

# 制約付き組合せ最適化問題に向けた Quantum Approximate Optimization Algorithm の改良

中田百科<sup>1</sup> 藤村怜香 棚橋耕太郎<sup>1</sup>

**概要:** 量子コンピュータ技術は、古典コンピュータと比較して組合せ最適化問題を高速に解くことができると期待されている。中でも、量子ゲート方式を用いた組合せ最適化の近似アルゴリズムである QAOA (Quantum Approximate Optimization Algorithm) は、NISQ (Noisy Intermediate-Scale Quantum device) でも動作すると期待されており盛んに研究されている。QAOAにより効率的に解を探索するためには状態遷移を記述するミキサースハミルトニアン<sup>1</sup>の適切な設計が重要であるが、一部の特殊な制約を除き一般の制約に対する設計方法はこれまで知られていなかった。そこで本研究では複数の多項式制約に対応したミキサースハミルトニアン<sup>1</sup>の設計方法を提案し、その有効性を示した。

**キーワード:** 制約付き組合せ最適化, QAOA

## Improved Quantum Approximate Optimization Algorithm for Constrained Combinatorial Optimization

### 1. はじめに

組合せ最適化問題は、広告配信、物流、製造など実世界に多数のアプリケーションを持つ。近年、データ流通量の増加に伴いこれらの問題の規模も増加しており、現実的な時間内での求解が困難になる事例が多くなってきた。この解決のため、近似解法のヒューリスティクスや、特に最近では量子コンピュータ技術などが盛んに開発されている。量子コンピュータは古典計算機とは異なり、0と1のビットの状態を同時に保有できることにより全ビット列の組合せを並列処理できる性質を持つ。このため、量子コンピュータによって組合せ最適化問題が高速に解けると期待されている。

量子コンピュータ、特に量子ゲート方式を用いた組合せ最適化にはいくつか手法が存在し、グローバのアルゴリズムを用いたグローバ適応探索 (Grover Adaptive Search, GAS) [1]、そして変分量子回路を用いた量子・古典ハイブリッドアルゴリズム[2]が知られている。特に変分量子回路による方法は、現在のノイズあり中規模量子デバイス、すなわち NISQ (Noisy Intermediate-Scale Quantum device) でも動作すると期待されており、盛んに研究されている。この方法は量子回路を変分パラメータを用いてモデル化した上で、組合せ問題の目的関数に対応するコストハミルトニアン<sup>1</sup>のエネルギー期待値を算出し、その値が最小になるように変分パラメータを最適化する手法である。変分パラメータの学習が成功すれば、組合せ問題の最適解の良い近似解を出力するような量子回路を獲得することができる。その

中でも、QAOA (Quantum Approximate Optimization Algorithm) は量子回路のモデルとしてミキサースハミルトニアン<sup>1</sup>による時間発展とコストハミルトニアン<sup>1</sup>による時間発展を繰り返し組み込んだ手法である。前者は現在の状態を別の状態に遷移させる役割を担い、後者は組合せ問題の目的関数に応じたダイナミクスを与える。各時間発展のタイムスケールが変分パラメータに相当する。

通常、ミキサースハミルトニアン<sup>1</sup>には一様横場のハミルトニアン<sup>1</sup>が用いられるが[3]、問題に応じて適切に設計することで、効率的な解探索を実現することができる。例えば制約付きの組合せ最適化の場合、図 1 に示すようにミキサースにより制約を満たすような状態間での遷移のみを実現すれば、良質な実行可能解を求めることができると期待される。ところが、一部の特殊な制約を除き一般の制約に対するミキサースの設計指針はまだ知られていない状況である。そこで本研究では、複数の多項式制約に対する新たなミキサースの設計方法を提案し、複雑な制約に対応した QAOA アルゴリズムを開発した。以下、図 1 に示した制約を満たす状態の空間を実行可能解空間 $\Omega_F$ と呼ぶことにする。

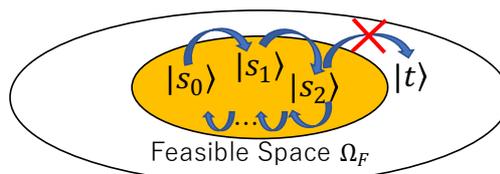


図 1 制約を満たす空間内の遷移の様子

<sup>1</sup> (株) リクルート  
Recruit Co., Ltd.

