼ばれる, ノイズがあり, 量子ビットの数も限られている量子コンピュータでは実行が困難である. 当然, ノイズの影響から計算できる回路の深さも限られている. そこで, 振幅増幅を並列的に実行し, 量子コンピュータから結果を別々に取り出した後, それらの測定結果を合わせて最尤推定を行うことで, 量子位相推定を行わない量子振幅推定手法を著者らは [57] で提案し, これを 3.2 節で説明する. そして, 3.3 節では, 数値シミュレーションを行う. 本提案手法は最尤推定を用いることから, 漸近有効性, すなわち試行回数が多い時に Cramér–Rao を達成できることが知られているが, 一方で現実的には有限の試行回数である必要があり, 高々 100 回程度で良いことを示す. 3.4 節では, 計算量に関する議論を行い, 続く 3.5 節では, 実際にモンテカルロ積分の一例を用いて, どの程度, 量子ビット数や量子回路の深さを減らせるかを議論する. そして, 3.6 節では, 本章のまとめを行う.

3.1 量子振幅推定について

本節では, Brassard らの [11] による量子位相推定を用いた量子振幅推定手法を説明する. まず, 推定する値を $\theta \in [0, \pi/2]$ とする. そして, 全てゼロの $n+1$ 個の量子ビット $|0\rangle_{n+1} = |0\rangle \otimes \cdots \otimes |0\rangle$ に作用するユニタリー演算子 A を次で置くことで, 推定する θ を振幅に埋め込むことができると仮定する.

$$A\,|0\rangle_{n+1} = \sin\theta\,|\tilde{\Psi}_1\rangle_n\,|1\rangle + \cos\theta\,|\tilde{\Psi}_0\rangle_n\,|0\rangle =: |\Psi\rangle_{n+1} \tag{3.1}$$

ここで, $|\tilde{\Psi}_1\rangle_n$ と $|\tilde{\Psi}_0\rangle_n$ は, n 量子ビットの"good"と"bad"の状態であり, この量子状態を $|\Psi\rangle_{n+1}$ と置く. 量子振幅推定の問題とは, この状況で, θ あるいは, $\sin^2\theta$ の値を効率よく推定する問題である. そして, 量子振幅増幅演算子（Grover 演算子）G を次で定義する.

$$G = -A\,U_0\,A^\dagger\,U_f \tag{3.2}$$

ただし, ユニタリー演算子 U_0 および U_f は次で定義する.

$$\begin{aligned} U_0 &= I_{n+1} - 2\,|0\rangle_{n+1}\,\langle 0|_{n+1}\,, \\ U_f &= I_{n+1} - 2I_n \otimes |1\rangle\,\langle 1| = I_n \otimes \sigma_z \end{aligned} \tag{3.3}$$

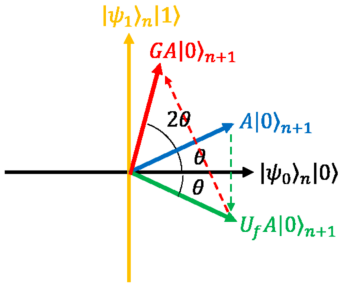


図 3.1　量子振幅増幅演算子の回転

演算子 $G = -A\,U_0\,A^\dagger U_f$ を量子状態 $|\Psi\rangle_{n+1} = A\,|0\rangle_{n+1}$ に作用させることを説明する図である．$A\,|0\rangle_{n+1}$ に U_f を作用させることは，$|\tilde{\Psi}_1\rangle$ 軸で反転させることである．さらに，$-A\,U_0\,A^\dagger$ を作用させることは，$A\,|0\rangle_{n+1}$ を軸として反転させることとなる．結果，$GA\,|0\rangle_{n+1}$ は，$A\,|0\rangle_{n+1}$ からは 2θ 回転していることを示している．

なお，I_n は n 次元の恒等演算子である．そして，U_f は，最後の量子ビットが $|1\rangle$ の時に符号を反転させる，すなわち，”good”状態 $|\tilde{\Psi}_1\rangle_n\,|1\rangle$ に対して符号を反転させる，つまり $U_f\,|\tilde{\Psi}_1\rangle_n\,|1\rangle = -\,|\tilde{\Psi}_1\rangle_n\,|1\rangle$ である．一方で，最後の量子ビットが $|0\rangle$ の時には恒等写像であるため，”bad”状態 $|\tilde{\Psi}_0\rangle_n\,|0\rangle$ には状態を変えない，つまり $U_f\,|\tilde{\Psi}_0\rangle_n\,|0\rangle = |\tilde{\Psi}_0\rangle_n\,|0\rangle$ となる．また，U_0 は $|0\rangle_{n+1}$ の符号を反転する，つまり $U_0\,|0\rangle_{n+1} = -\,|0\rangle_{n+1}$ を満たす．ここで，G を一回 $|\Psi\rangle_{n+1} = A\,|0\rangle_{n+1}$ に作用させた状態を考えると，

$$G\,|\Psi\rangle_{n+1} = \sin(3\theta)\,|\tilde{\Psi}_1\rangle_n\,|1\rangle + \cos(3\theta)\,|\tilde{\Psi}_0\rangle_n\,|0\rangle \tag{3.4}$$

となる．これは，幾何的に捉えると図 3.1 のような回転であり，式 (3.1) の状態で，$n+1$ 番目の量子ビットが $|1\rangle$ である確率が $\sin^2\theta$ であることに対して，式 (3.4) の状態で，$n+1$ 番目の量子ビットが $|1\rangle$ である確率が $\sin^2 3\theta$ となる．そして，G を m_k 回，$|\Psi\rangle_{n+1}$ に作用させた状態 $|\Psi_G(\theta,k)\rangle$ を考えると，

$$|\Psi_G(\theta,k)\rangle := G^{m_k}\,|\Psi\rangle_{n+1} = \sin((2m_k+1)\theta)\,|\tilde{\Psi}_1\rangle_n\,|1\rangle + \cos((2m_k+1)\theta)\,|\tilde{\Psi}_0\rangle_n\,|0\rangle \tag{3.5}$$

となる．ここで，$|\tilde{\Psi}_1\rangle_n\,|1\rangle$ と $|\tilde{\Psi}_0\rangle_n\,|0\rangle$ で張られる空間における G の固有状態 $|\Psi_\pm\rangle_{n+1}$ および固有値 $\lambda_\pm$ は，次のように書くことができる．

$$|\Psi_\pm\rangle_{n+1} = \frac{1}{\sqrt{2}}\left(|\tilde{\Psi}_0\rangle_n\,|0\rangle \mp i\,|\tilde{\Psi}_1\rangle_n\,|1\rangle\right),\quad \lambda_\pm = e^{\pm 2i\theta} \tag{3.6}$$

ここで量子位相推定を行うことで，θ の値を推定できる．つまり，ユニタリ演算子と固有ベクトルが与えられた時に固有値を算出できる．具体的には，式 (3.6) より，

$$|\Psi\rangle_{n+1} = \frac{1}{\sqrt{2}}\left(e^{i\theta}\,|\Psi_+\rangle_{n+1} + e^{-i\theta}\,|\Psi_-\rangle_{n+1}\right) \tag{3.7}$$

と書ける．そして，新たに補助量子ビットを m_k 個用意した状態 $|\Psi\rangle_{n+1}\,|0\rangle_{m_k}$ に，その追加した m_k 個の量子ビットにそれぞれアダマールゲート H を作用させると次の状態となる．

$$|\Psi\rangle_{n+1} \otimes H^{\otimes m_k}\,|0\rangle_{m_k} = \frac{1}{\sqrt{2^{m_k}}}\sum_{x=0}^{2^{m_k}-1}|x\rangle_{m_k}\left(e^{i\theta}\,|\Psi_+\rangle_{n+1} + e^{-i\theta}\,|\Psi_-\rangle_{n+1}\right) \tag{3.8}$$

さらに，追加した m_k 個の補助量子ビットを制御ビットとする制御量子振幅増幅演算子を行うと，

$$\frac{e^{i\theta}}{\sqrt{2}}\sum_{x=0}^{2^{m_k}-1}e^{2ix\theta}\,|x\rangle_{m_k}\,|\Psi_+\rangle_{n+1} + \frac{e^{-i\theta}}{\sqrt{2}}\sum_{x=0}^{2^{m_k}-1}e^{-2ix\theta}\,|x\rangle_{m_k}\,|\Psi_-\rangle_{n+1} \tag{3.9}$$

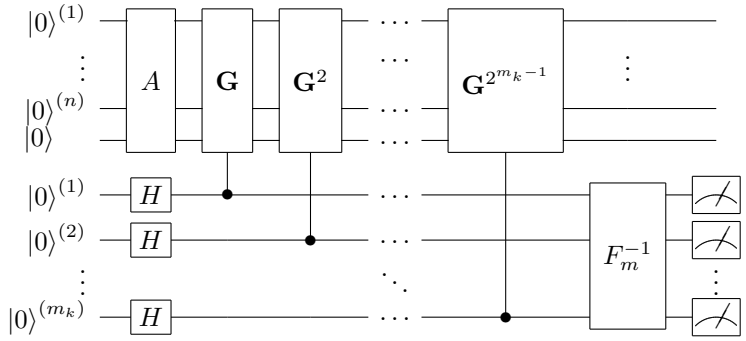


図 3.2　従来の振幅推定手法の量子回路 (図 1.1 再掲)

制御された振幅増幅演算子と, 逆量子フーリエ変換 $F_{m_k}^{-1}$ で構成されており, $n+1+m_k$ 量子ビットを用いる.

となる. そして, 逆量子フーリエ変換 $F_{m_k}^{-1}$ を追加した m_k 個の補助量子ビットに作用させると, 次の形となる.

$$\begin{aligned}&\frac{e^{i\theta}}{\sqrt{2}}\sum_{x=0}^{2^{m_k}-1} e^{2ix\theta}F_{m_k}^{-1}\left|x\right\rangle_{m_k}\left|\Psi_+\right\rangle_{n+1}+\frac{e^{-i\theta}}{\sqrt{2}}\sum_{x=0}^{2^{m_k}-1} e^{-2ix\theta}F_{m_k}^{-1}\left|x\right\rangle_{m_k}\left|\Psi_-\right\rangle_{n+1}\\ =&\frac{e^{i\theta}}{\sqrt{2}}\sum_{y=0}^{2^{m_k}-1} b_+(y)\left|y\right\rangle_{m_k}\left|\Psi_+\right\rangle_{n+1}+\frac{e^{-i\theta}}{\sqrt{2}}\sum_{y=0}^{2^{m_k}-1} b_-(y)\left|y\right\rangle_{m_k}\left|\Psi_-\right\rangle_{n+1}\end{aligned} \tag{3.10}$$

この係数 b_+, b_- は, $2^{m_k}\theta/\pi$ あるいは, $2^{m_k}(1-\theta/\pi)$ 付近にピークを持つ関数であり, m_k 個の補助ビットを観測すると, $\sin^2\theta=\sin^2(\pi-\theta)=:a$ の近似値を得ることができる. なお, [11] によって, 任意の整数 l に対して, θ が $\pm\pi k/2^{m_k}$ の誤差で推定できたとすると, a の誤差 ϵ は,

$$\epsilon=|a-\hat{a}|\leq 2\pi l\sqrt{a(1-a)}/2^{m_k}+l^2(\pi/2^{m_k})^2 \tag{3.11}$$

であり, 特に $l=1$ とすると, この誤差の下限は $8/\pi^2\sim 81\%$ であることが示されている. また, この演算の回路図を図 3.2 に示す. なお, 1 回の量子振幅増幅演算子 G で A と $A^\dagger$ がそれぞれ 1 回ずつの呼び出しが必要であることから, $G^{m_k}\left|\Psi\right\rangle_{n+1}$ を計算するためには, A または $A^\dagger$ の呼び出し回数 $Nq_G^{(k)}$ は,

$$Nq_G^{(k)}=2m_k+1 \tag{3.12}$$

となり, さらに, M 個の m_k に対する場合の呼び出し回数 Nq_G は,

$$Nq_G=\sum_{k=1}^{M}Nq_G^{(k)}=\sum_{k=1}^{M}(2m_k+1) \tag{3.13}$$

となる. ここで $m_k=2^k$ とすると, $Nq_G\sim\mathcal{O}(2^M)$ となり, 式 (3.11) から, $Nq_G\sim\mathcal{O}(\epsilon^{-1})$ であることが導かれる. 古典的なサンプリングで値の推定を行う場合, $Nq_G\sim\mathcal{O}(\epsilon^{-1/2})$ であることから 2 乗の加速をしていることがわかる.

しかし, この手法は, 制御振幅増幅演算子を用いることにより, いくつものゲートと, 追加の補助量子ビットが必要となる. 特に, この補助量子ビットの数は, 要求される推定精度が高いほど, たくさんの数が必要になる. 一方で, 2021 年現在および近未来の量子コンピュータデバイスを用いて計算をする場合には, 量子ビットの数は限りがあり, ノイズがあることから操作できるゲートの数は限られる等の様々な制約があり, この手法を実行する上で困難になる可能性が高いことを示している. [37] では, ゲートの数の見積もりを行なっており, 高々 10 を超える量子ビットを用いる際でさえ, 数千から数万の CNOT ゲートが必要であり, 更に実機の量子ビット間の接続によって追加のゲート操作が数倍程度かかる例を示している. そこで, [57] で提案した量子位相推定を使わない, つまり, 制御振幅増幅演算子および逆量子フーリエ変換を用いない量子振幅推定手法を次の節で述べる.

3.2 最尤推定を用いた量子振幅推定アルゴリズムについて

この節では, [57] で提案した最尤推定を用いた量子振幅推定手法を説明する. これは, 量子振幅増幅した状態に対して, 補助ビットに制御演算子を作用させて, 逆量子フーリエ変換で振幅を取り出すのではなく, 複数の量子振幅増幅をそれぞれ実行および測定を行い, それらの結果を合わせ, 最尤推定を行うことで振幅の推定を行うものである.

まず, 振幅増幅演算子 G を m_k 回採用させた状態 $|\Psi_G(\theta,k)\rangle = G^{m_k}|\Psi\rangle_{n+1}$ を示している式 (3.5) を再掲する.

$$|\Psi_G(\theta,k)\rangle = G^{m_k}|\Psi\rangle_{n+1} = \sin((2m_k+1)\theta)\,|\tilde{\Psi}_1\rangle_n\,|1\rangle + \cos((2m_k+1)\theta)\,|\tilde{\Psi}_0\rangle_n\,|0\rangle \tag{3.14}$$

この状態における最後の, つまり, $n+1$ 番目の量子ビットが 1 あるいは 0 であることを測定する. つまり $E_{G,1} = \mathbf{I}_n \otimes |1\rangle\langle 1|$ と $E_{G,0} = \mathbf{I}_n \otimes |0\rangle\langle 0|$ という測定演算子で計算すると, $i \in \{0,1\}$ を観測する確率 $p_G^{(k)}(i;\theta) = \mathrm{Tr}(\rho_G(\theta,k)E_{G,i})$ を次で得ることができる. なお, $\rho_G(\theta,k) = |\Psi_G(\theta,k)\rangle\langle\Psi_G(\theta,k)|$ である.

$$\begin{aligned} p_G^{(k)}(1;\theta) &= \mathbb{P}(\{h_k=1\}) = \sin^2((2m_k+1)\theta), \\ p_G^{(k)}(0;\theta) &= \mathbb{P}(\{h_k=0\}) = 1-\sin^2((2m_k+1)\theta) = \cos^2((2m_k+1)\theta) \end{aligned} \tag{3.15}$$

ここで, $h_k \in \{0,1\}$ は二値の確率変数である. そして, $\mathbf{h}$ を $(h_1,\cdots,h_M) \in \{0,1\}^M$ として, $\mathbf{h}$ に対する確率密度関数を次で置く.

$$P_G(\mathbf{h};\theta) = \prod_{k=1}^{M} (p_G^{(k)}(1;\theta))^{h_k} (p_G^{(k)}(0;\theta))^{1-h_k} \tag{3.16}$$

そして, 各 m_k 毎に, N_{shot} 回測定し, その結果を $\mathbf{H} = \{\mathbf{h}^1,\cdots,\mathbf{h}^{N_{\mathrm{shot}}}\}$ とする. これは, 確率分布 $P_G(\mathbf{h};\theta)$ からのサンプルの集合, あるいは, $\mathbf{h}$ の集合である. そして, 尤度関数 $L_G(\theta;\mathbf{H})$ を以下で定義をする.

$$L_G(\theta;\mathbf{H}) = \prod_{j=1}^{N_{\mathrm{shot}}} P_G(\mathbf{h}^j;\theta) \tag{3.17}$$

すると, 次の式により, θ を最尤推定することができる.

$$\hat{\theta}_{\mathrm{ML}} = \arg\max_{\theta} L_G(\theta;\mathbf{H}) = \arg\max_{\theta} \ln L_G(\theta;\mathbf{H}) \tag{3.18}$$

なお, 前述の尤度関数を示す式 (3.17) は, 次の変形をすることができる.

$$\begin{aligned} L_G(\theta;\mathbf{H}) &= \prod_{j=1}^{N_{\mathrm{shot}}} P_G(\mathbf{h}^j;\theta) \\ &= \prod_{j=1}^{N_{\mathrm{shot}}} \prod_{k=1}^{M} (p_G^{(k)}(1;\theta))^{h_k^j} (p_G^{(k)}(0;\theta))^{1-h_k^j} \\ &= \prod_{k=1}^{M} \prod_{j=1}^{N_{\mathrm{shot}}} (p_G^{(k)}(1;\theta))^{h_k^j} (p_G^{(k)}(0;\theta))^{1-h_k^j} \\ &=: \prod_{k=1}^{M} L_G^{(k)}(\theta;\mathbf{h}_k) \end{aligned} \tag{3.19}$$

ここで, $L_G^{(k)}(\theta;\mathbf{h}_k) := \prod_{j=1}^{N_{\mathrm{shot}}} (p_G^{(k)}(1;\theta))^{h_k} (p_G^{(k)}(0;\theta))^{1-h_k}, \mathbf{h}_k = \{h_k^1,\cdots,h_k^{N_{\mathrm{shot}}}\}$ と置くことで, m_k 毎の尤度関数の積であると見なすことができる. その考え方を図示したものが, 図 3.3 である. 改めて, 本手法では, 異なる m_k 毎に, 量子回路を実行し, それらの結果を合わせて尤度関数を構築し, 最尤推定を行うものである. 重

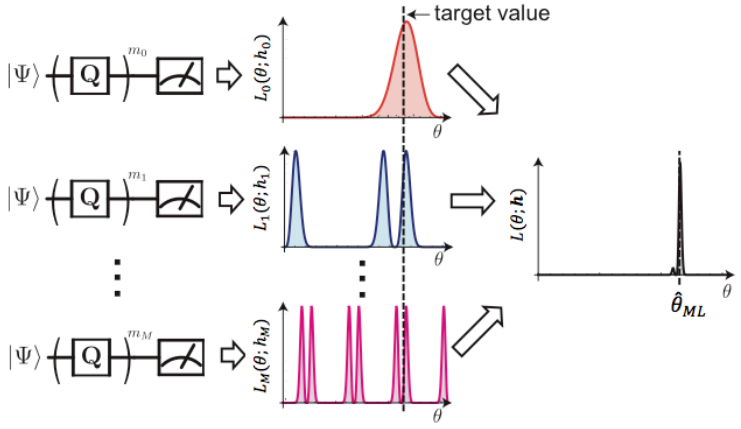


図 3.3　最尤推定を用いた手法のイメージ図

各 m_k 毎に, 尤度関数を得て, それらの積を取ることでピークを求めることができる.

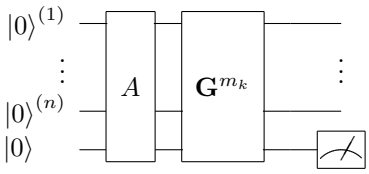


図 3.4　制御された振幅増幅演算子を用いない振幅増幅回路 (再掲)

制御された振幅増幅演算子ではなく, 単なる振幅増幅演算子を用いることで, 大幅な CNOT ゲートを減らすことができる. さらに, 量子フーリエ変換も用いない. m 回の量子回路自体の実行が必要となるものの, 量子ビット自体は $n+1$ 個のみで, 測定する量子ビットも 1 個のみとなる.

要な点の一つとして, 図 3.4 で示す回路を用いることで m_k 毎に量子回路を実行すれば良いことがわかる. これは, [11] らの手法では全ての m_k を一つの量子回路 (図 3.2 参照) で実行する必要があったことに対して, m_k 毎に分割実行が可能となるため, 量子ビットの数やゲート操作の数を明らかに減らすことができる. 3.5 節で具体的な例を用いて説明する. なお, m_k の種類が一つしかない場合は, 尤度関数の最大値が複数生じることがあり, 異なる m_k を複数用意することが必要である. また, 任意の k に対して, $m_k = 0$ とすると, 量子の優位性が消失してしまうことがわかる. 詳細に関しては, 本節の後半で説明する.

さて, 本手法を評価するために, θ に関する推定誤差を考える必要がある. 一般に, 最尤推定法は, 漸近有効性つまり, Cramér–Rao の下限を漸近的に達成することが知られている. ここで, Fisher 情報量を計算することで, Cramér–Rao の下限を求め, 推定精度の下限を評価する. まず, 任意の m_k に対して, Fisher 情報量 $J_G^k(\theta)$ を次で与えられる.

$$J_G^{(k)}(\theta) = \mathbb{E}\Big[\Big(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta)\Big)^2\Big] \tag{3.20}$$

ここで, $\mathbb{E}[\cdots]$ は, 式 (3.15) の確率分布に対する期待値である. そして, 本手法において, Fisher 情報量を式 (3.15) より具体的に計算すると以下で与えられる.

$$\begin{aligned} J_G^{(k)}(\theta) &= \mathbb{E}\Big[\Big(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta)\Big)^2\Big] \\ &= \sum_{h_k\in\{0,1\}}\Big(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta)\Big)^2 p_G^{(k)}(h_k;\theta) \\ &= 4(2m_k+1)^2 \end{aligned} \tag{3.21}$$

表 3.1　振幅増幅の回数 m_k の種類と標準偏差の一覧

#	振幅増幅の回数	標準偏差
1	$m_k = 0$ (ランダムサンプリング)	$\mathcal{O}(Nq_G^{-1/2})$
2	$m_k = k$ (略称：LIS)	$\mathcal{O}(Nq_G^{-3/4})$
3	$m_k = a^k$ (略称：EIS)	$\mathcal{O}(Nq_G^{-1})$

そして, M 個の m_k に対する Fisher 情報量 $J_G(\theta)$ は

$$J_G(\theta) = \sum_{k=1}^{M} J_G^{(k)}(\theta) = 4\sum_{k=1}^{M}(2m_k+1)^2 \tag{3.22}$$

となる. そして, Cramér–Rao の不等式より, θ に対する推定値 $\hat{\theta}$ の標準偏差 ϵ は,

$$\epsilon^2 = \mathbb{E}\big[(\theta-\hat{\theta})^2\big] \geq \frac{1}{N_{\text{shot}}} J_G(\theta)^{-1} = \frac{1}{4N_{\text{shot}}\sum_{k=1}^{M}(2m_k+1)^2} \tag{3.23}$$

となる. ここで, N_{shot} は, 独立した測定を繰り返した回数であり, 独立同一分布であることがわかる.

改めて, 一般的に, 最尤推定によって推定した $\hat{\theta}_{\text{ML}}$ は, 漸近的に, この不等式の下限を達成することができることが知られていることから, 式 (3.23) で与えられる下限が, m_k の数列の選び方によってどう変わるかを議論する. 具体的には, 1.$m_k = 0$ とした場合, 2.$m_k = k$ とした場合, 3.$m_k = a^k$（ただし, a は 2 以上の整数）とした場合, について述べる.

なお, コストは, θ を計算することができる演算子 A の呼び出し回数で考える. 式 (3.12) および (3.13) で示しているように, 振幅増幅演算子 G は, A と $A^\dagger$ を一つずつ持ち, $|\Psi\rangle_{n+1} = A|0\rangle_{n+1}$ であることを鑑みると, m_k を一つ固定した際の呼び出し回数 $Nq_G^{(k)}$, および, 本手法全体での呼び出し回数 Nq_G は次の通りとなる.

$$\begin{aligned} Nq_G^{(k)} &= 2m_k+1 \\ Nq_G &= \sum_{k=1}^{M} Nq_G^{(k)} = \sum_{k=1}^{M}(2m_k+1) \end{aligned} \tag{3.24}$$

1. $m_k = 0$ の場合
   任意の k に対して, $m_k = 0$ とする. つまり, 振幅増幅をしない場合に該当するため, これは量子ではない古典的なランダムサンプリングに該当する. この場合, $J_G(\theta) = 4\sum_{k=1}^{M}(2*0+1)^2 = 4M$ となるため, 推定値 $\hat{\theta}$ に対する標準偏差 $\epsilon \to 1/\sqrt{4N_{\text{shot}}M}$ となる. そして, A の呼び出し回数 $Nq_G = \sum_{k=1}^{M}(2*0+1) = M$ であることから, 標準偏差 ϵ は, $\mathcal{O}(Nq_G^{-1/2})$ である.
2. $m_k = k$ の場合 (LIS, Linearly incremental sequence)
   Fisher 情報量は, $J_G(\theta) = 4\sum_{k=1}^{M}(2k+1)^2 = 4/3(4M^3+12M^2+11M)$ となるため, 推定値 $\hat{\theta}$ に対する分散は, $\epsilon \to 1/\sqrt{4/3N_{\text{shot}}(4M^3+12M^2+11M)} = \mathcal{O}(M^{-3/2})$ となる. また, 演算子 A の呼び出し回数は, $Nq_G = \sum_{k=1}^{M}(2k+1) = M^2+2M$ となることから, 分散 ϵ は, $\mathcal{O}(Nq_G^{-3/4})$ である. つまり, 1.$m_k = 0$ の場合より優位であることがわかる.
3. $m_k = a^k$ の場合 (EIS, Exponentially incremental sequence)
   同様に計算すると, 演算子 A の呼び出し回数は, $Nq_G = \sum_{k=1}^{M}(2a^k+1) \sim 2a^{M+1}$ であり, Fisher 情報量 $J_G(\theta)$ は, $4\sum_{k=1}^{M}(2a^k+1)^2 \sim \mathcal{O}(Nq_G^{-1})$ となる. これは, Heisenberg 限界を達成することができる.

そして, これらを表 3.1 にまとめる. ただし, これらの標準偏差の値は, あくまで最尤推定を用いていることから漸近的に, つまり試行回数が十分に大きな範囲では達成可能であるが, 有限回の範囲で, 達成できるかは保証されていないことに注意しなければならない. そこで, 3.3 節で数値シミュレーションを行い, 検証をする.

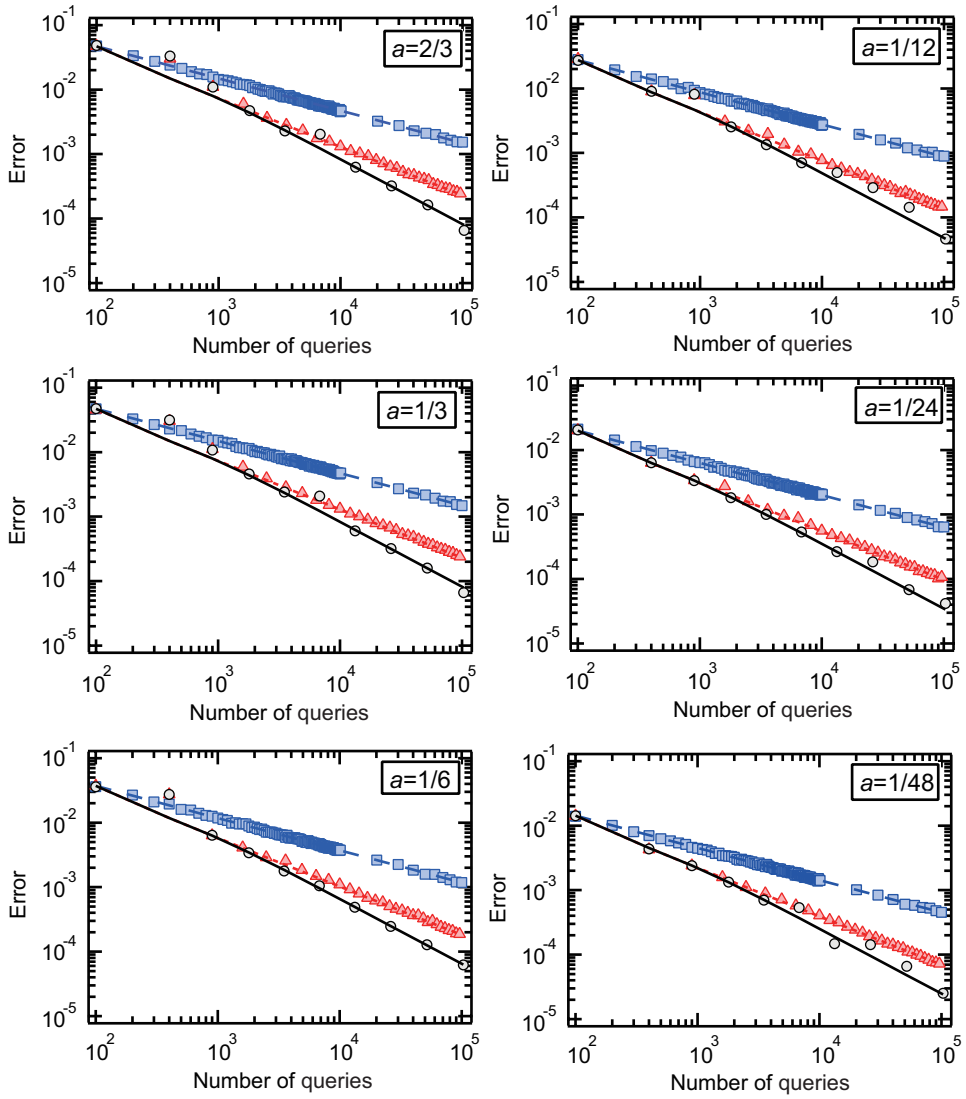


図 3.5　最尤推定を用いた振幅推定法のシミュレーション結果

横軸に演算子 A の呼び出し回数, 縦軸に推定精度の誤差のグラフ. 青色の四角点が古典的な場合, 赤色の三角の点が $m_k = k$ の場合, 黒色の丸い点が $m_k = 2^k$ の場合のシミュレーション結果であり, それぞれの線は Cramér-Rao の下限を示す.

3.3　数値実験

この節では, 最尤推定を用いた量子振幅推定手法に関しての数値シミュレーションを行う. 具体的には, 複数の θ に対して, 本手法を用いて推定した $\hat{\theta}$ と標準偏差 $\epsilon = \left(1/N_{\mathrm{rep}-1}\sum_{i=1}^{N_{\mathrm{rep}}}(\theta_i - \theta)^2\right)^{1/2}$ の評価を行い, 数値シミュレーションの結果と, 理論下限である Cramér–Rao の下限が, 漸近的に一致することを確認する.

まず, シミュレーションの設定を説明する. 測定回数 N_{shot} は 100 とし, 各確率に対して, h_k は Bernoulli 分布にてサンプリングを行い, 振幅増幅の回数 m_k は, 節 3.2 で示した 1.$m_k = 0$ つまり, 古典的なランダムサンプリング, 2.$m_k = k$, つまり線形に増やすケース, そして, 3. つまり, べき乗で増やすことで Heisenberg 限界を達成できるケースの 3 つの場合を行う. 推定する値 θ としては $[0.955, 0.615, 0.420, 0.292, 0.205, 0.145]$ を用

表 3.2　数値シミュレーションにおけるパラメータの一覧

項目	文字	値
測定回数	N_{shot}	100
振幅増幅の回数	m_k	$\{0, k, 2^k\}$
推定する値	θ	$[0.955, 0.615, 0.420, 0.292, 0.205, 0.145]$
誤差を評価するための繰返回数	N_{rep}	$1,000$

い, 各試行を $N_{\mathrm{rep}} = 1,000$ 回ずつ行うことで, 標準偏差 ϵ を評価する. なお, 推定する区間は, 推定された値の近傍を付近を, 徐々に狭めながら探索をグリッドサーチで行う. 前述の各パラメータを表 3.2 にまとめ, 結果を図 3.5 に示す. 図 3.5 では, 演算子 A の呼び出し回数 Nq_G と標準偏差の関係を示している. 青色の四角点が古典的な場合, 赤色の三角点が $m_k = k$ の場合, 黒色の丸い点が $m_k = 2^k$ の場合のシミュレーション結果であり, それぞれの線は Cramér–Rao の下限を示す. ここから次の 3 点のことがわかる. まず, この最尤推定を用いた量子振幅推定手法がうまく機能していることを示している. つまり, シミュレーション結果と理論的な下限である Cramér–Rao の下限が一致していることである. そして, これは最尤推定を用いていることから, 漸近一致, つまり測定回数が十分に大きい時には, Cramér–Rao の下限と一致することが知られているが, この場合, 高々 100 の測定回数でもうまく機能していることがわかる. 続いて, 青色の四角点と, 黒色の丸点を比べると, 確かに量子振幅推定法を用いた場合に二乗加速していることがわかる. 最後に, 推定したい θ の値によらず, うまく機能してることも確認できる. なお, 一部, シミュレーション結果と下限が離れている点があるものの, 測定回数を増やすことで近づくことを確認している.

3.4　計算量に関わる議論

この節では, 計算量についての議論を行う. まず, 振幅増幅演算子の数列 $m_k = a^k$ の時, 素朴な方法で, 尤度関数の最大化にかかる計算量は, $\mathcal{O}((1/\epsilon)\ln(1/\epsilon))$ である. これは対数尤度関数 $\ln L_G(\theta; \mathbf{H}) = \ln \prod_{k=1}^{M} L_G^{(k)}(\theta; \mathbf{h}_k)$ を評価するのに, $\mathcal{O}(M)$ の計算量がかかるからである. そして, 標準偏差 ϵ は, Cramér–Rao の下限から, $\mathcal{O}(N_{\mathrm{q}}^{-1})$ である. これは, 対数尤度関数を評価する計算量が $N_{\mathrm{q}} \sim 2^M N_{\mathrm{shot}}$ であるからである. ここで, 対数尤度関数の最大値を見つけるために, $1/\epsilon$ の刻み数の Brute-force search を行うと, $\mathcal{O}((1/\epsilon)\ln(1/\epsilon))$ の計算量がかかる. なお, $m_k = k$ の場合には, 同様の計算により, $\mathcal{O}(\epsilon^{-5/3})$ と評価できる. 注意すべき点として, $m_k = 0$ の場合, つまり古典的なサンプリングを行う際には, $\prod_{k=0}^{M} L_k(h_k; \theta)$ における最小値を, Brute-force search を用いて計算する必要がない. 何故ならば, 目標値は単純に $\hat{\theta} = \sum_{k=0}^{M} h_k / \sum_{k=0}^{M} N_{\mathrm{shot}}$ で得られるからである. そして, 推定誤差は, Cramér–Rao の下限から, $\mathcal{O}(N_{\mathrm{q}}^{-1/2})$ を得る. $N_{\mathrm{q}} = N_{\mathrm{shot}} M$ であることを思い出すと, 古典的な場合の計算量は, $\mathcal{O}(\epsilon^{-2})$ となる. m_k 毎に応じて, 表 3.3 に, 後処理の計算量と A の呼び出し回数に対応する計算量の結果をまとめる.

3.5　モンテカルロ積分への適用

これまでの節で, 位相推定を用いない振幅推定手法を説明した. この節では, 具体的なアプリケーションとして, モンテカルロ積分への適用を [35] を元に説明する. そして, 具体的な例として三角関数のリーマン積分を行うとともに, 位相推定および制御ユニタリー演算子が不要であることによるゲート操作, 量子ビットの数をどの程度減らすことができるのかについて述べる. なお, この三角関数の積分のソースコードは著者らによって, Qiskit Community Tutorials[85] に公開されている.

表 3.3　m_k に応じた計算量の一覧

必要である推定精度を ϵ とした際に, m_k の種類に応じた計算量
（量子の部分に該当する A の呼び出し回数および, 古典的な後処理）の一覧を表に示す.

