

2025 年“开源之夏”项目申请书

陈思源

2025 年 6 月 6 日

1 项目简介

项目名称：基于 MindSpore Quantum 实现自适应偏置场热启动 QAOA 算法

简介：在当前的含噪声中等规模量子计算时代，由于大规模容错量子计算机尚未实现，像 QAOA 这样的混合量子经典算法是实现量子优势的关键。QAOA 尤其适用于解决像最大割 (MaxCut) 这样的组合优化问题。为了提升标准 QAOA 的性能，研究者们提出了多种改进策略，其中最主要的是 (i). 热启动：利用一个强大的经典算法（如 Goemans-Williamson 算法）预先找到一个近似解，并用这个解来初始化量子算法;(ii) 自适应偏置：在 QAOA 的混合哈密顿量中加入可调节的偏置场，在优化过程中动态更新，以引导算法找到更好的解。最近的一篇论文 arXiv:2503.20048v1 提出了一种名为 WS-ab-QAOA 的新算法，它同时结合了热启动和自适应偏置这两种策略。研究表明，在解决 3-正则图上的 MaxCut 问题时，该新算法的表现优于其他 QAOA 变体和经典的 GW 算法。特别是在千比特规模的模拟中，即使是层数最浅 ($p=1$) 的 WS-ab-QAOA，其对解的改进效果也需要经典 GW 算法进行约 $10^{5.5}$ 次随机投影才能与之匹敌，这展示了在经典算法的基础上使用量子算法进行精修的巨大潜力。本次项目旨在基于 MindSpore Quantum 量子计算框架对其进行复现与拓展。

2 研究背景与研究思路

2.1 MAXCUT 问题及其经典算法

最大割问题是著名的 NP-complete 问题之一，它可以转化为如下二次整数规划问题

Problem 1 (MAXCUT).

$$\begin{aligned} & \text{maximize } W = \frac{1}{2} \sum_{i < j} w_{ij}(1 - y_i y_j) \\ & \text{subject to: } y_i \in \{-1, +1\}, \forall i \in V \end{aligned}$$

上述二次整数规划问题有一族合理的松弛，它将 Boolean 变量 y_i 替换为 m 维的归一化的矢量 $\mathbf{u}_i \in S_m = \{\mathbf{u} \in \mathbb{R}^m \mid \|\mathbf{u}\|_2 = 1\}$ ，乘积 $y_i y_j$ 替换为矢量内积 $\mathbf{u}_i \cdot \mathbf{u}_j$

Problem 2 (m -relaxed MAXCUT).

$$\begin{aligned} & \text{maximize } W = \frac{1}{2} \sum_{(i,j) \in E} w_{ij}(1 - \mathbf{u}_i \cdot \mathbf{u}_j) \\ & \text{subject to: } \mathbf{u}_i \in S_m, \forall i \in V \end{aligned}$$

取 $m = 1$ 对应去除整数约束的原问题，算法会输出一个 $n = |V|$ 维的向量 $\mathbf{c} = (c_1, \dots, c_n)$ 且每个分量在区间 $[0, 1]$ ；取 $m = n$ ，对应下一章详细叙述的 GW algorithm；取 $1 < m < n$ 对应 Burer-Monteiro 算法。

2.1.1 GW algorithm

当 $m = n$ 时，问题 2 的最优解 $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ 可以转化为如下 SDP 问题 [GW95]

Problem 3 (SDP-MAXCUT).

$$\begin{aligned} & \text{minimize } R = \text{tr}(\mathbf{w} \cdot \mathbf{Y}) \\ & \text{subject to: } \mathbf{Y}_{ii} := \text{tr}(\mathbf{D}_i \cdot \mathbf{Y}) = 1, \forall i \in [n] \\ & \mathbf{Y} \succeq 0 \end{aligned}$$

这是一个半正定优化问题，可以通过内点法等方法高效找到唯一的最优解 \mathbf{Y}^* ，此后通过奇异值分解求得 n 个 $m = n$ 维归一化的向量 $\{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ 。个维归一化的向量并不是符合整数约束的变量，为了使其符合约束，定义如下随机算法。在此基础上可以证明定理 2.1。