## 2. 従来手法

本節ではまず QAOA の一般的手法を説明し、その後 QAOA を用いた制約付きの組合せ最適化の従来手法を説明する。

### 2.1 QAOA

QAOA では、ミキサーハミルトニアン  $\hat{H}_M$  による時間発展  $U_M(\beta) = e^{-i\beta\hat{H}_M}$  とコストハミルトニアン  $\hat{H}_C$  による時間発展  $U_C(\gamma) = e^{-i\gamma\hat{H}_C}$  を繰り返す量子回路モデルを用いる。

$$U_{QAOA} = U_M(\beta_T)U_C(\gamma_T) \cdots U_M(\beta_1)U_C(\gamma_1) \quad (1)$$

ミキサーハミルトニアンとしては通常量子アニーリングと同様に  $\hat{H}_M = \sum_{i=1}^n X_i$  の一様横磁場ハミルトニアンが与えられる。コストハミルトニアンは問題の目的関数  $F(\mathbf{x})$  を  $\mathbf{x} \rightarrow (1-Z)/2$  の変換により Pauli の  $Z$  演算子を用いて記述したものであり、以下二次式を想定する。各時間発展の時間が変分パラメータ  $\beta, \gamma$  であり、コストハミルトニアンのエネルギー期待値が最小になるように値を更新する。繰り返し数  $T$  が少ない場合は浅い量子回路で済むため NISQ でも動作すると期待されている。以下、 $T$  を QAOA の層数と呼ぶことにする。なお、式(1)のゲートを作用させる初期状態には、各ビットにアダマールゲートを作用させてできる状態、すなわち全状態の一様足し合わせ状態  $|+\rangle$  が一般的に用いられる。

### 2.2 ペナルティ項による方法

QAOA を制約付きの問題に適用する場合、制約式に対するペナルティ項を用いる方法が一般的である。すなわち、満たすべき制約式  $P_k(\mathbf{Z}) = 0$  ( $k = 1, \dots, M$ ) と十分大きい正の係数  $\lambda$  を用いて、元のコストハミルトニアン  $\hat{H}_{C0}$  を

$$\hat{H}_C = \hat{H}_{C0} + \lambda \sum_{k=1}^M (P_k(\mathbf{Z}))^2 \quad (2)$$

のように修正するものであり、この方法は量子アニーリングでも頻繁に使用される。例えば制約式  $P_k(\mathbf{Z})$  が一次式の場合には式(2)は二次式となり、2.1 節に記した通常の QAOA を用いて解くことができる。

ただし、ペナルティ項により多数の結合が生じるため実装に必要なゲート数が増えること、係数  $\lambda$  の設計をマニュアルで決める必要があること、制約を破る解がサンプルされ得ることなど多数の課題がある。特に最後の課題に起因して、大規模な回数観測を行わなければ高質な実行可能解の獲得が困難になる可能性がある。

### 2.3 ミキサーの修正による方法

前節とは別の方法で、制約付きの問題を解く方法を説明する。すなわち、ミキサーハミルトニアンを制約に応じて修正する方法である。この時ミキサーハミルトニアンには

以下の2つの条件を満たすことが要求される[4]。すなわち、

$$\langle p | \hat{H}_M | q \rangle = \langle q | \hat{H}_M | p \rangle = 0 \quad (\forall p \in \Omega_F, \forall q \notin \Omega_F) \quad (3)$$

$$\exists r \in \mathbb{Z}^+ \text{ s.t. } \langle p | (\hat{H}_M)^r | q \rangle \neq 0 \quad (\forall p, q \in \Omega_F) \quad (4)$$

である。前者は制約を満たす状態と制約を破る状態間の遷移確率が0となる条件であるため、時間発展中に実行可能解空間外との遷移が禁止されることを要請する。後者は制約を満たす状態間の遷移確率が全て有限という条件のため、実行可能解空間の全域を探索できることを要請する。従って、ミキサーハミルトニアンが式(3)及び(4)を満たせば実行可能解空間  $\Omega_F$  のみ、かつその全域を探索することが可能となる。なお式(4)の  $\mathbb{Z}^+$  は自然数の集合を表す。

文献[4][5]では one-hot 制約等に対して上記の2条件を満たすミキサーを設計しているが、他の種類の制約や、あるいは制約が複数存在する場合に適用できるものではない。また、文献[6]は一次式の線形制約に対するミキサーを集合被覆問題を用いて導出しようとした研究であるが、問題の次元が増えた際には提案手法では式(4)を満たすように設計することが難しくなるという結論に至っている。