振幅増幅の回数の数列 m_k	A の呼び出し回数	古典的な計算量
古典的な場合		
$(m_k = 0\ \forall k)$	$\mathcal{O}(\epsilon^{-2})$	$\mathcal{O}(\epsilon^{-2})$
線形に増やす場合 (LIS)		
$(m_0 = 0, m_1 = 1, m_2 = 2, \cdots, m_M = M)$	$\mathcal{O}(\epsilon^{-4/3})$	$\mathcal{O}(\epsilon^{-5/3})$
指数関数的に増やす場合 (EIS)		
$(m_0 = 0, m_1 = 2^0, m_2 = 2^1, \cdots, m_M = 2^{(M-1)})$	$\mathcal{O}(\epsilon^{-1})$	$\mathcal{O}(\epsilon^{-1} \ln \epsilon^{-1})$

3.5.1　振幅推定を用いたモンテカルロ積分

まず, 1 次元積分 $\mathbb{E}[f(x)] = \int_0^1 q(x)f(x)dx$ を考える. ただし, $f : [0,1] \to [0,1]$ は滑らかな実関数とし, $q : [0,1] \to [0,1]$ は $\int_0^1 q(x)dx = 1$ を満たす確率分布とする. そして, この積分値は, 次で近似計算が可能である.

$$S(f) = \sum_{j=0}^{2^n-1} p(x_j)f(x_j) \tag{3.25}$$

ここで, x_j は, $x_j = (j + 1/2)/2^n$ で与えられれ, p は確率質量関数で定義され, $p(x_j) = \int_{x_j-1/2^{n+1}}^{x_j+1/2^{n+1}} q(x)dx$ を満たす. なお, $f(x)$ は離散化しており, 近似誤差が生じる, すなわち $S(f) \neq \mathbb{E}[f(x)]$ となる. しかしながら, 本解析では, 実際の量子デバイスにおける本アルゴリズムの性能を評価することを目的にしており, $\mathbb{E}[f(x)]$ との比較ではなく, $S(f)$ とモンテカルロ積分で得られた値との誤差を評価する. そして, 振幅推定は, 以下の演算子を用いて実行する.

$$\mathcal{P}\left|0\right\rangle_n = \sum_{j=0}^{2^n-1} \sqrt{p(x_j)}\left|j\right\rangle_n, \quad \mathcal{T}\left|j\right\rangle_n\left|0\right\rangle = \left|j\right\rangle_n\left(\sqrt{f(x_j)}\left|1\right\rangle + \sqrt{1-f(x_j)}\left|0\right\rangle\right) \tag{3.26}$$

そして, これらの演算子 $\mathcal{P}, \mathcal{T}$ を $(n+1)$ 量子ビットの初期状態 $\left|0\right\rangle_n\left|0\right\rangle$ に適用すると,

$$\begin{aligned}\mathcal{T}(\mathcal{P}\otimes\mathbf{I}_1)\left|0\right\rangle_n\left|0\right\rangle &= \sum_{j=0}^{2^n-1}\sqrt{p(x_j)}\left|j\right\rangle_n\left(\sqrt{f(x_j)}\left|1\right\rangle + \sqrt{1-f(x_j)}\left|0\right\rangle\right) \\ &= \sqrt{S(f)}\left|\tilde{\Psi}_1\right\rangle\left|1\right\rangle + \sqrt{1-S(f)}\left|\tilde{\Psi}_0\right\rangle\left|0\right\rangle,\end{aligned} \tag{3.27}$$

を得ることができる. ここで, $\left|\tilde{\Psi}_1\right\rangle$ および $\left|\tilde{\Psi}_0\right\rangle$ は次で定義される.

$$\left|\tilde{\Psi}_1\right\rangle = \frac{1}{\sqrt{S(f)}}\sum_{j=0}^{2^n-1}\sqrt{p(x_j)f(x_j)}\left|j\right\rangle_n, \quad \left|\tilde{\Psi}_0\right\rangle = \frac{1}{\sqrt{1-S(f)}}\sum_{j=0}^{2^n-1}\sqrt{p(x_j)(1-f(x_j))}\left|j\right\rangle_n \tag{3.28}$$

式 (3.27) は, 式 (3.1) の状態を表している. つまり, $n+1$ 番目の量子ビットが $\left|1\right\rangle$ である状態を観測すると, 振幅の二乗, つまり $S(f)$ を求めることができる. よって, 誤差 ϵ の近似積分の値 $S(f)$ を求める際に, $O(1/\epsilon)$ の演算子 A の呼び出し回数で求めることができる. ここで, この演算の回路を図 3.6 に示し, 従来の位相推定を用いる回路を図 3.7 に示す. 図からもわかるように明らかに量子回路一回あたりの量子ビット数や量子回路のゲートの数は, 図 3.6 の方が少ないことがわかる. そこで, 次の項で一例を用いて, 数値比較を行う.

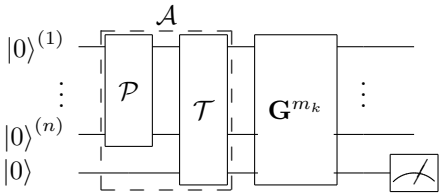


図 3.6　モンテカルロ積分を, 振幅増幅のみで行う回路

$\mathcal{T}(\mathcal{P}\otimes\mathbf{I}_1)=\mathcal{A}$ で $S(f)$ を埋め込み, 振幅増幅演算子 G を m_k 回作用させ, $n+1$ 番目の量子ビットを測定する. これを異なる複数の m_k に行い, 測定結果から最尤推定を行うことで, $S(f)$ を得る.

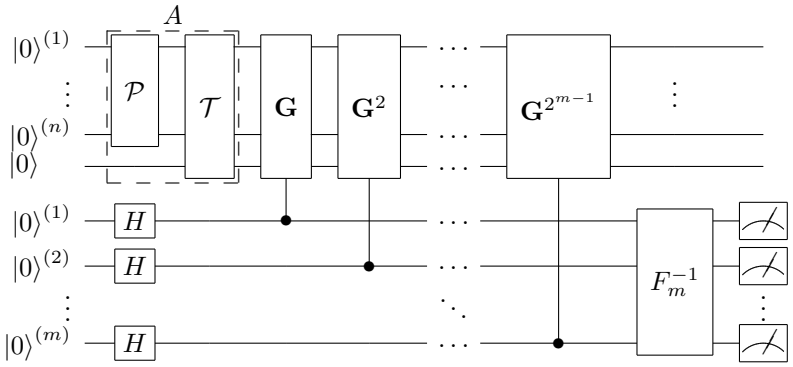


図 3.7　モンテカルロ積分を, 従来の位相推定を用いる振幅推定で行う回路

図 3.6 と同様に $\mathcal{P}\otimes\mathbf{I}_1=A$ で $S(f)$ を埋め込み, 制御された振幅増幅演算子たちで, 補助ビットに作用し, 逆量子フーリエ変換 F_m^{-1} を行うことで, 振幅を観測する.

3.5.2　三角関数を用いたモンテカルロ積分の例

この項では, モンテカルロ積分のシンプルな例として, 三角関数の積分

$$\int_0^1 \sin^2(bx)\,dx \tag{3.29}$$

を扱う. ただし, b は定数とし, 近似的に次を計算する.

$$S(f)=\sum_{j=0}^{2^n-1}\frac{1}{2^n}\sin^2\left(bx_j\right). \tag{3.30}$$

そして, 前項で用いた演算子 $\mathcal{P}$ および $\mathcal{T}$ は次にようになる.

$$\begin{aligned}&\mathcal{P}:|0\rangle_n\to\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle_n\\&\mathcal{T}:|x\rangle_n|0\rangle\to|x\rangle_n\left(\sin\left(\frac{bx}{2^n}\right)|1\rangle+\cos\left(\frac{bx}{2^n}\right)|0\rangle\right)\end{aligned} \tag{3.31}$$

そして, 演算子 $\mathcal{P}$ は n 個のアダマールゲートで, 演算子 $\mathcal{T}$ は, 図 3.8 で示すように, 制御 Y 回転ゲートで実現できる.

そして, 具体的な例として, 本提案する制御演算子を用いない振幅増幅の回路と, 従来の制御演算子を用いた振幅推定の回路を, 式 (3.31) を用いて, 図 3.9, 3.10 を示す. なお, 簡単のために, 量子ビットの数 $n=2$, 振幅増幅の回数 $m_k=1$, $b=\pi/4$ とし, 量子ビット毎は全結合とする. この図から明らかに, 提案手法の方がゲート数や量子ビット数が少なくてすむことがわかる.

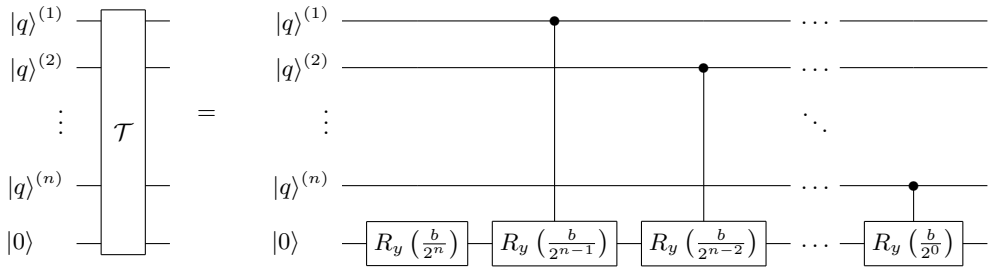


図 3.8　演算子 $\mathcal{T}$ を実装した回路

式 (3.31) にて定義されている演算子 $\mathcal{T}$ を $|x\rangle$ に作用する回路. 初期状態 $|x\rangle_n$ を各量子ビット毎に $|q\rangle^{(1)}$, $|q\rangle^{(2)}$,$\cdots$, $|q\rangle^{(n)}$ と表す. そして $R_y(\theta)$ は, 角度 θ の Y 軸回転を示す.

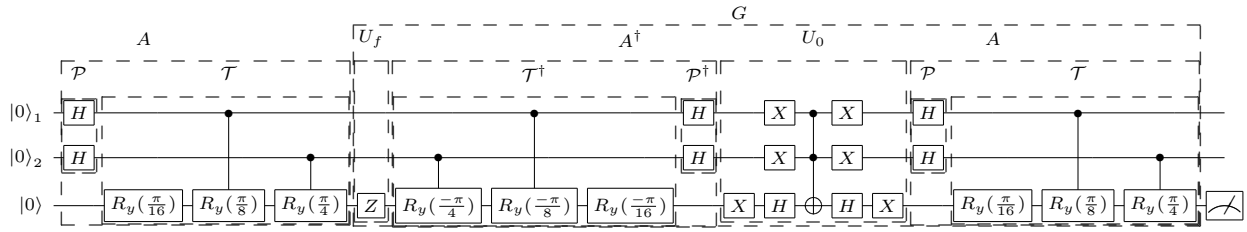


図 3.9　制御演算子を用いない振幅増幅の回路 ($m_k = 1$ の時) の例

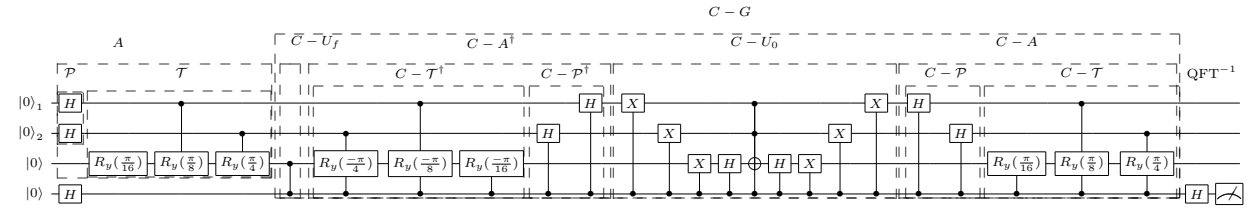


図 3.10　制御演算子を用いた振幅推定の回路 ($m_k = 1$) の例

そして, 表 3.4 に, それぞれの手法において, 振幅増幅演算子 G の数に応じて, CNOT ゲートの数, 量子ビットの数を示す. なお, 現在の NISQ と呼ばれるノイズのある量子コンピュータにおいては, CNOT ゲートがシングル量子ビットゲートに比べてノイズの大きさが大きいため, CNOT ゲートの数をベンチマークの値とする. 本アルゴリズムの CNOT ゲート数は, 従来の振幅推定に比べて約 7〜18 倍小さくなる. また, 従来の振幅推定法では, 演算回数が増えると必要な量子ビット数が増えるが, 本章で導入した手法では一定であることがわかる.

改めて, この二つの手法の様々な項目について 3.5 にまとめる. 従来の [11] では, 振幅増幅した結果を一度に取り出すため, 実行回数は一回ですむものの, 一回あたりの実行する量子ビットの数や測定する量子ビットの数, 必要なゲート数は多くなる. なお, 表の a は, 約 7〜18 の値である.

3.6　本章のまとめ

本章では, 最尤推定を用いることで, 量子位相推定を用いない量子振幅推定アルゴリズム [57] を紹介した. 3.1 節では, Brassard らによる量子振幅推定アルゴリズム [11] を紹介し, 3.2 節では, 最尤推定を用いた量子振幅推定アルゴリズムを説明し, いくつかの論点を整理した. この手法のポイントは, 振幅増幅された量子状態の測定結果を用いて尤度関数を構築し, そして, 最尤推定にて θ の推定を行うものである. そして 3.3 節では, 数値シミュレーションによって手法の有効性を検証した. Cramér–Rao の下限とシミュレーション結果が, 高々 100 回程度の測定でうまくいくこと, そして提案手法が古典的なものより二乗の加速が得られることをと示し

表 3.4　既存手法と提案手法における CNOT ゲートの数と量子ビットの数の比較

	CNOT ゲートの数			量子ビットの数	
演算子 G の回数	従来の手法	提案手法	比率	従来の手法	提案手法
0	-	4	-	-	3
2^0	135	18	7.5	7	3
2^1	399	32	12.4	8	3
2^2	927	60	15.5	9	3
2^3	1,981	116	17.1	10	3
2^4	4,085	228	17.9	11	3
2^5	8,287	452	18.3	12	3
2^6	16,683	900	18.5	13	3
2^7	33,465	1,796	18.6	14	3
2^8	67,017	3,588	18.9	15	3

表 3.5　振幅推定法の比較の概要 (再掲)

項目	Brassard らの手法	著者らの手法
実行回数	1 回	m 回
演算子 A の呼び出し回数	$N_{\mathrm{shot}} \sum_k m_k$ 回	$N_{\mathrm{shot}} \sum_k m_k$ 回
必要な量子ビット	$n+1+m$ 個	$n+1$ 個
測定する量子ビット	m 個	1 個
必要なゲート数	$a * N_{gate}$ 個	N_{gate} 個

た. 続く 3.4 節では計算量に関わる議論を行い, 最後に 3.5 節では, 活用例としてモンテカルロ積分の例を述べた. 特に, CNOT ゲートや量子ビットの数を大幅に減らせることができることを示した.

改めて, Brassard らが提案した量子振幅推定アルゴリズム [11] は, 量子振幅増幅と量子位相推定を用いることで, 一度に計算をすることができるものだったが, 一方で, 一度に全ての計算を行うことから, 量子回路において非常に長い回路を実行する必要があり, さらに計算精度を上げようとした場合には, 量子ビット数が増えるものであった. 一方で現在のハードウェアは量子ビットにも限りがあり, 実行できる回路の深さも限られている. そこで著者らによって最尤推定を用いることで, 量子位相推定を用いない量子振幅推定アルゴリズム [57] が提案され, 必要な量子ビット数の削減（定数化）および, 並列的に実行することで, 一度に実行する回路の深さの削減が実現した. これは現在の量子コンピュータでも量子振幅推定アルゴリズムが実行できる可能性を向上させたものである. また, 仮にエラー訂正ができる量子コンピュータができたとしても, 量子ビット等のリソースが無制限に使える訳ではなく, より少ないハードウェアで実行できることは重要な課題の一つであり, 現在の限られた量子コンピュータおよび, 将来のエラー訂正ができる量子コンピュータの両方に, 有用なアルゴリズムである.

これを契機に, いくつかの量子位相推定を用いずに量子振幅推定を実行するアルゴリズムが提案されている [58, 59, 60, 61]. Aaronson ら [58] によって, 量子フーリエ変換を用いない振幅推定手法を提案し, 推定精度の上限に関しても厳密な証明を与えた. しかしながら, これは, 逐次的に実行する必要があり, さらに測定のサンプリングが数百万回必要であり, 実際には実行が困難である可能性がある. また, 続く, Grinko ら [59] あるいは 中路 [60] によっても, サンプリングの上限をつけるとともに, その回数を改善させている. 著者らによる [57] および, [59], [60] は Qiskit finance[86] にも実装されており, 簡単に比較検証が可能である. なお, 我々の

提案した最尤推定を用いた手法に上限をつけられておらず, これは今後の課題である.

3.5 節で示している三角関数の積分に関しては, [87] ではスピンエコー法を用いることで, より短い回路で実装することが提案されており, また, [44] では, 求めたい積分値を近似的にフーリエ級数展開することで実装する方法が提案されているなど, より実際のモンテカルロ積分をする上で必要なハードウェアリソースをいかに削減するかという議論が行われている.

最後に, 現在の NISQ[22] と呼ばれる量子コンピュータには, ノイズがある. 詳細は次の第 4 章で説明するが, 実際に量子コンピュータを使って用いて本アルゴリズムを実行すると, 1.2 の 1.3 および次節でも再掲しているようにノイズによる影響を受けることがわかる. つまりノイズに対しての影響を加味したアルゴリズムの進展が必要であり, それについて, 次の第 4 章で, 著者ら [65] の論文の議論を説明する.

第 4 章

ノイズ下における量子振幅推定

この章では, [65] に掲載さている, ノイズがある状況下での量子振幅推定手法について説明する.

前章では, 最尤推定を用いた量子振幅推定手法を導入し, 量子振幅増幅を並列的に実行できるように, 言い換えると, 量子ビット数を減らし, 量子回路の深さを減らすことを実現し, 現在の限られている量子コンピュータでの量子振幅推定手法の実行できる可能性を増やした. 一方で, 現在の量子コンピュータでは, ノイズがあるため, この影響を考える必要がある. 実際に, 現在の超電導量子コンピュータ IBM Quantum *ibm_valencia*[16] で実行すると, 図 4.1 のようになる. 詳細は, 4.3 節で述べるが, 橙色の細い線が, ノイズがない理想的な量子コンピュータをシミュレーションしたもので, 緑色の丸い点が実機の結果である. 横軸である振幅増幅の回数 m_k を増やした時に, 実機の結果はほぼ 0.5 に収束しており, シミュレーションの結果と実機の結果には乖離があることを示している. つまり, 実機におけるノイズの影響を加味する必要があることを示唆しており, この章では, ノイズがある状況下での最尤推定を用いた量子振幅推定手法について解析を行う.

まず, 4.1 節では, ノイズモデルの導入を行う. ここでは特に脱分極ノイズがある状況下での最尤推定を用いた量子振幅推定を議論する. これは図 4.1 の結果をサポートするモデルである. なお, ここでは脱分極ノイズの

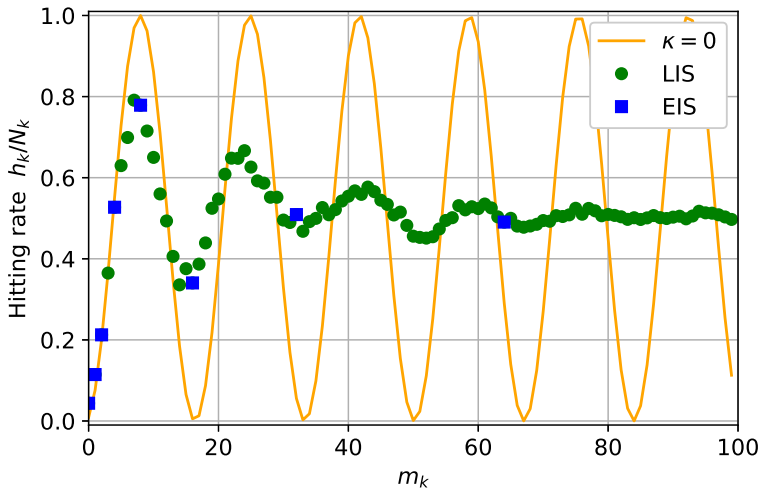


図 4.1　実機を用いた振幅増幅の結果（再掲）

超電導量子コンピュータ IBM Quantum *ibm_valencia* を用いて振幅増幅を行なった結果. 横軸には振幅増幅の回数 m_k, 縦軸には最後の量子ビットが 1 を観測した確率であり, 橙色がシミュレーションによる結果, 緑色の丸点及び青色の四角点が実機から得られた結果である.

大きさを未知のパラメータとして導入し, 複数パラメータの推定問題としてモデルを構築する. このモデルに対して最尤推定を行うため, 尤度関数を示すとともに, Cramér–Rao の下限を求めるために Fisher 情報行列を計算する. 次に, 4.2 節では, 前節の状況下, つまり脱分極ノイズがある状況下で, 振幅増幅の回数 m_k を増やした時の推定精度がどうなるか, あるいは脱分極ノイズの大きさが変わった時に, 推定精度がどうなるかを議論する. そして, 4.3 節では実際の超電導デバイス IBM Quantum *ibm_valencia* を用いて実験を行う. 特に 2 量子ビットおよび 3 量子ビットを用いて, 3.5 節で示した三角関数のモンテカルロ積分を行う. 注意すべき点として, 本節では, 脱分極ノイズモデルを仮定して, 議論を行うが, 実際の量子コンピュータが脱分極ノイズかはわからないものの, モデルと実験結果から, 脱分極ノイズモデルは悪くないモデルであることがわかる. 4.4 節では, 例外的なパラメータの議論を行う. これは, 推定したいパラメータとノイズパラメータからなる複数パラメータを推定する問題から生じるものであり, 4.2 節で議論した関係が成り立たない場合である. つまり, 脱分極ノイズが小さくなっても, 推定精度が下げられない場合がある. ただし, これは $1 \sim 2\%$ 程度の割合でしか発生せず, 回避方法も述べる. 続いて, 4.5 節では, 4.2 節で議論した関係式を用いてハードウェアの必要な性能を求める. これは将来, どの程度, 大規模な量子コンピュータができれば, どのようなことができるようになるかがわかる. 4.6 節では計算量に関する議論を行い, 最後に 4.7 節で本章のまとめを行う.

4.1 脱分極ノイズのある量子振幅推定

この節では, ノイズがある量子振幅推定手法を議論する. まず, ノイズなどにより, パラメータが複数ある場合に, 最尤推定を用いる量子振幅推定手法を考える. パラメータたちをベクトル表記として, $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_M]^\top$ とする. なお, $\top$ は転置を表す. そして, 最後の $n+1$ 番目の量子ビットが $0, 1$ になる確率をそれぞれ, $p_G^{(k)}(0;\boldsymbol{\theta}), p_G^{(k)}(1;\boldsymbol{\theta})$ と書き, $\mathbf{h}$ に対する確率 $P_G(\mathbf{h};\theta)$ は, 以下である.

$$P_G(\mathbf{h};\boldsymbol{\theta}) = \prod_{k=1}^{M} (p_G^{(k)}(1;\boldsymbol{\theta}))^{h_k} (p_G^{(k)}(0;\boldsymbol{\theta}))^{1-h_k} \tag{4.1}$$

そして, 尤度関数 $L(\boldsymbol{\theta};\mathbf{H})$ は,

$$L(\boldsymbol{\theta};\mathbf{H}) = \prod_{j=1}^{N_{\text{shot}}} P_G(\mathbf{h}^j;\boldsymbol{\theta}) \tag{4.2}$$

である. ただし, $\mathbf{H} = \{\mathbf{h}^1, \cdots, \mathbf{h}^{N_{\text{shot}}}\}$ である. 最尤推定法を用いることで, $\hat{\boldsymbol{\theta}}_{\text{ML}}$ を以下で推定することができる.

$$\hat{\boldsymbol{\theta}}_{\text{ML}} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta};\mathbf{H}) = \arg\max_{\boldsymbol{\theta}} \ln L(\boldsymbol{\theta};\mathbf{H}) \tag{4.3}$$

一般に, 推定誤差の共分散行列 $\text{Cov}(\hat{\boldsymbol{\theta}}) = \mathbb{E}[(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^{\text{T}}]$ は, $\hat{\boldsymbol{\theta}}$ を不偏推定値とすると, 次の Cramér–Rao の不等式を満たす.

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \frac{1}{N_{\text{shot}}} (J_G(\boldsymbol{\theta}))^{-1} \tag{4.4}$$

ただし, $J_G(\boldsymbol{\theta})$ は, Fisher 情報行列で, 次で与えられる.

$$J_G(\boldsymbol{\theta}) = \mathbb{E}\left[\left(\frac{\partial}{\partial \boldsymbol{\theta}} \ln P_G(\mathbf{h};\boldsymbol{\theta}) \right) \left(\frac{\partial}{\partial \boldsymbol{\theta}} \ln P_G(\mathbf{h};\boldsymbol{\theta}) \right)^\top \right] \tag{4.5}$$

一般的に, 最尤推定により, 式 (4.4) で与えられる下限は, 漸近的に達成可能であることが知られている. なお, Fisher 情報行列 $J_G(\boldsymbol{\theta})$ の i, j 成分は次で与えられる.

$$[J_G(\boldsymbol{\theta})]_{i,j} = \mathbb{E}\left[\left(\frac{\partial}{\partial \theta_i} \ln P_G(\mathbf{h};\boldsymbol{\theta}) \right) \left(\frac{\partial}{\partial \theta_j} \ln P_G(\mathbf{h};\boldsymbol{\theta}) \right)^\top \right] \tag{4.6}$$

改めて, 一般には, データを生成する真の確率分布は未知である. そのため, 最尤推定法を用いる際には, 適切なモデルを構築することが重要なことである. そして, ノイズのある量子コンピュータの出力結果を考慮したいが, 実際の量子コンピュータのノイズを完全に特定することは簡単ではない. そこで, 本節においては, 実際の量子コンピュータから得られる未知のノイズを考える上で, 脱分極ノイズモデルを仮定する. 脱分極ノイズモデルは, ミニマル, また, 最悪ケースとしてのノイズモデルとして利用され [67, 68, 69], さらに, 量子デバイスのベンチマーク指標の一つであるランダマイズドベンチマーク [70, 71, 72] でも用いられるモデルである. また, 脱分極ノイズモデルは, 任意のユニタリー演算と可換であり, 計算が非常にしやすい, つまり, 解析的に扱うことができるというメリットがある.

まず, 脱分極ノイズチャンネル [12, 88] を次で定義する.

$$\mathcal{D}(\rho) = r\rho + (1-r)\frac{I_{n+1}}{d}. \tag{4.7}$$

ここで, ρ は密度行列であり, $\mathcal{D}(\rho)$ は, 脱分極を表す CPTP 写像であり, $1-r$ は量子ビットが脱分極になる確率, d はこの量子系における次元, つまり, $d = 2^{n+1}$ である. なお, r は $0 \leq r \leq 1$ であり, $r = 1$ のノイズがない状態, $r = 0$ の時は完全混合状態になっていることを示す. そして, 本節では r は未知パラメーターとして扱う. つまり, 振幅に埋め込む θ と, ノイズパラメータ r の二つのパラメータを推定する複数パラメータ推定問題として扱う. これは, 一次元の推定問題より, 本質的に困難な問題である.

そして, 一般に, ノイズのない, 振幅増幅チャンネルを表す CPTP 写像は, 密度行列を用いて次で表すことができる.

$$\mathcal{G}(\rho) = \mathbf{G}\rho\mathbf{G}^\dagger. \tag{4.8}$$

式 (4.7) と (4.8) より, ノイズがある状況下の振幅増幅チャンネルは, 次で与えられる.

$$\rho_{G,\mathrm{noise}} = \mathcal{G}\mathcal{D}(\rho_G) = \mathcal{D}\mathcal{G}(\rho_G) = r\mathbf{G}\rho_G\mathbf{G}^\dagger + (1-r)\frac{I_{n+1}}{d} \tag{4.9}$$

さらに, ノイズ付き振幅増幅を m_k 回繰り返すことを考えると

$$\begin{aligned}\rho_{G,\mathrm{noise}}^{(k)} = (\mathcal{G}\mathcal{D})^{m_k}(\rho) &= r^{m_k}\mathbf{G}^{m_k}\rho_G\mathbf{G}^{\dagger m_k} + (1-r^{m_k})\frac{I_{n+1}}{d} \\ &= r^{m_k}\rho_{G,r}^{(k)} + (1-r^{m_k})\frac{I_{n+1}}{d}\end{aligned} \tag{4.10}$$

となることがわかる. なお, ρ_G は, 式 (3.1) にある $|\Psi\rangle_{n+1}$ を用いて $\rho_G = |\Psi\rangle_{n+1}\langle\Psi|$ であり, $\rho_{G,r}^{(k)}$ は, 式 (3.5) にある $|\Psi_G(\theta,k)\rangle$ を用いて, $\rho_{G,r}^{(k)} = |\Psi_G(\theta,k)\rangle_{n+1}\langle\Psi_G(\theta,k)|$ ある. ここで, 未知のパラメータ $\boldsymbol{\theta}$ は, $\kappa = -\ln r$ とすると, $\boldsymbol{\theta} = [\theta,\kappa]^\top$ と書ける. そして, 式 (4.10) における $n+1$ 番目の量子ビットが 0 か 1 である状態を測定する, つまり, 測定演算子 $E_{G,0} = I_n \otimes |0\rangle\langle 0|$ と $E_{G,1} = I_n \otimes |1\rangle\langle 1|$ を作用させることで, 確率 $p_G^{(k)}(i;\theta,\kappa) = \mathrm{Tr}(\rho_{G,\mathrm{noise}}^{(k)}E_{G,i})$ は, 式 (3.15) にある $\mathrm{Tr}(\mathbf{G}^{m_k}\rho_G\mathbf{G}^{\dagger m_k}E_{G,0}) = \cos^2((2m_k+1)\theta)$, $\mathrm{Tr}(\mathbf{G}^{m_k}\rho_G\mathbf{G}^{\dagger m_k}E_{G,1}) = \sin^2((2m_k+1)\theta)$ であることを用いと, 次のように計算できる.

$$\begin{aligned}p_G^{(k)}(0;\boldsymbol{\theta}) = p_G^{(k)}(0;\theta,\kappa) &= \mathrm{e}^{-\kappa m_k}\cos^2((2m_k+1)\theta) + \frac{1-\mathrm{e}^{-\kappa m_k}}{2} = \frac{1}{2} + \frac{1}{2}\mathrm{e}^{-\kappa m_k}\cos(2(2m_k+1)\theta) \\ p_G^{(k)}(1;\boldsymbol{\theta}) = p_G^{(k)}(1;\theta,\kappa) &= \mathrm{e}^{-\kappa m_k}\sin^2((2m_k+1)\theta) + \frac{1-\mathrm{e}^{-\kappa m_k}}{2} = \frac{1}{2} - \frac{1}{2}\mathrm{e}^{-\kappa m_k}\cos(2(2m_k+1)\theta)\end{aligned} \tag{4.11}$$

そして, h_k を $\{0,1\}$ の二値変数とし, $\mathbf{h} = (h_1,\cdots,h_M) \in \{0,1\}^M$ に対する確率 $P_G(\mathbf{h};\theta,\kappa)$ を以下で置く.

$$P_G(\mathbf{h};\theta,\kappa) = \prod_{k=1}^{M}(p_G^{(k)}(1;\theta,\kappa))^{h_k}(p_G^{(k)}(0;\theta,\kappa))^{1-h_k} \tag{4.12}$$

更に, 各 m_k 毎に, N_{shot} 回測定し, その結果を $\mathbf{H} = \{\mathbf{h}^1, \cdots, \mathbf{h}^{N_{\mathrm{shot}}}\}$ とすると, 尤度関数 $L_G(\theta, \kappa; \mathbf{H})$ を以下とする.

$$L_G(\theta, \kappa; \mathbf{H}) = \prod_{j=1}^{N_{\mathrm{shot}}} P_G(\mathbf{h}^j; \theta, \kappa) \tag{4.13}$$

改めて, 本節においては, $\boldsymbol{\theta} = [\theta, \kappa]^\top$ が未知のパラメータであり, 最尤推定を用いることは, 次の式を計算することである.

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}; \mathbf{H}) = \arg\max_{\boldsymbol{\theta}} \ln L(\boldsymbol{\theta}; \mathbf{H}) \tag{4.14}$$

そして, 最尤推定法を用いることで, Cramér–Rao の不等式を漸近的に達成でき, その下限を評価するために, Fisher 情報行列 $J_G^{(k)}(\theta, \kappa)$ は次で計算される.

$$\begin{aligned}
\left(J_G^{(k)}(\theta,\kappa)\right)_{1,1} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)^2\right] \\
&= \sum_{h_k\in\{0,1\}}\left(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)^2 p_G^{(k)}(h_k;\theta,\kappa) \\
&= (2m_k+1)^2\frac{4\sin^2\left(2\left(2m_k+1\right)\theta\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta\right)} \\
\left(J_G^{(k)}(\theta,\kappa)\right)_{1,2} = \left(J_G^{(k)}(\theta,\kappa)\right)_{2,1} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)\left(\frac{\partial}{\partial\kappa}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)\right] \\
&= \sum_{h_k\in\{0,1\}}\left(\frac{\partial}{\partial\theta}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)\left(\frac{\partial}{\partial\kappa}\ln p_G^{(k)}(h_k;\theta,\kappa)\right) p_G^{(k)}(h_k;\theta,\kappa) \\
&= m_k(2m_k+1)\frac{\sin\left(4\left(2m_k+1\right)\theta\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta\right)} \\
\left(J_G^{(k)}(\theta,\kappa)\right)_{2,2} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\kappa}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)^2\right] \\
&= \sum_{h_k\in\{0,1\}}\left(\frac{\partial}{\partial\kappa}\ln p_G^{(k)}(h_k;\theta,\kappa)\right)^2 p_G^{(k)}(h_k;\theta,\kappa) \\
&= m_k^2\frac{\cos^2\left(2\left(2m_k+1\right)\theta_a\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta_a\right)}
\end{aligned} \tag{4.15}$$

ただし, $\left(J_G^{(k)}(\theta,\kappa)\right)_{i,j}$ は, Fisher 情報行列 $J_G^{(k)}(\theta,\kappa)$ の (i,j) 成分を表す. そして, M 個の m_k に対する

Fisher 情報行列 $J_G(\theta,\kappa)$ は次のようになる.

$$
\begin{aligned}
(J_G(\theta,\kappa))_{1,1} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\theta}\ln P_G(\mathbf{h};\theta,\kappa)\right)^2\right] \\
&= \sum_{k=0}^{M}(2m_k+1)^2\frac{4\sin^2\left(2\left(2m_k+1\right)\theta\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta\right)}, \\
(J_G(\theta,\kappa))_{1,2} = (J_G(\theta,\kappa))_{2,1} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\theta}\ln P_G(\mathbf{h};\theta,\kappa)\right)\left(\frac{\partial}{\partial\kappa}\ln P_G(\mathbf{h};\theta,\kappa)\right)\right] \\
&= \sum_{k=0}^{M}m_k(2m_k+1)\frac{\sin\left(4\left(2m_k+1\right)\theta\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta\right)}, \\
(J_G(\theta,\kappa))_{2,2} &= \mathbb{E}\left[\left(\frac{\partial}{\partial\kappa}\ln P_G(\mathbf{h};\theta,\kappa)\right)^2\right] \\
&= \sum_{k=0}^{M}m_k^2\frac{\cos^2\left(2\left(2m_k+1\right)\theta\right)}{e^{2\kappa m_k}-\cos^2\left(2\left(2m_k+1\right)\theta\right)}.
\end{aligned}
\tag{4.16}
$$

改めて, 振幅推定の問題において, 推定したいパラメータは θ であり, θ の推定誤差を評価する必要がある. 式 (4.4) の Cramér–Rao の不等式より, θ の推定誤差の下限は, Fisher 情報行列の逆行列の (1,1) 成分を用いて次のように書ける.

$$
\epsilon = \sqrt{\mathbb{E}\left[(\theta-\hat{\theta})^2\right]} \geq \frac{1}{\sqrt{N_{\mathrm{shot}}}}\sqrt{\left(J_G(\theta,\kappa)^{-1}\right)_{1,1}} =: \epsilon_{\min}(\theta,\kappa). \tag{4.17}
$$

一般的に, 最尤推定における Cramér–Rao の下限は, 漸近的に達成可能であることが知られており, 次の節では, $\epsilon_{\min}(a,\kappa)$ を用いて, 要求される推定精度を満たすためのノイズパラメータ κ の大きさについて議論を行う.

この節の最後に, 他のノイズモデルに関する議論を行う. 複数のパラメータを用いる Markovain ノイズモデルは, Kraus 演算子を用いることで表すことができる. しかしながら, 一般的には, 振幅増幅演算子と可換ではなく, 解析的な Cramér–Rao の下限の表記を得ることができない. このような推定精度を議論する場合において, 量子計測の分野では, いくつかの数値計算手法が提案されている [88, 89, 90]. なお, 脱分極チャンネル以外の特定のノイズチャンネルであったとしても, 解析的な表現を得ることは難しく, 例えば, [64] では, 振幅推定の問題において, 脱分極ノイズに加えて, 振幅ダンピング等のノイズモデルを考慮しているが, ノイズパラメータの大きさが既知である場合を扱っているにも関わらず, 解析的な形での計算が困難であり, 推定誤差を数値的に評価している.

