

Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicator

中路 紘平 (Kouhei Nakaji)^{1,a)}

概要: 近年, ノイジーな中規模量子コンピューター (NISQ) の開発が目覚しく発展している. このような状況の下, いかにノイズの影響を減らしつつ, 意味のある計算が行えるか? というところに大きな注目が集まっている. 量子状態の振幅へのデータロード回路は, 多くの量子アルゴリズムで用いられている一方, 非常に多くの演算を必要とするため, アルゴリズムの実行プロセスの中で最も深刻なノイズのソースとなり得る. 本稿では, 量子変分アルゴリズムを用いて, ノイズの影響を受けにくく, かつ少ない演算処理でデータロードが可能なデータロード回路を構築する手法について紹介する.

1. 研究の背景

1.1 量子コンピューター研究の背景

ゲート方式の量子コンピューター (以下, 量子コンピューター) は, 従来の計算機 (以下, 古典コンピューター) が有限時間で解くことのできない問題を極めて効率的に解くことができる計算機として世界中で期待を集めている. このような期待のもと, これまで量子コンピューターを用いた画期的な計算手法 (=量子アルゴリズム) が多数提案されてきた. そのような量子アルゴリズムの例としては, 素因数分解を行う Shor のアルゴリズム [1], 検索を行う Grover のアルゴリズム [2], さらに線形方程式を解く HHL アルゴリズム [3] などが挙げられる. これらアルゴリズムが実現されたあかつきには, 化学, 金融, AI を始めとして, 実社会に幅広いインパクトを与えることは確実である.

以上のアルゴリズムは, 誤り耐性機能付きの理想的な量子コンピューターを想定して開発されたことに注意が必要である. 誤り耐性とは計算結果に対するノイズの影響を修正する機能のことを指す. ノイズ修正機能があることによって, たとえ計算過程でノイズが入り込んだとしても, 最終的には信頼できる計算結果を得られるわけである.

しかしながら, 近年, **誤り耐性機能を持たず**, ノイジーな中規模量子コンピューターに注目が集まっている. というのも, ここ数年で量子コンピューターの実機性能が飛躍的に向

上し, 数十量子ビットの誤り耐性機能無し量子コンピューターが利用可能になってきたためである. このような, ノイジーな誤り耐性機能無し中規模量子コンピューターは Noisy Intermediate Scale Quantum (NISQ) [4] と呼ばれている. NISQ においては, 演算回数 (=量子ゲートの数) が増えれば増えるほど演算結果へのノイズの影響が強くなる. 結果, 膨大な演算が必要な, 従来提案されてきたアルゴリズムの多く (Shor のアルゴリズム等) は, ノイズに有用な結果を塗り潰されてしまうため, 軒並み NISQ では実行不可能である. 残念なことに, 今後数十年開発される実機は, 引き続きノイジーな NISQ であることが予想されている. つまり, 量子コンピューターを今後数十年で実用化するには, NISQ におけるノイズの問題をなんとか克服しなければならない.

以上のような状況の下, NISQ におけるノイズの問題を克服すべく世界中で盛んに研究が行われている. その中で重要な方向性としては, **いかに量子ゲートの数を減らしつつ, 意味ある演算を行えるか?** ということである. 例えば, 近年の大きな研究トピックである変分量子アルゴリズム (参考:[5]) は, 浅い (量子ゲート数が少ない) 量子回路を使うことでノイズの影響を小さくしつつも, 古典コンピューターを巧みに使うことで意味のある結果を導き出すことのできる有望な手法である. なお, 量子ゲートの数を減らすことは, 誤り耐性機能付きの量子コンピューターにおいても演算の高速化のために当然重要であるから, 浅い量子回路を活用する手法は, NISQ 向けの手法でありながらも, 理想的な量子コンピューターへの適用が期待できる.