また、制約を満たす全実行可能解の状態の一様足し合わせ  $|S\rangle = \frac{1}{\sqrt{|\Omega_F|}} \sum_{\mathbf{x} \in \Omega_F} |\mathbf{x}\rangle$  を用意した上で、ミキサーを

$$\begin{aligned} \hat{H}_M &= |S\rangle\langle S| \\ U_M(\beta) &= e^{-i\beta\hat{H}_M} \end{aligned} \quad (5)$$

により構築する方法も最近開発され、グローバリーミキサーと呼ばれている[7]。ここで、 $|\Omega_F|$  は制約を満たす状態の数を表す。グローバリーミキサーはミキサーが満たすべき式(3)、(4)の2つの条件を満たす。ただし、同文献では巡回セールスマン問題といった、全実行可能解の一様足し合わせ状態  $|S\rangle$  の生成が比較的容易なもののみを対象とした検討を行っており、一般の制約に対する具体的なミキサー表式を求めてはいない。そこで本研究では複数の多項式制約という幅広いクラスの制約条件を対象として、グローバリーミキサーの具体的な表式を導出することを検討した。

## 3. 提案手法

### 3.1 ミキサーの定式化

本研究では、二値の説明変数  $\mathbf{x} \in \{0,1\}^n$  に対し複数の多項式制約

$$P_k(\mathbf{x}) \equiv \sum_{J \subset \{0,1,\dots,n-1\}} a_{kJ} \prod_{j \in J} x_j = 0 \quad (k = 1, \dots, M) \quad (6)$$

が課された、目的関数  $F(\mathbf{x})$  を最小化する問題を考える。ここで、 $J$  は  $\{0,1,\dots,n-1\}$  の部分集合であり、例えば  $J = \{0,1\}$  なら  $a_{kJ}$  は  $x_0x_1$  の項の係数を表す。式(5)のグローバリーミキサーを構築するため、これらの多項式制約を満たす全実行可能解の一様足し合わせ状態  $|S\rangle$  を以下の手順で作成する。

まず、多項式制約式が  $P(\mathbf{x}) \equiv \sum_{J \subset \{0,1,\dots,n-1\}} a_J \prod_{j \in J} x_j$  の1つのみの場合を考える。説明変数に対応するビット  $|0\rangle_n$  と、制

約式の値を書き込む補助ビット $|0\rangle_m$ にアダマールゲートを作用させた $|+\rangle = H^{\otimes n+m}|0\rangle_n|0\rangle_m = \frac{1}{\sqrt{2^{n+m}}}\sum_{x,k}|x\rangle_n|k\rangle_m$ を作成する。ここで、補助ビットのビット数 $m$ は制約式 $P(\mathbf{x})$ の値を書き込むための精度であり、二進数表示の桁数に相当する。すなわち、制約式の値の水準を $D$ とすると $m = O(\log_2 D)$ となる。補助ビットに制約式の値を書き込むため、補助ビットに作用するゲート $U_P(\theta) = R_Z(2^{m-1}\theta) \otimes R_Z(2^{m-2}\theta) \cdots \otimes R_Z(\theta)$ を定義する。ここで $R_Z(\theta) = e^{-i\theta Z/2}$ は $Z$ 軸に関する $\theta$ 回転のゲートである。多項式制約中の各単項式 $\prod_{j \in J} x_j$ に現れる変数 $x_j$ のビットを制御ビットとして補助ビットに $U_P\left(\frac{2\pi}{2^m} a_j\right)$ を作用させることで、以下のように位相に多項式の値 $P(\mathbf{x})$ を書き込むことができる。

$$\begin{aligned} |+\rangle &\rightarrow \frac{1}{\sqrt{2^{n+m}}} \sum_{x,k} e^{i\frac{\sum_{j \in \{0,1,\dots,n-1\}} a_j \prod_{j \in J} x_j}{2^m}} |x\rangle_n |k\rangle_m \\ &= \frac{1}{\sqrt{2^{n+m}}} \sum_{x,k} e^{i\frac{kP(\mathbf{x})}{2^m}} |x\rangle_n |k\rangle_m \end{aligned} \quad (7)$$

次に、量子逆フーリエ変換 $U_{QFT}^\dagger$ を用いて、多項式の値を補助ビットに格納する。

$$U_{QFT}^\dagger \frac{1}{\sqrt{2^{n+m}}} \sum_{x,k} e^{i\frac{kP(\mathbf{x})}{2^m}} |x\rangle_n |k\rangle_m = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle_n |P(\mathbf{x})\rangle_m \quad (8)$$