4.2 脱分極ノイズ下での, 実現可能な推定誤差

前節では, ノイズがある状況下での量子振幅推定手法の提案を行った. ノイズの大きさを未知パラメータとして導入し, 複数のパラメータ推定問題として定式化をし, 脱分極ノイズチャンネルを仮定した場合の, 確率, 尤度関数, Fisher 情報行列を解析的に導いた. そして, Fisher 情報行列の逆行列の $(1,1)$ 成分を用いて, Cramér–Rao の不等式から達成可能な推定精度の誤差の下限 $\epsilon_{\min}(\theta,\kappa)$ を求めた. これにより, 推定したいパラメータ θ, ノイズの大きさ κ, 振幅増幅の回数つまりは, 演算子 A の呼び出し回数 $Nq_G = \sum_k N_{\mathrm{shot}}(2m_k+1)$, 達成可能な推定精度の下限 $\epsilon_{\min}(\theta,\kappa)$ の関係を議論するこができるようになり, 本節ではこれらの関係について議論を行う. これは, 具体的な問題を解く上で, 要求される推定精度から, 必要な演算回数やノイズの大きさを議論することができ, 非常に役に立つことである.

まず, 演算子の呼び出し回数 $Nq_G = \sum_{k=0}^{M} N_k(2m_k+1)$ と Cramér–Rao の下限から得られる推定精度の誤差の下限 $\epsilon_{\min}(\theta,\kappa)$ の関係を, 図 4.2 で示す. 推定する θ を 0.659 とし, 任意の k に対して測定回数 N_{shot} を

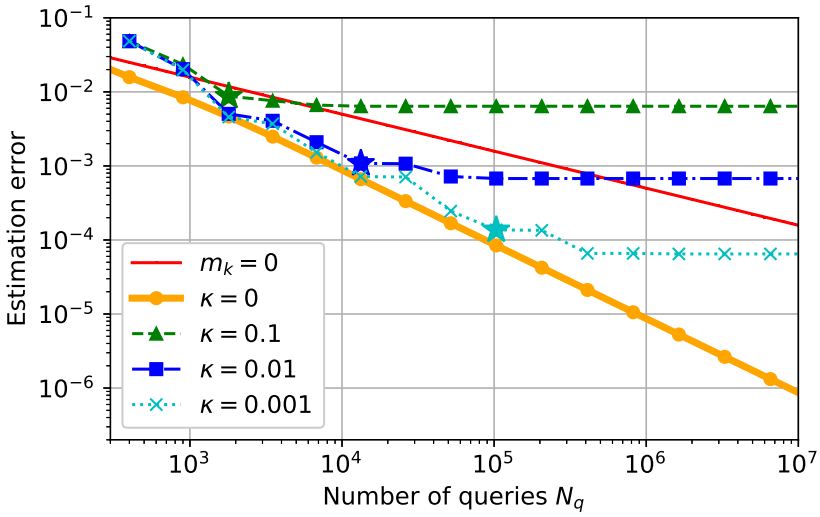


図 4.2　脱分極ノイズがある場合の A の呼び出し回数と推定精度の関係

この図は, Cramér–Rao の下限 $\epsilon_{\min}(a,\kappa)$ と, 演算子 A の呼び出し回数の関係を示している. そして, 赤色の細い線は, 古典的な場合, つまり全ての k に対して $m_k = 0$ の Cramér–Rao の下限であり, それ以外の 4 本の線は, 量子的な場合の Cramér–Rao の下限であり, 様々なノイズの大きさ κ の場合を表している. なお, 詳細は, 本 4.2 節の本文で説明する.

100 とする. 細い赤色の実線は, 古典的な手法の場合の, 推定精度の誤差の下限に該当し, それは, $m_k = 0$, つまり, $Nq_G = \sum_{k=0}^{M} N_k$ で与えられる. それ以外の 4 本の線は, 前節で説明した量子振幅推定を用いた推定精度の誤差の下限を示しており, 振幅増幅の回数は, $m_k = \lfloor 2^{k-1} \rfloor$ とする. 太い橙色の実線は, ノイズがない場合, つまり $\kappa = 0$ の時である. そして, ノイズがある場合として, 三角の点がある緑色の破線, 四角の点がある青色の鎖線, 罰点がある水色の点線の 3 つの場合があり, ノイズの大きさをそれぞれ, $\kappa = 10^{-1}, 10^{-2}, 10^{-3}$ とする. 改めて, ノイズのない太い橙色の実線は, 演算子 A の呼び出し回数 N_{q} と ϵ の関係は, $N_{\mathrm{q}} \sim O(1/\epsilon)$ となり, これは Heisenberg スケーリングを達成できていることを示している.

ここで, 脱分極ノイズがある時の振る舞いについて, 図 4.2 から, 主に次の三つのことがわかる. まず一つ目が, 推定精度の誤差は, N_{q} が限られた範囲では, ほぼ Heisenberg 限界を達成できる. 具体的には, $\kappa = 0.01$ の四角の点がある青色の鎖線においては, 大凡 $N_{\mathrm{q}} \sim 10^4$ までの範囲で, $\kappa = 0.001$ の罰点がある水色の点線においては, 大凡 $N_{\mathrm{q}} \sim 10^5$ までの範囲で, 橙色の太い実線, つまりノイズのない場合とほぼ同じ推定精度の誤差を実現できる. これは, 脱分極ノイズがあったとしても, 古典的な手法の場合 (赤色の細い実線) を上回り, Heisenberg 限界に近いものを達成できる可能性があることを示しているものである.

一方で, N_{q} の大きな範囲では, つまり, 振幅増幅の回数である m_k が大きい値である時, 推定誤差がある値より小さくならなくなる. [*1] これが二つ目のポイントであり, この範囲では本手法を止める必要があることを示している. すなわち, 以下の関係式を満たす最大の整数として定義される $\bar{m}$ より, 振幅増幅の回数 m_k が小さければ, Heisenberg 限界がほぼ達成されることがわかる [88].

$$(2\bar{m}+1)(1-\mathrm{e}^{-\kappa}) \leq 1. \tag{4.18}$$

[*1] この状況は, $\kappa \neq 0$ の時に常に起きる. それは, 次の不等式が d'Alembert の収束判定法を用いることで導かれることからわかる.
$\epsilon_{\min}(\theta,\kappa) \geq (\mathcal{I}_{1,1}(\theta,\kappa))^{-1/2} \geq \left(\sum_{k=0}^{M} N_k 4(2m_k+1)^2 \frac{e^{-2\kappa m_k}}{1-e^{-2\kappa m_k}}\right)^{-1/2}$

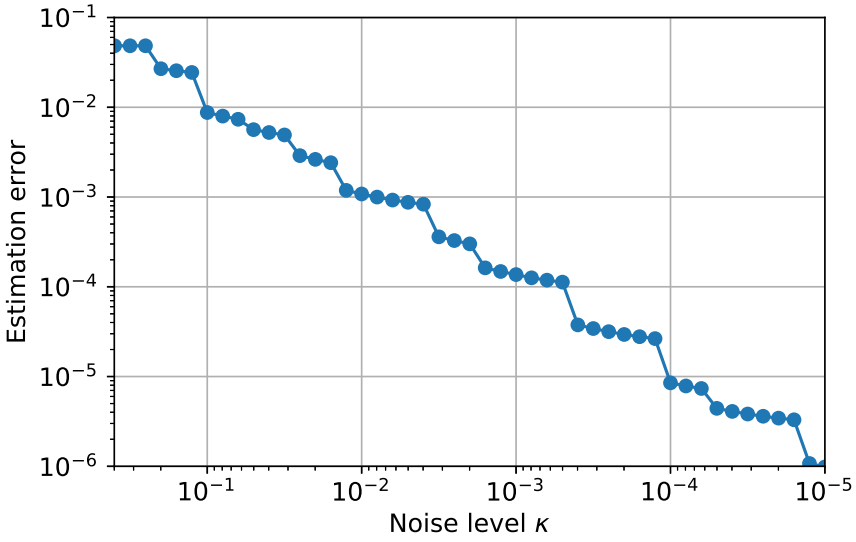


図 4.3　脱分極ノイズがある場合のノイズの大きさ κ と推定精度の関係

ノイズの大きさ κ と実現可能な推定精度 ϵ の関係を示している．なお, 推定したい値 $\theta = 0.659$, 振幅増幅の回数 $m_k = \lfloor 2^{k-1} \rfloor$, 測定回数 $N_k = 100, \forall k$ とする．

この式 (4.18) は, サンプルの数が十分に多い場合に, $N+1$ 番目の量子ビットが 1 を測定する確率がノイズの影響を受けないことを保証するための条件である．そして, 図 4.2 における星印は, ノイズの大きさ κ が与えられた時に, Heisenberg 限界を概ね達成できる $\bar{m}$ から導かれる N_{q} である．ただし, これはパラメーターが 1 つの場合に, つまり κ が既知の場合に導出したものであり, $\epsilon_{\min}(\theta, \kappa)$ はパラメータが 2 つの場合の, Fisher 情報行列から導いたものである．

最後に, ノイズの大きさ κ が小さくなればなるほど, $\epsilon_{\min}(\theta, \kappa)$ が小さくできることがわかる．よりこれを説明するために, 図 4.3 では, ノイズの大きさ κ と達成可能な推定精度の誤差 ϵ の関係を示す．なお, 推定する θ を 0.659, 任意の k に対して測定回数 N_{shot} を 100, 振幅増幅の回数は $m_k = \lfloor 2^{k-1} \rfloor$ とし, 図 4.2 と同様の条件である．例えば, もし, $O(1/\epsilon)$ の A の呼び出し回数を用いて, 推定精度の誤差 $\epsilon = 10^{-4}$ を要求される場合, ノイズの大きさ κ は, 10^{-3} より小さい必要があることがわかる．重要なことに, 図 4.3 は, ϵ と κ の間に擬似線形があることを示している．つまり, もし推定精度の誤差を 10 倍小さくしたければ, κ も 10 倍小さくする必要があることがわかる．これは, 式 (4.18) が二つのパラメータを推定する場合でも成り立つと推測されることを示唆している．

そして, ここから, 量子振幅推定を行う際の, 要求されるハードウェアの性能を求めることができ, 具体的に, モンテカルロ積分の例を用いて, 4.5 節で, 詳細に議論する．なお, 注意すべきことに, 多くの θ に関しては図 4.2,4.3 の傾向を示すが, 一部の θ については, $\epsilon_{\min}(\theta, \kappa)$ が大きな値を示す特殊な場合があり, それは 4.4 節で示す．

また, 振幅増幅の回数 m_k には改良の余地がある．例えば, $m_k = \lfloor r^{k-1} \rfloor (r$ は $r > 1$ を満たす実数) とした場合には, r の値に応じて $\bar{N}_{\mathrm{q}}$ の値が変化する．つまり, これは, m_k の更新戦略を変更することで, ノイズが存在する場合の達成可能な推定誤差を改善できることを示している．さらに重要なことに, 本節での 2 パラメータの推定問題では, r の値によって, 推定精度は大きく制限されることになる．これは, 4.4 節で述べる．なお, [73] では, ノイズパラメータは既知と仮定した上で, $m_k = k^{r'}$ の場合について議論を行っている．

本節の議論では, 前節で述べた数学的モデルを用いたものであり, 実際の量子コンピュータでは, このモデル

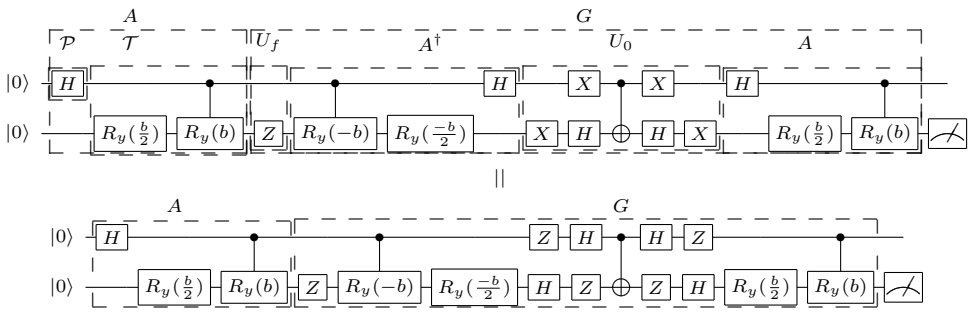


図 4.4　2 量子ビットを用いた場合の回路 ($m_k = 1$ の場合)

2 量子ビットを用いた, つまり $n = 1$ の場合の, 量子回路を示す. 上図は図 3.9 を $n = 1$ の場合で, 再掲したものであり, 下図は実際に実機で実行する際に, $HX = ZH, XH = HZ$ で置き換えることでゲートの数を減らした回路図である. なお, $A = \mathcal{T}(\mathcal{P} \otimes I_1)$,$G = -A\ U_0\ A^\dagger\ U_f$ である.

と完全に一致する訳ではない. つまり, これだけで要求される推定精度の誤差に達成できる κ の条件と言えるかはわからないので, 実際の量子コンピュータを用いて実験することで, 妥当性を説明する.

4.3　実デバイスを用いた実験

4.1 節では, 脱分極ノイズモデルを仮定した振幅推定法について導入を行い, 4.2 節では, A の呼び出し回数 Nq_G と推定精度の関係, およびノイズの大きさと推定精度の関係を議論した. 本節では, 実際の量子コンピュータを使って実験を行う. 具体的には, IBM Quantum *ibmq_valencia* [16] を使い, 最尤推定を用いた振幅推定法を行い, 脱分極ノイズモデルの妥当性を評価する. 特に, 振幅推定法を使うことで, 二乗の加速をすることが知られているモンテカルロ積分計算 [14] による検証を行う. モンテカルロ積分計算の具体的な手法は 3.5 節で説明したものであり, まず, 4.3.1 項で, 2 量子ビットを用いたケースを示し, 4.3.2 項で 3 量子ビットを用いた結果を示す. また, 2 量子ビットを用いたケースで, パラメータを変えた結果を付録 A 章に示す. なお, 具体的なソースコードに関しては [85] に公開している.

4.3.1　2 量子ビットを用いた実験

この項では, IBM Quantum *ibmq_valencia* の 2 つの量子ビットを用いた実験を行う. 具体的には, 式 (3.30) を計算した結果を, 一番, 単純な場合として $n = 1$ の場合, つまり $x = 0$ と $x = 1$ の二値のみから近似された離散和 $S(f)$ を考える. なお, IBM Quantum *ibmq_valencia* の 0 番目と 1 番目の量子ビットを用いて実験を行なった.

まず, この実験の計算したい確率 $p_G^{(k)}(1;\theta,\kappa)$ は, 式 (4.11) でも示しているが, 改めて次の通りである.

$$p_G^{(k)}(1;\theta,\kappa) = \frac{1}{2} - \frac{1}{2}\mathrm{e}^{-\kappa m_k}\cos(2(2m_k+1)\theta) \tag{4.19}$$

そして, 量子回路は, ユニタリー演算子 $\mathcal{P}$, $\mathcal{T}$, および G を用いて, 図 4.4 のように書ける. なお, 実際の量子コンピュータを用いる際には, 2 点, 技術的な工夫をしている.

- 演算子 Q の中央にある CNOT ゲートの前後にある 1 量子ビットゲート操作については, 図 3.9 に対して, 量子ゲートの基本操作から成り立つ関係 $HX = ZH, XH = HZ$ を元に置き換えをしている. これは, Z ゲートがフレームチェンジにより実装されており [83], パルスが当たらない仕様であることから, ノイズの影響を減らすためにである.
- 一方で, 複数の演算子 G を跨いでは, コンパイルによるゲートの置き換えを行わないようにしている.

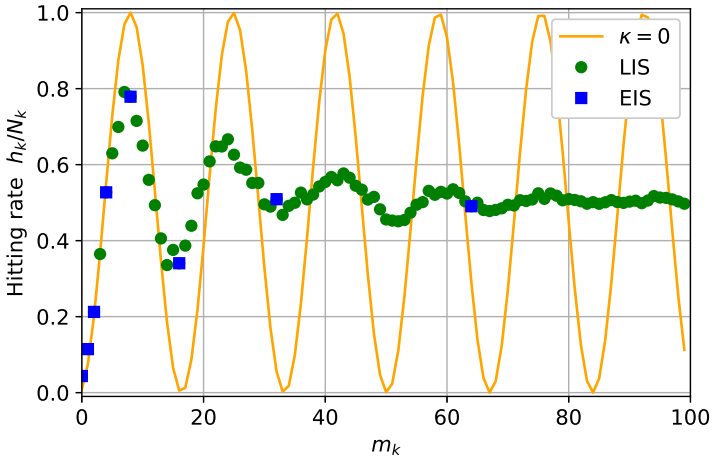


図 4.5　実機を用いた振幅増幅の結果（再掲）

超電導量子コンピュータ IBM Quantum *ibm_valencia* を用いて振幅増幅を行なった結果. 横軸には振幅増幅の回数 m_k, 縦軸には最後の量子ビットが 1 を観測した確率, つまり h_k/N_k であり, 橙色がシミュレーションによる結果（つまりノイズがない理想的な環境の場合）, 緑色の丸点及び青色の四角点が実機から得られた結果であり, それぞれ $m_k = k$ および $m_k = 2^k$ の点である. なお, 測定回数は全ての k に対して $N_k = 8,192$ としている.

表 4.1　2 量子ビット (実機) を用いた際のパラメータ一覧

項目	文字	値
測定回数	N_{shot}	100
振幅増幅の回数	m_k	2^k
推定する値	θ	0.659
誤差を評価するための繰返回数	N_{rep}	1,064

今回の実験ではコンパイラーの性能を試したい訳ではなく, ノイズの影響を確認することが主な目的だからである.

まず, 図 4.5 は, 本章の冒頭で示した図を再掲する. これは, $b = \pi/20$ とした場合, つまり $S(f) = \sin^2\theta = 0.0077$ の場合で, $N_{\text{shot}} = 8,192$ である. 横軸が振幅増幅 m_k の回数で, 縦軸が補助ビット（つまり二番目の量子ビット）が 1 を観測した確率である. そして, 緑色の丸い点が $m_k = k$ の場合, 特に青色の四角は $m_k = 2^k$ の場合の実際の超電導量子コンピュータから得た結果である. また, 橙色の実線がノイズがない理想的な量子コンピュータをシミュレーションした場合の結果である. この図 4.5 から, 振幅増幅を繰り返し実行していくにつれて, 指数関数的に減衰していき 0.5 に収束していく傾向があることがわかり, 極小モデルとして, この減衰振動をモデル化するために, 4.1, 4.2 節で議論をしている脱分極ノイズ (4.7) を導入する.

そして, 式 (4.19) のモデルに基づき, θ と κ を推定するために, 最尤推定を用いる振幅推定法を行う. 改めて, 本手法は, 式 (4.19) にある確率を用いて, 式 (4.12), (4.13) から, 尤度関数を構築し, それを式 (4.14) のように最大化を行う. なお, ここでは振幅増幅の回数の種類 M を 6 パターン, つまり, $M = 1, 2, 3, 4, 5, 6$ を検証した. そして, 振幅増幅の回数 $m_k = 2^k$ であり, 測定回数は $N_{\text{shot}} = 100$ とし, $b = 2\pi/5$ つまり $\theta = 0.659$ とした. 実験環境をまとめて表 4.1 に示し, 実験結果を図 4.6 に示す.

図 4.6 では, 演算子 A の呼び出し回数 $Nq_G \left(= \sum_{k=0}^{M} N_k(2m_k + 1)\right)$ と, 推定誤差の関係を示す. 細い赤

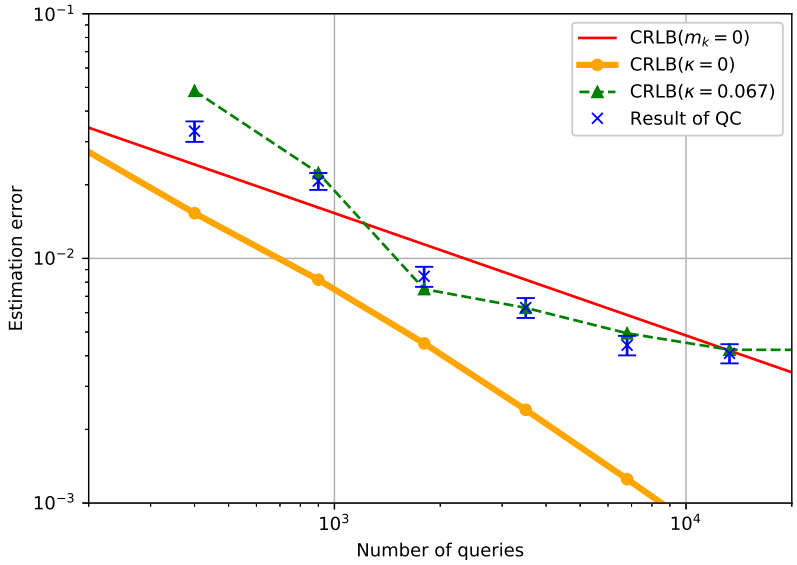


図 4.6　2 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係

赤色の細い実線, および橙色の太い実線は, Cramér–Rao の下限であり, それぞれ, 古典的な場合 ($m_k = 0$) とノイズのない量子の場合 ($m_k = 2^k$, $\kappa = 0$) である. そして, 緑色の点線は, ノイズの大きさ $\kappa = 0.067$ とした時の Cramér–Rao の下限であり, κ の値は実機のデータから求めたものである. 青色のバツ点は, 真の値と, 実機から最尤推定を用いて推定した値の標準偏差である.

色, 及び, 太い橙色の実線は, 式 (4.17) から導かれる Cramér–Rao の不等式から導かれる, ノイズがない場合 ($\kappa = 0$) の下限 $\epsilon_{\min}(a, \kappa)$ であり, それぞれ, 古典的な場合, つまり, $m_k = 0$ の場合と, 量子の Heisenberg 限界を達成する場合, つまり, $m_k = 2^k$ の場合である. そして, 青い罰点は, 最尤推定法を用いた実機の結果の標準偏差を示しており, エラーバーは, 標準誤差の 3 倍を表している. 例えば, $M = 4$ における青の罰点, つまりは, $N_{\mathrm{q}} \sim 3.5 \times 10^3$ の点では, 振幅増幅の回数 m_k が $k = 0, \ldots, 4$ の時, つまり $m_k = 0, 1, 2, 4, 8$ のデータを用いて最尤推定をしており, その推定精度の誤差の値が 0.65×10^{-2} となっている. なお, 同じ実験を $1,064$ 回繰り返して, 平均を計算することでブレ幅を抑えるようにするとともに, その結果を用いて, エラーバーで, 3 倍の標準誤差を示している. そして, 緑色の破線は, ノイズがある時の Cramér–Rao の下限を示している. なお, そのノイズの大きさである κ は, 実験から得られた全ての結果, つまり, 100 回の測定を $1,064$ 回の繰り返した $106,400$ のデータで, $M = 6$ まで, つまり $m_k = 0, 1, 2, 4, 8, 16, 32$ の結果を全て用いて, 一回の最尤推定を行って得られた $\kappa = 0.067$ と, 推定したい真の $\theta = 0.659$ から計算して得たものである.

図 4.6 からは, 主に二つのことがわかる. 一つ目は, 実機から推定した青色の罰点は, 理論的な下限である緑線とほぼ一致している. 若干のずれは, ノイズが完璧にはモデリングできないことによるものであり, ゲート操作の回転誤差や, 近隣の量子ビットの不必要な接続などから起こる脱分極ノイズ以外のノイズによるものと推測される. 二つ目は, 一部の実機の結果である青色の罰点（及び, 緑色の実機のノイズの大きさを反映した Cramér–Rao の下限）は, 古典的な赤色の細い実線より下回っていることである. ノイズの大きさが十分に小さくないことから, ノイズのない量子コンピュータを用いた橙色の太い実線とは差があるものの, 一方で, 古典的な手法より良い精度を得られている点がいくつかある. ゆえに, ノイズがある量子コンピュータであっても, 演算子 A の呼び出し回数においては, 古典的なコンピュータを上回る可能性があることを示唆している. そして, 最後に, 他の量子ビットや他の θ を持ちいた場合でも同様の結果が得られることを, 付録 A 章に示す.

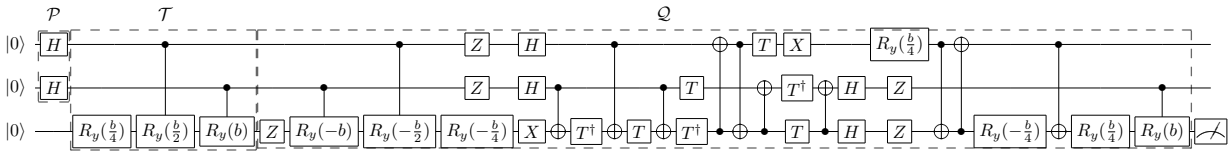


図 4.7　3 量子ビットを用いた場合の回路 ($m_k = 1$ の場合)

3 量子ビットを用いた場合, つまり $n = 2$ の場合の, 振幅増幅 ($m_k = 1$) の回路を示す. なお, 一番上の量子ビットと中央の量子ビットは物理的に接続していないため, SWAP ゲートをうまく入れることで, CNOT ゲートの数の増加を抑える実装にしている.

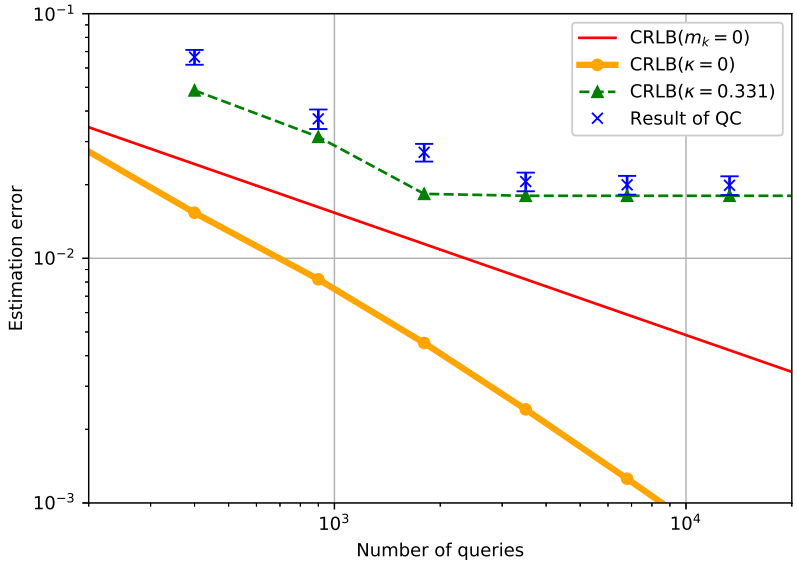


図 4.8　3 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係

赤色の細い実線, および橙色の太い実線は, Cramér–Rao の下限であり, それぞれ, 古典的な場合 ($m_k = 0$) とノイズのない量子の場合 ($m_k = 2^k$, $\kappa = 0$) である. そして, 緑色の点線は, ノイズの大きさ $\kappa = 0.331$ とした時の Cramér–Rao の下限であり, κ の値は実機のデータから求めたものである. 青色のバツ点は, 真の値と, 実機から最尤推定を用いて推定した値の標準偏差である.

4.3.2　3 量子ビットを用いた実験

この項では, 3 つの量子ビットを用いた場合の実験結果を示す. つまり, 式 (4.19) において, $n = 2$, つまり $2^n = 2^2$ の分割をした離散積分を考える. ここでは, $b = 2\pi/5$ であり, 推定したい $\theta = 0.665$ とし, 実際に実行した量子回路の, $m_k = 1$ の場合を, 図 4.7 に示す. なお, *ibmq_valencia* においては, 3 つの量子ビットを使う場合には, 全てのビットが相互結合していないことには注意が必要である. 単純に実装を行うと, 多くの SWAP ゲートが発生してしまう可能性があり, 結果 CNOT ゲートの数が増え, ノイズの影響を受けやすくなってしまう可能性がある. そこで, 接続が限定されることによるゲート数の増加が減るように実装をしている. そして, ここでは, 図 4.7 の一番上の量子ビットを実機の 0 番目に, 図の真ん中の量子ビットを実機の 2 番目に, そして, 一番下の量子ビットを実機の 1 番目に来るように配置した.

3 量子ビットを用いた場合の結果を図 4.8 で示す. 図 4.8 では, 4.3.1 項の図 4.6 と同様に示している. つま

り, 赤色の細い実線, 橙色の太い実線, 緑色の破線が Cramér–Rao の下限であり, それぞれ古典的な場合, ノイズのない理想的な量子の場合, 実機のノイズから推定される κ を用いた量子の場合である. そして, 青色のバツ点およびエラーバーは実機を用いた結果から得られる標準偏差および 3 倍の標準誤差を示したものである.

2 量子ビットの場合と同様にノイズの大きさ κ を推定すると, $\kappa = 0.331$ となる. これは, 2 量子ビットの場合の $\kappa = 0.067$ に比べて非常に大きいため, ノイズがある時の Cramér–Rao の下限である緑色の破線は高い位置になり, 赤い細い実線を下回ることができない. つまり, 現状で, この三角関数の積分において 3 量子ビットを用いた場合, 振幅推定アルゴリズムでは量子の優位性を得ることができないことがわかる. 一方で, 実機から得られた実験結果である青色のバツ点と, Cramér–Rao の下限である緑色の破線は, よく一致しており, 脱分極ノイズモデルが悪くないモデルであることを示している. また, ノイズの大きさ κ が必要程度小さくなくては, 量子の優位性を受けられないことを実験的にも確認できることであり, これは 4.2 節での議論の重要性を裏付けるものである.

なお, 通常に Qiskit でコンパイルを行うと, 2 量子ビットの場合, 量子回路が非常に短縮化されてしまうことがある. これは任意の 2 量子ビット状態が 3 つの CNOT ゲートとシングルゲートたちで示すことができることを知られているからである. 更に, 3 量子ビットの場合は, 6 つの CNOT にすることができることも知られており, 4 量子ビットの場合は接続性を加味した場合 17CNOT であることが知られている [91]. しかしながら, この問題は拡張性が保証されておらず, 量子ビットの増加に伴い, CNOT 数がどう増えるかが不明であることから, 今回はあえて, コンパイルを行わずに, ノイズの影響を調べることにしている.

4.4 例外的な変数についての議論

4.1, 4.2 節で, 脱分極ノイズモデルを仮定した上での, 最尤推定を用いた振幅推定を議論し, 4.3 節では, 実際の超電導量子コンピュータである IBM Quantum を用いて, デモンストレーションを行い, 本手法の有用性の検証を行った. これからの節では, 本問題設定におけるいくつかの重要な議論を行う. まず, 本節では, 例外的な変数に関する議論を行う. つまり, 4.2 節において, 演算子 A の呼び出し回数 N_q と推定精度の関係, 及び, 脱分極ノイズの大きさ κ と推定精度の関係について議論したが, 図 4.2 及び図 4.3 のような傾向を示さない θ と κ の変数の組みが存在する. これに関して, 本 4.4 節で議論を行う. そして, 4.5 節ではハードウェアの見積もりを行い, 4.6 節では計算量に関する議論を行う.

4.4.1 例外的な変数について

本 4.4.1 項では, 例外的な変数についての説明を行い, 次 4.4.2 項では, この変数たちの頻度について議論を行い, 頻度が少ないことを示す. 最後の 4.4.3 項では, 例外的な変数の回避方法について, 二つ手法を提案する.

4.2 節における議論は, ノイズの大きさである κ と $\epsilon_{\min}(\theta, \kappa)$ の間に擬似線形が成り立ち, そして, 振幅増幅の回数 m_k が, $m_k \lesssim \bar{m}$ の範囲において, 演算子 A の呼び出し回数 $N_{\mathrm{q}} = \sum_{k=0}^{M} N_k(2m_k + 1)$ に対して, 推定精度の分散 ϵ がほぼ Heisenberg 限界になるという条件に基づいている. しかしながら, これは常に成り立つ条件ではない. 例えば, $\theta = \pi/8$, $m_k = 2^{k-1}$ の場合においては, 推定精度の分散 $\epsilon_{\min}(\theta, \kappa)$ は, κ に対して擬似線形にはならないことを図 4.9 の橙色の破線 (三角点) で示している. これは, ノイズの大きさ κ が十分に小さい場合でも, θ を正確に推定できないことを示している. この現象は, 複数のパラメータ推定問題に起因するものであり, 一般に, $\boldsymbol{\theta}$ に対する推定精度の分散共分散行列が, Cramér–Rao の不等式 (4.4) を満たす, つまり, $\mathrm{Cov}(\hat{\boldsymbol{\theta}}) \geq J_G^{-1}(\boldsymbol{\theta})$ を満たすことによるものである. もし, $\det J_G(\boldsymbol{\theta})$ がゼロに近い値であれば, 推定を効率的に行えないことは明らかである. なお, この現象は, 1 変数の推定問題では起こらないものである. また, ノイズのある量子コンピュータにおいて, 優位性を得るためには, 推定する変数にノイズから派生するパラメータを組み込む必要があり, このような解析が重要な議論である.

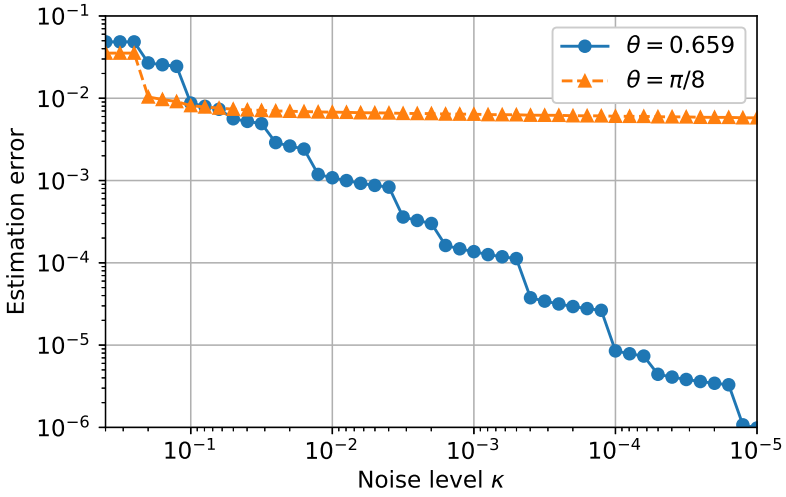


図 4.9　例外的な場合のノイズの大きさ κ と推定誤差の関係

推定誤差の下限 $\epsilon_{\min} = \sqrt{(J_G^{-1})_{1,1}}$ と脱分極ノイズの大きさ κ の関係を示す．青色の実線 (丸点) は図 4.3 で示している結果であり，$\theta = 0.695$ の場合である．橙色の破線 (三角点) は，推定する値 $\theta = \pi/8$ とした結果である．なお，$\{m_k\} = \{0, 2^0, 2^1, 2^2, \cdots\}$，$N_{\text{shot}} = 100$ とする．

4.4.2　例外的な変数の出現頻度

続いて，本項では，例外的な変数の頻度について議論を行う．まず，例外的な変数がどの程度存在するのかを直感的に理解するために，縦軸と横軸それぞれに，θ と κ を用意し，等高線で推定精度の誤差の下限である $\epsilon_{\min} = \sqrt{(J_G^{-1})_{1,1}}$ を図 4.10 に表示する．この等高線は，ほぼ全ての θ に対して，κ が小さくなると，$\epsilon_{\min}$ が線形に小さくなることを示している．つまり，κ が 1 桁小さくなると，$\epsilon_{\min}$ が 1 桁下がることがわかる．しかしながら，κ の値に依存しない例外的な θ の値が存在することが改めて確認できる．例えば $\theta = \pi/8$ においては，縦に長い筋があり，κ の値によらず，$\epsilon_{\min}$ がほぼ変わらないことがわかる．

この事象は，前項で述べた通り，Fisher 情報行列 $J_G(\boldsymbol{\theta})$ の行列式がゼロに近い値であることに起因している．それを確認するために，図 4.11 では，横軸に θ を取り，縦軸には，$\epsilon_{\min}$ の値を青色の実線で，例外度合いを表す $\alpha = J_{1,2}^2 / J_{1,1} J_{2,2}$ の値を赤色の点線で示す．そして，(A) では $\kappa = 10^{-2}$ の時，(B) では $\kappa = 10^{-3}$ の時である．α は $0 < \alpha \leq 1$ であり，$\alpha = 1$ は $\det J_G(\theta, \kappa) = 0$ を意味し，θ の推定が困難である時である．図 (A) と (B) において，$\alpha = 1$ の例外的な状況においては，推定精度の誤差が比較的大きな値を示している．

そして，この例外的な変数の割合を線密度にて，定量的に分析を行う．つまり，$[0, \pi/2]$ の一様分布から，θ のサンプルを $N_{ld} = 10^5$ 個用意し，$\alpha > 0.9$ となる割合を計算することで，線密度を計算する．κ の値を変えた際の線密度を表 4.2 に示す．この表より，ほぼ κ の値に寄らず，$1\% \sim 2\%$ であることがわかる．これは次の事象が複合的に作用していることによるものである．つまり，κ の値に応じて，$\alpha > 0.9$ を満たす数が反比例する一方で，例外的な領域の平均の長さが比例していることによるものである．例外的な変数の線密度は，例外的な領域の平均の長さと数の積に寄るものであい，κ には依存していない．

ただし，この線密度は $\kappa \to 0$ の極限で有限の値となるが，一方で，$\kappa = 0$ の場合には例外的な変数は存在しない．これは本質的にノイズがあるか否かに起因しており，ノイズがない場合には，$J_{G\ 1,1}$ を用いて，推定精度の誤差の下限を求めることができるが，ノイズの大きさがわからない場合には，$(J_G^{-1})_{1,1}$ を用いるべきである．

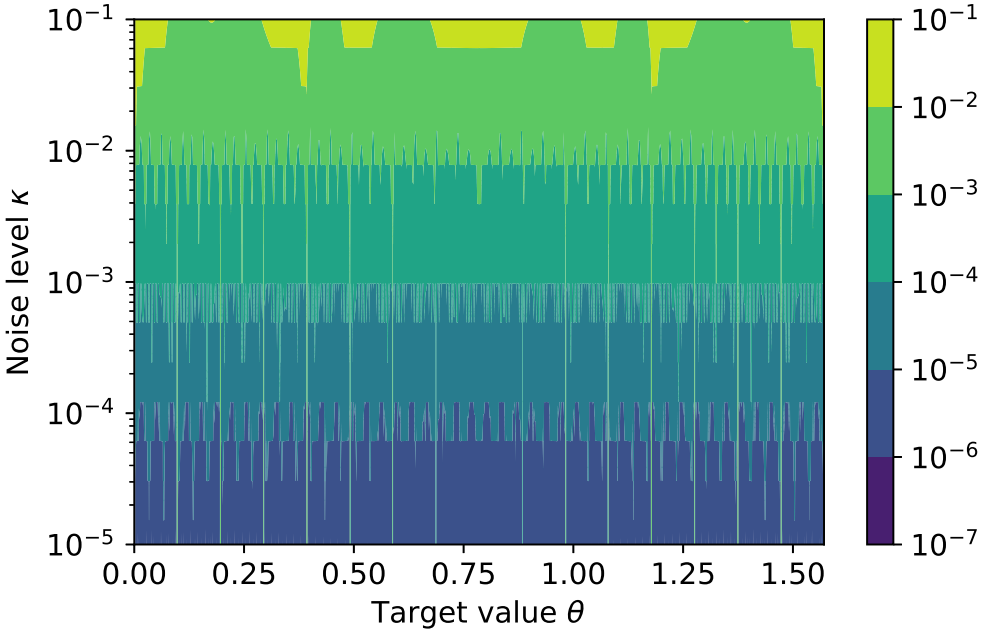


図 4.10　例外的な場合の, 目標の値と, ノイズの大きさ κ と推定誤差の関係

横軸に推定したい値を, 横軸にノイズの大きさとして, 推定誤差の下限 $\epsilon_{\min} = \sqrt{(\mathcal{J}_{\mathcal{G}}{}^{-1})_{1,1}}$ を等高線で示す.