¹ 慶應義塾大学 理工学研究科
〒 223-8522 神奈川県横浜市港北区日吉 3 丁目 1 4 - 1

^{a)} kohei.nakaji@keio.jp

1.2 データロードの問題

さて、多くの量子アルゴリズムは、古典データを量子状態の振幅に埋め込む「データロード回路」を必要とする。実はこのデータロード回路を厳密に実現しようとする、量子ビット数の指数乗個のゲートが必要であり、NISQ ではノイズが多すぎて実行が難しく、理想的な量子コンピューターにおいては計算処理の最大のボトルネックになり得る。

例えば、上述した HHL アルゴリズムは、 $A\mathbf{x} = \mathbf{b}$ という線形方程式を解くアルゴリズムであり、データロードを除けば量子ビット数 n の多項式回程度の演算で実行可能である (\mathbf{x} の読み出しが必要ない場合)。しかしながら、アルゴリズム実行に際しては、 $O(2^n)$ 個の量子ゲートを用いて下記の状態

$$|\mathbf{b}\rangle = \sum_{j=0}^{2^n-1} b_j |j\rangle \quad (1)$$

を読み込む必要があり、結果としてデータロードを含めると、古典アルゴリズムに対する優位性が失われてしまう。ここで b_j は \mathbf{b} の j 番目の成分であり ($\sum_j b_j^2 = 1$ と規格化しておく)、 $|j\rangle$ は 2 進数のビット列で表された状態を 10 進数に変換して記述したものである。

以上のデータロードの問題、すなわち古典データを量子状態の振幅にロードする際に膨大な $O(2^n)$ 個のゲートが要求されてしまう問題を解決するのが、本稿で紹介する我々の研究 [6] の目的である。我々は [6] で、データロードを行う量子回路を変分量子アルゴリズムを使って構築する画期的な手法: **Approximate Amplitude Encoding(AAE)** を提案した。我々の提案手法を使うと、データロードに必要な量子ゲートの数を著しく減らすことができる。以下では、その提案手法について紹介する。

2. 提案手法

2.1 提案手法のゴール

本提案手法のゴールは、 $\sum_j d_j^2 = 1$ に規格化された実 N 次元ベクトル $\mathbf{d} = \{d_0, d_1, \dots, d_{N-1}\}$ で表される古典データを、少ない量子ゲートで量子状態の振幅に埋め込むことである。言い換えれば、 $n \sim \log_2 N$ として、 n 量子ビットの状態

$$|\mathbf{Data}\rangle = \sum_{j=0}^{N-1} d_j |j\rangle, \quad (2)$$

を効率的にロードできる量子回路の構築が、本研究のゴールである。

上述の通り、(2) を厳密に実現するような量子回路を構築しようすると、 $O(2^n)$ 個の量子ゲートが必要であり、多くの量子アルゴリズムの古典アルゴリズムに対する優位性を無くしてしまうだけでなく、NISQ での実行を不可能にしてしまう。このような問題を解決すべく、変分量子アルゴリズムを用いて $|\mathbf{Data}\rangle$ を近似的にロードするパラメトライズ

された量子回路 $U(\theta)$ を鍛えるのが提案手法の肝である。ここで、 θ は複数のパラメータ $\theta = \{\theta_1, \theta_2, \dots, \theta_R\}$ を表す。通常の変分量子アルゴリズムと同様、パラメータ θ は後述するコスト関数を最小化するようにアップデートしていく。図 1 に示すように、 $U(\theta)$ としてはパラメータが埋め込まれた回転ゲートの層と 2 量子ビットを結合するゲートの層が交互に繰り返す Hardware Efficient Ansatz を用いる。但し、 $U(\theta)$ を作用させた後の状態を計算基底で展開した際の振幅が実数になるように、各回転ゲートとしては R_y ゲート、2 量子ビットを結合するゲートとしては CNOT ゲートを用いる。

以上の $U(\theta)$ を用いて、達成したいゴールを二通りの条件として書き下す (いずれかの条件が満たされていれば良い)。一つ目の条件は、変分量子アルゴリズムの結果として、 n 量子ビットに作用する $U(\theta)$ が

$$U(\theta_*)|0\rangle^{\otimes n} = e^{i\alpha} |\mathbf{Data}\rangle \quad (3)$$

を近似的に満たす、というものである。ここで、 θ_* は変分量子アルゴリズムの結果として得られたパラメータ、 α は何らかの位相の値である。以上を満たしていれば、 $U(\theta)$ は明らかに望むデータロード回路になっている。二つ目の条件は、 V を浅い回路と測定処理に対応する演算子、 $|y\rangle$ を補助ビットの状態として、 $n+1$ 量子ビットに作用する $U(\theta)$ が

$$VU(\theta_*)|0\rangle^{\otimes n}|0\rangle = e^{i\alpha} |\mathbf{Data}\rangle|y\rangle, \quad (4)$$