式(8)の右辺の状態を以下 $|\Omega\rangle \equiv U_F|0\rangle_n|0\rangle_m$ と表し、これまでに使用したゲートをまとめて $U_F$ と書くことにする。

この状態 $|\Omega\rangle$ には制約を破る状態、すなわち補助ビットが $|k\rangle_m$  ( $k \neq 0$ )の状態も含まれている。従って、全実行可能解の一樣足し合わせ状態 $|\mathcal{S}\rangle$ を作るには、制約を破る状態の振幅を低減させ、制約を満たす状態のみを抽出する必要がある。そこで本研究では、以下のグローバールゴリズムのゲート演算 $G = U_{diff}U_{inv}$

$$\begin{aligned} U_{inv} &= I - 2|0\rangle_m\langle 0| \\ U_{diff} &= 2|\Omega\rangle\langle\Omega| - I \\ &= 2U_F|0\rangle\langle 0|U_F^\dagger - I \\ &= U_F(2|0\rangle\langle 0| - I)U_F^\dagger \end{aligned} \quad (9)$$

を適切な回数 $d$ だけ作用させることで、全実行可能解の一樣足し合わせ状態 $|\mathcal{S}\rangle$ を近似的に作成する。

$$\begin{aligned} |0\rangle_n|0\rangle_m &\rightarrow |\mathcal{S}\rangle \equiv G^d U_F |0\rangle_n |0\rangle_m \\ &\equiv \frac{1}{\sqrt{|\Omega_F|}} \sum_{x \in \Omega_F} |x\rangle_n |0\rangle_m \end{aligned} \quad (11)$$

以下右辺の状態を $|\mathcal{S}\rangle$ と表す。なお式(10)では $|0\rangle_n|0\rangle_m = |0\rangle$ と略記した。ここで回数 $d$ は実行可能解の状態確率が最大になるように決定される。

次に複数の制約条件がある場合を考える。全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ は、主に以下の2つの方法で求めることができる。

### (1) 一つの多項式制約に集約する方法

一つ目は、各制約条件を一つの多項式制約に集約し、そ

の制約式に対して上記の手続きを1回のみ行う方法である。例えば、各制約の二乗和で表される多項式 $P(\mathbf{x}) = \sum_{k=1}^M (P_k(\mathbf{x}))^2$ を考えると、全制約条件 $P_k(\mathbf{x}) = 0$  ( $k = 1, \dots, M$ )とこの1つの多項式制約条件 $P(\mathbf{x}) = 0$ は等価になる。したがって、この1つの多項式 $P(\mathbf{x})$ に対して式(11)のゲート演算群を作用させることで、全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ を作成することができる。

### (2) 制約毎に補助ビットに書き込む方法

二つ目は、各制約 $P_k(\mathbf{x}) = 0$ に対して上記の手続きを繰り返す方法である。すなわち、各制約に対して別々の補助ビット $|0\rangle_{m_k}$  ( $k = 1, \dots, M$ )を用意し、そこに式(7)、式(8)を用いて各制約式の値を書き込む。以下、全制約の値を書き込むゲートを改めて $U_F$ と表し直す。このようにして構築した $|\Omega\rangle \equiv U_F|0\rangle_n|0\rangle_m$ に対し、グローバールのアルゴリズムのゲート演算 $G$ を作用させる方法である。

このように、全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ を構築する方法は複数考えられる。実用上は、補助ビットの数や異なるビット間の相互作用項の数等を勘案した上で方法を決定すれば良い。例えば、一つ目の方法では二乗和をとる際にビット間に多数の結合が生じるため、結合の自由度が高いハードウェアであれば実装できる可能性がある。また二つ目の方法ではビット間の結合は少なく済むものの、制約ごとに個別の補助ビットが必要になるため、一つ目の方法よりも補助ビット数が増加する可能性がある。この場合にはビット数が多いハードウェアでの実装が望ましいであろう。