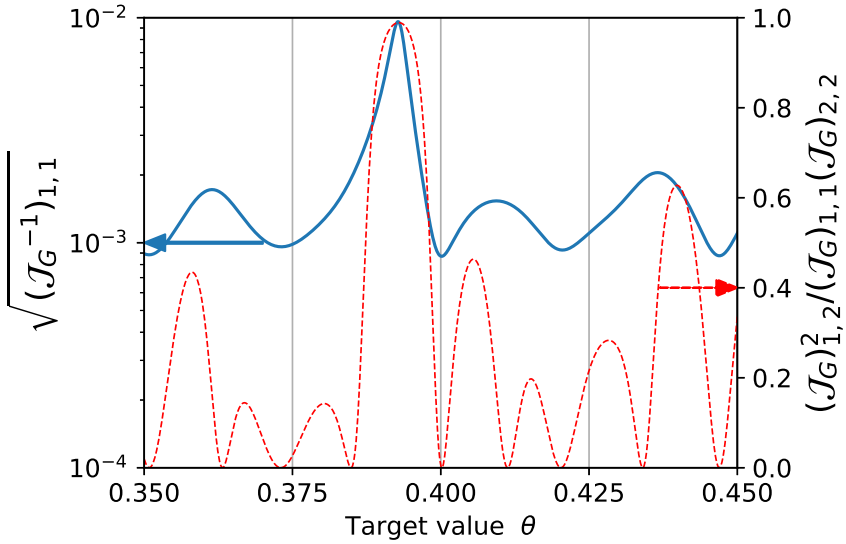
4.4.3　例外的な変数の回避方法

この節の最後の項では, 例外的な変数の回避方法を二つ, 提案する. 一つは, 振幅増幅の回数の数列である m_k を工夫することによるもので, m_k を変えると Fisher 情報行列の行列式が変化することによるものである. もし, 例外的な変数が検出された場合には, その状況を回避するために, 別の m_k の数列を試すことである. 例えば, $\{m_k\} = \{0, \lfloor 2.5^0 \rfloor, \lfloor 2.5^1 \rfloor, \lfloor 2.5^2 \rfloor, \cdots\}$ とすることで, $\theta = \pi/8$ の場合を回避することができる. 実際, ノイズの大きさ κ と推定精度の誤差を図 4.12 の赤の点線で示しており, 準線形の関係になることがわかる. これは, m_k を適切に選ぶことで, Fisher 情報行列の行列式が小さくならなくできることの証しである. つまり, これは経験的 Fisher 情報量を計算し, 例外的な変数を検知した場合には回避することができると考えられる.

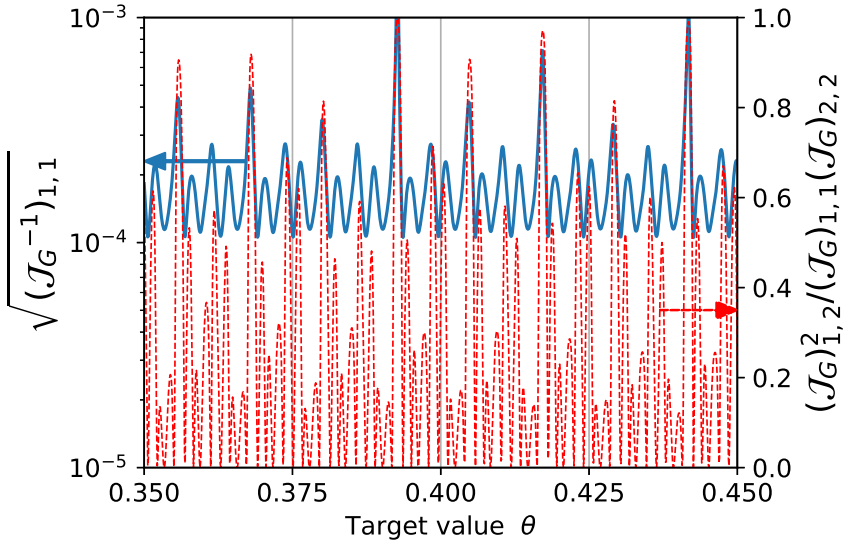
そして, 二つ目の回避する方法は, 推定する θ を変えるものである. 例外的な変数を検知した場合, 新たな補助量子ビットを導入することによるものであり, $R_y(\phi)$ 回転を次の式で行うものである.

$$\begin{aligned}\mathcal{A}\,|0\rangle_{n+2} &= \sin\theta\,|\tilde{\Psi}_1\rangle_n\,|1\rangle\,|0\rangle + \cos\theta\,|\tilde{\Psi}_0\rangle_n\,|0\rangle\,|0\rangle \\ &\xrightarrow{Ry(\phi)} \cos\phi\sin\theta\,|\tilde{\Psi}_1\rangle_n\,|1\rangle\,|0\rangle + \sin\phi\sin\theta\,|\tilde{\Psi}_1\rangle_n\,|1\rangle\,|1\rangle \\ &\quad + \cos\phi\cos\theta\,|\tilde{\Psi}_0\rangle_n\,|0\rangle\,|0\rangle + \sin\phi\cos\theta\,|\tilde{\Psi}_0\rangle_n\,|0\rangle\,|1\rangle\,.\end{aligned} \tag{4.20}$$

これにより, 例えば, 最後の $n+1$ 番目と $n+2$ 番目の量子ビットの状態が $|1\rangle\,|1\rangle$ である状態を測定すると, 観測する値が $\sin^2\phi\sin^2\theta$ となり, 例外的な場合を回避することができる.



(A) $\kappa = 10^{-2}$



(B) $\kappa = 10^{-3}$

図 4.11　例外的な場合の推定誤差と, 例外度合いの関係

推定誤差 $\epsilon_{\min} = \sqrt{(J_G^{-1})_{1,1}}$ を, 青色の実線で表し, 例外度合いを示す $\alpha = (J_G)^2_{1,2}/(J_G)_{1,1}(J_G)_{2,2}$ を赤色の点線で示す. なお, ノイズの大きさである κ は, (A) の時に $\kappa = 10^{-2}$, (B) の時に $\kappa = 10^{-3}$ である.

表 4.2　例外的な場合の発生割合

$[0, 1]$ の範囲における例外的なパラメータの表れる線密度を表にしたものである．線密度が二項分布に従うとし，その統計誤差を示す．

κ	例外的な場合の線密度
10^{-10}	1.56 ± 0.04 %
10^{-9}	1.84 ± 0.04 %
10^{-8}	1.90 ± 0.04 %
10^{-7}	1.56 ± 0.04 %
10^{-6}	2.01 ± 0.04 %
10^{-5}	1.70 ± 0.04 %
10^{-4}	1.67 ± 0.04 %
10^{-3}	2.02 ± 0.04 %
10^{-2}	1.28 ± 0.04 %
10^{-1}	0.00 ± 0.00 %

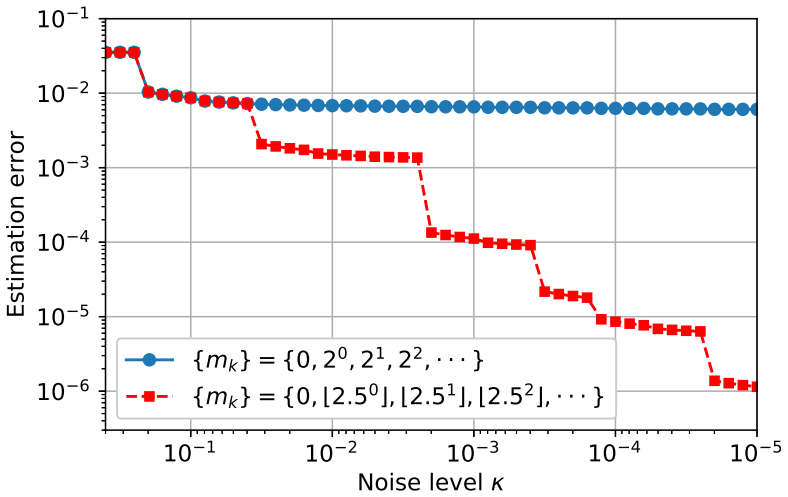


図 4.12　例外的な場合のノイズの大きさ κ と推定誤差の関係 (m_k を変えた場合)

例外的な場合 ($\theta = \pi/8$) における，κ と $\epsilon_{\min}$ の関係を示す．青色の実線 (丸点) は $\{m_k\} = \{0, 2^0, 2^1, 2^2, \cdots\}$ の時を，赤色の点線 (四角点) は $\{m_k\} = \{0, \lfloor 2.5^0 \rfloor, \lfloor 2.5^1 \rfloor, \lfloor 2.5^2 \rfloor, \cdots\}$ の時を示す．

4.5　ハードウェアの見積もり

この節では，最尤推定を用いた振幅推定法を用いる際に，必要なハードウェアの性能の見積もりを行う．改めて，図 4.3 から，Heisenberg 限界を達成できる状況下における，推定精度の誤差 ϵ を達成することに必要な最大のノイズの大きさ $\bar{\kappa}$ を示したものである．そして，このノイズの大きさ $\bar{\kappa}$ を，量子回路の基本ゲートのエラーに結びつけ，更にアルゴリズムに必要な時間を評価する．なお，4.4 節で議論した例外的な変数に関しては検知でき，対応できるものとして，議論から外すものとする．計算のステップは以下の 4 のステップとなる．

1. まず, 振幅推定の問題と, 目標となる推定精度の誤差 ϵ が与えられるとする. それは, 振幅増幅の演算子である $\mathcal{G}$ が（モンテカルロ計算の場合には, $\mathcal{P}$ と $\mathcal{T}$ の演算子が）与えられ, つまり, それらの演算子を構成する基本ゲートの集合を得ることができる. 例えば, 図 4.4 や図 4.7 のような形である.
2. 次に, 与えられている目標となる推定精度の誤差 ϵ に対して, 図 4.3 を用いて, 任意の k に対して N_k を固定した場合に, 許容される最大のノイズの大きさ $\bar{\kappa}$ を求めることができる. そして, この計算に必要な $\bar{\kappa}$ と, 実際に実機を用いて得ることができる $\hat{\kappa}$ を比較することで, 必要な推定値 $\hat{\theta}$ が得られるかどうかを確認することができる. 更に, 式 4.18 により, Heisenberg 限界が達成可能な振幅増幅の回数の最大値 $\bar{m}$ を得ることができる. 正確には, $\bar{m} = 0.5/(e^{\bar{\kappa}} - 1)$ であり, 例えば, $\bar{\kappa} = 0.1$ の時には $\bar{m} = 5$ の振幅増幅の回数しか Heisenberg 限界を達成できないが, $\bar{\kappa} = 0.001$ であれば, $\bar{m} = 500$ 回の, 振幅増幅が可能となる.
3. 続いて, 振幅増幅の回数の最大値である $\bar{m}$ と, 演算子 $\mathcal{G}$ を構成するゲートの数 L から, それぞれの振幅増幅の回数 $m_1, m_2, \ldots, \bar{m}$ に応じた実行時間を見積もることができる. $\bar{m}$ の際の実行時間が, 実機の実行可能な計算時間以内であるか否かで, 実行可能であるかを判断することができる. そして, 各振幅増幅の回数 $m_1, m_2, \ldots, \bar{m}$ の合計を考えることで, 本アルゴリズムを用いる実時間上のメリットがあるか否かを判断することができる.
4. 最後に, 各ゲートのエラーの大きなについて考える. 全ての量子ビットが, 脱分極ノイズのみが作用していると仮定すると, 演算子 $\mathcal{G}$ の成功確率 $p = e^{-\bar{\kappa}}$ は, $\prod_{i=1}^{L}(1 - \epsilon_i)$[18] として見積もることができる. ただし, L は演算子 $\mathcal{G}$ を構成するゲートの数であり, ϵ_i は, i 番目の量子ビットの脱分極ノイズのエラーの起こる確率である. つまり, この式から一つのゲートのエラーの値を大まかに見積もることができ, このゲートエラーと実際の実機のゲートエラーの大きさ（ただし, これは標準的な無作為なベンチマークテストによって求めたものである）と比較することで, アルゴリズムの実行可能性を検証することができる.

詳細な計算方法と具体例における計算の値を, 表 4.3 に示す. なお, 以下の例でも述べるが, いくつの仮定をしている.

- 振幅増幅の回数の数列である m_k は, 2^k とする.
- 一つの振幅増幅 m_k を固定した際の測定回数 $N_{\mathrm{shot}} = 100$ とする.
- 2 量子ビットゲートのエラーの大きさ ϵ_{d} は, 1 量子ビットゲートのエラーの大きさ ϵ_{s} の 10 倍とする. つまり, $\epsilon_{\mathrm{d}} = 10\epsilon_{\mathrm{s}}$ とする.
- 量子ビット間の接続は全結合とする.

一例として, 2 変数間で相関のある 5 変数の多重積分, 例えば, $\int(x_1x_2 + x_2x_3 + x_3x_4 + x_4x_5)d\mathbf{x}$ のような積分を考える. 5 変数以上の重積分のような計算をする際には, グリッドに基づいた数値積分ではなく, モンテカルロ計算を用いることが一般であるため, この例を用いている. ここで, 上記ステップを一つずつ説明していく.

1. Heisenberg 限界を達成する演算回数を用いて, 目標とする推定精度の誤差 $\epsilon = 0.001$ とする.
2. すべての k に対して, $N_{\mathrm{shot}} = 100$ とすると, 図 4.3 から $\bar{\kappa} = 0.005$ となる. そして, 量子ビットは 99 個が必要であり, （48 個のゲート分解に使う補助量子ビットを含む）, そして, 振幅増幅の回数は, $\bar{m} = 99$ となる.
3. そして, 実際のいくつかの量子コンピュータでは, ゲートオペレーションの時間が公開されており, 例えば前節で用いた *ibmq_valencia* という超電導デバイスでは, 1 量子ビットゲートの時間は 7.1×10^{-8} 秒, 2 量子ビットゲートの時間は 2.8×10^{-7} 秒, そして, 測定時間は, 3.5×10^{-6} 秒である. 更に, 1 ショット毎のインターバルの時間を, 測定してから次の回路を実行するまでの時間を, 実際の実行時間の 10 倍

表 4.3　ハードウェアの見積もりの各数値の一覧

項目	計算式	値	備考
N_{nq} (積分用の量子ビットの数)	$\log_2 1/\epsilon$	10	$\epsilon = 0.001$
N_{tnq} (量子ビットの総数)	$2N_{\mathrm{nq}}N_{\mathrm{int}} - 1$	99	$N_{\mathrm{nq}}N_{\mathrm{int}} + 1$ は振幅推定用 $N_{\mathrm{nq}}N_{\mathrm{int}} - 2$ は $\mathbf{U}_f$ の分解の補助ビット
$\bar{\kappa}$ (ノイズの大きさ)	$5\epsilon/\sqrt{N_k}$	0.005	図 4.3 参照.
N_y (CCR_y の数)	$N_{\mathrm{nq}}(N_{\mathrm{nq}} - 1)N_{\mathrm{int}}^2/2$	1,000	$\mathcal{T}$ による
N_s ($\mathcal{G}$ の 1 量子ビットのゲート数)	$2(N_{\mathrm{nq}}N_{\mathrm{int}} + 1 + 6N_y) + 12N_{\mathrm{nq}}N_{int} - 15$	12,687	$2(N_{\mathrm{nq}}N_{\mathrm{int}} + 1)$ は, $\mathcal{P}$ と $\mathcal{P}^{-1}$ に起因. $2(6N_y * 2)$ は, $\mathcal{T}$ と $\mathcal{T}^{-1}$ に起因. $12N_{\mathrm{nq}}N_{\mathrm{int}} - 15$ は, $\mathbf{S}_0$ の分解.
N_d ($\mathcal{G}$ の CNOT ゲートの数)	$8N_y * 2 + 6N_{\mathrm{nq}}N_{\mathrm{int}} - 5$	16,295	$8N_y \times 2$ は, $\mathcal{T}$ と $\mathcal{T}^{-1}$ に起因. $6N_{\mathrm{nq}}N_{\mathrm{int}} - 5$ は, $\mathbf{S}_0$ の分解.
ϵ_s (1 量子ビットゲートエラー) ϵ_d (CNOT gate error)	$\mathrm{e}^{-\bar{\kappa}} = (1 - \epsilon_s)^{N_s}(1 - \epsilon_d)^{N_d}$	2.8×10^{-8} 2.8×10^{-7}	近似計算を含み, $\epsilon_s = 0.1\epsilon_d$ を仮定する.
$\bar{m}$ ($\mathcal{G}$ の最大数)	$(2\bar{m} + 1)(1 - \mathrm{e}^{-\bar{\kappa}}) < 1$	99	式 (4.18) 参照
t_{AA} ($\mathcal{G}$ の計算時間)	$t_{AA} = t_s N_s + t_d N_d$	5.4×10^{-3}	t_s と t_d は, 1, 2 量子ビットのゲートの時間で, IBM Quantum の実機の値で 7.1×10^{-8}, 2.8×10^{-7} 秒である.
$t_{\bar{m}}$ ($\bar{m}$ の時の実行時間)	$t_{AA} \times \bar{m} + t_m$	0.54	t_m は測定時間で 3.5×10^{-6} 秒である.
t_t (総実行時間)	$\sum_{k=1}^{N}((t_{AA}m_k + t_m + t_i) * N_k)$	1,082	N は, $m_N = \bar{m}$ に起因. t_i は, ショット毎の時間間隔で 10 倍で置く. $N_k = 100$ は全ての k に対するショット数.

を仮定する. この条件の元で, 振幅増幅の回数 $\bar{m}$ に対しては, 一回の回路の実行時間が 0.54 秒である. そして, 本アルゴリズム全体の実行時間は 1,082 秒となる.

4. 最後に, Heisenberg 限界に従うゲートエラーの大きさについて議論する. そして, 2 量子ビットゲートのエラーの大きさ ϵ_{d} は, 1 量子ビットゲートのエラーの大きさ ϵ_{s} の 10 倍とすると, それぞれの大きさは, $\prod_{i=1}^{L}(1 - \epsilon_i)$ から, $\epsilon_{\mathrm{d}} = 2.8 \times 10^{-7}$, $\epsilon_{\mathrm{s}} = 2.8 \times 10^{-8}$ となる. なお, 現在の *ibmq_valencia* の実機の値は, おおよそ, $\epsilon_{\mathrm{d}} \sim 1.0 \times 10^{-2}$ と, $\epsilon_{\mathrm{s}} \sim 1.0 \times 10^{-3}$ である.

重要な点は, 現在の利用可能な実際の量子コンピュータにおけるゲートエラーの大きさと, 与えられた問題における必要なゲートエラーの大きさの差を評価することができることである. また, 量子コンピュータの実行時間が, 0.54 秒必要であり, これは現在のデバイスより大幅に長い可能性がある. これらのギャップは非常に大きいように見えるが, 一方で, このギャップが明確になることでどの程度のハードウェアの進展が必要かがわかることは非常に意味のあることである. 更に, ハードウェアの進展だけでなく, アルゴリズムとしての改良もこのギャップを埋めることに繋がることであり, 例えば, [87] の研究では, 回路の深さと量子ビットの数を減らすことができる方法が提案されている.

この節の最後に, 3.3 節で行った実機を用いた際のノイズの大きさについて, 公開情報に基づいたノイズの大きさとが同程度であることを述べる. 3.3 節で用いた回路では, 4.3.1 項の 2 量子ビットの場合（図 4.4 を参照）では, 一回の振幅増幅の回路には, 5 個の CNOT ゲートが含まれており, 4.3.2 項の 3 量子ビットの場合（図 4.7 を参照）では, 16 個の CNOT ゲートが含まれている. なお, 図 4.7 には, R_y ゲートがあるが, これは CNOT ゲート二つから構成されている. そして, 2 量子ビットの場合では, CNOT ゲートのエラーの大きさが 0.00565 であり, $\kappa \sim -\ln((1 - 0.00565)^5) \sim 0.0283$ となる. また, 3 量子ビットの場合では, 異なる量子ビット間でのゲートエラーの値は異なるため二種類の CNOT ゲートがあり, それぞれのゲートのエラーの大きさが 0.008923 と, 0.01119 であり, またそれぞれ 8 個ずつ使っていることから, $\kappa \sim -\ln((1 - 0.008923)^8 * (1 - 0.01119)^8) \sim 0.162$ となる. これに 1 量子ビットのエラーなどを加味すると,

$\kappa = 0.067$ と $\kappa = 0.331$ の推定値に近づくことがわかる.

4.6 計算量に関する議論

本 4 章の最後の議論として, 計算量を取り扱う. つまり, ノイズがあり, そのノイズの大きさがわからない状況下における, 振幅増幅を行なった後の最尤推定における古典的な計算量が, $O((\ln(1/\epsilon))^{5/2})$ となり, それが量子での計算部分より少ないことを示す. なお, 本 4.6 節では, 前の 4.5 節での議論と同様に, 4.4 節で扱った例外的な変数に関しては, 検出し, 対処できるものとして, 議論を進める.

改めて, 脱分極ノイズモデルが存在する状況においては, 目標である振幅 θ と, ノイズの大きさ κ の二つの変数が存在するとしているため, 二次元の最尤推定を行う必要があることを 4.1 節で示した. 例えば, グリッドサーチを用いる場合, 推定精度 ϵ を達成するためには, 尤度関数の評価が $O(1/\epsilon^2)$ 回, 必要である [92]. これは, 全体の計算量が $O(1/\epsilon^2)$ となり, つまりこれは古典的なモンテカルロ計算と同じ計算量となるため, 量子の優位性がなくなることを意味している.

この計算量の問題を回避するために, 各 $k \in [0, M]$ 番目の最尤推定を行う際に対して, 適応型グリッドサーチを用いることとする. すなわち, 各 k 番目の最尤推定を行う際に, 一つ前の $k-1$ 番目の Fisher 情報量から推定される誤差の C_ϵ 倍の信頼区間の範囲でのみ, 一定の分割数でグリッドサーチを実行した. これは, Chebysev の不等式を用いることで, 各 k 番目で推定される誤差は, 少なくとも $1 - 1/C_\epsilon^2$ 以下の確率で, C_ϵ 倍の信頼区間にあることが保証されるためである. したがって, 推定精度の誤差 ϵ を得るためには, 最尤推定の回数 M が $O(\ln(1/\epsilon))$ となる. よって, 推定がうまくいくことができる全体の確率は, $(1 - 1/C_\epsilon^2)^M$ であり, $C_\epsilon = \Theta((\ln(1/\epsilon))^{1/2})$ の時には, $\Omega(1)$ となる. そして, 全体の最尤推定における計算量は, $C_\epsilon \cdot M \cdot O(\ln(1/\epsilon))$ となる. なお, 最後の項は, 尤度関数を評価するものである. これにより, 古典的な事後処理における計算量は, $O((\ln(1/\epsilon))^{5/2})$ で与えられる.

例外的な変数の場合を除いては, ノイズの大きさ κ の推定精度が大きいとしても, θ の推定精度の誤差には, 影響がない. これにより, 二次元の最尤推定を一次元の最尤推定にすることができるかもしれない. 例えば, この場合, 大きなグリッドサイズでノイズの大きさ κ を大まかに推定し, θ に対する一次元の最尤推定を高い精度で行うことで, より少ない計算量で推定を行うことができる.

更に, 測定の回数が十分に多い場合には, 二次元の最尤推定は加重最小二乗推定で近似することができる. これは, Levenberg-Marquard[93, 94] のようなアルゴリズムを用いることで, 実際により効率的に解くことができる.

4.7 本章のまとめ

本章では, ノイズがある状況下における最尤推定を用いた振幅推定手法を説明した. 特に, 数学的に扱いやすい脱分極ノイズモデルを仮定し, 解析的に尤度関数, Fisher 情報行列, Cramér–Rao の下限を 4.1 節で述べ, 続いて 4.2 節では, 演算子 A の呼び出し回数と, 推定精度, ノイズの大きさの関係を議論した. 4.3 節では, 実際の超電導量子コンピュータ IBM Quantum *ibmq_valencia* を持ちて, 少ない量子ビットながら, 本手法の妥当性を確認した.

一方で, 本問題においては, 推定したいパラメータ θ とノイズの大きさ κ の二つの変数を求める複数の推定問題になる. 一般に複数のパラメータを推定する問題は簡単ではなく, 本問題においては, 例外的な変数が存在するという事象が起こる. 具体的には, ある特定の θ の場合, どれだけノイズの大きさ κ が小さくなっても推定精度を下げることができないものであり, 4.4 節で取り扱った. ただし, この例外的な変数は, 約 $1 \sim 2\%$ 程度の割合でしか存在せず, 更に, 実際に例外的な場合にも振幅増幅の数列 m_k を変えること, あるいは新たな補助量

子ビットを導入することで, 推定する値 θ を変えることで回避することができることを説明した.

4.5 節では, 大きな系でのハードウェアの見積もり, 特に一般に古典的な場合に, モンテカルロシミュレーションが用いられる 5 変数の多重積分を用いた見積もりを行なった. 現在のデバイスとは相応のギャップがあるものの, 具体的な数値として比較できていることが重要であると考える. そして, 4.6 節では, 計算量に関する議論を行った.

当然, ノイズの種類は様々であり, [64] では振幅ダンピングなどの, 他のノイズモデルを仮定した場合の研究も行われている. しかしながら, これらは Kraus 演算子表記となり, 解析的な分析が困難であり, 数値的な分析となっている. 更に, 振幅増幅の数列である m_k の改良にも非常に重要なテーマの一つである. 例えば, [73] では, $m_k = a^k$ とすることで, Heisenberg 限界を達成することは諦めるが, 量子の優位性を得つつ, 必要である推定精度と数列を関係付けることをしている.

現在の量子コンピュータは, NISQ[22] と呼ばれるもので, 量子ビットの数や重ね合わせ状態を維持できる時間などが限定的なデバイスしかなく, 特にノイズの影響を考慮にいれた上で使いこなすことが求められている. 一方でノイズを, 特に脱分極ノイズモデルを導入すると, Heisenberg 限界を前提として, 達成可能な推定精度が制限されてしまう. そこで次の第 5 章では, 達成可能な推定精度を改善する手法を提案する. 具体的には, 修正した振幅増幅演算子を導入することで, 脱分極ノイズがある状況下において, 推定精度を改善することができる.

第 5 章

修正した量子振幅増幅演算子

この章では, [75] で示している修正した量子振幅増幅演算子を用いた量子振幅推定手法について, 述べる.

第 3 章では, 量子位相推定を使わない振幅推定法について述べ, 更に第 4 章では, 第 3 章で述べた振幅推定法に関して, 脱分極ノイズモデルを導入した際の振る舞いを議論した. 一方で, 本章でも述べる通り, 脱分極ノイズがある状況下において第 4 章の方法では, 量子 Fisher 情報量と (古典)Fisher 情報量には差がある状況である, つまりアルゴリズムとしてまだ改善する余地があることを示している. そこで, 本章では, 新たに修正した量子振幅増幅演算子を用いた量子振幅推定手法を提案する. これは, 4 章での量子振幅演算子では, 量子 Fisher 情報量を理論的に達成できないものの, 本章で提案する修正した量子振幅増幅演算子では, 量子 Fisher 情報量を達成できる可能性があるものである.

まず, 5.1 節で, 修正した量子振幅増幅演算子 Q を導入する. これは, 従来の振幅増幅の演算子を少しずらした演算子であり, 前章までの議論とは別の基底で貼られる空間での回転を行っていると考えるものである. そして, この新たな演算子 Q は, ノイズがない状況下で, 従来の演算子 G と同様の性能を出す, つまり Heisenberg 限界が達成可能であることを示す. 次に, 脱分極ノイズが存在する場合の議論を 5.2 節で行う. 新たな演算子 Q を用いた場合の方が, (古典)Fisher 情報量の上限が大きいことを示す. また, 量子 Fisher 情報量に関しても解析を行う. なお, 本章では脱分極ノイズの大きさ κ は既知であるとして, 取り扱うこととする. そして, 5.4 で数値実験を行い, 本手法が機能することを示す. 最後に 5.5 で本章のまとめを行う.

5.1 修正した量子振幅増幅演算子

この節では, 修正した振幅増幅演算子 Q を導入し, ノイズがない状況下での振る舞い, つまりは Fisher 情報量がどうなるかを述べる. まず, 修正した振幅増幅演算子 Q を次で定義する.

$$Q := -U_0 A^\dagger U_f A \tag{5.1}$$

これは, 3.1 節の式 (3.2) でも述べた振幅増幅演算子 $G = -AU_0A^\dagger U_f$ と似た演算子であるが, 回転する空間の基底が異なる. つまり, 演算子 G は, $|\Psi_1\rangle_n |1\rangle$ と $|\Psi_0\rangle_n |0\rangle$ の基底で貼られた空間を 2θ の回転をするものであるが, 演算子 Q は, $A|0\rangle_{n+1}$ と $|\Phi\rangle_{n+1}$ の基底で貼られた空間を 2θ の回転をするものである. それは, $Q = A^\dagger GA$ であることからわかり, 幾何学的に図 5.1 を示す.

そして, Q を $|0\rangle_{n+1}$ に作用させた状態を考えると,

$$Q|0\rangle_{n+1} = \cos(2\theta)|0\rangle_{n+1} + \sin(2\theta)|\Phi\rangle_{n+1} \tag{5.2}$$

であることがわかる. 更に, Q を m_k 回 $|0\rangle_{n+1}$ に作用させた状態を, $|\Psi_Q(\theta,k)\rangle$ とすると, 次で計算できる.

$$|\Psi_Q(\theta,k)\rangle := Q^{m_k}|0\rangle_{n+1} = \cos(2m_k\theta)|0\rangle_{n+1} + \sin(2m_k\theta)|\Phi\rangle_{n+1} \tag{5.3}$$

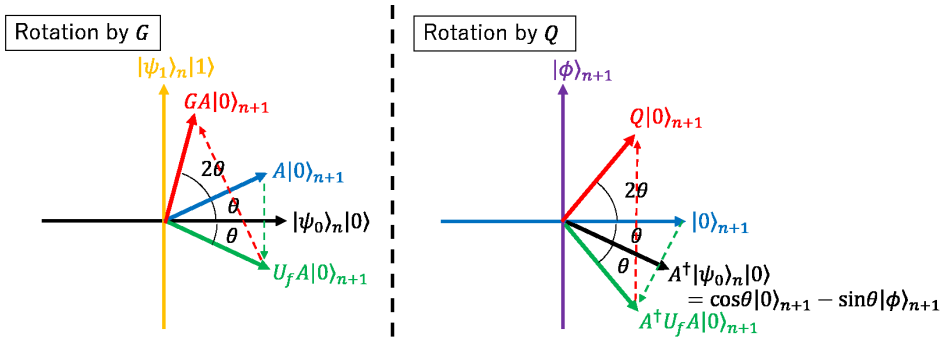


図 5.1　修正した振幅増幅演算子の回転の概要 (再掲)

従来の振幅増幅演算子 G（左側）と修正した振幅増幅演算子 Q（右側）の回転の概要図. どちらも初期状態を青色, 演算後を赤色で表し, 2θ だけ回転していることを示している.