を近似的に満たす、というものである。式 (4) が満たされている場合、右辺の始めの n -qubit が、まさにロードしたい状態になっており、補助量子ビットを無視すれば、後段の計算処理の初期状態として利用できるから、こちらの場合も、同様に望むデータロード回路が得られたことになる。

通常、回転ゲートの層と 2 量子ビットを結合するゲートの層をセットにして 1 層と数えるが、(3) または (4) を満たす $U(\theta)$ を ℓ 層の回路で実現できたとすれば、 $O(\ell n)$ のゲートで近似的にデータロードを行う量子回路が得られたことになる。したがって、 ℓ を $O(1) \sim O(\text{poly}(n))$ と設定する限りは、厳密に状態をロードするよりもはるかに少ない量子ゲート数で、データロードを行えるわけである。

では、どのようにして (3) または (4) を満たす $U(\theta_*)$ を得ればよいだろうか？ というのが次のクエスチョンになるが、その答えに移る前に、本提案手法で用いる変分量子アルゴリズムについて簡単に振り返る。

2.2 変分量子アルゴリズム

変分量子アルゴリズムでは、パラメトライズされた量子回路 $U(\theta)|0\rangle^{\otimes n}$ を用いて、何らかのコスト関数

$$C(U(\theta)|0\rangle^{\otimes n}) \quad (5)$$

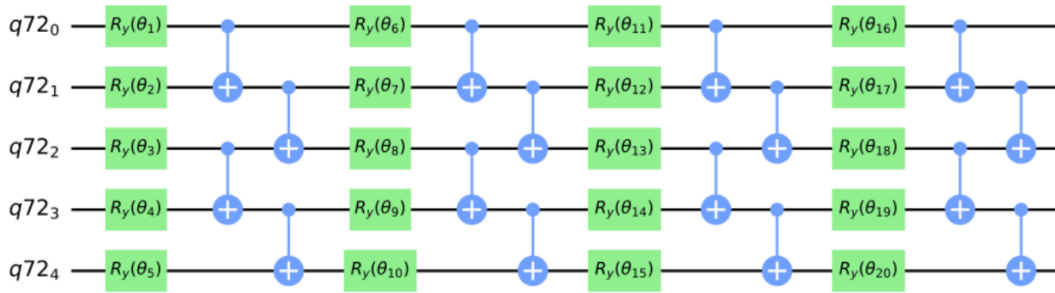


図 1 hardware Efficient Ansatz の例 (4 層). 回転ゲートにパラメータを埋め込んでいる.

を最小化するようにパラメータ θ を最適化していく^{*1}. コスト関数としては, $U(\theta)|0\rangle^{\otimes n}$ が満たすべき条件から設計する. 例えば, 化学ハミルトニアン¹の最小固有状態を求める変分量子固有値ソルバーでは, コスト関数として化学のハミルトニアンが用いられる. パラメータ更新の方法としては, しばしば最急降下法と呼ばれる手法が用いられ, 各パラメータ θ_r を

$$\theta_r \rightarrow \theta_r - \eta \frac{\partial C}{\partial \theta_r} \quad (r = 1, 2, \dots, R) \quad (6)$$

のように更新していく. ここで, η は学習率であり, 勝手に設定可能なパラメータである. 具体的な計算プロセスとしては, まず θ を初期化し, (a) 量子回路によって状態 $U(\theta)|0\rangle^{\otimes n}$ を構築した後, 各勾配 $\partial C / \partial \theta_r$ を推定する. (b) 次に, (6) にしたがって, パラメータを更新する. さらに (a), (b) を繰り返して θ を更新し, コスト関数を最小化する θ を得る.

2.3 提案手法

以上を踏まえ, 我々の提案手法について述べていく. 提案手法においては, 以上の最適化の結果 (3) または (4) が実現されるようなコスト関数 $C(U(\theta)|0\rangle^{\otimes n})$ を用いて, 上述した変分量子アルゴリズムを実行し, 最適な $U(\theta_*)$ を求める. では, どのようなコスト関数を選べば良いか? というのが提案手法の肝である.