以上の手続きにより求めた、全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ を用いて、グローバールミキサーを構築する。式(5)の $|\mathcal{S}\rangle$ に $|\mathcal{S}\rangle$ を代入して整理すると、

$$\begin{aligned} U_M(\beta) &= e^{-i\beta|\mathcal{S}\rangle\langle\mathcal{S}|} \\ &= \sum_{k=0}^{\infty} \frac{(-i\beta)^k}{k!} (|\mathcal{S}\rangle\langle\mathcal{S}|)^k \\ &= I + \sum_{k=1}^{\infty} \frac{(-i\beta)^k}{k!} |\mathcal{S}\rangle\langle\mathcal{S}| (|\mathcal{S}\rangle\langle\mathcal{S}|)^{k-1} |\mathcal{S}\rangle\langle\mathcal{S}| \\ &= I + \sum_{k=1}^{\infty} \frac{(-i\beta)^k}{k!} |\mathcal{S}\rangle\langle\mathcal{S}| \\ &= I - (1 - e^{-i\beta}) |\mathcal{S}\rangle\langle\mathcal{S}| \\ &= I - (1 - e^{-i\beta}) G^d U_F |0\rangle\langle 0| U_F^\dagger (G^\dagger)^d \\ &= G^d U_F (I - (1 - e^{-i\beta}) |0\rangle\langle 0|) U_F^\dagger (G^\dagger)^d \end{aligned} \quad (12)$$

を得る。上式では $|0\rangle_n|0\rangle_m = |0\rangle$ と略記した。

以上により、複数の多項式制約に対応したグローバールミキサーの具体的な表式を求めることができた。なおこの量子回路には $2|0\rangle\langle 0| - I$ や $I - (1 - e^{-i\beta})|0\rangle\langle 0|$ のユニタリゲートが含まれるが、これらはMulti-controlled Zゲートを用いることで実装することができる[7]。

### 3.2 コストハミルトニアンと初期状態

本研究では、ほぼ実行可能解の空間内のみを遷移させるミキサーを使用するため、式(2)のようにコストハミルトニアンにペナルティ項を付与する必要はない。従って対象とする問題の目的関数 $F(\mathbf{x})$ を $x \rightarrow (1-Z)/2$ の変換によりPauliのZ演算子を用いて記述した $\hat{H}_{C0}$ のみを、本提案手法におけるコストハミルトニアンとする。

以上のミキサーハミルトニアンとコストハミルトニアンを用いて式(1)の $U_{QAOA}$ を構築することができる。このゲートを作用させる初期状態としては、2.1節の通常のQAOAと同様に全状態の一樣足し合わせ状態 $|+\rangle$ を用いても良いが、本研究では式(11)に示した全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ を用いる。これにより、初期状態の時点で高い確率で実行可能解をサンプルできるようになると期待される。

### 3.3 適用範囲について

本提案手法は2.1節の通常のQAOAと異なり、量子フーリエ変換及びグローバールのアルゴリズムを使用するため、ビット数の多い問題ではゲートの複雑性が増し実装が困難になる場合がある。そこで本節では、こういった問題に対して本手法が適用可能かを議論する。

量子フーリエ変換に関してはビットの数を $m$ 個として $O(m^2)$ 個の量子ゲートを要し、近似的な回路であっても $O(m \log m)$ 個のゲートが必要となる[8]。本研究では量子フーリエ変換は補助ビットに作用するため、補助ビットの数になるべく少なく済むような、制約式の値の水準が小規模な問題が望ましい。またグローバールのアルゴリズムは、全状態数と目標の状態数をそれぞれ $N_0$ 、 $N_1$ 個として $d = O(\sqrt{N_0/N_1})$ 回のゲート演算 $G$ を作用させる必要がある[9]。したがって、 $G$ の作用回数が少なく済むよう、制約を満たす状態の数の比率が高い問題が望ましい。

ただし、量子ゲートの数以外にも制御ゲートの個数など、量子コンピュータのハードウェアに実装する上で考慮すべき点は多々存在する。これらについては今後検討を行う予定である。

## 4. シミュレーション結果

本提案手法に対しQiskit[10]によるQiskit Aerシミュレーションを実施した結果について述べる。いずれの検証においても、3.1節及び3.2節で述べたミキサーハミルトニアン、コストハミルトニアンと時間発展ゲートを使用した。なお、3.1節で述べた通り全実行可能解の一樣足し合わせの近似状態 $|\mathcal{S}\rangle$ の作成には(1)、(2)の2通りの方法が考えられるが、以下では特に二つ目の方法、すなわち制約毎に補助ビットに書き込む方法を用いて、実装及びシミュレーションを行った。