Algorithm 1 hyperplane rounding

Require: 图 $G = (V, E)$ ，目标函数 C ，以及 SDP 松弛后的最优解向量集合 $\{\mathbf{v}_i\}$

Ensure: 顶点集合 V 的划分 $S \cup (V \setminus S)$

- 1: 在单位球面 S_n 上均匀随机抽取一个向量 \mathbf{r}
 - 2: 定义集合 $S := \{i \in V \mid \mathbf{v}_i \cdot \mathbf{r} \geq 0\}$
 - 3: 返回划分 S 和 $V \setminus S$
-

基于合理的复杂度假设，GW 算法是 MAXCUT 问题最优的近似算法。

Theorem 2.1. 运行 *GW-random rounding* 算法 $N = \frac{A^2}{\varepsilon^2} \ln \frac{2}{\delta}$ 次，可以以 $(1 - \delta)$ 的置信概率拿到至少一个 $W > \alpha \times OPT - \varepsilon$, $\alpha \approx 0.87856$ 的划分。

2.1.2 BM-MC_k 算法

当 $1 < m < n$ 时, m-relaxed MAXCUT 不再能转化为一个 SDP 问题。具体到 $m = 2$ 的情况, 需要求解的非凸优化问题变成了:

$$\begin{aligned} &\text{maximize} \quad \langle \sigma, \mathbf{w}\sigma \rangle \\ &\text{subject to} \quad \sigma = [\sigma_1, \dots, \sigma_n]^T \in \mathbb{R}^{n \times 2} \\ &\qquad\qquad\qquad \|\sigma_i\|_2^2 = 1, \quad i \in [n] \end{aligned}$$

这个问题是在一个由 n 个圆周构成的积流形上寻找最优解, 有算法 2 高效求解 [Mei+17]。为了评定上述算法的有效性, 对 m 维向量集合 $\{\mathbf{v}_i\}$ 定义函数

Algorithm 2 BM-MC₂

Require: 矩阵 $\mathbf{w} \in \mathbb{R}^{n \times n}$

Ensure: 在约束 $\|\sigma_i\|_2^2 = 1, \quad i \in [n]$ 下, 最大化 $\langle \sigma, \mathbf{w}\sigma \rangle$

- 1: 设定目标函数矩阵 $A \leftarrow -A_G$
 - 2: **初始化:** 随机生成矩阵 $\sigma^{(0)} \in \mathbb{R}^{n \times 2}$
 - 3: **投影到流形:** 对每一行 $i \in [n]$, 更新 $\sigma_i^{(0)} \leftarrow \sigma_i^{(0)} / \|\sigma_i^{(0)}\|_2$
 - 4: 设置迭代计数器 $t \leftarrow 0$
 - 5: **while** 未满足收敛条件 (例如 $\|\text{grad}f(\sigma^{(t)})\|_F$ 足够小或达到最大迭代次数) **do**
 - 6: 计算对角矩阵 $\Lambda^{(t)} \leftarrow \text{ddiag}(A\sigma^{(t)}(\sigma^{(t)})^\top)$
 - 7: 计算黎曼梯度: $\text{grad}f(\sigma^{(t)}) \leftarrow 2(A - \Lambda^{(t)})\sigma^{(t)}$
 - 8: **if** $\|\text{grad}f(\sigma^{(t)})\|_F > \mu_G$ **then**
 - 9: $u^{(t)} \leftarrow \text{grad}f(\sigma^{(t)}) / \|\text{grad}f(\sigma^{(t)})\|_F$
 - 10: **else**
 - 11: 使用 Power Method 计算黎曼海森矩阵 $\text{Hess}f(\sigma^{(t)})$ 的近似最大特征向量 $u^{(t)}$
 - 12: **end if**
 - 13: 选择合适的步长 $\eta^{(t)}$
 - 14: 沿方向 $u^{(t)}$ 更新位置: $\tilde{\sigma} \leftarrow \sigma^{(t)} + \eta^{(t)}u^{(t)}$
 - 15: 投影回流形: 对每一行 $i \in [n]$, 更新 $\sigma_i^{(t+1)} \leftarrow \tilde{\sigma}_i / \|\tilde{\sigma}_i\|_2$
 - 16: $t \leftarrow t + 1$
 - 17: **end while**
-

$$\text{HP}(\{\mathbf{v}_i\}) = \frac{1}{\pi} \sum_{(i,j) \in E} w_{ij} \arccos(\mathbf{v}_i \cdot \mathbf{v}_j), \quad (1)$$

$$\text{BM-MC}_m(\{\mathbf{v}_i\}) = \frac{1}{2} \sum_{(i,j) \in E} w_{ij} (1 - \mathbf{v}_i \cdot \mathbf{v}_j). \quad (2)$$