最適化の結果 (3) を満たすというだけであれば, ロードしたい状態と量子回路によって作られる状態の忠実度を 1 に近づけるようなコスト関数:

$$C(U(\theta)|0\rangle^{\otimes n}) = 1 - |\langle \text{Data} | U(\theta)|0\rangle^{\otimes n}|^2 \quad (7)$$

にとれば良いと思われるかもしれない. しかしながら, コスト関数をこのように選ぶことはできない. というのも, 右辺およびその勾配を計算するために, 毎回状態 $|\text{Data}\rangle$ を構築せねばならないためである (もし $|\text{Data}\rangle$ を簡単に構築できるのであれば, そもそも量子変分アルゴリズムを使う必要がない!). この例からもわかる通り, 最小化すれば (3) または (4) が満たされるようなコスト関数を選ぶだけでは不

分で, 少ない計算量で評価が可能なものを持つてくる必要がある.

実は, (3) または (4) と等価な別の条件をコスト関数で表現すれば, 学習後に (3) または (4) が満たされ, かつ少ない計算量で評価が可能なコスト関数を得ることができる. そのような条件は, ロードしたいデータに依って二つのケースに分かれる. ケース 1 は, d_j の符号が全て同じ (0 も含んでもよい) 場合であり, ケース 2 はそれ以外である. 例えば 2 量子ビットの場合で, $\{d_0 = 0.8, d_1 = 0, d_2 = 0, d_3 = 0.6\}$ や $\{d_0 = -0.8, d_1 = 0, d_2 = 0, d_3 = -0.6\}$ はケース 1 に該当し, $\{d_0 = 0.8, d_1 = 0, d_2 = 0, d_3 = -0.6\}$ はケース 2 に該当する. なお, ケース 1 に関しては, (3) と等価な条件, ケース 2 に関しては (4) と等価な条件を導くことができる.

式 (3) または式 (4) と等価な条件

ケース 1 の場合, H をアダマール演算子, $d_j^H = \sum_{k=0}^{N-1} d_k \langle j | H^{\otimes n} | k \rangle$ とすると, n 量子ビットに作用するオペレータ $U(\theta)$ に関する以下の条件

$$|\langle j | U(\theta) | 0 \rangle^{\otimes n}|^2 = d_j^2 \quad (8)$$

$$|\langle j | H^{\otimes n} U(\theta) | 0 \rangle^{\otimes n}|^2 = (d_j^H)^2 \quad (9)$$

は, (3) と等価である. その理由は, 下記定理が成立するためである.

定理 1. ケース 1 において, もし $U(\theta)$ が式 (8) と式 (9) を満たせば, $U(\theta)|0\rangle^{\otimes n} = \sum_j d_j |j\rangle$ or $U(\theta)|0\rangle^{\otimes n} = -\sum_j d_j |j\rangle$ が成り立つ.

証明. まず, \mathbf{a} を, その要素が $\mathbf{a}_j = \langle j | U(\theta) | 0 \rangle$ となる N 次元ベクトルとして定義する. すると, (8) と (9) は下記のように書き直すことができる:

$$\mathbf{a}_j^2 = \mathbf{d}_j^2 \quad (\forall j) \quad (10)$$

$$\left(\sum_{k=0}^{N-1} H_{jk}^{\otimes n} \mathbf{a}_k \right)^2 = \left(\sum_{k=0}^{N-1} H_{jk}^{\otimes n} \mathbf{d}_k \right)^2 \quad (\forall j), \quad (11)$$

ここで, $H_{jk}^{\otimes n} \equiv \langle j | H^{\otimes n} | k \rangle$ と定義した. (11) の左辺で $j = 0$ とおくと

^{*1} 量子機械学習の場合のコストはやや異なるが, 今回の話題とは直接関係しないため, 本稿では立ち入らない.

$$\left(\sum_{k=0}^{N-1} H_{0k}^{\otimes n} a_k\right)^2 = \frac{1}{2^n} \left(\sum_{k=0}^{N-1} a_k\right)^2 \leq \frac{1}{2^n} \left(\sum_{k=0}^{N-1} |a_k|\right)^2 \quad (12)$$