### 4.1 探索の様子を図示

本提案手法による探索の効果を視覚的に理解するため、学習途中の回路で得られる解がどのように変遷するかを検証した。変数 $\mathbf{x} \in \{0,1\}^4$ に対する目的関数を $F(\mathbf{x}) = -x_0x_3 + 2x_1x_2$ 、制約条件を $P_1(\mathbf{x}) = x_0 + x_1 - 1$ と $P_2(\mathbf{x}) = x_2 - x_0 - x_3 + 1$ とする問題に対しシミュレーションを実施し、学習途中の変分パラメータの値の回路における状態サンプリング結果を図2に示した。

(a)、(b)、(c)は変分パラメータ探索の初期、中期、終了段階にて得られた代表値の回路の結果を表す。横軸は変数 $\mathbf{x}$ のとりうる状態を表し、縦軸は10000回のサンプリングで得られる頻度から算出した状態毎の観測確率である。ここで提案手法の層数を $T = 1$ 、すなわちコストハミルトニアン及びミキサーハミルトニアンの時間発展ゲートの各作用回数を1とした。式(12)のミキサーにおけるグローバールアルゴリズムの回数は $d = 1$ とした。また補助ビットは制約 $P_1(\mathbf{x})$ 、 $P_2(\mathbf{x})$ に対してそれぞれ1、2個ずつの合計3つを使用した。変分パラメータの探索にはCOBYLA[11]を使用し、実装にはSciPyのモジュールを使用した[12]。探索の初期値は $(\beta, \gamma) = (0, 0)$ 、最大イテレーション数は1000とし、それ以外はSciPyのデフォルト設定値を使用した。(a)、(b)、(c)における変分パラメータの値はそれぞれ $(\beta, \gamma) = (0.10, -0.08)$ 、 $(-1.86, -1.55)$ 、 $(-10.16, -2.39)$ である。また、変分パラメータの学習の際にエネルギー期待値は1000回のサンプル平均により算出した。

本問題において制約を満たす状態、すなわち実行可能解は $\mathbf{x} = (0,1,0,1)$ 、 $(1,0,0,0)$ 、 $(1,0,1,1)$ の3つであるが、変分パラメータ学習の初期時点ではほぼこれらの状態に収束していることが分かる。これは、式(11)に示した全実行可能解の一

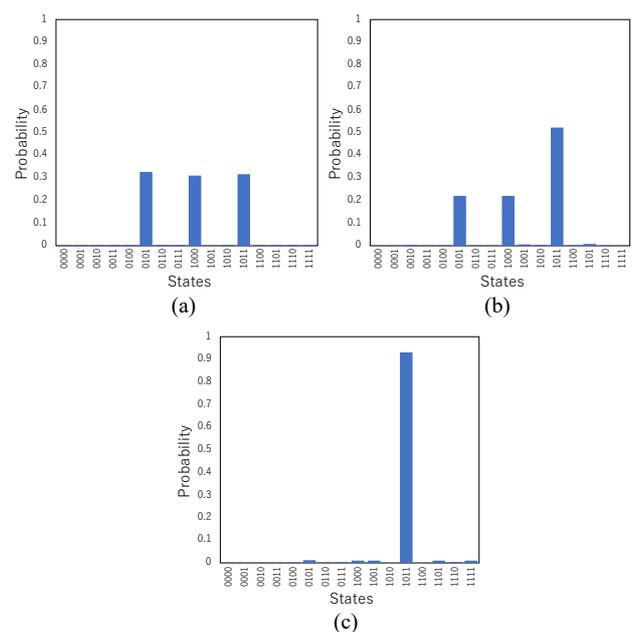


図2 学習途中の回路の状態ヒストグラム

様足し合わせの近似状態 $|\delta\rangle$ を初期状態としたためである。また、そこから変分パラメータの学習に伴い目的関数の値が最小になる大局解 $\mathbf{x} = (1,0,1,1)$ の状態の確率が上昇し、90%を超える確率に収束する様子が分かる。

#### 4.2 従来手法との比較と変分パラメータの初期値依存性の評価

次に、前節と同様の問題に対して従来手法の QAOA と比較を行う。制約付きの問題を扱う従来手法としては 2.2 及び 2.3 節で述べた 2 通りの方法が考えられるが、後者は今回の制約を扱うことができないため、2.2 節のペナルティ項を用いる方法を採用する。すなわち初期状態として全状態の一樣足し合わせ状態 $|+\rangle$ を用い、ミキサースハミルトニアンとして一樣横磁場ハミルトニアン $\hat{H}_M = \sum_{i=1}^n X_i$ を用い、コストハミルトニアンに式(2)のペナルティ項を付与する QAOA 手法との比較を実施する。1 層から 4 層までの QAOA と、提案手法を用いて解の探索を実施し、10000 回の状態サンプリングで得られた大局解の観測回数に基づいて観測確率を算出した。その結果を図 3 に示す。