ただし, $|\Phi\rangle_{n+1}$ は以下である.

$$\begin{aligned} |\Phi\rangle_{n+1} &= A^\dagger(-\sin\theta\,|\psi_0\rangle_n\,|0\rangle + \cos\theta\,|\psi_1\rangle_n\,|1\rangle) \\ &= \frac{1}{\sin 2\theta}\,(Q - \cos 2\theta)\,|0\rangle_{n+1}\,. \end{aligned} \tag{5.4}$$

本手法において, 測定演算子は, $E_{Q,0} = |0\rangle_{n+1}\langle 0|_{n+1}$ と $E_{Q,1} = \mathbf{I}_{n+1} - E_{Q,0}$ を用いる. これは, $n+1$ 個の量子ビットがすべて 0 か, そうでないかで測定している. よって, 式 (5.3) の状態において, $\{E_{Q,0}, E_{Q,1}\}$ で測定すると, 次の確率分布を得ることができる.

$$\begin{aligned} p_Q^{(k)}(0;\theta) &= \mathbb{P}(\{h_k = 0\}) = 1 - \sin^2((2m_k)\theta) = \cos^2((2m_k)\theta) \\ p_Q^{(k)}(1;\theta) &= \mathbb{P}(\{h_k = 1\}) = \sin^2((2m_k)\theta) \end{aligned} \tag{5.5}$$

なお, $h_k \in \{0,1\}$ は二値の確率変数である. そして, 3.2 節での議論と同様の議論にて, $\mathbf{h} = (h_1, \cdots, h_M) \in \{0,1\}^M$ に対する確率を次で置く.

$$P_Q(\mathbf{h};\theta) = \prod_{k=1}^{M} (p_Q^{(k)}(1;\theta))^{h_k} (p_Q^{(k)}(0;\theta))^{1-h_k} \tag{5.6}$$

更に, 各 m_k 毎に, N_{shot} 回測定し, その結果 $\mathbf{H}$ を $\{\mathbf{h}^1, \cdots, \mathbf{h}^{N_{\text{shot}}}\}$ とすると, 尤度関数 $L_Q(\theta;\mathbf{H})$ を以下で定義できる.

$$L_Q(\theta;\mathbf{H}) = \prod_{j=1}^{N_{\text{shot}}} P_Q(\mathbf{h}^j;\theta) \tag{5.7}$$

すると, 次の式により θ を最尤推定することができる.

$$\hat{\theta}_{\text{ML}} = \arg\max_\theta L_Q(\theta;\mathbf{H}) = \arg\max_\theta \ln L_Q(\theta;\mathbf{H}) \tag{5.8}$$

そして, フィシャー情報量は,

$$\begin{aligned} J_Q^{(k)}(\theta) &= \mathbb{E}\Big[\Big(\frac{\partial}{\partial\theta}\ln p_Q^{(k)}(h_k;\theta)\Big)^2\Big] \\ &= \sum_{h_k\in\{0,1\}} \Big(\frac{\partial}{\partial\theta}\ln p_Q^{(k)}(h_k;\theta)\Big)^2 p_Q^{(k)}(h_k;\theta) \\ &= 4(2m_k)^2 \end{aligned} \tag{5.9}$$

となる．ここで, $|\Psi_Q(\theta,k)\rangle$ の状態に対する, 演算子 A の呼び出し回数 $Nq_Q^{(m)}=2m$ であることから, 演算子 G,Q に対する Fisher 情報量を Nq の関数として考えると, 次のように書ける．

$$\begin{aligned} J_G^{(Nq_G^{(m)})} &= 4(Nq_G^{(m)})^2 \\ J_Q^{(Nq_Q^{(m)})} &= 4(Nq_Q^{(m)})^2 \end{aligned} \tag{5.10}$$

つまり, G,Q の場合も Nq に対する関数としては, 同じ形で表すことができる．これは, ノイズがない場合に G と Q を用いた手法のどちらにおいて, 優劣がないことを示している．また, 演算子 Q を用いた場合でも, G の場合と同様の推定精度を得ることができる, つまり, 本手法も Heisenberg 限界を達成できることがわかる．

5.2 脱分極ノイズがある場合

前節では, 新たな演算子 Q を導入し, ノイズがない状況下では, 従来の演算子 G と同様の精度を出すことができることを示した．本節では, 4 章でも導入した脱分極ノイズが存在する場合の議論を行う．脱分極ノイズがある状況における演算子 Q を用いた場合の Fisher 情報量 $J_{Q,\kappa}^{N_\mathrm{q}}(\theta)$ を算出するとともに, θ に寄らない Fisher 情報量の上限 $J_{Q,\kappa}^{N_\mathrm{q}}|_\mathrm{env}$ を求め, 演算子 G を用いた場合の上限 $J_{G,\kappa}^{N_\mathrm{q}}|_\mathrm{env}$ と比較する．なお, 本章における議論では, ノイズの大きさ κ は既知であるとして扱う．

まず, 4.1 節の式 (4.7) で導入した脱分極ノイズチャンネルを再掲する．

$$\mathcal{D}(\rho)=\mathrm{e}^{-\kappa}\rho+(1-\mathrm{e}^{-\kappa})\frac{I_{n+1}}{d}. \tag{5.11}$$

ただし, κ は脱分極ノイズの大きさを表す．そして, 一般に演算子 Q に対するノイズのない, 振幅増幅チャンネルを表す CPTP 写像は,

$$\mathcal{Q}(\rho)=\mathbf{Q}\rho\mathbf{Q}^\dagger. \tag{5.12}$$

で書ける．そして, ノイズ付き振幅増幅を m_k 回繰り返すことを考えると,

$$\rho_{Q,\mathrm{noise}}^{(k)}=(\mathcal{QD})^{m_k}(\rho_Q)=\mathrm{e}^{-\kappa m_k}\rho_{Q,\kappa}^{(k)}+(1-\mathrm{e}^{-\kappa m_k})\frac{I_{n+1}}{d}. \tag{5.13}$$

となる．ただし, 振幅増幅 1 回につき, 脱分極ノイズが作用するものとし, $\rho_{Q,\kappa}^{(k)}=|\Psi_Q(\theta,k)\rangle_{n+1}\langle\Psi_Q(\theta,k)|$ である．そして, 測定演算子 $E_{Q,0}$ 及び $E_{Q,1}$ を用いて測定をすると, それぞれの確率 $p_{Q,\kappa}^{(k)}(0;\theta)$, $p_{Q,\kappa}^{(k)}(1;\theta)$ は次となる．

$$\begin{aligned} p_{Q,\kappa}^{(k)}(0;\theta) &= \mathrm{e}^{-\kappa m_k}\cos^2(2m_k\theta)+(1-\mathrm{e}^{-\kappa m_k})\frac{1}{d} \\ p_{Q,\kappa}^{(k)}(1;\theta) &= \mathrm{e}^{-\kappa m_k}\sin^2(2m_k\theta)+(1-\mathrm{e}^{-\kappa m_k})\frac{d-1}{d} \end{aligned} \tag{5.14}$$

ここで, 改めて演算子 A の呼び出し回数 $N_\mathrm{q}{}_Q^{(m)}=2m$ であることから, N_q に対する確率とみなすと,

$$\begin{aligned} p_{Q,\kappa}^{N_\mathrm{q}}(0;\theta) &= \mathrm{e}^{-\kappa N_\mathrm{q}/2}\cos^2(N_\mathrm{q}\theta)+(1-\mathrm{e}^{-\kappa N_\mathrm{q}/2})\frac{1}{d} \\ p_{Q,\kappa}^{N_\mathrm{q}}(1;\theta) &= \mathrm{e}^{-\kappa N_\mathrm{q}/2}\sin^2(N_\mathrm{q}\theta)+(1-\mathrm{e}^{-\kappa N_\mathrm{q}/2})\frac{d-1}{d} \end{aligned} \tag{5.15}$$

となる．そして, Fisher 情報量 $J_{Q,\kappa}^{N_\mathrm{q}}(\theta)$ は, 次で計算できる．

$$J_{Q,\kappa}^{N_\mathrm{q}}(\theta)=\frac{4{N_\mathrm{q}}^2\sin^2(N_\mathrm{q}\theta)\cos^2(N_\mathrm{q}\theta)\mathrm{e}^{-\kappa N_\mathrm{q}}}{\left(\frac{1}{d}+\left(\cos^2(N_\mathrm{q}\theta)-\frac{1}{d}\right)\mathrm{e}^{-\kappa Nq/2}\right)\left(\frac{d-1}{d}+\left(\sin^2(N_\mathrm{q}\theta)-\frac{d-1}{d}\right)\mathrm{e}^{-\kappa N_\mathrm{q}/2}\right)} \tag{5.16}$$

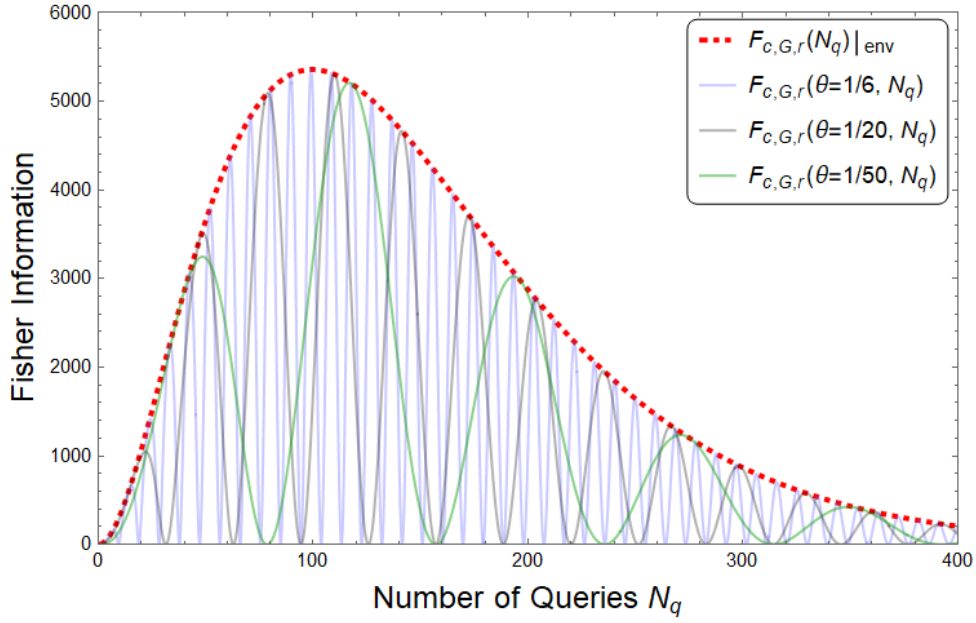


図 5.2　演算子 A の呼び出し回数と Fisher 情報量の関係

3 つの実線は, 式 (5.17) のフィッシャー情報量を表し, それぞれ $\theta \in \{1/6, 1/20, 1/50\}$ の値である. そして, 赤色の点線は, 式 (5.18) の包絡線のフィッシャー情報量である. なお, ノイズの大きさは, $\mathrm{e}^{-\kappa} = 0.99$ である.

なお, 演算子 G に対して, ノイズの大きさ κ が既知である時の Fisher 情報量 $J^{N_{\mathrm{q}}}_{G,\kappa}(\theta)$ は, 演算子 A の呼び出し回数 $N_{\mathrm{q}G}^{(m)} = 2m+1$ であることを加味すると,

$$J^{N_{\mathrm{q}}}_{G,\kappa}(\theta) = \frac{4{N_{\mathrm{q}}}^2 \sin^2(N_{\mathrm{q}}\theta)\cos^2(N_{\mathrm{q}}\theta)\mathrm{e}^{-\kappa N_{\mathrm{q}}}}{\left(\frac{1}{2} + \left(\cos^2(N_{\mathrm{q}}\theta) - \frac{1}{2}\right)\mathrm{e}^{-\kappa N q/2}\right)\left(\frac{1}{2} + \left(\sin^2(N_{\mathrm{q}}\theta) - \frac{1}{2}\right)\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}\right)} \tag{5.17}$$

である. ここで $d=2$ の時は, $J^{N_{\mathrm{q}}}_{G,\kappa}(\theta) = J^{N_{\mathrm{q}}}_{Q,\kappa}(\theta)$ である. ただし, これらの大きさは, θ の値によって異なるため, θ に寄らない上限による議論を行う. これらの Fisher 情報量 $J^{N_{\mathrm{q}}}_{G,\kappa}(\theta)$ と $J^{N_{\mathrm{q}}}_{Q,\kappa}(\theta)$ は, 次の式で与えられる, θ に寄らない上限 $J^{N_{\mathrm{q}}}_{G,\kappa}|_{\mathrm{env}}$ と $J^{N_{\mathrm{q}}}_{Q,\kappa}|_{\mathrm{env}}$ を持ち, それは図 5.2 に示すような包絡線になる.

$$\begin{aligned}
J^{N_{\mathrm{q}}}_{G,\kappa}(\theta) \leq J^{N_{\mathrm{q}}}_{G,\kappa}|_{\mathrm{env}} =& 4N_{\mathrm{q}}^2\mathrm{e}^{-\kappa N_{\mathrm{q}}} \\
J^{N_{\mathrm{q}}}_{Q,\kappa}(\theta) \leq J^{N_{\mathrm{q}}}_{Q,\kappa}|_{\mathrm{env}} =& 4N_{\mathrm{q}}^2\mathrm{e}^{-\kappa N_{\mathrm{q}}/2} + 8{N_{\mathrm{q}}}^2\frac{d-1}{d^2}\left(1-\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}\right)^2 \\
& - \frac{8N_{\mathrm{q}}^2\left(1-r^{N_{\mathrm{q}}}\right)\sqrt{(d-1)\left(d-1+\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}\right)\left((d-1)\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}+1\right)}}{d^2}
\end{aligned} \tag{5.18}$$

ここで, G と Q の場合における Fisher 情報量の上限値を比べる. $J^{N_{\mathrm{q}}}_{Q,\kappa}|_{\mathrm{env}}$ の最後の項に相加相乗平均を用いることで, 以下がわかる.

$$\begin{aligned}
J^{N_{\mathrm{q}}}_{Q,\kappa}|_{\mathrm{env}} \geq& 4N_{\mathrm{q}}^2\mathrm{e}^{-\kappa N_{\mathrm{q}}/2} + 8{N_{\mathrm{q}}}^2\frac{d-1}{d^2}\left(1-\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}\right)^2 \\
& - \frac{8N_{\mathrm{q}}^2\left(1-r^{N_{\mathrm{q}}}\right)}{d^2}\left\{\frac{\left(d-1+\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}\right)+(d-1)\left((d-1)\mathrm{e}^{-\kappa N_{\mathrm{q}}/2}+1\right)}{2}\right\} \\
=& 4N_{\mathrm{q}}^2\mathrm{e}^{-\kappa N_{\mathrm{q}}} \\
=& J^{N_{\mathrm{q}}}_{G,\kappa}|_{\mathrm{env}}
\end{aligned} \tag{5.19}$$

つまり, 本質的に, 演算子 Q を用いた場合の方が, 値の大きな Fisher 情報量が得ることができることがわかる. そして, Cramér–Rao の不等式より, $J^{N_\mathrm{q}}_{G,\kappa}|_\mathrm{env}$, $J^{N_\mathrm{q}}_{Q,\kappa}|_\mathrm{env}$ は, 推定誤差の下限を与える. また, この下限は, チャンネル推定の問題で示されている通り, N_shot の状態をいくつかに分割し, 直前の測定結果に応じて測定演算子を調整する適応的な測定を行うことで, 漸近的に達成できることが知られている [95, 96, 97]. そして, 前述の式は, ノイズの大きさが非常に小さい場合でも, N_q が大きいところでは, Heisenberg 限界が漸近的に達成できないことを示している. ただし, ノイズの大きさによるものの, 初期状態である式 (3.1) のサンプリングにより, 推定精度は定数倍, 改善することが可能である. このサンプリングは, $N_\mathrm{q} = 1$ の場合であり, つまり Fisher 情報量はほぼ 4 である. 一方で, ノイズが小さい場合, $N_\mathrm{q} \gg 1$ の時, つまり振幅増幅を行なった場合, Fisher 情報量は, 4 よりも大きくなることがわかる. よって, 漸近的な範囲ではなく, N_q が有限の範囲での Fisher 情報量を考える必要がある.

5.3 量子 Fisher 情報量について

この節では, 量子 Fisher 情報量についての議論を行う. 5.1 節で演算子 Q を導入し, ノイズがない状況では, 演算子 G と同様の性能を出せることを示した. そして, 5.2 節では, 脱分極ノイズが存在する状況下での Fisher 情報量の上限について議論をし, 演算子 Q の方が演算子 G より大きいことを示した. この節では量子 Fisher 情報量について計算し, 演算子 G,Q の (古典)Fisher 情報量との差分を議論する. 特に d が十分に大きい時には, 演算子 Q の上限と量子 Fisher 情報量がほぼ一致することを示す.

まず, ノイズのないケースの量子 Fisher 情報量について述べる. これまでの議論は, 古典 Fisher 情報量について扱っており, 演算子 G を用いた場合では, $E_{G,0}$ および $E_{G.1}$ という固定された測定演算子で測定をしている. ただし, 測定演算子の選択には自由度があり, 他の測定がより大きな古典 Fisher 情報量を得る可能性がある. しかし, 任意の測定から得られる不偏推定量 $\hat{\theta}$ は, 量子 Cramér–Rao の不等式に従うことが知られている [98, 99, 100].

$$\mathbb{E}\left[(\theta - \hat{\theta})^2\right] \geq \frac{1}{N_\mathrm{shot} F^{N_\mathrm{q}}_G(\theta)} \tag{5.20}$$

ただし, $F^{N_\mathrm{q}}_G(\theta)$ は, 量子 Fisher 情報量で, 次で定義されるものである.

$$F^{N_\mathrm{q}}_G(\theta) = \mathrm{Tr}\left(\rho^{N_\mathrm{q}}_G(\theta) L_S\left(\rho^{N_\mathrm{q}}_G(\theta)\right)^2\right). \tag{5.21}$$

なお, $L_S(\rho^{N_\mathrm{q}}_G(\theta))$ は, symmetric logarithmic derivative (SLD) と呼ばれている次を満たすエルミート演算子である.

$$\dot{\rho}^{N_\mathrm{q}}_G(\theta) = \frac{1}{2}\left(\rho^{N_\mathrm{q}}_G(\theta) L_S(\rho^{N_\mathrm{q}}_G(\theta)) + L_S(\rho^{N_\mathrm{q}}_G(\theta))\rho^{N_\mathrm{q}}_G(\theta)\right). \tag{5.22}$$

なお, 他の量子 Fisher 情報量の定義もあるものの, 一つの変数の推定問題に対しては, SLD がタイトな境界となることが知られている [100, 101]. ここで, オーバードットは, θ による微分である. そして, 出力状態が純粋状態である場合, 量子 Fisher 情報量は次式で明示的に与えられる.

$$\begin{aligned} F^{N_\mathrm{q}}_G(\theta) &= 4\left(\langle\dot{\psi}_G(\theta, N_\mathrm{q})|\dot{\psi}_G(\theta, N_\mathrm{q})\rangle - |\langle\dot{\psi}_G(\theta, N_\mathrm{q})|\psi_G(\theta, N_\mathrm{q})\rangle|^2\right) \\ &= 4N^2_\mathrm{q} \end{aligned} \tag{5.23}$$

ここで, 古典 Fisher 情報量 $J^{N_\mathrm{q}}_G(\theta)$ は式 (3.21) から, 量子 Fisher 情報量 $F^{N_\mathrm{q}}_G(\theta)$ に一致することがわかる. これはノイズがない場合には, 最後の量子ビットを測定する演算子 $E_{G,0}$ と $E_{G,1}$ が最適であることを示している. 更に, 演算子 Q についても同様の議論ができる. よって, 演算子 Q に対する量子フィシャー情報量を $F^{N_\mathrm{q}}_Q(\theta)$ とすると,

$$J^{N_\mathrm{q}}_G(\theta) = F^{N_\mathrm{q}}_G(\theta) = J^{N_\mathrm{q}}_Q(\theta) = F^{N_\mathrm{q}}_Q(\theta) = 4N^2_\mathrm{q} \tag{5.24}$$

となる. なお, これまで述べてきた振幅推定の問題は, ユニタリー演算子 G に埋め込まれたパラメータ推定問題とみなすことができ, これは, 量子計測分野の主要な目標の 1 つである [102, 103]. 量子計測分野では, N 個の独立した初期状態に対して, 一つのユニタリー演算を行った場合の推定誤差は $1/\sqrt{N}$ である. これは標準的な量子の限界, あるいは, ショットノイズと呼ばれているものであり, エンタングルメントや複数の逐次演算を行うことで, 推定誤差が 2 次的改善され, $1/N$ にできることが知られており, これは Heisenberg 限界と呼ばれている. 一方で, この 2 次的な加速は, 無限小のノイズであったとしても, 機能しなくなることが知られており, [104, 105, 89] これは, 量子振幅推定においても発生する事象である.

そして, 脱分極ノイズがある状況における量子 Fisher 情報量について述べる. チャネル推定問題における既存研究 [106, 107] と同様に, 脱分極ノイズがある状態 (4.10) が指数関数的に表現できるということを用いて, SLD 演算子は, 次のように計算できる.

$$L_S(\rho_{G,\kappa}^{N_\mathrm{q}}(\theta)) = \frac{2\mathrm{e}^{-\kappa N_\mathrm{q}/2}}{\frac{2}{d} + \left(1-\frac{2}{d}\right)\mathrm{e}^{-\kappa N_\mathrm{q}/2}} \dot{\rho}_G^{N_\mathrm{q}}(\theta) \tag{5.25}$$

そして, 量子 Fisher 情報量 $F_{G,\kappa}^{N_\mathrm{q}}(\theta)$ は,

$$F_{G,\kappa}^{N_\mathrm{q}}(\theta) = \frac{4{N_\mathrm{q}}^2\mathrm{e}^{-\kappa N_\mathrm{q}}}{\frac{2}{d} + \left(1-\frac{2}{d}\right)\mathrm{e}^{-\kappa N_\mathrm{q}/2}} \tag{5.26}$$

となる. 同様に, 演算子 Q に対して, 状態 (5.13) における量子 Fisher 情報量 $F_{Q,\kappa}^{N_\mathrm{q}}(\theta)$ は, 次のようになり, これは $F_{G,\kappa}^{N_\mathrm{q}}(\theta)$ と同じである.

$$F_{Q,\kappa}^{N_\mathrm{q}}(\theta) = \frac{4{N_\mathrm{q}}^2\mathrm{e}^{-\kappa N_\mathrm{q}}}{\frac{2}{d} + \left(1-\frac{2}{d}\right)\mathrm{e}^{-\kappa N_\mathrm{q}/2}} \tag{5.27}$$

重要なことは, $F_{G,\kappa}^{N_\mathrm{q}}(\theta)$ および $F_{Q,\kappa}^{N_\mathrm{q}}(\theta)$ は, θ によらない値であることである.

ここで, 演算子 G,Q それぞれに対して, 古典 Fisher 情報量, 古典 Fisher 情報量の上限, 量子 Fisher 情報量の関係は, 任意の θ に対して, 次の通りとなる.

$$\begin{aligned} J_{G,\kappa}^{N_\mathrm{q}}(\theta) \le J_{G,\kappa}^{N_\mathrm{q}}|_\mathrm{env} \le F_{G,\kappa}^{N_\mathrm{q}}(\theta) \\ J_{Q,\kappa}^{N_\mathrm{q}}(\theta) \le J_{Q,\kappa}^{N_\mathrm{q}}|_\mathrm{env} \le F_{Q,\kappa}^{N_\mathrm{q}}(\theta) \end{aligned} \tag{5.28}$$

演算子 G については, 量子ビットが一つの場合（$n=1$）, つまり, $d=2$ の時は, $J_{G,\kappa}^{N_\mathrm{q}}|_\mathrm{env} = F_{G,\kappa}^{N_\mathrm{q}}(\theta)$ となる. 一方で, $n \ge 2$ の時には, $J_{G,\kappa}^{N_\mathrm{q}}|_\mathrm{env} < F_{G,\kappa}^{N_\mathrm{q}}(\theta)$ となる. 更に, 十分に量子ビットが多い時, すなわち n が無限大の極限では,

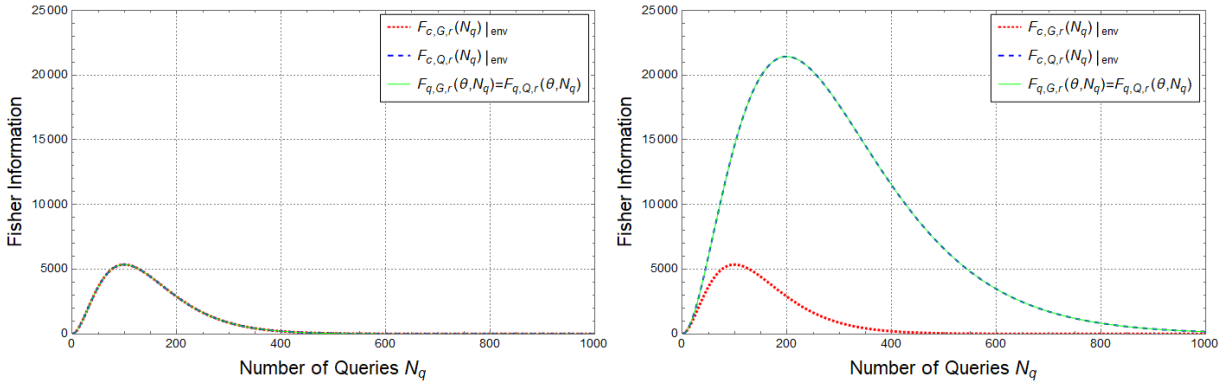
$$F_{G,\kappa}^{N_\mathrm{q}}(\theta) = 4{N_\mathrm{q}}^2\mathrm{e}^{-\kappa N_\mathrm{q}/2} \tag{5.29}$$

となり, 古典 Fisher 情報量の上限である $J_{G,\kappa}^{N_\mathrm{q}}|_\mathrm{env} = 4N_\mathrm{q}^2\mathrm{e}^{-\kappa N_\mathrm{q}}$ であることより, 量子 Fisher 情報量の方が, $\mathrm{e}^{-\kappa N_\mathrm{q}/2}$ 倍大きいことがわかる. すなわち, 演算子 G を用いた方法では, $n=1$ の場合を除き, 量子 Fisher 情報量を得ることができないことを示している.

なお, 入力状態 $A|0\rangle_{n+1}$ は, 演算子 G の二つの固有状態を等しく重み付けしている重ね合わせ状態であり, これは, 単一プローブにユニタリー演算子 G を適用するチャネル推定問題の逐次戦略において, 最適な入力状態として知られている [108, 107]. したがって, 量子 Fisher 情報量 (5.26) は, 演算子 G を用いるすべての方法に対して最適であり, 更に, 量子 Fisher 情報量 (5.26) は, 演算子 A を用いる場合にも最適であることを [75] で示している.

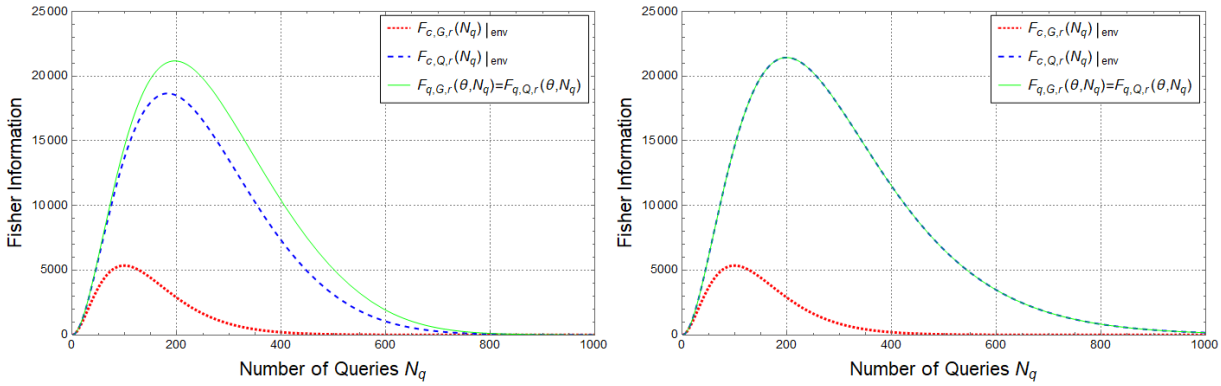
次に, 演算子 Q を用いた場合に, 量子 Fisher 情報量を得ることができるかについて議論をする. $d=2$ の場合には, $J_{Q,\kappa}^{N_\mathrm{q}}|_\mathrm{env} = F_{Q,\kappa}^{N_\mathrm{q}}(\theta)$ であり, 更に, 次の式を示すことができる.

$$\lim_{d\to\infty}\left\{J_{Q,\kappa}^{N_\mathrm{q}}|_\mathrm{env} - F_{Q,\kappa}^{N_\mathrm{q}}(\theta)\right\} = 0 \tag{5.30}$$



(a) $n = 1$ $(d = 2)$

(c) $n = 100$ $(d = 2^{100})$



(b) $n = 10$ $(d = 2^{10})$

(d) $n \to \infty$ $(d \to \infty)$

図 5.3　演算子 A の呼び出し回数と量子フィッシャー情報量の関係

演算子 G を用いた手法の古典フィッシャー情報の包絡線 $J^{N_\mathrm{q}}_{G,\kappa}|_{\mathrm{env}}$ は赤色の点線で, 演算子 Q を用いた手法の古典フィッシャー情報の包絡線 $J^{N_\mathrm{q}}_{Q,\kappa}|_{\mathrm{env}}$ は青色の点線で, 量子フィッシャー情報量が緑色の破線である.
なお, ノイズの大きさは, $\kappa = 0.01$ とした.

これは, 量子ビット数が多い場合には, 古典 Fisher 情報量の上限が, 量子 Fisher 情報量に漸近することを示しており, 本質的な演算子 G と Q の違いである. この G と Q の違いを直感的に理解するために, 改めて, それぞれの $E_{G,1}$ と $E_{Q,0}$ で測定する確率は,

$$\begin{aligned} p^{N_\mathrm{q}}_{G,\kappa}(1;\theta) &= \mathrm{e}^{-\kappa N_\mathrm{q}/2}\cos^2(N_\mathrm{q}\theta) + (1-\mathrm{e}^{-\kappa N_\mathrm{q}/2})\frac{1}{2} \\ p^{N_\mathrm{q}}_{Q,\kappa}(0;\theta) &= \mathrm{e}^{-\kappa N_\mathrm{q}/2}\cos^2(N_\mathrm{q}\theta) + (1-\mathrm{e}^{-\kappa N_\mathrm{q}/2})\frac{1}{d} \end{aligned} \tag{5.31}$$

である. ここで, 演算子 Q を用いる場合には, 量子ビット数の増加に対して, 第 2 項にあるノイズの効果が指数関数的に減少するため, 第 1 項から与えられる情報を効率的に得ることがわかる. 一方で, 演算子 G を用いる場合は, d の大きさによらずノイズの影響を受けるため, N_q が大きい範囲では, 第 2 項は 1/2 となるため, ノイズに埋もれてしまう可能性が高いものである.

そして, これらの Fisher 情報量を具体的な例で計算をして, 振る舞いを確認する. 図 5.3 は, 演算子 G と Q を用いる振幅推定法における, 古典および量子の Fisher 情報量と, N_q の関係を示したものであり, 赤色の点線が演算子 G を用いた場合の古典 Fisher 情報量の上限 $J^{N_\mathrm{q}}_{G,\kappa}|_{\mathrm{env}}$, 青色の線が演算子 Q を用いた場合の古典 Fisher 情報量 $J^{N_\mathrm{q}}_{Q,\kappa}|_{\mathrm{env}}$, 緑色の線が量子 Fisher 情報量であり, 前述の通り, 演算子 G および Q を用いた場合

で一致した値 $F^{N_{\rm q}}_{G,\kappa}(\theta) = F^{N_{\rm q}}_{Q,\kappa}(\theta)$ である．測定回数は 1 回, ノイズの大きさ $\kappa = 0.01$ とし, そして, 図 5.3 の (a), (b), (c), (d) は, それぞれ, 量子ビットの数が, 1, 10, 100, ∞ の 4 つの場合である．

(a) 1 量子ビットの場合, つまり $n = 1$ であり, $d = 2$ の場合, 演算子 G および Q を用いた場合の, 古典および量子の Fisher 情報量の, すべて一致していることがわかる．つまり, $J^{N_{\rm q}}_{G,\kappa}|_{\rm env} = J^{N_{\rm q}}_{Q,\kappa}|_{\rm env} = F^{N_{\rm q}}_{G,\kappa}(\theta) = F^{N_{\rm q}}_{Q,\kappa}(\theta)$ である．

(b) $n = 10$ の場合, $J^{N_{\rm q}}_{G,\kappa}|_{\rm env}$ と $J^{N_{\rm q}}_{Q,\kappa}|_{\rm env}$ では明確な差があり, 特に後者の Q を用いる場合は, 少ない量子ビットしか使っていないケースであったとしても, ほぼ理論限界である量子 Fisher 情報量 $F^{N_{\rm q}}_{Q,\kappa}(\theta)$ に近い値を示していることがわかる．

(c) $n = 100$ の場合, $J^{N_{\rm q}}_{Q,\kappa}|_{\rm env}$ は, ほぼ極限の $F^{N_{\rm q}}_{Q,\kappa}(\theta)$ に達していることがわかり, それは, (d) の場合とほぼ一致していることがわかる．一方で, $J^{N_{\rm q}}_{G,\kappa}|_{\rm env}$ は, (b) の場合とほぼ変わらないことがわかる．

(d) $n = \infty$ の場合, $J^{N_{\rm q}}_{Q,\kappa}|_{\rm env} = F^{N_{\rm q}}_{Q,\kappa}(\theta)$ であることが確認できる．

ここで, 特に (c) の場合に注目すると, $J^{N_{\rm q}}_{Q,\kappa}|_{\rm env}$ は, $F^{N_{\rm q}}_{Q,\kappa}(\theta)$ とほぼ同じ形状であることから, $N_{\rm q} = 2/\kappa$ の場合に, 最大値 $16/(e^2\kappa^2)$ を取ることがわかる．一方で, $J^{N_{\rm q}}_{G,\kappa}|_{\rm env}$ は, $N_{\rm q} = 1/\kappa$ の場合に, 最大値 $4/(e^2\kappa^2)$ を取る, つまり, G の場合に比べて Q の場合の方が, 4 倍大きな Fisher 情報量を得ることができることがわかる．なお, これはノイズの大きさが任意の場合で同様に議論が成り立つ．

したがって, 演算子 Q を用いた手法では, 従来の G を用いた手法を超えるだけでなく, 量子 Fisher 情報量の逆数で与えられる推定精度の下限を達成する推定量を与えることがわかる．また, $|0\rangle_{n+1}$ が Q の最適な初期状態であり, これは, 今回得られた量子 Fisher 情報量が, 演算子 Q を用いるあらゆる手法にとって最適な値であることを示している．なお, 一般に, 量子 Fisher 情報量の逆数で与えられる推定精度を達成するためには, 測定結果に応じて振幅増幅操作の回数を変える等の, 適応的な戦略をとる必要がある．しかしながら, 次節で示す数値シミュレーションにより, 振幅増幅演算の回数を事前に用意した非適応的な戦略の場合であっても, 演算子 Q を用いた手法の方が, 演算子 G を用いた場合よりも優位であることを示す．

本節の最後で, 測定エラーに関する議論を行う．改めて, 演算子 G を用いた従来の手法では, 最後の 1 量子ビットの測定しか行わないが, 演算子 Q を用いた本節で扱った手法では, 全ての量子ビットを測定する必要があり, 測定エラーの影響を受けやすくなる．例えば, 全ての量子ビットで独立に, ϵ の測定エラーが生じると仮定すると, Fisher 情報量には $(1-\epsilon)^n$ の誤差が生じることとなる．そして, 現在のデバイスでは測定エラーは $\epsilon = 0.01$ 程度であることを鑑み, $(1-\epsilon)^n = 1/2$ を解くと $n = 70$ となるため, 70 量子ビットあたりで, Q を用いた手法の優位性が失われることがわかる．よって, Q を用いた振幅増幅手法を活用するためには, 測定のエラー緩和 [109, 110, 111] を適用することが必要であることを示唆している．一方で, 測定のエラー緩和には一般に, 指数関数的な時間のかかる事後処理が必要となる課題がある．ここで, 今回の場合, θ の推定値を得る際には, 全ての量子ビットがゼロである場合か否かを測定することとなるため, テンソル積を用いたエラーモデルを用いることで, 効率的に行うことができる [112].