となる. 等号成立条件は $a_k \geq 0$ ($\forall k$) または $a_k \leq 0$ ($\forall k$) である. この等号成立条件は $\mathbf{a} = \mathbf{d}$ または $\mathbf{a} = -\mathbf{d}$ と等価であることが (10) とケース 1 の条件: $\mathbf{d}_j \geq 0$ ($\forall j$) or $\mathbf{d}_j \leq 0$ ($\forall j$) から言える. 逆に言えば, もし等号成立条件を満たさない場合は,

$$\begin{aligned} \frac{1}{2^n} \left(\sum_{k=0}^{N-1} a_k\right)^2 &< \frac{1}{2^n} \left(\sum_{k=0}^{N-1} |a_k|\right)^2 = \frac{1}{2^n} \left(\sum_{k=0}^{N-1} d_k\right)^2 \\ &= \left(\sum_{k=0}^{N-1} H_{0k}^{\otimes n} d_k\right)^2, \end{aligned} \quad (13)$$

となるが, これは (11) で $j = 0$ とおいたものに矛盾する. したがって, (12) の等号成立条件は満たされる, すなわち $\mathbf{a} = \mathbf{d}$ or $\mathbf{a} = -\mathbf{d}$ である. \square

ここで, d_j^H は d_j に Walsh Hadamard 変換 [7] を施すことで, 古典計算機を用いて計算可能である. 以上の定理は (8) と (9) がおおそ満たされるように変分量子アルゴリズムを実行すれば, 期待通りのデータロード回路が得られることを示唆している. なお, (8) と (9) は両方とも, 「左辺で表される確率分布が右辺で表される確率分布と等しい」という条件であると解釈でき, 本章の最後に述べる通り, 学習の結果これらの条件が満たされるようなコスト関数を設計することができる.

ケース 2 の場合は少し工夫をすることで, (4) と等価な条件を導くことができる. 下記のように, 計算基底で見た場合に振幅が非負のものと負のものを分離して表すことにする:

$$|Data\rangle = |Data^+\rangle + |Data^-\rangle, \quad (14)$$

ここで, $|Data^+\rangle$ は非負の振幅のみを持ち, $|Data^-\rangle$ は負の振幅のみを持つ. すると, 補助ビットを導入することで, $(n+1)$ 量子ビットの状態を以下のように作ることができる:

$$|\bar{\psi}\rangle \equiv |Data^+\rangle|0\rangle - |Data^-\rangle|1\rangle. \quad (15)$$

この状態の振幅が全て非負であり, かつ $\langle\bar{\psi}|\bar{\psi}\rangle = 1$ が満たされることは簡単な計算で確かめることができる. さて, $|\bar{\psi}\rangle = \sum_{j=0}^{2N-1} \bar{d}_j |j\rangle$ のように $2N$ 次元ベクトル $\bar{\mathbf{d}}$ を使って計算基底で展開する. すると, $|\bar{\psi}\rangle$ はケース 1 の条件 (=全ての振幅の符号が同じ) を満たすから, 定理 1 により, もし

$$|\langle j|U(\theta)|0\rangle^{\otimes n+1}|^2 = \bar{d}_j^2 \quad (16)$$

$$|\langle j|H^{\otimes n+1}U(\theta)|0\rangle^{\otimes n+1}|^2 = (\bar{d}_j^H)^2 \quad (17)$$

が満たされれば $U(\theta)|0\rangle^{\otimes n+1} = \pm|\bar{\psi}\rangle$ となる. ここで, $\bar{d}_j^H =$

$\sum_{k=0}^{N-1} \bar{d}_k \langle j|H^{\otimes n}|k\rangle$ である. 一旦 $|\bar{\psi}\rangle$ が得られれば, 次の手続きで $|Data\rangle$ が得られる. すなわち, アダマール変換を $(n+1)$ 番目の量子ビットに作用することで

$$\begin{aligned} I^{\otimes n} \otimes H |\bar{\psi}\rangle &= \frac{|Data^+\rangle - |Data^-\rangle}{\sqrt{2}} |0\rangle \\ &+ \frac{|Data^+\rangle + |Data^-\rangle}{\sqrt{2}} |1\rangle, \end{aligned} \quad (18)$$

が得られ, $(n+1)$ 番目の量子ビットを測定し $|1\rangle$ に事後選択することで始めの n 量子ビットが $|Data\rangle$ と一致する. 以上をまとめると, 次の定理が成立する.