横軸は変分パラメータ $\beta, \gamma$ の探索の初期値の組数を表しており、 $[0, 2\pi]$ から一樣分布でその個数分の初期値を生成して COBYLA による探索を実施し、最も低いエネルギー期待値を与える変分パラメータ値を持つ回路を用いて大局解の観測確率を求めた。なお初期値の偏りを低減するため、図 3 には上記を 100 回繰り返して得られる観測確率の平均値をプロットした。また、変分パラメータの学習の際、エネルギー期待値は 1000 回のサンプル平均により算出した。生成する初期値の組数は 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 250, 500 とした。提案手法におけるコストハミルトニアン及びミキサースハミルトニアンの時間発展ゲートの層数は 1 とし、式(12)のミキサースにおけるグローバールゴリズムの回数は $d = 1$ とした。補助ビットは制約 $P_1(\mathbf{x})$ ,  $P_2(\mathbf{x})$ に対してそれぞれ 1, 2 個ずつの合計 3 つを使用した。COBYLA における最大イテレーション数については、従来手法及び提案手法ともに 1000 に設定した。本検証に先立ち SciPy を用いて最適化を実施した結果、50 回以内で収束する傾向が見られたため、これは十分な回数と考えられる。

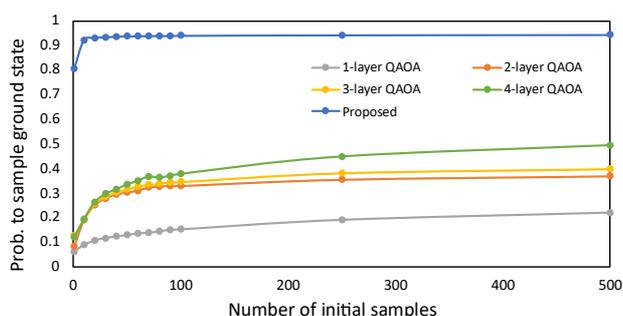


図 3 各手法における大局解の観測確率と初期値依存性

図 3 に示すように、提案手法は 1 層のみでも大局解の観測確率が高く、4 層の QAOA の約 2 倍の確率であることが分かる。また、提案手法では初期値の組数が少ない場合においても高い観測確率を誇っており、変分パラメータの初期値依存性が低いことが分かった。これは、提案手法のコストハミルトニアンが式(2)と異なりペナルティ項を含んでおらず、エネルギー期待値関数における最適化が容易になったためと考えられる。

なお、図 3 から QAOA は層数を増やすほど大局解の観測確率が向上する傾向が見られるが、この現象は QAOA の一般的な性質に一致する[13]。したがってより多層の QAOA を用いれば提案手法に匹敵する大局解の観測確率を達成できる可能性があるが、層数に比例して変分パラメータの数が増加するため、学習が困難になるという課題が生じる。

#### 4.3 様々な問題に対する検証

以上は特定の問題に対する検証であったが、本節では乱数で生成した様々な問題に対して大局解の観測確率を算出し、従来手法との比較を行う。変数 $\mathbf{x} \in \{0,1\}^6$ に対する目的関数を $F(\mathbf{x}) = \sum_{i \leq j} J_{ij} x_i x_j$ , 制約条件を $P_1(\mathbf{x}) = x_0 + x_1 - 1$ と $P_2(\mathbf{x}) = x_2 - x_0 - x_3 + 1$ ,  $P_3(\mathbf{x}) = x_4 + x_5 - 1$ とする問題に対しシミュレーションを実施し、大局解の観測確率を算出した。ここで、各係数 $J_{ij}$ は $\{-2, -1, 0, 1, 2\}$ から一樣分布で決定した。この問題を 100 回作成し、大局解の観測確率の平均をとった結果を表 1 に示す。ここで、観測確率の算出方法は前節と同様であるが、変分パラメータの初期値の組数は 100 個に固定した。また、提案手法におけるコストハミルトニアン及びミキサースハミルトニアンの時間発展ゲートの層数は 1 とし、式(12)のミキサースにおけるグローバールゴリズムの回数は $d = 1$ とした。補助ビットは制約 $P_1(\mathbf{x})$ ,  $P_2(\mathbf{x})$ ,  $P_3(\mathbf{x})$ に対してそれぞれ 1, 2, 1 個ずつの合計 4 つを使用した。COBYLA における最大イテレーション数については、従来手法及び提案手法ともに 1000 に設定した。本検証に先立ち SciPy を用いて最適化を実施した結果、50 回以内で収束する傾向が見られたため、これは十分な回数と考えられる。