5.4 数値実験

前節までで, 演算子 Q を用いた振幅増幅法を導入し, 従来の演算子 G を用いた場合と比べて, 古典および量子の Fisher 情報量にどのような差が生じるかについて述べた．本節では, 実際に数値実験を行い, 推定精度を Fisher 情報量の逆数と比較をする．なお, 推定にあたっては, 3 章及び, 4 章で述べた最尤推定法を用いた振幅推定を行う．また, 振幅増幅の回数 m_k は非適応型とすることから, 推定するパラメータ θ による依存性が存在する．つまり, 量子 Fisher 情報量から得られる下限を達成できる保証はないことには注意が必要である．

各条件として, 振幅増幅の回数の列を, 任意の k に対して, $m_k = \lfloor (6/5)^{k-1} \rfloor$ とする．これは, Heisenberg 限

界を近似的に達成できるものである．そして, ノイズの大きさを $\kappa = 0.01$, 量子ビットの数 $n = 100$（すなわち, $d = 2^{100}$）, 測定回数 $N_{\mathrm{shot}} = 100$ とする．これは, 図 5.3 の (c) の場合と同様であり, これらの変数を表 5.1 にまとめる．なお, この実験を 200 回繰り返し, 推定値 $\hat{\theta}$ の平均二乗誤差の平方根を評価する．また, Q を用いた手法では, $m_0 = 0$ の場合が, 変数 θ の値によらないため, G の場合を用いて評価している．

表 5.1　修正した振幅推定法の数値シミュレーションのパラメータ一覧

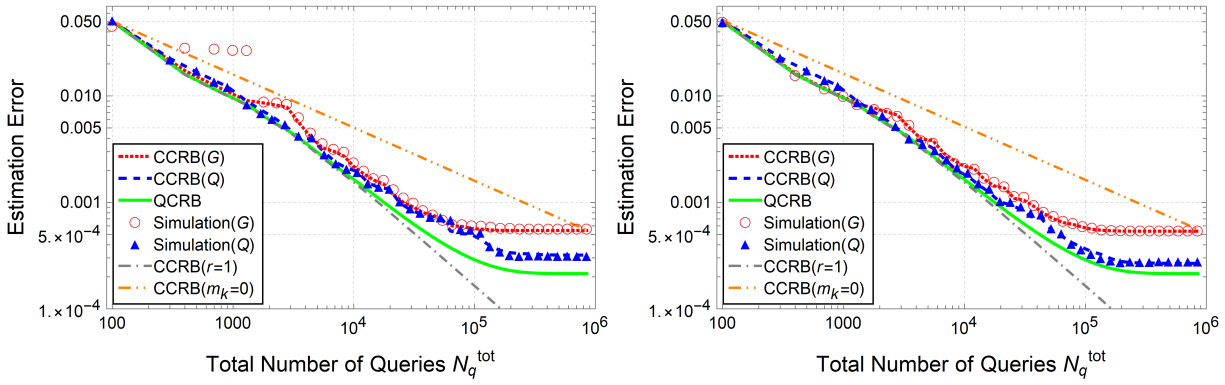
項目	文字	値
測定回数	N_{shot}	100
振幅増幅の回数の数列	$m_k, (k = 0, 1, 2, \ldots)$	$\lfloor (6/5)^{k-1} \rfloor$
ノイズの大きさ	κ	0.01
ヒルベルト空間の次元	d	2^{100}
推定する値	$a = \sin^2\theta$	$\{2/3, 1/3, 1/6, 1/12, 1/24, 1/48\}$

図 5.4 は, 目標となる θ の値を変えた時の, 演算子 A の呼び出し回数 $N_{\mathrm{q}}^{(m_k)}$ の和をとった, A の呼び出しの総回数 $N_{\mathrm{q}}^{\mathrm{tot}} = \sum_k N_{\mathrm{shot}} N_{\mathrm{q}}^{(m_k)}$ と, 推定精度の誤差（二乗平均の推定誤差の平方根）との関係を示していたものである．演算子 G と Q による, 最尤推定を用いた振幅推定法の数値シミュレーションの推定精度の誤差を, それぞれ赤色の丸点と青色の三角点で示す．そして, 量子フィッシャー情報量から導かれる Cramér–Rao の下限, 演算子 G, Q を用いた Cramér–Rao の下限を, それぞれ緑色の実線, 赤色の点線, 青色の破線で示す．ノイズがない時 ($\kappa = 0$) の振幅増幅をした場合と, 振幅増幅を用いない ($m_k = 0$) 場合の下限を, 灰色の一点鎖線および橙色の二点鎖線で示す．

ここでまず, 数値実験の結果は, 古典 Fisher 情報量から導かれる Cramér–Rao の下限にほぼ一致していることがわかる．そして, 一部の外れている結果を除いて, $N_{\mathrm{q}}^{\mathrm{tot}}$ が小さい範囲, つまり, ノイズの影響が小さい範囲では, 演算子 G を用いた手法 (赤色) と演算子 Q を用いた手法 (青色) による推定精度の誤差はほぼ同じである．更に，この範囲でこれらの結果は, 量子 Fisher 情報による下限 (緑色) ともほぼ一致し，いずれも Heisenberg 限界を概ね達成していることを示している．具体的には, G を用いる方法では $N_{\mathrm{q}}^{\mathrm{tot}} \sim 0.5 \times 10^5$ までの範囲で, Q を用いる方法では $N_{\mathrm{q}}^{\mathrm{tot}} \sim 10^5$ までの範囲で, それぞれ Heisenberg 限界をほぼ達成できている．そして, N_{q} が大きくなると, ノイズの影響で推定誤差が一定の値に収束することがわかるが, この収束する値は G の場合と Q の場合には差があり, Q を用いた場合の方が 2 倍程度, 推定精度がよくなることがわかる．さらに, 量子 Fisher 情報量との差は, G の場合は約 2.5 から 3 倍程度あるものの, 一方で Q の場合は 1.2 から 1.5 倍程度であり, 量子 Cramér–Rao の限界に近づいていることがわかる．最後に, グラフの右端, つまり, 振幅増幅の回数 m_k が約 700 回の増幅に対応する部分では, 脱分極ノイズが比較的大きな 1% 程度であったとしても, Q を用いた場合の方が, 適応的な測定を行なっていないにも関わらず, 古典的な場合および G を用いた場合よりも優位であることを示している．

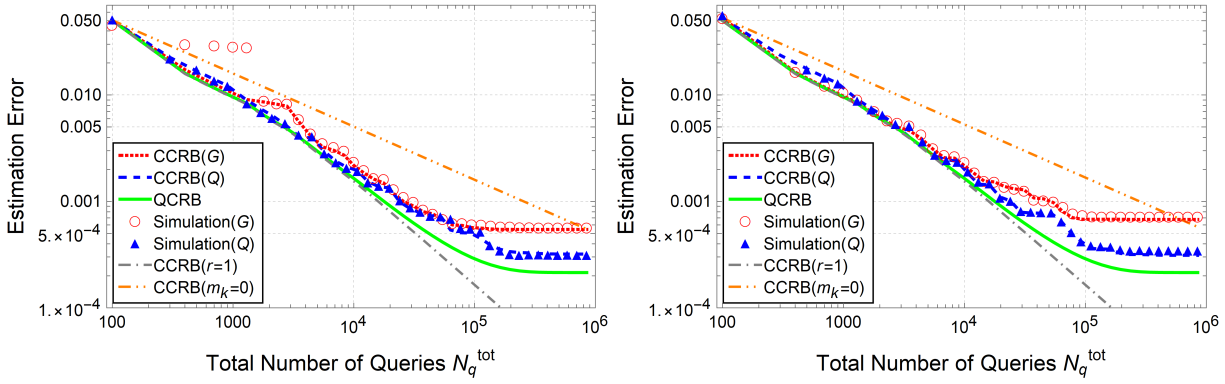
5.5　本章のまとめ

本章では, 5.1 節で修正した振幅増幅演算子 Q を導入し, ノイズがない場合には従来の振幅増幅演算子 G と同様に, 二乗加速する量子アルゴリズムの優位性が得られることを示した．続く 5.2 節では, 脱分極ノイズがある場合の比較を行い, 演算子 G,Q を用いた場合のそれぞれの Fisher 情報量の上限を比べると, 演算子 Q を用いた場合の方が大きくなることを示した．そして, 5.3 説では, 量子 Fisher 情報量について議論を行い, $d > 1$ の範囲では, 振幅増幅演算子 G を用いた場合には量子 Fisher 情報量が達成できない一方で, 修正された振幅増幅演算子 Q を用いた場合は達成可能であることを示した．ただし, 量子 Fisher 情報量を達成するためには, 一



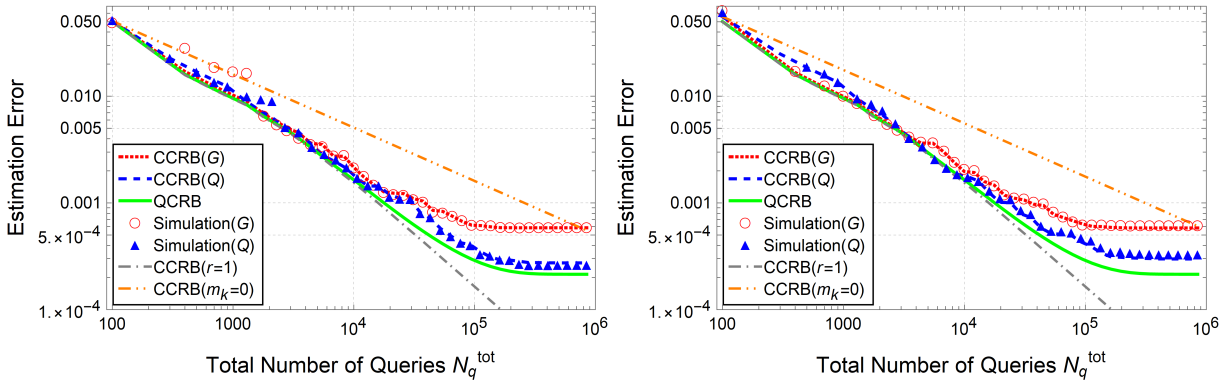
(a) $a = 2/3$

(d) $a = 1/12$



(b) $a = 1/3$

(e) $a = 1/24$



(c) $a = 1/6$

(f) $a = 1/48$

図 5.4　修正した振幅推定法の数値シミュレーションの結果

演算子 A の総呼び出し回数 $N_{\mathrm{q}}^{\mathrm{tot}}$ と, 推定精度の誤差の関係を示す. なお, パラメータは, 表 5.1 に示す. 演算子 G と Q による, 最尤推定を用いた振幅推定法の数値シミュレーションの推定精度の誤差を, それぞれ赤色の丸点と青色の三角点で示す. そして, 量子フィッシャー情報量から導かれる Cramér–Rao の下限, 演算子 G, Q を用いた Cramér–Rao の下限を, それぞれ緑色の実線, 赤色の点線, 青色の破線で示す. ノイズがない時 ($\kappa = 0$) の振幅増幅をした場合と, 振幅増幅を用いない ($m_k = 0$) 場合の下限を, 灰色の一点鎖線および橙色の二点鎖線で示す.

般的には, 適応型の測定が必要であるが, 5.4 節の数値シミュレーションで示した通り, 非適応型の測定戦略であったとしても, 演算子 G を用いた場合より, 修正した演算子 Q の方が量子 Fisher 情報量に近づくことを確認した.

なお, この量子 Fisher 情報量を達成するための適応型の戦略は今後の課題の一つである. これは, 実用上も非常に重要であり, ノイズのある量子コンピュータを使う上では, 基本的にノイズの値はハードウェアで決まってしまうため, エンドユーザーからは簡単には変えることができない. そこで, ソフトウェア, アルゴリズムを改良することで, 最大限性能を引き出すことが重要なことの一つであると考える. さらに, 本章では, ノイズの大きさは既知であるとして扱ったが, 4 章のように未知として扱うことも非常に重要なテーマの一つであると考える.

第 6 章

ノイズパラメータを推定しない量子振幅推定

この章では, [76] で示しているノイズパラメータを推定しない量子振幅推定法について述べる.

3 章では, 最尤推定を用いる振幅推定法について述べ, 位相推定を用いない手法を導入した. そして, 脱分極ノイズモデルを仮定した際の振幅推定法を 4 章で議論し, 続く 5 章では, 振幅増幅演算子を工夫し, 手法の改善を行った.

しかしながら, これらの手法の難しさは, 完璧なパラメトリックモデルを与えることが不可能であるという事実にある. 4 章, 5 章で扱った脱分極ノイズモデルは, 例えば, ランダマイズド・ベンチマーキング [70, 71, 72] などにも用いられる一般的なノイズの一つである. そして, 特に 4.3 節では実際の量子コンピュータの実機を用いて, モデルの妥当性を検証したものの, 実際には, 脱分極ノイズ以外のノイズもあり, 式 (4.12) で示している $P_G(\mathbf{h};\theta,\kappa)$ と, 未知である真の分布との間には, ギャップが存在するはずである. このギャップを小さくするために, つまりは, 複数の未知のノイズをより特徴づけるために, より複雑なパラメトリックモデル (確率分布)$P_G(\mathbf{h};\theta,\boldsymbol{\beta})$ を考える. ここで, $\boldsymbol{\beta}=(\beta_1,\beta_2,\ldots,\beta_M)$ は, 複数のノイズパラメータを表すベクトルとする. これにより, モデルの持つ表現力を高くすることができ, 導入したモデルと未知である真の分布のギャップは小さくすることができる. しかしながら, 一方で, パラメータの数は $M+1$ 個となり, 最尤推定法を用いた場合には, $M+1$ 次元の最適化問題を解く必要が生じる. つまり, 多次元非凸関数 $L_G(\theta,\boldsymbol{\beta};\mathbf{H})$ の尤度関数を, $(\theta,\boldsymbol{\beta})$ に対して最大化をしなければならない. これは, 一般に M が大きい場合には, 統計的な推定問題として, 難しい問題である. そこで, パラメータ直行化法 (parameter orthogonalization method)[77] という手法を用いることで, 局外パラメータ (nuisance parameter)$\boldsymbol{\beta}$ を考慮せずに, 効率かつ正確に, 興味のあるパラメータ (parameter of interest)θ を 1 次元の最適化問題として, 推定することができる.

なお, $M\to\infty$ という極限を考える, つまり, $\boldsymbol{\beta}$ を β_t という関数に置き換えることで, セミパラメトリックモデルとなり, 無限次元のノイズ効果を暗に捉えることができる. セミパラメトリック推定理論では, 無限次元の関数であっても除去し, 推定したいパラメータ θ を効率的に構築する可能性がある方法が知られている [79]. 最近の研究においても, パラメータ直行化法の量子版 [80, 78] やセミパラメトリック理論 [81, 82] も発展を続けているが, 本章では古典的な理論の活用を行う.

本章では, パラメータ直交化法を用いて, 脱分極ノイズモデルを含むより広範囲なノイズモデルに対して, 振幅推定問題を議論する. まず, 6.1 節で, パラメータ直交化法の概要を説明する. そして, 拡張されたモデルを持つ振幅推定問題に, パラメータ直交化法を用いることを考えるが, このモデルは縮退するため, つまり, 対応する Fisher 情報行列の行列式がゼロになるため, パラメータ直交化法を適用することができない. そこで, この問題を解決するために, 厳密には異なるものの, 振幅増幅演算子 G と良く似た演算子 R を導入し, 4 章で議論をした G を m_k 回作用させた状態に加え, G を m_k-1 回作用させて最後に R を作用させた状態を考える. そ

して, それらを合わせた確率分布を考えることで, Fisher 情報行列の行列式がゼロになることを避けることができる. これを 6.2 節で述べるが, これは実際の量子コンピュータ IBM Quantum を用いた結果を見ても大きな違和感のないものであることを示す. 続いて, 6.3 節では, 6.2 節に導入したモデルに対して, パラメータ直交化法を適用する. パラメータ直交化法の難しいところは, 多変数の微分方程式を解く必要があるところである. 言うまでもなく, 多変数の微分方程式が一般には解析的に解くことができないものの, 本モデルでは, 解析的な解を示すことができる. その結果, 最尤推定を用いた θ の推定は, $M+1$ 次元から, 1 次元の最適化問題にすることができる. 別の言い方をすると, ノイズパラメータである $\boldsymbol{\beta}$ の推定を行わずに, 興味のあるパラメータ θ を推定することができる. そして, シミュレーションと量子コンピュータの実機を用いて, 本手法の検証を 6.4 節で行う. 実際の超電導量子コンピュータ IBM Quantum (*ibm_kawasaki*) を用いた実験では, 識別できない現実のノイズの元でも本手法が動作することを示す. 最後に 6.5 節で, 本章のまとめを行う.

6.1 パラメータ直交化法について

本節では, パラメータ直交化法について, [77] を元に, 本質的な部分を説明する. 前述の通り, 一般に複数のパラメータ推定問題は, 多次元の最適化問題を解くことは難しいとされている. よって, それを避けるために, パラメータ直交化法を用いて, 推定したいパラメータと局外パラメータを分離し, 推定したいパラメータ θ だけの 1 次元の最適化問題に帰着させることができる.

まず, 一般的な確率分布として, $p(x;\theta,\boldsymbol{\beta})$ を考える. ただし, θ は推定したいパラメータであり, $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_M)$ は局外パラメータである. ここで, $(\theta,\boldsymbol{\beta})\to(\xi_1,\xi_2,\cdots,\xi_{M+1})=\boldsymbol{\xi}$ で, 次を満たす変数変換を考える.

$$\theta=\theta(\xi_1)=\xi_1,\ \beta_k=\beta_k(\xi_1,\xi_2,\cdots,\xi_{M+1}) \tag{6.1}$$

ここで, この変数変換は,

$$(J_\xi)_{1,k}=0 \tag{6.2}$$

を満たすようにする. なお, J_ξ は, 変数変換後のパラメータ $\boldsymbol{\xi}$ に対する Fisher 情報行列であり, (i,j) 成分を $(J_\xi)_{i,j}$ と表す. すると, $i=2,3,\ldots,M+1$ に対して, 以下の微分方程式が導かれる. (導出は付録 B で示す.)

$$J_{1,i}+\sum_{k=2}^{M+1}J_{i,k}\frac{\partial\beta_{k-1}}{\partial\xi_1}=0, \tag{6.3}$$

ただし, 一般に式 (6.3) は, 解析的な解を持たない. そして, パラメータ直交化の条件 $(J_\xi)_{1,k}=0$ $(\forall k\geq 2)$ は, 対応する尤度関数を $L(\boldsymbol{\xi};\boldsymbol{x})$ とすると, 次のように書き換えることができる.

$$\begin{aligned}
0=(J_\xi)_{1,k}&=\mathbb{E}\Big[\frac{\partial\ln p(x;\boldsymbol{\xi})}{\partial\xi_1}\frac{\partial\ln p(x;\boldsymbol{\xi})}{\partial\xi_k}\Big]\\
&=-\mathbb{E}\Big[\frac{\partial^2}{\partial\xi_1\partial\xi_k}\ln p(x;\boldsymbol{\xi})\Big]\\
&\approx-\frac{1}{N_x}\sum_g\frac{\partial^2}{\partial\xi_1\partial\xi_k}\ln p(x_g;\boldsymbol{\xi})\\
&=-\frac{1}{N_x}\frac{\partial^2}{\partial\xi_1\partial\xi_k}\ln L(\boldsymbol{\xi};\boldsymbol{x})
\end{aligned} \tag{6.4}$$

つまり, 推定したいパラメータである $\theta=\xi_1$ に対する尤度方程式 $(\partial/\partial\xi_1)\ln L(\boldsymbol{\xi};\boldsymbol{x})=0$ は, 任意の $k\geq 2$ である ξ_k によらないことがわかる. これは, 任意の $k\geq 2$ に対して, サンプル数 N_x が大きくなる漸近的な極限において, θ の臨界点 (微分がゼロとなる点) が, ξ_k に依存しないことを示している. それゆえ, $k\geq 2$ に対して大凡で選んだ $\bar{\xi}_k$ を用いて, 一次元の最適化問題である $\max_{\xi_1}\ln L(\xi_1,\bar{\xi}_2,\ldots,\bar{\xi}_{M+1};\boldsymbol{x})$ を解くことで, $\theta=\xi_1$

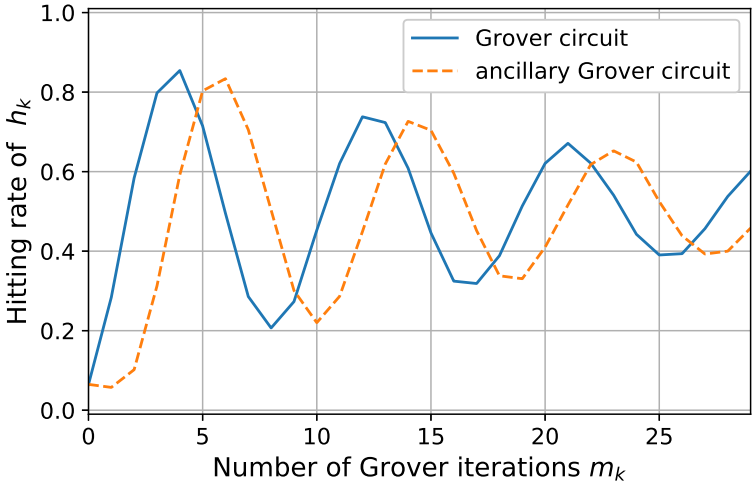


図 6.1　振幅増幅回路と補助回路に対する振幅増幅回数を増やした時の $|1\rangle$ の出る割合

ibm_kawasaki(1 番目及び 4 番目の量子ビットを使用) という実機を用いた場合の, 振幅増幅回路と補助回路を用いた際の “1” の出る確率を示す. なお, 横軸は振幅増幅の回数であり, $m_k = k$ とし, 測定回数 $N_{\mathrm{shot}} = 8,192$ とする. 青色の実線が, 式 (3.5) の, 振幅増幅演算子 G を用いた場合の結果であり, 橙色の破線が, 演算子 R_1 も用いた補助的な振幅増幅回路の式 (6.11) の結果である.

の最尤推定値を求めることができる. そして, $\xi_1(=\theta)$ における Cramér–Rao の不等式から導かれる下限, つまり変数変換後の Fisher 情報行列の逆行列の $(1,1)$ 成分である $(J_\xi^{-1})_{1,1}$ は, $(\theta, \boldsymbol{\beta})$ の多次元の最尤推定によって達成される値, つまり変数変換前の Fisher 情報行列の逆行列の $(1,1)$ 成分である $(J^{-1})_{1,1}$ と同じである. なお, 推定したいパラメータが複数ある場合には, パラメータ直交化法の一般的な標識は不明である.

6.2　補助回路の導入

前 6.1 節で, パラメータ直交化法について説明をした. この節では, 振幅推定法にパラメータ直交化法を用いるために, 未知のノイズがある状況での, パラメトリック分布モデルを定義することを行う.

まず, ノイズがない理想的なケースでは, 振幅増幅の回数 m_k に対して, 最後の量子ビットが 1 である確率 $p_G^{(k)}(1;\theta)$ は, 式 (3.15) に示している通り, 次式となる.

$$p_G^{(k)}(1;\theta) = \mathbb{P}(\{h_k = 1\}) = \sin^2((2m_k+1)\theta) \tag{6.5}$$

ここで, 改めて, 図 4.1 でも示している通り, IBM Quantum の超電導デバイス *ibm_kawasaki* を用いて, 振幅増幅の回路を実行すると, 図 6.1 の青色の実線となる. これは減衰した振動を取ることを示しており, 式 4.11 でも示してる通り, 脱分極ノイズモデルを仮定すると, 最後の量子ビットが 1 である確率 $p_G^{(k)}(1;\theta,\kappa)$ は,

$$p_G^{(k)}(1;\theta,\kappa) = \frac{1}{2} - \frac{1}{2}\mathrm{e}^{-\kappa m_k}\cos(2(2m_k+1)\theta) \tag{6.6}$$

となる. なお, $\kappa = 0$ の時は, つまりノイズがない時に該当し, それは, (6.5) と一致する. しかしながら, 実際には, 脱分極ノイズ以外のノイズも存在するため, より一般的な次のモデルに拡張を行う.

$$p_G^{(k)}(\theta,\beta_k) = \frac{1}{2} - \frac{1}{2}\beta_k\cos(2(2m_k+1)\theta). \tag{6.7}$$

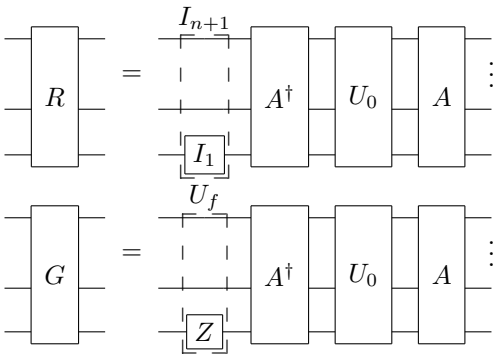


図 6.2　振幅増幅されない演算子 R および振幅増幅演算子 G の回路図

式 (6.10) で定義された演算子 $R = -AU_0A^\dagger I_{n+1}$ および, 振幅増幅演算子 $G = -A\ U_0\ A^\dagger\ U_f$ の回路図を示したものである.

これは, 振幅増幅の回数 m_k に応じて, 脱分極ノイズとは異なる種類のノイズを導入することができる. なお, 式 (6.7) は, 振幅増幅の演算子の実行時間を t とすると, 連続的な時間発展のモデルとして, $p_G(t;\theta,\beta(t)) = (1-\beta(t)\cos(2(2t+1)\theta))/2$ に帰着することができる. これは, 未知の関数 $\beta(t)$ を持つセミパラメトリックモデルと見做せるが, ここでは, 有限回だけのサンプリングを行うので, 有限次元のパラメトリックモデルになる.

そして, 異なる m_k から得られる確率を合わせた確率 $P_G(\mathbf{h};\theta,\boldsymbol{\beta})$ は,

$$P_G(\mathbf{h};\theta,\boldsymbol{\beta}) = \prod_{k=1}^{M} [p_G^{(k)}(\theta,\beta_k)]^{h_k} [1-p_G^{(k)}(\theta,\beta_k)]^{1-h_k} \tag{6.8}$$

であり, 尤度関数 $L_G(\theta,\boldsymbol{\beta};\mathbf{H})$ は

$$L_G(\theta,\boldsymbol{\beta};\mathbf{H}) = \prod_{j=1}^{N_{\mathrm{shot}}} P_G(\mathbf{h}^j;\theta,\boldsymbol{\beta}), \tag{6.9}$$

となる. なお, h_k, $\mathbf{h}^j$, および $\mathbf{H}$ は, 3.2 節と同じ意味である. したがってモデル分布は, 推定したいパラメータ θ と, 局外パラメータ $\boldsymbol{\beta} = (\beta_1,\ldots,\beta_M)$ で表すことができる.

ここで, 最尤推定法を用いる, つまり, 尤度関数の最大値を求める最適化問題 $\max_{\theta,\boldsymbol{\beta}} L_G(\theta,\boldsymbol{\beta};\mathbf{H})$ を解くことができれば良いが, 特に M が大きい場合には失敗することがある. パラメータ直交化法を適用することで, 最適化問題の次元を下げることができる. しかし, この場合, 出力数が M 個であるのに対して, パラメータ数 $M+1$ 個であるため, Fisher 情報行列が縮退する, つまり, Fisher 情報行列の行列式がゼロになってしまうため, 直交化法を用いることができない.

そこで, モデルの正則性を回復するために, ノイズパラメータの数を維持したまま, 追加の出力を出すことが必要がある. 言い換えると, 振幅増幅演算子 G^m の全体にかかるノイズと同じノイズに支配されながら, 異なる測定結果をもたらす量子回路があれば必要となる. よって, 次の「振幅増幅されない」演算子 R を導入する.

$$R = -AU_0A^\dagger I_{n+1} \tag{6.10}$$

この演算子 R は, 振幅増幅演算子 $G = -A\ U_0\ A^\dagger\ U_f$ における 演算子 $U_f = I_n \otimes \sigma_z$ を, 恒等演算子 I_{n+1} に置き換えたものであり, これらの演算子の回路図を図 6.2 に示す. なお, 現在の, IBM の超電導デバイスのように, 計算基底の測定 (つまり, Z 基底の測定) を行う量子デバイスでは, Z ゲートはフレームチェンジによって実現されている. つまり, 演算子 $\mathcal{U}_f$ は, システム上でパルス的な動作が行われていないことが知られている [83]. そして, この技術は, 超電導デバイスだけでなく, NMR 方式 [84] やイオントラップ方式 [70] のような別のタイプの量子デバイスでも適用されている. それゆえ, 演算子 G と R は同じノイズを受けていると仮定することは自然であり, この仮定は, 次で説明する実験結果にも裏付けられる.

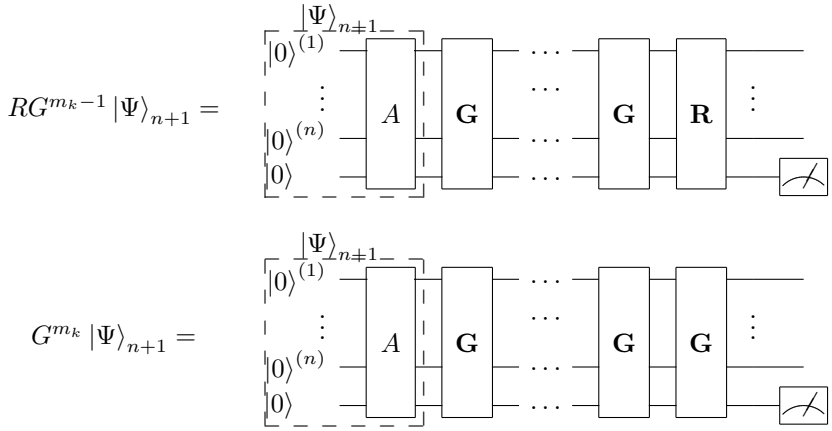


図 6.3　演算子 R を用いた補助的な振幅増幅回路 $RG^{m_k-1}|\Psi\rangle_{n+1}$ および振幅増幅回路 $G^{m_k}|\Psi\rangle_{n+1}$

振幅増幅回路 $G^{m_k}|\Psi\rangle_{n+1}$ における最後の演算子 G を振幅増幅しない演算子 R に変えた補助的な振幅増幅回路 $RG^{m_k-1}|\Psi\rangle_{n+1}$ を示したものである.

ここで, 図 6.3 で示すように, 演算子 R を用いて振幅増幅回路 $G^{m_k}|\Psi\rangle_{n+1}$ における最後の演算子 G を振幅増幅しない演算子 R に変えた補助的な振幅増幅回路 $RG^{m_k-1}|\Psi\rangle_{n+1}$ を考える. すると, ノイズがない場合, 次のように書ける.

$$\begin{aligned} RG^{m_k-1}|\Psi\rangle_{n+1} =& (-AU_0A^\dagger)(-AU_0A^\dagger U_f)G^{m_k-2}|\Psi\rangle_{n+1} \\ =& U_f G^{m_k-2}|\Psi\rangle_{n+1} \end{aligned} \tag{6.11}$$

現実のノイズの場合, 最終的な状態は $G^{m_k}|\Psi\rangle_{n+1}$ と同じノイズの影響を受ける. したがって, ノイズがある状況下での, RG^{m_k-1} における最後の量子ビットが 1 となる確率は,

$$\begin{aligned} p_{R_1}^{(k)}(\theta,\beta_k) &= \frac{1}{2} - \frac{1}{2}\beta_k \cos(2(2(m_k-2)+1)\theta) \\ &= \frac{1}{2} - \frac{1}{2}\beta_k \cos(2(2m_k-3)\theta). \end{aligned} \tag{6.12}$$

となる. これは位相の遅れがある振動であることを示している. そして, この補助的な振幅増幅回路 $RG^{m_k-1}|\Psi\rangle_{n+1}$ の実機による結果を図 6.1 における橙色の破線で示す. この結果は, 二つの回路 $G^{m_k}|\Psi\rangle_{n+1}$ と $RG^{m_k-1}|\Psi\rangle_{n+1}$ が同じノイズを受けているという仮定をサポートする結果である.

そして, $RG^{m_k-1}|\Psi\rangle_{n+1}$ の回路を, 合わせた確率は次で与えられる.

$$P_{R_1}(\boldsymbol{\ell};\theta,\boldsymbol{\beta}) = \prod_{k=1}^{M} [p_{R_1}^{(k)}(\theta,\beta_k)]^{\ell_k} [1 - p_{R_1}^{(k)}(\theta,\beta_k)]^{1-\ell_k} \tag{6.13}$$

更に, この二つの回路を用いて構成する尤度関数は次となる.

$$L(\theta,\boldsymbol{\beta};\mathbf{H},\mathbf{L}) = \prod_{i=1}^{N_{\text{shot}}} P_G(\mathbf{h}^i;\theta,\boldsymbol{\beta}) \prod_{j=1}^{N'_{\text{shot}}} P_{R_1}(\boldsymbol{\ell}^j;\theta,\boldsymbol{\beta}) \tag{6.14}$$

ここで, $\mathbf{L} = \{\boldsymbol{\ell}^1, \cdots, \boldsymbol{\ell}^{N'_{\text{shot}}}\}$ は, 補助的な振幅増幅回路の確率分布から得られる 2 値の確率変数 $\boldsymbol{\ell} = (\ell_1, \cdots, \ell_M)$ からサンプリングされた測定結果の集合であり, N'_{shot} は, RG^{m_k-1} の実行した (測定した) 回数である. そして, この補助的な回路と元の回路を用いることで, 対応する Fisher 情報行列の行列式はゼロでなくなり, パラメータ直交化法が適用できる正則モデルが確実に得られることになる.

6.3 パラメータ直交化法を用いた振幅推定法

6.1 節でパラメータ直交化法を説明し, 6.2 節では, 振幅推定手法への適用を議論した. ただし, 自然な適用では Fisher 情報行列の行列式がゼロになってしまうため, そこで補助回路を導入することで, 本問題を解消を解消することができる. そこで実際に本節では, 前 6.2 節で導入したモデルへのパラメータ直交化法を適用する. つまり, 微分方程式 (6.3) を満たす変数変換 (6.1) を求めることである. まず, 改めて満たすべき微分方程式を再掲する.

$$J_{1,i} + \sum_{k=2}^{M+1} J_{i,k} \frac{\partial \beta_{k-1}}{\partial \xi_1} = 0 \tag{6.15}$$

ここで, 前節でのモデルを適用すると, 全ての $k = 1, 2, \cdots, M$ に対して, 微分方程式は次のように書ける.

$$\begin{aligned} &\frac{\partial \beta_k(\boldsymbol{\xi})}{\partial \xi_1} \left(\frac{A_G^{(k)}(\xi_1)}{1 - A_G^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})} + \frac{A_{R_1}^{(k)}(\xi_1)}{1 - A_{R_1}^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})} \right) \\ &\quad = -\frac{\beta_k(\boldsymbol{\xi})}{2} \left(\frac{1}{1 - A_G^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})} \frac{\partial A_G^{(k)}(\xi_1)}{\partial \xi_1} + \frac{1}{1 - A_{R_1}^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})} \frac{\partial A_{R_1}^{(k)}(\xi_1)}{\partial \xi_1} \right) \end{aligned} \tag{6.16}$$

ただし, $N_{\text{shot}} = N'_{\text{shot}}$ とし, $A_G^{(k)}(\xi_1)$ および $A_{R_1}^{(k)}(\xi_1)$ を次で置く.

$$A_G^{(k)}(\xi_1) := \cos^2(2(2m_k + 1)\xi_1), \quad A_{R_1}^{(k)}(\xi_1) := \cos^2(2(2m_k - 3)\xi_1)$$

そして, 微分方程式 (6.16) から, すべての $k = 1, 2, \cdots, M$ に対して, $\beta_k(\boldsymbol{\xi})$ は次が導かれる.

$$\{1 - A_G^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})\}\{1 - A_{R_1}^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi})\} = c_k(\boldsymbol{\xi}') \tag{6.17}$$

なお, $c(\boldsymbol{\xi}')$ は, 局外パラメータ $\boldsymbol{\xi}' = (\xi_2, \cdots, \xi_{M+1})$ に関する任意の関数である. そして, 式 (6.17) は, 以下のように解くことができる.

$$\beta_k^2(\boldsymbol{\xi}) = \frac{A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1) \pm \sqrt{(A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1)^2 - 4A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)(1 - c(\boldsymbol{\xi}'_k))}}{2A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)} \tag{6.18}$$

ここで, $\beta_k(\boldsymbol{\xi})$ が $\boldsymbol{\xi}$ の実数の関数かつ, $\beta_k(\boldsymbol{\xi}) \geq 0$ であることから, 次の条件を満たす.

$$1 - \frac{(A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1))^2}{4A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)} \leq c_k(\boldsymbol{\xi}') \leq 1.$$

そして, $(A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1))^2 / (4A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)) \geq 1$ であることから, $0 \leq c_k(\boldsymbol{\xi}') \leq 1$ である時は, 解である全ての $\beta_k(\boldsymbol{\xi}_k)$ は実数である. さらに, $0 \leq p_G^{(k)}(\theta, \beta_k) \leq 1$ かつ, $0 \leq p_{R_1}^{(k)}(\theta, \beta_k) \leq 1$ であることから, 全て k に対して次の条件を満たすことが必要である.

$$0 \leq A_G^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi}) \leq 1, \quad 0 \leq A_{R_1}^{(k)}(\xi_1)\beta_k^2(\boldsymbol{\xi}) \leq 1.$$

よって, $\beta_k(\boldsymbol{\xi})$ は次で与えられる.

$$\beta_k(\boldsymbol{\xi}) = \sqrt{\frac{A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1) - \sqrt{(A_G^{(k)}(\xi_1) + A_{R_1}^{(k)}(\xi_1))^2 - 4A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)(1 - c_k(\boldsymbol{\xi}'_k))}}{2A_G^{(k)}(\xi_1)A_{R_1}^{(k)}(\xi_1)}} \tag{6.19}$$

パラメータ直交化法により, 式 (6.19) に現れる $c_k(\boldsymbol{\xi}')$ は, $0 \leq c_k(\boldsymbol{\xi}') \leq 1$ を満たす任意の関数であり, また, これは測定回数の多い漸近的な極限においては, $\theta = \xi_1$ には推定には影響を与えない. それゆえに, 大まかに選択した $c_k(\boldsymbol{\xi})'$ を尤度関数の $\beta_k(\boldsymbol{\xi})$ に代入し, 一次元の最大化問題を解くことで, θ を効率的に, またほぼ正確に計算することができる.

6.4 数値実験および実機を用いた検証

前節までで, 振幅推定の問題におけるパラメータ直交化法の導入について議論した. 特に, 脱分極ノイズモデルより一般的なモデルへ拡張し, さらに補助回路を導入することで, Fisher 情報行列の行列式がゼロになることを回避した. そして, 実際にパラメータ直交化法を適用すると, 驚くべきことに解析的に条件である微分方程式 (6.3) が解けることを示した. 本節では, 実際に, 提案した手法を確認するために, まず 6.4.1 項で, 数値シミュレーションを行い, 本手法の妥当性を示す. そして, 6.4.2 項では, 実際の超電導量子コンピュータ IBM Quantum *ibm_kawasaki* を用いた実験を行う.

6.4.1 数値実験

まず, サンプル数, つまり測定回数 $N_{\rm shot}$ が有限の状況下, つまり, 非漸近領域において, パラメータ直交化法が有用であるかを数値的に検証する. 特に, $\theta = \xi_1$ に関する尤度方程式の解を次のように求める.

$$\frac{\partial}{\partial \xi_1} \ln L(\boldsymbol{\xi}; \mathbf{H}, \mathbf{L}) = 0 \tag{6.20}$$

ここで, $\beta_k(\boldsymbol{\xi})$ は, 式 (6.19) で与えられる. よって, これらの解は, $N_{\rm shot}$ が比較的小さい場合であっても, 全ての $k \geq 2$ に対して, ほとんど ξ_k および $c_k(\boldsymbol{\xi}')$ に依存しないことがわかる. 実際, 量子コンピュータを用いた計算を行う上では, 測定回数 $N_{\rm shot}$ をできる限り小さくする必要があり, それゆえ, この解析は非常に重要である.