定理 2. ケース 2 において, $(n+1)$ 量子ビットに作用する $U(\theta)$ が式 (16) と式 (17) を満たすとする. すると, もし $(I^{\otimes n} \otimes H)U(\theta)|0\rangle^{\otimes n}$ の $n+1$ 番目の量子ビットを測定した結果が $|1\rangle$ であれば, 測定後の状態の始めの n 量子ビットは位相を除いて $|Data\rangle$ と一致する. つまり, 事後処理を $V = (I^{\otimes n} \otimes |1\rangle\langle 1|)(I^{\otimes n} \otimes H)$ とすれば, 式 (16) と式 (17) の条件は (4) と等価である.

ケース 1 と同じく, 以上の定理が示唆しているのは, (16) と (17) がおおそ満たされるように変分量子アルゴリズムを実行すれば, 事後選択によって, 期待通りのデータロード回路が得られる, ということである. なお, 事後選択の成功確率は (18) から簡単に計算でき, その値は $1/2$ となる. 図 2 に, ケース 1, ケース 2 それぞれの学習ルール, データロード方法についてまとめている.

学習方法

最後に $U(\theta)$ の学習方法について述べる. 上述の通り, ケース 1 の場合は式 (8) と 式 (9) が近似的に満たされるように, ケース 2 の場合は, 式 (16) と式 (17) が近似的に満たされるように $U(\theta)$ を学習する. この目的のために, 我々は maximum mean discrepancy(MMD) と呼ばれるコスト関数を使う. MMD を変分量子アルゴリズムに応用するアイデアは既存研究 [8], [9] で提案されている.

MMD は二つの確率分布の違いをコストとして表現したものである. 答えとなる確率分布を $p(j)$, モデルの確率分布を $q_\theta(j)$ とすると, MMD は以下のように表される:

$$\begin{aligned} \mathcal{L}_{MMD}(q_\theta, p) &= \mathbf{E}_{k \sim q_\theta}^{j \sim q_\theta} [\kappa(j, k)] - 2\mathbf{E}_{k \sim p}^{j \sim q_\theta} [\kappa(j, k)] \\ &+ \mathbf{E}_{k \sim p}^{j \sim p} [\kappa(j, k)]. \end{aligned} \quad (19)$$

ここで, $\mathbf{E}_{k \sim q_\theta}^{j \sim q}[\cdot]$ は j を確率分布 q からサンプリングし, k を確率分布 p からサンプリングしたときの期待値を表す. また, $\kappa(j, k)$ はカーネル関数と呼ばれ, 半正定値性を満たす限りにおいては, 任意に選ぶことができる関数である. なお, カーネル関数が「特性的」であれば, 「 $\mathcal{L}_{MMD}(q_\theta, p) = 0$ ならば $q_\theta(j) = p(j) \forall j$ 」がいえることが知られている [10], [11] (ここでは「特性的」であるための条件については議論しない). しばしば利用されるガウスクーネルは, 特性的なカーネルの例である.

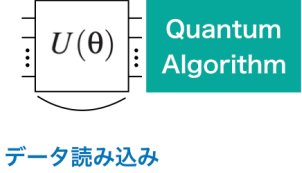
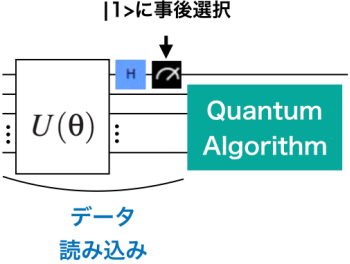
| | $U(\theta) 0\rangle$ のターゲット | $U(\theta)$ の学習ルール | データ読み込みの方法 |
|------|---|--|--|
| ケース1 | $ Data\rangle = \sum_{j=0}^{N-1} d_j j\rangle$ | <p>以下が近似的に満たされるように学習</p> $ \langle j U(\theta) 0\rangle ^2 = d_j^2$ $ \langle j H^{\otimes n}U(\theta) 0\rangle ^2 = \left(\sum_{k=0}^{N-1} d_k \langle j H^{\otimes n} k\rangle\right)^2$ |  <p>データ読み込み</p> |
| ケース2 | <p>但し</p> $ \psi\rangle \equiv Data^+\rangle 0\rangle - Data^-\rangle 1\rangle$ $ Data\rangle = Data^+\rangle + Data^-\rangle$ <p style="text-align: center;">非負の振幅 負の振幅</p> | <p>以下が近似的に満たされるように学習</p> $ \langle j U(\theta) 0\rangle ^2 = \bar{d}_j^2$ $ \langle j H^{\otimes n+1}U(\theta) 0\rangle ^2 = \left(\sum_{k=0}^{2^{N-1}} \bar{d}_k \langle j H^{\otimes n+1} k\rangle\right)^2$ |  <p>データ読み込み</p> |