表 1 により、提案手法は他手法の 3 倍程度の確率で大局解を観測できることが明らかとなった。ただし、4.2 節の問題より次元や制約の多い問題を扱ったため、図 3 の結果よりも観測確率が減少していることが分かる。本検証では 1 層のモデルを用いたが、大局解の観測確率をさらに向上させるには多層化等の工夫が必要と考えられる。

表 1 様々な問題に対する大局解の観測確率の平均

1-layer QAOA	2-layer QAOA	3-layer QAOA	4-layer QAOA	Proposed
0.085	0.107	0.101	0.118	0.345

## 5. おわりに

本報告では、複数の多項式制約条件という汎用的な制約付き組合せ最適化問題向けに、QAOA を改良する研究を実施した。具体的には、制約を満たす状態空間内のみを遷移させるようにミキサーを設計し、実行可能解を探索する新しい QAOA 手法を開発した。そのため、制約に対するペナルティ項を用いる従来の QAOA よりも効率的に解を探索することができるかと期待される。有効性検証のため Qiskit によるシミュレーションを実施し、従来手法に比べ高い確率で大局解をサンプリングできることが判明した。また、大局解の観測確率が変分パラメータの初期値の取り方にあまり依存せず、従来手法よりも変分量子回路の学習が容易であることが分かった。

今後は、より複雑な多項式制約の問題や多変数の問題への適用、及びモデルの多層化による高精度化等を検討していきたい。

## 参考文献

- [1] A. Gilliam, S. Woerner, and C. Gconciulea, “Grover adaptive search for constrained polynomial binary optimization.”, *Quantum*, vol. 5, p. 428, 2021.
- [2] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, and M. Ganzhorn et al., “Quantum Optimization Using Variational Algorithms on Near-Term Quantum Devices.”, *Quantum Science and Technology*, vol. 3, p. 030503, 2018.
- [3] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm.”, arXiv preprint, arXiv:1411.4028, 2014.
- [4] S. Hadfield, Z. Wang, B. O’Gorman, E. G. Rieffel, D. Venturelli and R. Biswas, “From the quantum approximate optimization algorithm to a quantum alternating operator ansatz.”, *Algorithms*, vol. 12, no. 2, p. 34, 2019.
- [5] Z. Wang, N. C. Rubin, J. M. Dominy, and E. G. Rieffel, “XY mixers: Analytical and numerical results for the quantum alternating operator ansatz.”, *Physical Review A*, vol. 101, no. 1, p. 012320, 2020.
- [6] H. Leipold, F. M. Spedalieri, “Constructing driver Hamiltonians for optimization problems with linear constraints.”, *Quantum Science and Technology*, vol.7, no.1, p. 015013, 2021.
- [7] A. Bartschi and S. Eidenbenz, “Grover Mixers for QAOA: Shifting Complexity from Mixer Design to State Preparation.”, in 2020 IEEE International Conference on Quantum Computing and Engineering, pp. 72-82, 2020.
- [8] D. Coppersmith, “An approximate fourier transform useful in quantum factoring.”, Technical Report RC19642, IBM, 1994.
- [9] M. Boyer, G. Brassard, P. Hoyer and A. Tappa, “Tight bounds on quantum searching.”, *Fortschritte der Physik*, vol. 46, issue 4–5, pp. 493–505, 1998.
- [10] H. Abraham, et al., “Qiskit: An open-source framework for quantum computing.”, 2019.
- [11] M. J. D. Powell, “A direct search optimization method that models the objective and constraint functions by linear interpolation.”, in *Advances in Optimization and Numerical Analysis*, eds. S. Gomez and J.-P. Hennart (Kluwer Academic: Dordrecht), pp. 51-67, 1994.
- [12] E. Jones, T. Oliphant, and P. Peterson, “SciPy: open source scientific tools for Python.”, <http://www.scipy.org/>.
- [13] G. E. Crooks, “Performance of the quantum approximate optimization algorithm on the maximum cut problem.”, arXiv preprint arXiv:1811.08419, 2018.