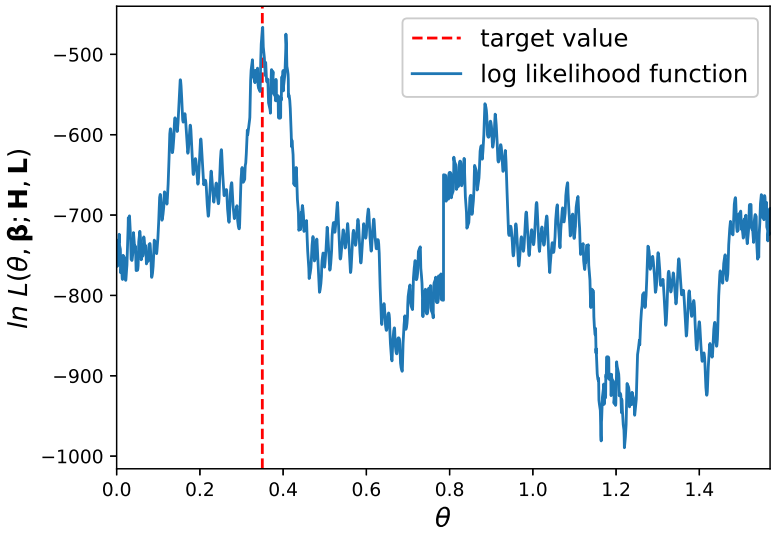
この数値実験のパラメータは次の通りとする. 目標となる $\theta_* = 0.35$ とし, 振幅増幅の回路および補助回路ともに, 測定の回数は $N_{\rm shot} = 50$ とする. そして, 振幅増幅の回数は, $m_k = 2^{k-1}$ $(k = 1, 2, \ldots, 8)$ とする. 真の確率分布は, 式 (6.6) における $\kappa = 0.01$ としたものであり, つまりは, 脱分極ノイズのみしか追加していないので, このパラメトリックモデルは β_k を適切に選択することで, この真の分布を表現することができる. 最後に, 式 (6.18) で与えられるパラメータ $\{c_1, \ldots, c_8\}$ は, 次の 2 つの場合を考える.

$$\textit{1}: \{0.844, 0.134, 0.956, 0.238, 0.236, 0.623, 0.793, 0.324\}$$
$$\textit{2}: \{0.571, 0.452, 0.475, 0.259, 0.107, 0.965, 0.362, 0.522\}$$

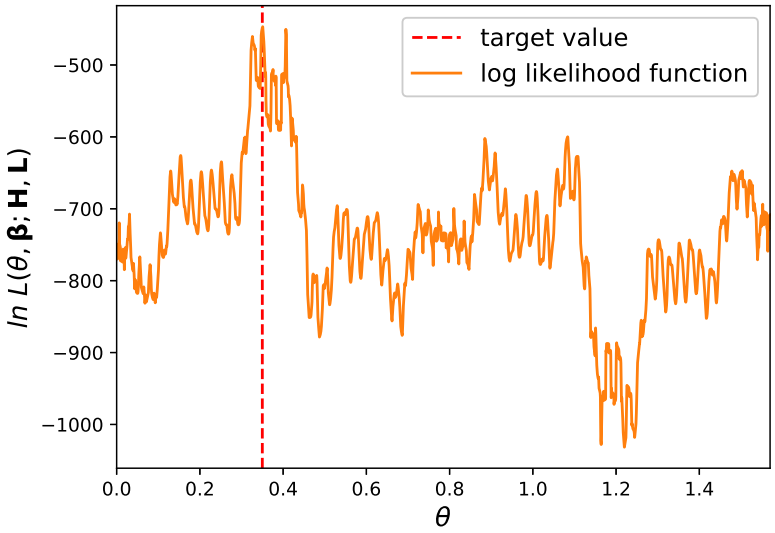
図 6.4(a), (b) はそれぞれ $\{c_1, \ldots, c_8\}$ がケース *1*, *2* の対数尤度関数 (6.14) の外形を, 青色の実線で示している. ただし, $\{c_1, \ldots, c_8\}$ 以外のパラメータは全て同じであり, 目標となる推定値は $\theta_* = 0.35$ であり, 赤色の破線で示している. これらの図から, 式 (6.20) の解である対数尤度の最大値は, 目標となる値に近いことがわかる. つまり, $\{c_k\}$ の値を考慮せずに, 一次元の最大化問題を解くことで解を得られることがわかる. 一方で, この尤度関数の形状は, 元の 9 次元の尤度関数は非常に複雑な形を持つことを示している. つまり, この尤度関数の複雑さは最尤推定がうまくいかない可能性があることを示唆しており, この多次元の最適化問題を扱わなくて良くなることが, パラメータ直交化法の最大の利点である.

そして, 最適解以外の $L(\boldsymbol{\xi}; \mathbf{H}, \mathbf{L})$ の臨界点, すなわち式 (6.20) の解は, ほとんど $\{c_k\}$ に依存しない. 図 6.5 は, 最適解付近のケース *1*, *2* の場合, つまり, 図 6.4 の拡大図を重ねたものであり, ケース *1* を青色の実線, ケース *2* の橙色の実線, 目標の値 $\theta_* = 0.35$ の赤色の破線で示している. この図 6.5 から, 対数尤度関数の形は, $\{c_k\}$ の値によって変わることを示している. 一方で注目すべきことは, 臨界点の位置がケース *1* とケース *2* で, 近いことである. 測定回数 $N_{\rm shot}$ が比較的小さいため, 若干の差はあるものの, 理論から予測されるように, $N_{\rm shot}$ を大きくすると, 差は小さくなることが確認できている.

次に, 本手法の推定精度について議論する. なお, パラメータについては $\{c_k\}$ の値以外, 前述と同じパラメータを用いる. 図 6.6 は, 横軸に演算子 A の呼び出し回数 $N_q = \Sigma_{k=1}^{M} N_{\rm shot}(2m_k + 1)$ を取り, 縦軸に推定精度を示したものである. 赤色の細い実線は, 古典的なランダムサンプリング, つまり $m_k = 0$ の時の Fisher 情報量



(a) Case *1*



(b) Case *2*

図 6.4　パラメータ直交化法を用いた後の対数尤度関数

青色の実線は, 式 (6.14) にある対数尤度関数を, 赤色の破線は推定したい値 $\theta_* = 0.35$ を示している.

の逆数から導かれる Cramér–Rao の下限であり, 橙色の太い実線は, ノイズのない理想的な量子振幅推定法を用いた場合の Cramér–Rao の下限である. この二つを比べると, 後者の方が二乗での早く誤差が減少することが改めて確認できる. そして, 三角点のある緑色の破線と四角点のある青色の点線は, それぞれ脱分極ノイズを仮定した際の, 二次元の最尤指定を用いた場合（4 章での手法）と, 本章で議論した場合の Cramér–Rao の下限である. また, 橙色の太い実線と, 緑色の破線, 青色の点線の三つの場合は, 振幅増幅の回数は $m_k = 2^{k-1}$ としており, 前者の二つの場合では測定回数を $N_{\text{shot}} = 100$ とし, 後者の場合では振幅増幅の回路および補助回路の両方の測定回数 $N_{\text{shot}} = N'_{\text{shot}} = 50$ としている. つまり, 各 m_k について, 100 回の測定を行なっている. なお, 青色の破線の場合, Fisher 情報行列の逆行列の (1,1) 成分を計算しているが, これはパラメータ直交化法を用いた前後で変化しない. そして, 紫色のバツ点は, 本章で提案している手法を用いてシミュレーションを行

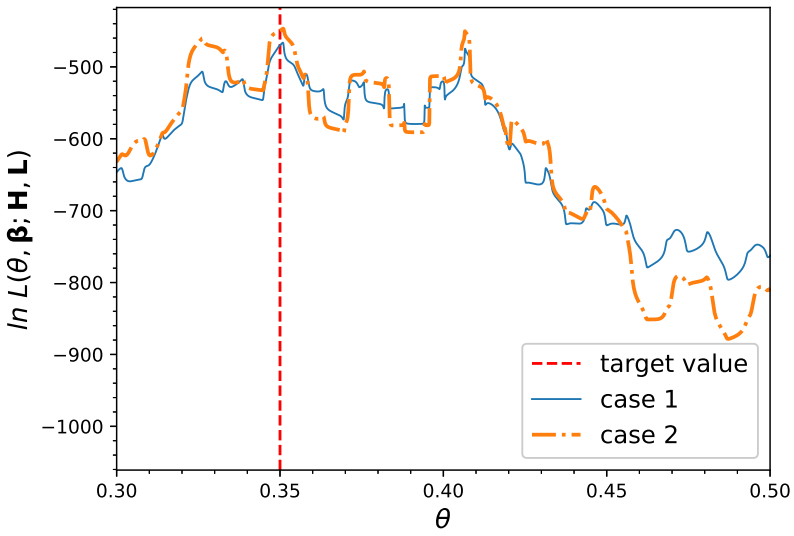


図 6.5　パラメータ直交化法を用いた後の対数尤度関数 (拡大図)

図 6.4(a),(b) の真の値 0.35 周辺の拡大図である.

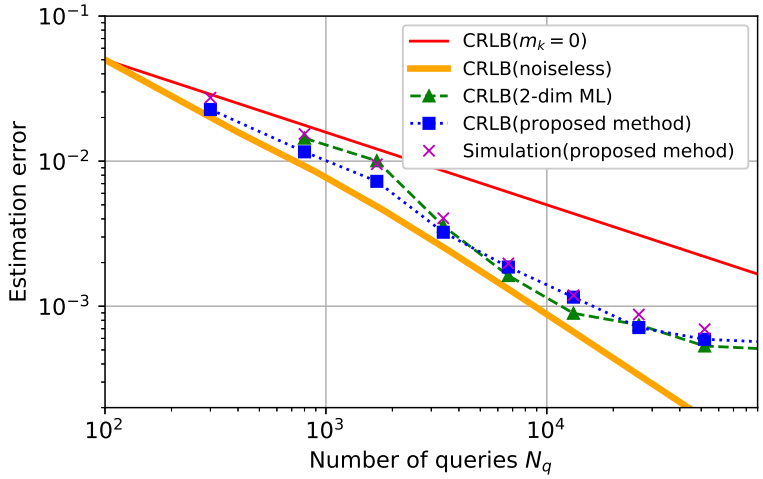


図 6.6　パラメータ直交化法を用いた数値シミュレーション結果

θ の推定精度と, 演算子 A の総呼び出し回数 N_q の関係を示す. 赤色の細い実線と, 橙色の太い実線は, ノイズのない Cramér–Rao の下限であり, それぞれ古典的な場合と, 量子的な場合を示している. 緑色の破線と, 青色の点線は, それぞれ, 4 章で説明した二次元の最尤推定を用いる手法と, 本章で説明しているパラメータ直交化法を用いた場合の Cramér–Rao の下限である. なお, 脱分極ノイズモデルを仮定している. そして紫色のバツ点は, 真の値 $\theta_* = 0.35$ と本章での手法を用いて数値シミュレーション計算された値 θ との標準誤差を示している.

表 6.1　2 量子ビット (実機) を用いた際のパラメータ一覧

項目	文字	値
測定回数	$N_{\mathrm{shot}} = N'_{\mathrm{shot}}$	50
振幅増幅の回数	m_k	2^k
推定する値	θ	0.175
誤差を評価するための繰返回数	N_{rep}	2,119

なったものであり, 真の値 $\theta_* = 0.35$ と推定した値 θ の分散を示している. なお, すべての k に対して $c_k = 0.3$ としている.

まず, 緑色の破線と青色の点線は, 非常に近いことがわかる. この結果は, 真の分布が脱分極ノイズだけからなることから, 本節で導入した, パラメータを増やしたモデルが真のモデルを正しく捉えることをできていることを示している. 緑色の破線については, 4 節でも議論した通り, ノイズがある状況でも両モデルともに, 赤色の細い実線, つまり古典的な場合より, 良い精度出せる領域があるということである. そして, もう一つの重要なことは, シミュレーションを行なった紫色のバツ点が, 理論的な下限である青色の点線にほぼ一致していることである. つまり, 本節で議論している手法がうまく機能していることを示している. そして, 多次元の, 特に本シミュレーションでは 9 次元の最適化問題を, 1 次元の最適化問題に帰着でき, うまく推定ができていることを示している. 図 6.5 で示した複雑な尤度関数の形は, 9 次元の関数を最大化し, 最尤推定を行うことが難しいことを示しており, 9 次元の最適化を行なった場合には, おそらく, シミュレーションの紫色のバツ点と, 理論下限である青色の点線の乖離が大きくなるものと推測される.

6.4.2　量子コンピュータ実機を用いた実験

6.4.1 項では, 振幅推定の問題にパラメータ直交化法を用いたシミュレーションを行い, 手法がうまくいくことを確認した. 本項では, 実際に超電導量子コンピュータ IBM Quantum(*ibm_kawasaki*) を用いて, 実際の実験で生じる未知のノイズを扱えるのかを議論する. なお, 実験の状況は, 3.5 節および, 4.3.1 項で用いたものである.

本項でのパラメータとしては, 推定したい $\theta = 0.175$ とし, つまり, $S = 3.03^{-2}$ とし, 振幅増幅の回数の数列は, $m_k = 2^{k-1}$, $(k = 1, 2, \ldots, 7)$ とする. そして, m_k に応じて異なるノイズが発生すると想定する, つまり, 7 つの局外パラメータ $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_7)$ が存在するとし, それに対してパラメータ直交化法を用いる. なお, パラメータ直交化法に用いる c_k は全ての k に対して 0.3 とし, 測定回数は演算子 G^{m_k} を用いる場合と, 演算子 RG^{m_k-1} を用いる場合の双方で, $N_{\mathrm{shot}} = N'_{\mathrm{shot}} = 50$ とする. これらの実験に用いたパラメータは次の表 6.1 にまとめる.

そして, 図 6.7 に, 実験結果を示す. この図は演算子 A の呼出回数の合計 $N_q = \Sigma_{k=1}^{M} N_{\mathrm{shot}}(2m_k + 1)$ と, 推定誤差の関係を示している. 前の図 6.6 と同様に, 赤色の細い実線と, 橙色の太い実線は, Cramér–Rao の不等式から導かれる下限であり, それぞれ古典的な場合, つまり $m_k = 0$ の場合と, $m_k = 2^{k-1}$ を用いたノイズのない量子の場合を示している. そして, 青色のバツ点は, 実機の結果から推定された θ と, 真の値 $\theta_* = 0.175$ との推定精度の誤差を示したものである. なお, 同じ状況下での実験を 2,119 回繰り返して行い, その標準誤差の 3 倍を, エラーバーで示している.

そして, この図から $N_q = 3 \times 10^3$ の範囲では, 多少の定数のオーバーヘッドがあるものの, N_q^{-1} に近い傾きを示している. このオーバーヘッドは, デバイスの不確かさやノイズの大きさが前項のシミュレーションよりも大きかったことによるものと考えられる. オーバーヘッドがあるにもかかわらず, $N_q = 3 \times 10^3$ 付近の実験結果は赤線を下回っている結果があることは, 古典的な手段より良い推定値を得られていることを示している.

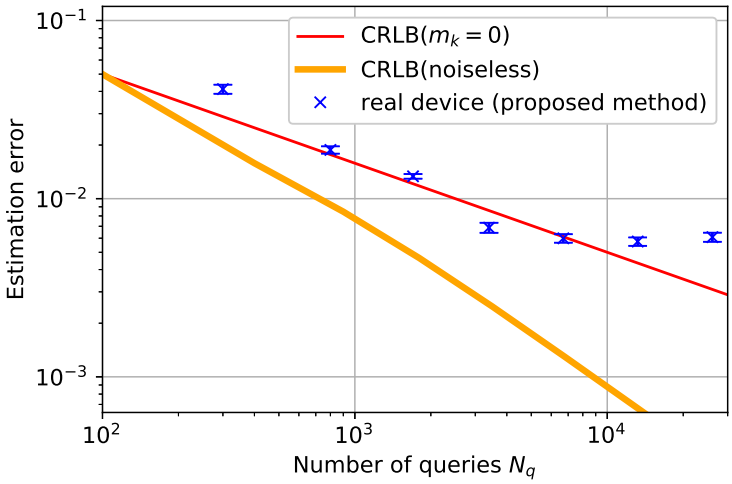


図 6.7　パラメータ直交化法を用いた実機での検証の結果

θ の推定精度と, 演算子 A の総呼び出し回数 N_q の関係を示す. 赤色の細い実線と, 橙色の太い実線は, ノイズのない Cramér–Rao の下限であり, それぞれ古典的な場合と, 量子的な場合を示している. 青色のバツ点は, 実機の結果であり, この手法を用いた際の推定精度の誤差を示している.

ただし, 上回っている点より更に振幅増幅の回数を増やしても, 更なる余分なノイズが発生してしまうため, 推定精度の改善は見込めないものである. また, 4.3 節で用いた量子コンピュータ *ibmq_valencia* と, 本項で用いた *ibm_kawasaki* におけるノイズの大きさは同程度であり, 振幅増幅の回数を増やしても推定精度の改善をできないということは矛盾しない結果である. そして, この結果は 1 次元の最大化問題を解くことで推定した値 θ を得ることができ, 現実の特定できないノイズ環境下でもうまく機能していることを示している. これは脱分極ノイズモデルより一般的なモデルを, 更に局外パラメータの数を増やすことで, より複雑なモデルに拡張できる可能性があることを示している. そして, パラメータ直交化法を用いることで, θ の推定値は, 局外パラメータによらず, また局外パラメータを推定することなく, ほぼ正確に計算できる可能性があることを示唆している.

6.5　本章のまとめ

本章では, パラメータ直交化法を用いた振幅推定法について述べた. 6.1 節でパラメータ直交化法に関する簡単なレビューを行い, 6.2 節では, 振幅推定問題へのパラメータ直交化法の適用を議論した. まず, 脱分極ノイズモデルより一般的なモデルを導入するが, その場合, Fisher 情報行列の行列式がゼロになることから, 推定が困難である. そこで, その問題を解決するために, 補助回路の導入を行い, 行列式がゼロになる問題を回避した. そして, 6.3 節では実際に, 導入した補助回路を用いたモデルに対してパラメータ直交化法を適用した. 驚くべきことに, パラメータ直交化法を用いる上で解くべき微分方程式が解析的に解を持つことを示した. 続く 6.4 節では, 数値シミュレーションと実機を用いた検証を行い, 本手法がうまく機能していることを示した.

この結果は, 更なるモデルの拡張の可能性を示している. つまり, 更なる局外パラメータを導入することで, より一般的なモデルを扱うことができる可能性があることを示している. しかしながら, 局外パラメータを導入することにより, Fisher 情報行列の行列式がゼロになる可能性もある. 本節では, 最後の一つの振幅増幅演算子 G を導入した振幅増幅しない演算子 R に置き換えることを行なったが, R を複数導入するモデルも考える

ことが可能であり, 様々な補助回路の導入することで, 本節での議論と同様に Fisher 情報行列の縮退を回避できる可能性がある. また, 5 章で導入した修正した振幅増幅演算子 Q を用いた議論も可能であると考えられる.

本節での議論は, 現在のノイズのある量子コンピュータを扱う上では, 非常に重要な方向性を示している可能性がある. 現在および, 近未来に渡っては, ノイズのある量子コンピュータを扱っていく必要があるが, ノイズは未知であり完全な特定は不可能である. そこで, 統計的推定理論は, このような現実的な問題設定における有用な手法を提供する可能性があり, また, 将来の Fault Tolerant 量子コンピュータに向けても, また, Fault Tolerant 量子コンピュータにおいても, 更に重要な役割を担う可能性が十分にあるものである.

第 7 章

結論

本稿では, ノイズがある場合の量子振幅推定法に関する解析を行うとともに, そのアルゴリズムを実際にノイズのある量子コンピュータを用いて実験, 検証を行った.

2 章では基本的な量子情報に関する事項を説明した.

3 章では, まず, [57] で著者らによって提案されている, 量子位相推定を用いない量子振幅推定法を説明した. 従来の提案 [11] では, 量子振幅推定法には, 量子位相推定と量子振幅増幅の二つのアルゴリズムが必要であるとされていた. [57] で提案した通り, 量子振幅増幅を並列的に実行し, 各々の結果を測定し, それらの測定結果から尤度関数を構築し, 最尤推定を行うことで, 量子位相推定を用いずに量子振幅推定をできる手法を提案した. この最尤推定を用いる量子振幅推定の特徴は以下である.

- 量子位相推定を用いずに, 古典的な手法に比べて, 計算量の二乗の優位性を維持したまま, 量子振幅推定を行うことができる.
- 制御された振幅増幅演算子の制御部分および, 逆量子フーリエ変換が不要となる.
- 量子ゲートの数や量子回路の深さが, 従来の手法に比べて大幅に少なくてすむ.
- 従来の手法では振幅増幅演算子を繰り返す回数の数だけ, 利用する量子ビットおよび測定する量子ビットの数が増えるが, 本手法では演算子の回数の数によらず利用する量子ビットの数を定数化でき, 更に, 測定する量子ビットが, 1 量子ビットで良くなる.
- 一方で量子コンピュータの実行回数は増えるものの, 計算量のコストである演算子 A の呼び出し回数は, 変わらない.

そして, 最尤推定を用いることで, 漸近一致性が保証されるとともに, シミュレーション結果は高々数十から百程度の測定回数で問題ないことを示した. なお, この手法は, 仮に誤り訂正可能な量子コンピュータ, あるいは Fault Tolerant 量子コンピュータ (FTQC) が実現したとしても非常に有用なものであると考えている. 現状の NISQ と呼ばれる限られた量子コンピュータを使う上では, 量子ビットの数も限られ, 量子ゲート数にも制限があり, アルゴリズムを工夫することが必要不可欠である. 一方で, 昨今のハードウェアの進展を鑑みると, 将来の大規模化, エラー訂正の実現が期待されているが, 仮に実現した際にも, 突然ノイズのない, すなわちエラー訂正された数万, 数十万量子ビットのデバイスができるわけではなく, エラー訂正の実現された 1 量子ビット, 10 量子ビットと徐々に伸びていくと考えることが自然である. つまり, 現在の NISQ に限らず, 量子コンピュータのハードウェアに求められる要求を, ソフトウェア・アルゴリズムの改善によって実現していくことが重要なテーマの一つであることは言うまでもないことである. なお, 3.5 節で例として出した三角関数の積分は, 著者らによって Qiskit community[85] に公開されているとともに, 本手法自体も Qiskit Finance Tutorials のトップ [66] にも実装されており, 金融問題へのアプローチの重要なアルゴリズムの一つである位置付けになっている.

続く 4 章では, ノイズの影響がある状況下での量子振幅推定法について述べた. これは, 著者らによって [65] で提案されているものである. 前述の通り, 現在, 量子コンピュータのハードウェアは日々, 進展が凄まじいものの, まだノイズがある状況であり, この状況下で何ができるのかを考えていくことが価値があることであると考える. そこで, 前述の最尤推定を用いる量子振幅推定法について, ノイズがある状況下での議論を行った. 一般的なノイズモデルの一つである脱分極ノイズを仮定していて, 推定精度がどうなるのかを議論した. この章での主な貢献は以下の通りである.

- 脱分極ノイズモデルを導入し, 最尤推定を用いた振幅推定法を構築した.
- さらに, この定式化から, 推定精度, 演算子 A の呼び出し回数, ノイズの大きさなどの関係を導いた.
- 実際の超電導量子コンピュータを用いて, 小規模ながら本モデルの有用性を検証した.
- 特に, ノイズの大きさは未知のパラメータとする, つまりマルチパラメータ推定問題として扱った. これは一般的には難しい問題であり, 本問題では推定が困難な例外的な場合が存在するが, その割合が少ないことと回避法を述べた.
- 本モデルを用いて, ハードウェアの見積もりに関する議論をした. どの程度のハードウェアがあれば何ができるか, あるいは行いたい計算をするために必要なハードウェアを見積もることができることを示した.

この 4 章での実機を用いた実験の中では, 僅かながら古典的な場合より, 実験結果が良い精度を示す場合があった. この結果は, たとえノイズがある量子コンピュータであったとしても, 有用な可能性があることを示唆しているものである. ただ, この有用性はあくまで計算量の観点のみであり, 将来での活用を見据える上では計算時間という尺度での検証も必要なものであり, 今後の課題の一つである.

そして, 5 章では, 著者らの [75] に基づき, 新たな振幅増幅演算子の議論を行なった. この振幅増幅演算子は従来のものとは別の基底で貼られた空間を回転する演算子である. すると, 導入した演算子を別の基底で測定することにより, 脱分極ノイズモデルを仮定した場合に, ヒルベルト空間の次元が 2 より大きい場合には, 従来の演算子では達成不可能な量子 Fisher 情報量を達成できる可能性があるものである. ポイントとしては以下である.

- 新たに, 修正した振幅増幅演算子を導入した. そしてノイズのない場合には, 従来の手法同様, 二乗加速が得られることを示した.
- 脱分極ノイズがある場合には, 修正した振幅増幅演算子を用いることで, 従来の演算子では達成不可能な量子 Fisher 情報量を達成できる可能性があることを示した.
- 量子 Fisher 情報量を達成するためには, 逐次的な測定計画が必要とされているが, ナイーブな測定法で従来手法より良い精度を得ることができることをシミューレーションで示した.

最後の議論として, 6 章で, [76] で述べている振幅推定法を説明した. 主な議論の流れは以下である.

- まず, 4 章で導入したモデルをさらに拡張・一般化を行なった.
- しかしながら, モデルを拡張する, つまりパラメータを増やすと, 推定問題としての難易度は上がる. そこで, 複数のパラメータ推定問題では, 統計的な手法がいくつも提案されており, ここではパラメータ直交化法の適用を検討した.
- ただし, 拡大したモデルでは, Fisher 情報行列の行列式がゼロになり, 推定ができない状況となる. そこで, 補助回路の導入を行い, 合わせた確率分布および尤度関数を導入することで, Fisher 情報行列の逆行列がゼロになることを回避し, パラメータ直交化法の適用を行なった.
- ここで推定したいパラメータが一つの場合には, ある微分方程式を解けば, パラメータ直交化法を適用で

きることが知られている．そして, この問題においては, この微分方程式が解析的な解を持つことを示した．

- 最後に, 数値実験と実機を用いた検証で本手法の有用性を確認した．

これまでのところでノイズに関して様々な議論を行なったが, ノイズに関しては議論が簡単ではない．なぜならば, 実際の量子コンピュータにおいて, ノイズのモデルを完璧に与えることが不可能だからである．そこで 4 章では脱分極ノイズを仮定したモデルを扱い, 6 章では更に拡張したモデルを扱った．どちらのモデルも, 実機での検証では決して悪い結果ではないものの, 更なる一般化は重要なテーマの一つである．また, 量子コンピュータと言えど, 測定結果は 0 か 1 の古典的なデータである．つまり, 古典的な統計手法を用いて解析をすることの意義は大切である．本章では, パラメータ直交化法を用いたが, プロファイル尤度関数を用いる方向性も議論の余地があると考える．さらに, これらの統計的な手法を他の量子アルゴリズムに適用することで, 広がりが出てくる可能性があるものと考える．

改めて, 本稿では主にノイズと, 振幅推定法の二つを研究テーマとして議論した．前者は, 現在および近未来の量子コンピュータを扱う上では避けて通れないテーマの一つであり, 前段落で述べた通りである．さらに, 例えばエラー緩和, エラー訂正, FTQC の実現に向けても, 実際の量子コンピュータにおけるノイズの特定やモデリングというのは, 重要であることは言うまでもない．そして, 後者の振幅推定法は, 量子アルゴリズムの中でも以下の点から重要なものの一つである．

- 古典的な手法に比べて計算量で二乗の優位性がある点．
- モンテカルロ計算や機械学習などの様々な応用が見込まれている点．

量子コンピュータを使う上では, 量子コンピュータだからこそできる計算を行うことが期待されているが, 振幅推定法は, それに十分に実現できる可能性があり, 更に応用エリアも汎用的かつ重要な分野で期待がされているものである．最後に, 量子コンピュータは萌芽期の技術であるが, だからこそ, 様々な分野の専門家が研究を行うことで, 新たな発見があり, 科学が進展するとともに, 計算科学の分野が発展していくことを期待する．そして, 本稿がその一助になることも期待する．

謝辞

本研究は, 著者が慶應義塾大学大学院理工学研究科後期博士課程在学中に, 同大学理工学部山本直樹教授の指導のもとに行ったものです. 山本先生には, 月並みながら, 感謝の言葉しかございません. まず, 社会人ドクターとして受け入れて頂いたことに始まり, 夜間・休日でも論文の確認を頂き, また, 研究の進め方を始め, 論文の書き方や, 査読への対応など研究における当たり前にできるべきことをご指導賜りました. 心より感謝申し上げます.

同大理工学部, 齊藤圭司教授, 小林景准教授, 早瀬潤子教授に副査をお受け頂き, 大変感謝致します. 量子情報・統計に関わる理論的な側面, 量子コンピュータを用いた実験的な側面と様々なところについて, ご助言を頂きました. ありがとうございました.

本研究活動自体が, IBM Quantum Network Hub @ Keio University という産学連携のプロジェクトに, 自身が所属する株式会社三菱 UFJ フィナンシャル・グループとして参画させて頂くにあたり, 同大伊藤公平塾長 (当時理工学部長) から, 派遣する共同研究員への博士課程入学をご提案頂いたことから始まっており, 貴重な機会を頂き, 大変感謝しております.

そして, 本プロジェクトにあたり, 同大量子コンピューティングセンターに関わる先生方, 学生の方々, 共同研究員の方々には, 大変お世話になりました. 特に, 山本先生を筆頭に, 金融チームのメンバーの方々には, 毎週様々な議論をさせて頂いており, 感謝申し上げます. その中でも, みずほリサーチ&テクノロジーズ株式会社から共同研究員として派遣されている宇野隼平上席主任コンサルタント, 金融チームをリードして頂いる同センターの鈴木洋一特任准教授には, 遅い時間まで何度も矢上キャンパスのコモンルームで, コロナ禍ではオンラインでも, ディスカッションさせて頂き, ありがとうございました. また, 日本 IBM から, 小野寺民也副所長, Raymond Harry Putra Rudy 研究員とも様々な議論をさせて頂き, 感謝致します. 言うまでもなく, 本研究は IBM Quantum という量子コンピュータの実機があったからこそできた研究であり, その利用にあたっては日本 IBM 渡辺日出雄部長を筆頭に日本及び, グローバルの IBM Qunatum のチームの皆様には大変にサポートいただき, 感謝申し上げます.

本博士論文では扱っていないものの, 博士課程在籍中には, 機械学習 [113, 114] や暗号 [115, 116] 等に関する研究にも取り組ませて頂きました. 前述の皆様を始め, 更に, [113] では, 三菱ケミカル株式会社から共同研究員として派遣されている高王己主席研究員, 同大博士課程の鈴木 雄大さん, 矢野碩志さん, 秋山真那斗さん, [114] では, ソニーグループから共同研究員として派遣されている手塚宙之研究員, 三井住友信託銀行株式会社から共同研究員として派遣されている光田尚生調査役, 同大博士課程の中路紘平さん, [115] では, 東京大学博士課程の大西健斗さん, 筑波大学の國廣昇教授, そして慶應義塾大学 Rodney Van Meter 教授, 佐藤特任講師と, [116] では, 情報通信研究機構の野島良室長, 篠原直行研究マネージャー, 青野良範主任研究員, 慶應義塾大学環境情報学部の Sitong Liu さんと, 研究をさせていただき, 感謝いたします. そして, 直接の共著論文という形にはなっていないものの, JSR 株式会社から共同研究員として派遣された大西裕也次長, 須藤翔太朗研究員, 佐久間怜研究員, 由本美香研究員, 三菱ケミカル株式会社から共同研究員として派遣された渡部絵里子研究員, 小林高雄主任研究員, 株式会社日立製作所から共同研究員として派遣された李憲之研究員, 日本 IBM から共同研究員として派遣された中村肇主任研究員, そして, 量子コンピューティングセンターの関口武治特任准教

授 (在籍時), 鹿野豊特任准教授 (在籍時), 渡邉宙志特任講師, 上西慧理子特任講師, 菅原道彦特任准教授, 川口英明特任講師の方々にも様々な議論をさせていただきました. 一貫して量子コンピュータという共通の課題がありつつも, 様々なテーマに取り組ませていただけたのは, まさにこの博士課程および共同研究員として量子コンピューティングセンターという環境があったからであり、運営を支えて頂いている新美新一プロジェクトマネージャー、秘書の相沢志乃さん、大上直子さんにも感謝申し上げます。また、博士課程在籍及び共同研究員として量子コンピューティングセンターでお世話になった方々に大変感謝申し上げます.

更に, 前述の本プロジェクトへの会社の参画を決め, 私を派遣することを決めて頂いた, 弊社の亀澤宏規取締役代表執行役社長グループ CEO, 亀田浩樹執行役常務グループ CIO を筆頭に, 派遣時から現在に至るまでの関係部, 及び自身が所属するシステム企画部の上司・同僚の方々には, 多大なるご配慮, ご支援を頂きました. 特に, 弊社では自社の社員を外部に常駐させ, 産学連携で研究を行うという, 前例のないプロジェクトの中で派遣いただいた当時の上司である越智俊城執行役員事務・システム企画部長 (当時), 高橋良一副部長 (当時), 松井雅弘次長 (当時), 栗山英樹次長 (当時), そして, 継続的に支援いただいている現在の上司である山本一也システム企画部長, 本間義道執行役員システム企画部部長, 高橋一興システム企画部部長, 森井克典副部長, 中島太一上席調査役, 秋山聖太調査役に, 本当に感謝申し上げます. 私が存じ上げないところでも社内のたくさんの方が様々な調整を頂き, 今日に至っており, 特に, プロジェクトの参加だけでなく, 会社負担での博士課程の在籍をさせて頂き, ありがとうございます. 引き続き, 微力ながら今後の量子コンピューティングの発展に寄与を目指し, 日本, 金融業界, そして, 自社での技術活用などでの還元により, 様々な社会課題の解決を目指して参る所存です.

最後に, 妻と長男のサポートがあったからこそ, この執筆に至っており, 特に妻には相談もせずに博士課程に入ることを決めてしまった状況であったものの, 理解し, 支えてくれたことに感謝致します.

付録

A　別の 2 量子ビットを用いた実験結果

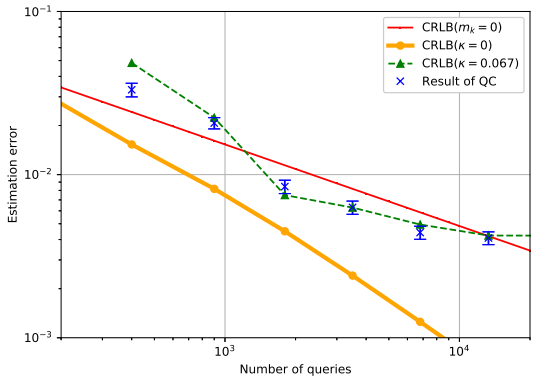
4.3.1 項では, 実際の量子コンピュータで 2 量子ビットを用いて, 三角関数の積分を行なった. ここでは, 4.3.1 項では用いたのと別の設定での実験結果を示し, 量子ビットを変えたこと, および推定する θ によらないことを示す. 各パラメータは表 A.1 に示す通りであり, 追加実験 *1* では本文とは利用する量子ビットを変え, 追加実験 *2* では更に推定する θ の値を変えている. この実験では, 2 量子ビットを用いるつまり, リーマン積分の分割数が 2 の場合である. つまり, $S(f) = (\sin^2 \pi/4b + \sin^2 3\pi/4b)/2$ であり, $S(f) = \sin^2 \theta$ である.

4.3.1 項での結果および追加実験の結果を図 A.1 に示す. この図は, 演算子 A の呼び出し回数と推定誤差の関係を示しており, 赤色の細い実線, 橙色の太い実線, 緑色の破線は, それぞれ古典的な場合, ノイズがない量子的な場合, 実機の結果から推定されるノイズのある量子的な場合の Cramér–Rao の下限を示している. そして, 青色のバツ点は, 真の値と, 実機から最尤推定を用いて推定した値の標準偏差であり, エラーバーは標準誤差の 3 倍を示したものである.

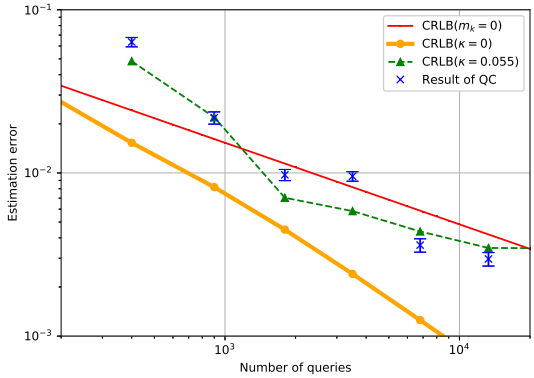
この結果は, 脱分極ノイズモデルを仮定した Cramér–Rao の下限 (緑色の破線) と近い値であり, モデリングが悪くないことを示している. そして, θ の値によって, ノイズの大きさ κ が異なることから, 実験ごとにノイズの大きさを推定することが重要であるとともに, ノイズの大きさ κ は使う量子ビットによっても異なるため, より良い推定を行うためには, 使う量子ビットの選択も重要である. なお, (b) と (c) では, エラーバーの値が違うように見えるが, これは縦軸が対数での表記になっているものであり, これらの差は高々 2 倍程度である.