図 2 学習ルール、データロード方法の概要

学習のゴールは、ケース 1、ケース 2 の両者ともに、量子回路の測定結果が従う確率分布を、ターゲットとなるベクトルに対応する確率分布におおよそ一致させることである。具体的には、ケース 1 の場合、学習のゴールは下記の確率分布

$$q_\theta(j) = |\langle j|U(\theta)|0\rangle|^2,$$

$$q_\theta^H(j) = |\langle j|H^{\otimes n}U(\theta)|0\rangle|^2$$

をそれぞれ以下のターゲット分布に近づけていくこと

$$p(j) = d_j^2, \quad p^H(j) = (\bar{d}_j^H)^2, \quad (20)$$

に対応する。また、ケース 2 の場合の学習のゴールは、下記の確率分布

$$q_\theta(j) = |\langle j|U(\theta)|0\rangle|^2,$$

$$q_\theta^H(j) = |\langle j|H^{\otimes n+1}U(\theta)|0\rangle|^2$$

をそれぞれターゲット分布

$$p(j) = \bar{d}_j^2, \quad p^H(j) = \bar{d}_j^{H^2}, \quad (21)$$

に近づけていくことに他ならない。

いずれのケースにおいても、二つの MMD を組み合わせた下記の関数:

$$\mathcal{L}(\theta) = \frac{\mathcal{L}_{MMD}(q_\theta, p) + \mathcal{L}_{MMD}(q_\theta^H, p^H)}{2} \quad (22)$$

は我々の目的に合ったコスト関数となる。まず、 $\mathcal{L}(\theta)$ は

$\mathcal{L}_{MMD}(q_\theta, p) = 0$ と $\mathcal{L}_{MMD}(q_\theta^H, p^H) = 0$ の両方が 0 となったときのみ最小値 0 となるが、特性的なカーネルを使う限りにおいて、これは $q_\theta(j) = p(j)$ かつ $q_\theta^H(j) = p^H(j)$ ($\forall j$) のときのみ $\mathcal{L}_{MMD}(q_\theta, p) = 0$ となることを意味する。つまり、量子変分アルゴリズムを用いて $\mathcal{L}(\theta)$ を 0 に近づけることができれば、ケース 1 の場合ならば (8) と (9)、ケース 2 の場合ならば (16) と (17) が近似的に満たされるわけである。

また、 $\mathcal{L}(\theta)$ の勾配の推定も容易である。演算子 P_r を 1 量子ビットに作用するエルミート演算子で、かつ $P_r^2 = 1$ を満たすとする (パウリ演算子は、この条件を満たす)。さらに各パラメータが θ_r が $U(\theta)$ に $\exp(-i\theta_r P_r/2)$ の形で埋め込まれているとする。このとき、 \mathcal{L} の θ_r に関する微分は

$$2 \frac{\partial \mathcal{L}}{\partial \theta_r} = \mathbf{E}_{k \sim q_\theta}^{j \sim q_{\theta_r}^+} \kappa(j, k) - \mathbf{E}_{k \sim q_\theta}^{j \sim q_{\theta_r}^-} \kappa(j, k) \\ - \mathbf{E}_{k \sim p}^{j \sim q_{\theta_r}^+} \kappa(j, k) + \mathbf{E}_{k \sim p}^{j \sim q_{\theta_r}^-} \kappa(j, k) \\ + \mathbf{E}_{k \sim q_\theta^H}^{j \sim q_{\theta_r}^{H+}} \kappa(j, k) - \mathbf{E}_{k \sim q_\theta^H}^{j \sim q_{\theta_r}^{H-}} \kappa(j, k) \\ - \mathbf{E}_{k \sim p^H}^{j \sim q_{\theta_r}^{H+}} \kappa(j, k) + \mathbf{E}_{k \sim p^H}^{j \sim q_{\theta_r}^{H-}} \kappa(j, k), \quad (23)$$

と明示的に計算することができる。ここで、シフトされた量子回路

$$U_{r\pm}(\theta) = U_{r\pm}(\{\theta_1, \dots, \theta_{r-1}, \theta_r, \theta_{r+1}, \dots, \theta_R\}) \\ = U(\{\theta_1, \dots, \theta_{r-1}, \theta_r \pm \pi/2, \theta_{r+1}, \dots, \theta_R\}), \quad (24)$$

を使って、シフトされた確率分布 $q_{\theta_r}^{\pm}(j) = |\langle j|U_{r\pm}(\theta)|0\rangle|^2$, $q_{\theta_r}^{H\pm}(j) = |\langle j|HU_{r\pm}(\theta)|0\rangle|^2$ を定義した. 勾配 (23) は j と k を確率分布 q_{θ} , $q_{\theta_r}^+$, $q_{\theta_r}^-$, q_{θ}^H , $q_{\theta_r}^{H+}$, $q_{\theta_r}^{H-}$, p , p^H からサンプリングすることで推定することができる.

以上を踏まえ、提案手法では $\mathcal{L}(\theta)$ をコスト関数として採用し、2.2 で述べた変分量子アルゴリズムを使ってデータロード回路を学習する. もし、適切に学習が進み、 $\mathcal{L}(\theta)$ を 0 に近づけることができたならば、望むデータロードサーキットを得ることができる. なお、 $\mathcal{L}(\theta) < \epsilon$ (ϵ は誤差) を実現するうえで $O(\text{poly}(n, 1/\epsilon))$ の測定を行えば十分であることを付け加えておく.

(参考) 数値計算による提案手法の検証

提案手法が実際にうまくワークするか否かを検証するために、[6] では金融データを用いた数値計算を行っている. 実際浅い回路でうまくデータがロードできること、およびロードした初期状態を用いて量子アルゴリズムが正しく動作することを確認している. もしご興味を持って頂いた方は、是非 [6] をご参照頂きたい.

3. まとめ

本稿では、古典データを効率的に埋め込む量子回路を学習する Approximate Amplitude Encoding (AAE) という手法を提案した. AAE の画期的な点は、2 つの maximum mean discrepancy を組み合わせたコスト関数を巧みに設計することで、古典データのロード回路を、変分量子アルゴリズムによって学習できるようにしたアイデアである. 我々が提案した AAE を活用することで、HHL などの様々なアルゴリズムの実現、ひいては NISQ の実社会への応用を著しく早めることができるはずである.

参考文献

- [1] Shor, P.: Algorithms for quantum computation: discrete logarithms and factoring, *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pp. 124–134 (1994).
- [2] Grover, L. K.: A fast quantum mechanical algorithm for database search, *STOC '96* (1996).
- [3] Harrow, A. W., Hassidim, A. and Lloyd, S.: Quantum algorithm for linear systems of equations, *Physical review letters*, Vol. 103, No. 15, p. 150502 (2009).
- [4] Preskill, J.: Quantum Computing in the NISQ era and beyond, *Quantum*, Vol. 2, p. 79 (online), DOI: 10.22331/q-2018-08-06-79 (2018).
- [5] Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S., Endo, S., Fujii, K., McClean, J., Mitarai, K., Yuan, X., Cincio, L. and Coles, P. J.: Variational Quantum Algorithms, *ArXiv*, Vol. abs/2012.09265 (2020).
- [6] Nakaji, K., Uno, S., Suzuki, Y., Raymond, R., Onodera, T., Tanaka, T., Tezuka, H., Mitsuda, N. and Yamamoto, N.: Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicator (2021).
- [7] Ahmed, N. and Rao, K. R.: Walsh-hadamard transform,

Orthogonal Transforms for Digital Signal Processing, Springer, pp. 99–152 (1975).

- [8] Liu, J.-G. and Wang, L.: Differentiable learning of quantum circuit born machines, *Physical Review A*, Vol. 98, No. 6, p. 062324 (2018).
- [9] Coyle, B., Mills, D., Danos, V. and Kashefi, E.: The born supremacy: Quantum advantage and training of an ising born machine, *npj Quantum Information*, Vol. 6, No. 1, pp. 1–11 (2020).
- [10] Sriperumbudur, B. K., Gretton, A., Fukumizu, K., Lanckriet, G. and Schölkopf, B.: Injective Hilbert Space Embeddings of Probability Measures, *COLT* (2008).
- [11] Fukumizu, K., Gretton, A., Sun, X. and Schölkopf, B.: Kernel Measures of Conditional Dependence, *NIPS* (2007).