表 A.1　追加実験におけるパラメーター一覧

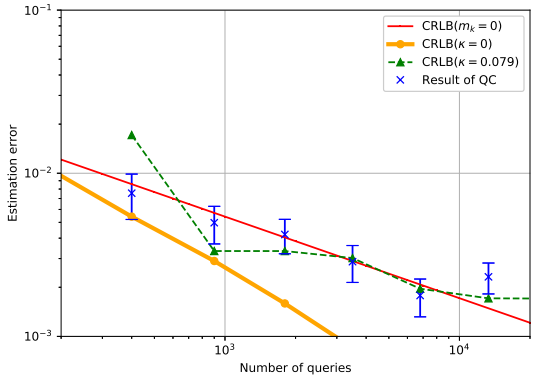
項目	文字	4.3.1 項	追加結果 *1*	追加結果 *2*
デバイス	-	IBM Quantum *ibm_valencia*		
測定回数	N_{shot}	100		
振幅増幅の回数	m_k	2^k		
測定回数	N_{shot}	100		
誤差を評価するための繰返回数	N_{rep}	1,064		
積分区間	b	$4/10\pi$		$1/10\pi$
積分値	$S(f)$	0.375		0.03
推定する値	θ	0.659		0.175
利用する量子ビット	-	0 番目, 1 番目	3 番目, 4 番目	



(a) 4.3.1 項 (再掲)



(b) 追加実験 *1*



(c) 追加実験 *2*

図 A.1　2 量子ビット (実機) を用いた際の演算子 A の呼び出し回数と推定誤差の関係

赤色の細い実線, および橙色の太い実線は, Cramér–Rao の下限であり, それぞれ, 古典的な場合 ($m_k = 0$) とノイズのない量子の場合 ($m_k = 2^k$, $\kappa = 0$) である. そして, 緑色の破線は, 実機のデータから求めたノイズの大きさを用いた Cramér–Rao の下限である. 青色のバツ点は, 真の値と, 実機から最尤推定を用いて推定した値の標準偏差である.

B　パラメータ直交化法の条件式の導出

本節では, 6.1 節で説明しているパラメータ直交化法について, 式 (6.3) の導出を行う. つまり, パラメータ直交化法で用いる条件である $(J_\xi)_{1,k} = 0$ を満たす条件が, 次であることを示す.

$$J_{1,i} + \sum_{k=2}^{M+1} J_{i,k} \frac{\partial \beta_{k-1}}{\partial \xi_1} = 0 \tag{B.1}$$

まず, 変数変換を τ を用いて改めて表記する.

$$\begin{aligned}
\theta &= \tau_0(\xi_1) = \xi_1, \\
\beta_1 &= \tau_1(\xi_1, \xi_2, \xi_3, \cdots, \xi_{M+1}), \\
\beta_2 &= \tau_2(\xi_1, \xi_2, \xi_3, \cdots, \xi_{M+1}), \\
&\vdots \\
\beta_M &= \tau_M(\xi_1, \xi_2, \xi_3, \cdots, \xi_{M+1}).
\end{aligned} \tag{B.2}$$

そして, $(J_\xi)_{1,k}$ をチェーンルールを用いて計算すると, 次の通りとなる. ただし, $f = \ln p(x; \theta, \boldsymbol{\beta})$, $g = \ln p(x; \boldsymbol{\xi})$ と表記する.

$$\begin{aligned}
(J_\xi)_{1,k} &= \mathbb{E}\Big[\frac{\partial g}{\partial \xi_1}\frac{\partial g}{\partial \xi_k}\Big] \\
&= \mathbb{E}\Big[\Big(\frac{\partial f}{\partial \theta}\frac{\partial \tau_0}{\partial \xi_1} + \sum_{h=2}^{M+1} \frac{\partial f}{\partial \beta_{h-1}}\frac{\partial \tau_{h-1}}{\partial \xi_1}\Big)\Big(\frac{\partial f}{\partial \theta}\frac{\partial \tau_0}{\partial \xi_k} + \sum_{i=2}^{M+1} \frac{\partial f}{\partial \beta_{i-1}}\frac{\partial \tau_{i-1}}{\partial \xi_k}\Big)\Big] \\
&= \mathbb{E}\Big[\Big(\frac{\partial f}{\partial \theta} + \sum_{h=2}^{M+1} \frac{\partial f}{\partial \beta_{h-1}}\frac{\partial \tau_{h-1}}{\partial \xi_1}\Big)\Big(\sum_{i=2}^{M+1} \frac{\partial f}{\partial \beta_{i-1}}\frac{\partial \tau_{i-1}}{\partial \xi_k}\Big)\Big] \\
&= \sum_{i=2}^{M+1} \frac{\partial \tau_{i-1}}{\partial \xi_k}\mathbb{E}\Big[\Big(\frac{\partial f}{\partial \theta} + \sum_{h=2}^{M+1} \frac{\partial f}{\partial \beta_{h-1}}\frac{\partial \tau_{h-1}}{\partial \xi_1}\Big)\frac{\partial f}{\partial \beta_{i-1}}\Big] \\
&= \sum_{i=2}^{M+1} \frac{\partial \tau_{i-1}}{\partial \xi_k}\Big(\mathbb{E}\Big[\frac{\partial f}{\partial \theta}\frac{\partial f}{\partial \beta_{i-1}}\Big] + \sum_{h=2}^{M+1} \mathbb{E}\Big[\frac{\partial f}{\partial \beta_{i-1}}\frac{\partial f}{\partial \beta_{h-1}}\Big]\frac{\partial \tau_{h-1}}{\partial \xi_1}\Big) \\
&= \sum_{i=2}^{M+1} \frac{\partial \tau_{i-1}}{\partial \xi_k}\Big(J_{1,i} + \sum_{h=2}^{M+1} J_{i,h}\frac{\partial \tau_{h-1}}{\partial \xi_1}\Big)
\end{aligned} \tag{B.3}$$

なお, $\frac{\partial \tau_0}{\partial \xi_1} = 1$, $\frac{\partial \tau_0}{\partial \xi_k} = 0$, であることを用いている. 求めたいパラメータ直交化法の条件は, 式 (B.3) がゼロであることであり, よって, これは式 (B.1) であることである.

参考文献

[1] Richard P Feynman. Simulating physics with computers. International journal of theoretical physics, Vol. 21, No. 6/7, pp. 467–488, 1982.

[2] David Deutsch. Quantum theory, the church-turing principle and the universal quantum computer. Vol. 400, pp. 97–117, 1985.

[3] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences, Vol. 439, No. 1907, pp. 553–558, 1992.

[4] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM review, Vol. 41, No. 2, pp. 303–332, 1999.

[5] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, Vol. 21, No. 2, pp. 120–126, 1978.

[6] Neal Koblitz. Elliptic curve cryptosystems. Mathematics of computation, Vol. 48, No. 177, pp. 203–209, 1987.

[7] Victor S Miller. Use of elliptic curves in cryptography. In Conference on the theory and application of cryptographic techniques, pp. 417–426. Springer, 1985.

[8] Andrew Steane. Multiple-particle interference and quantum error correction. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, Vol. 452, No. 1954, pp. 2551–2577, 1996.

[9] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. Physical Review A, Vol. 54, No. 2, p. 1098, 1996.

[10] Lov K Grover. A fast quantum mechanical algorithm for database search. In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, pp. 212–219, 1996.

[11] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. Contemporary Mathematics, Vol. 305, pp. 53–74, 2002.

[12] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[13] Stephen Jordan. Quantum algorithm zoo. https://quantumalgorithmzoo.org/.

[14] Ashley Montanaro. Quantum algorithms: an overview. npj Quantum Information, Vol. 2, No. 1, pp. 1–8, 2016.

[15] Yasunobu Nakamura, Yu A Pashkin, and JS Tsai. Coherent control of macroscopic quantum states in a single-cooper-pair box. nature, Vol. 398, No. 6730, pp. 786–788, 1999.

[16] IBM Quantum Experience. https://quantum-computing.ibm.com.

[17] Antonio D Córcoles, Abhinav Kandala, Ali Javadi-Abhari, Douglas T McClure, Andrew W Cross, Kristan Temme, Paul D Nation, Matthias Steffen, and JM Gambetta. Challenges and opportunities of near-term quantum computing systems. Proceedings of the IEEE, 2019.

[18] 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. Nature, Vol. 574, No. 7779, pp. 505–510, 2019.

[19] Robert S Smith, Michael J Curtis, and William J Zeng. A practical quantum instruction set architecture. arXiv preprint arXiv:1608.03355, 2016.

[20] Juan M Pino, Jennifer M Dreiling, Caroline Figgatt, John P Gaebler, Steven A Moses, MS Allman, CH Baldwin, M Foss-Feig, D Hayes, K Mayer, et al. Demonstration of the trapped-ion quantum ccd computer architecture. Nature, Vol. 592, No. 7853, pp. 209–213, 2021.

[21] Kenneth Wright, Kristin M Beck, Sea Debnath, JM Amini, Y Nam, N Grzesiak, J-S Chen, NC Pisenti, M Chmielewski, C Collins, et al. Benchmarking an 11-qubit quantum computer. Nature communications, Vol. 10, No. 1, pp. 1–6, 2019.

[22] John Preskill. Quantum computing in the nisq era and beyond. Quantum, Vol. 2, p. 79, 2018.

[23] Peter W Shor. Fault-tolerant quantum computation. In Proceedings of 37th Conference on Foundations of Computer Science, pp. 56–65. IEEE, 1996.

[24] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. SIAM Journal on Computing, 2008.

[25] Emanuel Knill, Raymond Laflamme, and Wojciech H Zurek. Resilient quantum computation. Science, Vol. 279, No. 5349, pp. 342–345, 1998.

[26] Aleksei Yur'evich Kitaev. Quantum computations: algorithms and error correction. Uspekhi Matematicheskikh Nauk, Vol. 52, No. 6, pp. 53–112, 1997.

[27] John Preskill. Reliable quantum computers. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, Vol. 454, No. 1969, pp. 385–410, 1998.

[28] John Preskill. Fault-tolerant quantum computation. In Introduction to quantum computation and information, pp. 213–269. World Scientific, 1998.

[29] David P DiVincenzo and Peter W Shor. Fault-tolerant error correction with efficient quantum codes. Physical review letters, Vol. 77, No. 15, p. 3260, 1996.

[30] Daniel Gottesman. Theory of fault-tolerant quantum computation. Physical Review A, Vol. 57, No. 1, p. 127, 1998.

[31] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. Physical review A, Vol. 52, No. 4, p. R2493, 1995.

[32] Andrew M Steane. Error correcting codes in quantum theory. Physical Review Letters, Vol. 77, No. 5, p. 793, 1996.

[33] Alexei Kitaev. Quantum measurements and the abelian stabilizer problem. arXiv preprint quant-ph/9511026, 1995.

[34] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. Physical review letters, Vol. 103, No. 15, p. 150502, 2009.

[35] Ashley Montanaro. Quantum speedup of Monte Carlo methods. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 471, No. 2181, p. 20150301, 2015.

[36] Patrick Rebentrost, Brajesh Gupt, and Thomas R Bromley. Quantum computational finance: Monte Carlo pricing of financial derivatives. Physical Review A, Vol. 98, No. 2, p. 022321, 2018.

[37] Stefan Woerner and Daniel J Egger. Quantum risk analysis. npj Quantum Information, Vol. 5, No. 1, pp. 1–8, 2019.

[38] Nikitas Stamatopoulos, Daniel J Egger, Yue Sun, Christa Zoufal, Raban Iten, Ning Shen, and Stefan Woerner. Option pricing using quantum computers. Quantum, Vol. 4, p. 291, 2020.

[39] Ana Martin, Bruno Candelas, Ángel Rodríguez-Rozas, José D Martín-Guerrero, Xi Chen, Lucas Lamata, Román Orús, Enrique Solano, and Mikel Sanz. Towards pricing financial derivatives with an ibm quantum computer. arXiv preprint arXiv:1904.05803, 2019.

[40] Daniel J Egger, Ricardo García Gutiérrez, Jordie Cahue Mestre, and Stefan Woerner. Credit risk analysis using quantum computers. IEEE Transactions on Computers, 2020.

[41] Koichi Miyamoto and Kenji Shiohara. Reduction of qubits in a quantum algorithm for monte carlo simulation by a pseudo-random-number generator. Physical Review A, Vol. 102, No. 2, p. 022424, 2020.

[42] Kazuya Kaneko, Koichi Miyamoto, Naoyuki Takeda, and Kazuyoshi Yoshino. Quantum speedup of monte carlo integration with respect to the number of dimensions and its application to finance. Quantum Information Processing, Vol. 20, No. 5, pp. 1–24, 2021.

[43] Shouvanik Chakrabarti, Rajiv Krishnakumar, Guglielmo Mazzola, Nikitas Stamatopoulos, Stefan Woerner, and William J Zeng. A threshold for quantum advantage in derivative pricing. Quantum, Vol. 5, p. 463, 2021.

[44] Steven Herbert. Quantum monte-carlo integration: The full advantage in minimal circuit depth. arXiv preprint arXiv:2105.09100, 2021.

[45] Koichi Miyamoto. Bermudan option pricing by quantum amplitude estimation and chebyshev interpolation. arXiv preprint arXiv:2108.09014, 2021.

[46] Koichi Miyamoto. Quantum algorithms for numerical differentiation of expected values with respect to parameters. arXiv preprint arXiv:2111.11016, 2021.

[47] Nikitas Stamatopoulos, Guglielmo Mazzola, Stefan Woerner, and William J Zeng. Towards quantum advantage in financial market risk using quantum gradient algorithms. arXiv preprint arXiv:2111.12509, 2021.

[48] Anupam Prakash. Quantum algorithms for linear algebra and machine learning. PhD thesis, UC Berkeley, 2014.

[49] Nathan Wiebe, Ashish Kapoor, and Krysta M. Svore. Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning. Quantum Inf. Comput., Vol. 15, pp. 316–356, 2015.

[50] Nathan Wiebe, Ashish Kapoor, and Krysta M. Svore. Quantum deep learning. Quantum Inf. Comput., Vol. 16, pp. 541–587, 2016.

[51] Nathan Wiebe, Ashish Kapoor, and Krysta M Svore. Quantum perceptron models. Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 4006–4014, 2016.

[52] Iordanis Kerenidis, Jonas Landman, Alessandro Luongo, and Anupam Prakash. q-means: A quantum algorithm for unsupervised machine learning. In Advances in Neural Information Processing Systems, pp. 4134–4144, 2019.

[53] Jing Liu, Haidong Yuan, Xiao-Ming Lu, and Xiaoguang Wang. Quantum Fisher information matrix and multiparameter estimation. Journal of Physics A: Mathematical and Theoretical, Vol. 53, No. 2, p. 023001, 2019.

[54] Hideyuki Miyahara, Kazuyuki Aihara, and Wolfgang Lechner. Quantum expectation-maximization algorithm. Physical Review A, Vol. 101, No. 1, p. 012326, 2020.

[55] Daniel S Abrams and Colin P Williams. Fast quantum algorithms for numerical integrals and stochastic processes. arXiv preprint quant-ph/9908083, 1999.

[56] Ilia Zintchenko and Nathan Wiebe. Randomized gap and amplitude estimation. Physical Review A, Vol. 93, No. 6, p. 062306, 2016.

[57] Yohichi Suzuki, Shumpei Uno, Rudy Raymond, Tomoki Tanaka, Tamiya Onodera, and Naoki Yamamoto. Amplitude estimation without phase estimation. Quantum Information Processing, Vol. 19, No. 2, p. 75, 2020.

[58] Scott Aaronson and Patrick Rall. Quantum approximate counting, simplified, in Symposium on Simplicity in Algorithms. pp. 24–32. SIAM, 2020.

[59] Dmitry Grinko, Julien Gacon, Christa Zoufal, and Stefan Woerner. Iterative quantum amplitude estimation. npj Quantum Information, Vol. 7, No. 1, pp. 1–6, 2021.

[60] Kouhei Nakaji. Faster amplitude estimation. Quantum Inf. Comput., Vol. 20, pp. 1109–1122, 2020.

[61] Ramgopal Venkateswaran and Ryan O'Donnell. Quantum approximate counting with nonadaptive grover iterations. arXiv preprint arXiv:2010.04370, 2020.

[62] Calyampudi Radhakrishna Rao. Linear statistical inference and its applications, Vol. 2. Wiley New York, 1973.

[63] Pedro J Salas. Noise effect on grover algorithm. The European Physical Journal D, Vol. 46, No. 2, pp. 365–373, 2008.

[64] Eric G Brown, Oktay Goktas, and Weng Kian Tham. Quantum amplitude estimation in the presence of noise. arXiv preprint arXiv:2006.14145, 2020.

[65] Tomoki Tanaka, Yohichi Suzuki, Shumpei Uno, Rudy Raymond, Tamiya Onodera, and Naoki Yamamoto. Amplitude estimation via maximum likelihood on noisy quantum computer. Quantum Information Processing, Vol. 20, No. 9, pp. 1–29, 2021.

[66] MD SAJID ANIS, Héctor Abraham, AduOffei, Rochisha Agarwal, Gabriele Agliardi, Merav Aharoni, Ismail Yunus Akhalwaya, Gadi Aleksandrowicz, Thomas Alexander, Matthew Amy, Sashwat Anagolum, Eli Arbel, Abraham Asfaw, Anish Athalye, Artur Avkhadiev, Carlos Azaustre, PRATHAMESH BHOLE, Abhik Banerjee, Santanu Banerjee, Will Bang, Aman Bansal, Panagiotis Barkoutsos, Ashish Barnawal, George Barron, George S. Barron, Luciano Bello, Yael Ben-Haim, M. Chandler Bennett, Daniel Bevenius, Dhruv Bhatnagar, Arjun Bhobe, Paolo Bianchini, Lev S. Bishop, Carsten Blank, Sorin Bolos, Soham Bopardikar, Samuel Bosch, Sebastian Brandhofer, Brandon, Sergey Bravyi, Nick Bronn, Bryce-Fuller, David Bucher, Artemiy Burov, Fran Cabrera, Padraic Calpin, Lauren Capelluto, Jorge Carballo, Ginés Carrascal, Adam Carriker, Ivan Carvalho, Adrian Chen, Chun-Fu Chen, Edward Chen, Jielun (Chris) Chen, Richard Chen, Franck Chevallier, Kartik Chinda, Rathish Cholarajan, Jerry M. Chow, Spencer Churchill, CisterMoke, Christian Claus, Christian Clauss, Caleb Clothier, Romilly Cocking, Ryan Cocuzzo, Jordan Connor, Filipe Correa, Abigail J. Cross, Andrew W. Cross, Simon Cross, Juan Cruz-Benito, Chris Culver, Antonio D. Córcoles-Gonzales, Navaneeth D, Sean Dague, Tareq El Dandachi, Animesh N Dangwal, Jonathan Daniel, Marcus Daniels, Matthieu Dartiailh, Abdón Rodríguez Davila, Faisal Debouni, Anton Dekusar, Amol Deshmukh, Mohit Deshpande, Delton Ding, Jun Doi, Eli M. Dow, Eric Drechsler, Eugene Dumitrescu, Karel Dumon, Ivan Duran, Kareem EL-Safty, Eric Eastman, Grant Eberle, Amir Ebrahimi, Pieter Eendebak, Daniel Egger, ElePT, Emilio, Alberto Espiricueta, Mark Everitt, Davide Facoetti, Farida, Paco Martín Fernández, Samuele Ferracin, Davide Ferrari,

Axel Hernández Ferrera, Romain Fouilland, Albert Frisch, Andreas Fuhrer, Bryce Fuller, MELVIN GEORGE, Julien Gacon, Borja Godoy Gago, Claudio Gambella, Jay M. Gambetta, Adhisha Gammanpila, Luis Garcia, Tanya Garg, Shelly Garion, Jim Garrison, Tim Gates, Leron Gil, Austin Gilliam, Aditya Giridharan, Juan Gomez-Mosquera, Gonzalo, Salvador de la Puente González, Jesse Gorzinski, Ian Gould, Donny Greenberg, Dmitry Grinko, Wen Guan, John A. Gunnels, Harshit Gupta, Naman Gupta, Jakob M. Günther, Mikael Haglund, Isabel Haide, Ikko Hamamura, Omar Costa Hamido, Frank Harkins, Kevin Hartman, Areeq Hasan, Vojtech Havlicek, Joe Hellmers, Łukasz Herok, Stefan Hillmich, Hiroshi Horii, Connor Howington, Shaohan Hu, Wei Hu, Junye Huang, Rolf Huisman, Haruki Imai, Takashi Imamichi, Kazuaki Ishizaki, Ishwor, Raban Iten, Toshinari Itoko, Alexander Ivrii, Ali Javadi, Ali Javadi-Abhari, Wahaj Javed, Qian Jianhua, Madhav Jivrajani, Kiran Johns, Scott Johnstun, Jonathan-Shoemaker, JosDenmark, JoshDumo, John Judge, Tal Kachmann, Akshay Kale, Naoki Kanazawa, Jessica Kane, Kang-Bae, Annanay Kapila, Anton Karazeev, Paul Kassebaum, Josh Kelso, Scott Kelso, Vismai Khanderao, Spencer King, Yuri Kobayashi, Kovi11Day, Arseny Kovyrshin, Rajiv Krishnakumar, Vivek Krishnan, Kevin Krsulich, Prasad Kumkar, Gawel Kus, Ryan LaRose, Enrique Lacal, Raphaël Lambert, Haggai Landa, John Lapeyre, Joe Latone, Scott Lawrence, Christina Lee, Gushu Li, Jake Lishman, Dennis Liu, Peng Liu, Yunho Maeng, Saurav Maheshkar, Kahan Majmudar, Aleksei Malyshev, Mohamed El Mandouh, Joshua Manela, Manjula, Jakub Marecek, Manoel Marques, Kunal Marwaha, Dmitri Maslov, Paweł Maszota, Dolph Mathews, Atsushi Matsuo, Farai Mazhandu, Doug McClure, Maureen McElaney, Cameron McGarry, David McKay, Dan McPherson, Srujan Meesala, Dekel Meirom, Corey Mendell, Thomas Metcalfe, Martin Mevissen, Andrew Meyer, Antonio Mezzacapo, Rohit Midha, Daniel Miller, Zlatko Minev, Abby Mitchell, Nikolaj Moll, Alejandro Montanez, Gabriel Monteiro, Michael Duane Mooring, Renier Morales, Niall Moran, David Morcuende, Seif Mostafa, Mario Motta, Romain Moyard, Prakash Murali, Jan Müggenburg, Tristan NEMOZ, David Nadlinger, Ken Nakanishi, Giacomo Nannicini, Paul Nation, Edwin Navarro, Yehuda Naveh, Scott Wyman Neagle, Patrick Neuweiler, Aziz Ngoueya, Johan Nicander, Nick-Singstock, Pradeep Niroula, Hassi Norlen, NuoWenLei, Lee James O'Riordan, Oluwatobi Ogunbayo, Pauline Ollitrault, Tamiya Onodera, Raul Otaolea, Steven Oud, Dan Padilha, Hanhee Paik, Soham Pal, Yuchen Pang, Ashish Panigrahi, Vincent R. Pascuzzi, Simone Perriello, Eric Peterson, Anna Phan, Francesco Piro, Marco Pistoia, Christophe Piveteau, Julia Plewa, Pierre Pocreau, Alejandro Pozas-Kerstjens, Rafał Pracht, Milos Prokop, Viktor Prutyanov, Sumit Puri, Daniel Puzzuoli, Jesús Pérez, Quant02, Quintiii, Isha R, Rafey Iqbal Rahman, Arun Raja, Roshan Rajeev, Nipun Ramagiri, Anirudh Rao, Rudy Raymond, Oliver Reardon-Smith, Rafael Martín-Cuevas Redondo, Max Reuter, Julia Rice, Matt Riedemann, Rietesh, Drew Risinger, Marcello La Rocca, Diego M. Rodríguez, RohithKarur, Ben Rosand, Max Rossmannek, Mingi Ryu, Tharrmashastha SAPV, Nahum Rosa Cruz Sa, Arijit Saha, Abdullah Ash-Saki, Sankalp Sanand, Martin Sandberg, Hirmay Sandesara, Ritvik Sapra, Hayk Sargsyan, Aniruddha Sarkar, Ninad Sathaye, Bruno Schmitt, Chris Schnabel, Zachary Schoenfeld, Travis L. Scholten, Eddie Schoute, Mark Schulterbrandt, Joachim Schwarm, James Seaward, Sergi, Ismael Faro Sertage, Kanav Setia, Freya Shah, Nathan Shammah, Rohan Sharma, Yunong Shi, Jonathan Shoemaker, Adenilton Silva, Andrea Simonetto, Divyanshu Singh, Parmeet Singh, Phattharaporn Singkanipa, Yukio Siraichi, Siri, Jesús Sistos, Iskandar Sitdikov, Seyon Sivarajah, Magnus Berg Sletfjerding, John A. Smolin, Mathias Soeken, Igor Olegovich Sokolov, Igor

Sokolov, Vicente P. Soloviev, SooluThomas, Starfish, Dominik Steenken, Matt Stypulkoski, Adrien Suau, Shaojun Sun, Kevin J. Sung, Makoto Suwama, Oskar Słowik, Hitomi Takahashi, Tanvesh Takawale, Ivano Tavernelli, Charles Taylor, Pete Taylour, Soolu Thomas, Kevin Tian, Mathieu Tillet, Maddy Tod, Miroslav Tomasik, Caroline Tornow, Enrique de la Torre, Juan Luis Sánchez Toural, Kenso Trabing, Matthew Treinish, Dimitar Trenev, TrishaPe, Felix Truger, Georgios Tsilimigkounakis, Davindra Tulsi, Wes Turner, Yotam Vaknin, Carmen Recio Valcarce, Francois Varchon, Adish Vartak, Almudena Carrera Vazquez, Prajjwal Vijaywargiya, Victor Villar, Bhargav Vishnu, Desiree Vogt-Lee, Christophe Vuillot, James Weaver, Johannes Weidenfeller, Rafal Wieczorek, Jonathan A. Wildstrom, Jessica Wilson, Erick Winston, WinterSoldier, Jack J. Woehr, Stefan Woerner, Ryan Woo, Christopher J. Wood, Ryan Wood, Steve Wood, James Wootton, Matt Wright, Lucy Xing, Jintao YU, Bo Yang, Daniyar Yeralin, Ryota Yonekura, David Yonge-Mallo, Ryuhei Yoshida, Richard Young, Jessie Yu, Lebin Yu, Christopher Zachow, Laura Zdanski, Helena Zhang, Christa Zoufal, aeddins ibm, alexzhang13, b63, bartek bartlomiej, bcamorrison, brandhsn, charmerDark, deeplokhande, dekel.meirom, dime10, dlasecki, ehchen, fanizzamarco, fs1132429, gadial, galeinston, georgezhou20, georgios ts, gruu, hhorii, hykavitha, itoko, jessica angel7, jezerjojo14, jliu45, jscott2, klinvill, krutik2966, ma5x, michelle4654, msuwama, ntgiwsvp, ordmoj, sagar pahwa, pritamsinha2304, ryancocuzzo, saswati qiskit, septembrr, sethmerkel, shaashwat, sternparky, strickroman, tigerjack, tsura crisaldo, vadebayo49, welien, willhbang, wmurphy collabstar, yang.luh, and Mantas Čepulkovskis. Qiskit: An open-source framework for quantum computing, 2021.

[67] Martin Koppenhöfer, Christoph Bruder, and Alexandre Roulet. Quantum synchronization on the ibm q system. Physical Review Research, Vol. 2, No. 2, p. 023026, 2020.

[68] Easwar Magesan. Depolarizing behavior of quantum channels in higher dimensions. arXiv preprint arXiv:1002.3455, 2010.

[69] Kyungjoo Noh, Liang Jiang, and Bill Fefferman. Efficient classical simulation of noisy random quantum circuits in one dimension. Quantum, Vol. 4, p. 318, 2020.

[70] Emanuel Knill, Dietrich Leibfried, Rolf Reichle, Joe Britton, R Brad Blakestad, John D Jost, Chris Langer, Roee Ozeri, Signe Seidelin, and David J Wineland. Randomized benchmarking of quantum gates. Physical Review A, Vol. 77, No. 1, p. 012307, 2008.

[71] Easwar Magesan, Jay M Gambetta, and Joseph Emerson. Scalable and robust randomized benchmarking of quantum processes. Physical review letters, Vol. 106, No. 18, p. 180504, 2011.

[72] David C McKay, Sarah Sheldon, John A Smolin, Jerry M Chow, and Jay M Gambetta. Three-qubit randomized benchmarking. Physical review letters, Vol. 122, No. 20, p. 200502, 2019.

[73] Tudor Giurgica-Tiron, Iordanis Kerenidis, Farrokh Labib, Anupam Prakash, and William Zeng. Low depth algorithms for quantum amplitude estimation. arXiv preprint arXiv:2012.03348, 2020.

[74] Tudor Giurgica-Tiron, Sonika Johri, Iordanis Kerenidis, Jason Nguyen, Neal Pisenti, Anupam Prakash, Ksenia Sosnova, Ken Wright, and William Zeng. Low depth amplitude estimation on a trapped ion quantum computer. arXiv preprint arXiv:2109.09685, 2021.

[75] Shumpei Uno, Yohichi Suzuki, Keigo Hisanaga, Rudy Raymond, Tomoki Tanaka, Tamiya Onodera, and Naoki Yamamoto. Modified grover operator for quantum amplitude estimation. New Journal of Physics, Vol. 23, No. 8, p. 083031, 2021.

[76] Tomoki Tanaka, Shumpei Uno, Tamiya Onodera, Naoki Yamamoto, and Yohichi Suzuki. Noisy quantum amplitude estimation without noise estimation. Phys. Rev. A, Vol. 105, p. 012411, 2022.

[77] David Roxbee Cox and Nancy Reid. Parameter orthogonality and approximate conditional inference. Journal of the Royal Statistical Society: Series B (Methodological), Vol. 49, No. 1, pp. 1–18, 1987.

[78] Jun Suzuki, Yuxiang Yang, and Masahito Hayashi. Quantum state estimation with nuisance parameters. Journal of Physics A: Mathematical and Theoretical, Vol. 53, No. 45, p. 453001, 2020.

[79] Janet M Begun, WJ Hall, Wei-Min Huang, and Jon A Wellner. Information and asymptotic efficiency in parametric-nonparametric models. The Annals of Statistics, Vol. 11, No. 2, pp. 432–452, 1983.

[80] Jun Suzuki. Nuisance parameter problem in quantum estimation theory: tradeoff relation and qubit examples. Journal of Physics A: Mathematical and Theoretical, Vol. 53, No. 26, p. 264001, 2020.

[81] Mankei Tsang, Francesco Albarelli, and Animesh Datta. Quantum semiparametric estimation. Physical Review X, Vol. 10, No. 3, p. 031023, 2020.

[82] Valeria Cimini, Francesco Albarelli, Ilaria Gianani, and Marco Barbieri. Semiparametric estimation in hong-ou-mandel interferometry. arXiv preprint arXiv:2109.09368, 2021.

[83] David C McKay, Christopher J Wood, Sarah Sheldon, Jerry M Chow, and Jay M Gambetta. Efficient z gates for quantum computing. Physical Review A, Vol. 96, No. 2, p. 022330, 2017.

[84] Emanuel Knill, R Laflamme, R Martinez, and C-H Tseng. An algorithmic benchmark for quantum information processing. Nature, Vol. 404, No. 6776, pp. 368–370, 2000.

[85] Tomoki Tanaka, Shumpei Uno, Yohichi Suzuki, and Rudy Raymond. https://github.com/Qiskit/qiskit-community-tutorials/blob/master/algorithms/SimpleIntegral_AEwoPE.ipynb, 2019. Accessed: 2021-10-01.

[86] https://qiskit.org/documentation/finance/tutorials/00_amplitude_estimation.html. Accessed: 2021-12-25.

[87] Almudena Carrera Vazquez and Stefan Woerner. Efficient state preparation for quantum amplitude estimation. Physical Review Applied, Vol. 15, No. 3, p. 034027, 2021.

[88] Zhengfeng Ji, Guoming Wang, Runyao Duan, Yuan Feng, and Mingsheng Ying. Parameter estimation of quantum channels. IEEE Transactions on Information Theory, Vol. 54, No. 11, pp. 5172–5185, 2008.

[89] Bruno de Moura Escher, Ruynet Lima de Matos Filho, and Luiz Davidovich. General framework for estimating the ultimate precision limit in noisy quantum-enhanced metrology. Nature Physics, Vol. 7, No. 5, pp. 406–411, 2011.

[90] Haidong Yuan and Chi-Hang Fred Fung. Quantum parameter estimation with general dynamics. npj Quantum Information, Vol. 3, No. 1, pp. 1–6, 2017.

[91] Ken M Nakanishi, Takahiko Satoh, and Synge Todo. Quantum-gate decomposer. arXiv preprint arXiv:2109.13223, 2021.

[92] Zelda B Zabinsky. Stochastic adaptive search for global optimization, Vol. 72. Springer Science & Business Media, 2003.

[93] Jorge Nocedal and Stephen Wright. Numerical optimization. Springer Science & Business Media, 2006.

[94] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. Journal of the society for Industrial and Applied Mathematics, Vol. 11, No. 2, pp. 431–441, 1963.

[95] RD Gill and S Massar. State estimation for large ensembles. Physical review. A, Atomic, molecular

and optical physics, Vol. 61, pp. 2312–2327, 2000.

[96] Akio Fujiwara. Quantum channel identification problem. Physical Review A, Vol. 63, No. 4, p. 042304, 2001.

[97] Masahito Hayashi and Keiji Matsumoto. Statistical model with measurement degree of freedom and quantum physics. In Asymptotic theory of quantum statistical inference: selected papers, pp. 162–169. World Scientific, 2005.

[98] Carl Helstrom. The minimum variance of estimates in quantum signal detection. IEEE Transactions on information theory, Vol. 14, No. 2, pp. 234–242, 1968.

[99] Samuel L Braunstein and Carlton M Caves. Statistical distance and the geometry of quantum states. Physical Review Letters, Vol. 72, No. 22, p. 3439, 1994.

[100] Alexander S Holevo. Probabilistic and statistical aspects of quantum theory, Vol. 1. Springer Science & Business Media, 2011.

[101] Hiroshi Nagaoka. On Fisher information of quantum statistical models. In Asymptotic Theory Of Quantum Statistical Inference: Selected Papers, pp. 113–124. World Scientific, 2005.

[102] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum-enhanced measurements: beating the standard quantum limit. Science, Vol. 306, No. 5700, pp. 1330–1336, 2004.

[103] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum metrology. Physical review letters, Vol. 96, No. 1, p. 010401, 2006.

[104] Jan Kołodyński and Rafał Demkowicz-Dobrzański. Phase estimation without a priori phase knowledge in the presence of loss. Physical Review A, Vol. 82, No. 5, p. 053804, 2010.

[105] Sergey Knysh, Vadim N Smelyanskiy, and Gabriel A Durkin. Scaling laws for precision in quantum interferometry and the bifurcation landscape of the optimal state. Physical Review A, Vol. 83, No. 2, p. 021804, 2011.

[106] Zhang Jiang. Quantum Fisher information for states in exponential form. Physical Review A, Vol. 89, No. 3, p. 032128, 2014.

[107] Yao Yao, Li Ge, Xing Xiao, Xiaoguang Wang, and CP Sun. Multiple phase estimation for arbitrary pure states under white noise. Physical Review A, Vol. 90, No. 6, p. 062113, 2014.

[108] Lorenzo Maccone. Intuitive reason for the usefulness of entanglement in quantum metrology. Physical Review A, Vol. 88, No. 4, p. 042109, 2013.

[109] J. M. Chow, L. DiCarlo, J. M. Gambetta, A. Nunnenkamp, Lev S. Bishop, L. Frunzio, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf. Detecting highly entangled states with a joint qubit readout. Phys. Rev. A, Vol. 81, p. 062325, Jun 2010.

[110] Yanzhu Chen, Maziar Farahzad, Shinjae Yoo, and Tzu-Chieh Wei. Detector tomography on ibm quantum computers and mitigation of an imperfect measurement. Physical Review A, Vol. 100, No. 5, p. 052315, 2019.

[111] Sergey Bravyi, Sarah Sheldon, Abhinav Kandala, David C Mckay, and Jay M Gambetta. Mitigating measurement errors in multiqubit experiments. Physical Review A, Vol. 103, No. 4, p. 042605, 2021.

[112] Bo Yang, Rudy Raymond, and Shumpei Uno. An efficient quantum readout error mitigation for sparse measurement outcomes of near-term quantum devices. arXiv preprint arXiv:2201.11046, 2022.

[113] Yudai Suzuki, Hiroshi Yano, Qi Gao, Shumpei Uno, Tomoki Tanaka, Manato Akiyama, and Naoki Yamamoto. Analysis and synthesis of feature map for kernel-based quantum classifier. Quantum

Machine Intelligence, Vol. 2, No. 1, pp. 1–9, 2020.

[114] Kouhei Nakaji, Shumpei Uno, Yohichi Suzuki, Rudy Raymond, Tamiya Onodera, Tomoki Tanaka, Hiroyuki Tezuka, Naoki Mitsuda, and Naoki Yamamoto. Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicator. arXiv preprint arXiv:2103.13211, 2021.

[115] Kento Oonishi, Tomoki Tanaka, Shumpei Uno, Takahiko Satoh, Rodney Van Meter, and Noboru Kunihiro. Efficient construction of a control modular adder on a carry-lookahead adder using relative-phase toffoli gates. IEEE Transactions on Quantum Engineering, Vol. 3, pp. 1–18, 2022.

[116] Yoshinori Aono, Sitong Liu, Tomoki Tanaka, Shumpei Uno, Rodney Van Meter, Naoyuki Shinohara, and Ryo Nojima. The present and future of discrete logarithm problems on noisy quantum computers. arXiv preprint arXiv:2111.06102, 2021.