定义算法 BM-MC_k 是 κ -close 的当且仅当 $\text{BM-MC}_m(\{\mathbf{v}_i\}) \geq \kappa \text{OPT}$, BM-MC_k 算法是 κ -approximate 的当且仅当 $\text{HP}(\{\mathbf{v}_i\}) \geq \kappa \text{OPT}$, 之前 GW 算法其实在声称 BM-MC_n 算法

可以做到 1-close，对于一般的 $1 < m < n$ ，由于是非 SDP 优化，没办法保证松弛问题的最优性，但是存在近似最优解，可以证明 BM-MC_k 算法满足如下近似比：

Theorem 2.2 ([Mei+17]). 存在算法 BM-MC_m 解决 m -Relaxed MAXCUT，且给出的结果一定是 $(1 - 1/(m-1))$ -closed 的。

2.2 QAOA 算法及其变体简介

2.2.1 Ordinary QAOA

如图 1, p 层的 QAOA 算法输出如下 $(\vec{\gamma}, \vec{\beta})$ 参数化的量子态

$$|\vec{\gamma}, \vec{\beta}, G\rangle := e^{-i\beta_p \hat{B}} e^{-i\gamma_p \hat{C}_G} (\dots) e^{-i\beta_1 \hat{B}} e^{-i\gamma_1 \hat{C}_G} |-\rangle^{\otimes n}, \quad (3)$$

其中 $\gamma_i, \beta_i \in [0, 2\pi)$ 是 p 维向量， $\hat{B} = \sum_j X_j$ 。接着通过最小化最大割问题哈密顿量 C_G 的均值调整参数 $(\vec{\gamma}, \vec{\beta})$ ，

$$F_{\min}(p, G) := \min_{\gamma, \beta \in [0, 2\pi)^p} \langle \gamma, \beta, G | \hat{C}_G | \gamma, \beta, G \rangle, \quad (4)$$

在 $p \rightarrow +\infty$ 时，门的层数足够多，利用与绝热演化的等价性可以证明变分可以完美地取到最高能本征态 [FGG14]，即

$$\forall G, \lim_{p \rightarrow \infty} C(p, G) = 1, \text{ with } C(p, G) := \frac{F_{\max}(p, G)}{\langle \Phi_G | \hat{C}_G | \Phi_G \rangle} \in [0, 1]. \quad (5)$$

但是受限于实验误差，我们更关心对于固定的层数，算法最差是怎么样的。即

$$E(p) := \max_G C(p, G) = \max_G \frac{\min_{\gamma, \beta \in [0, 2\pi)^p} \langle \gamma, \beta, G | \hat{C}_G | \gamma, \beta, G \rangle}{\langle \Phi_G | \hat{C}_G | \Phi_G \rangle} \quad (6)$$

随着时间的推移，许多 QAOA 的变体被提出。这些变体都关注于修改混合操作 $e^{-i\beta \hat{B}}$ 和修

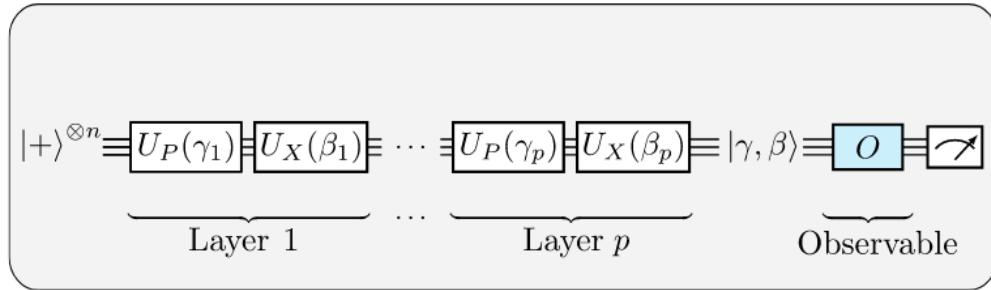


图 1: QAOA 算法线路

改初始输入态 $|-\rangle^{\otimes n}$ 以实现某些目标。仍然可以证明在绝热极限下（即线路深度 $p \rightarrow +\infty$ 时）算法会严格收敛到 MAXCUT 问题的解。然而，对于使用这些定制混合算符 (custom mixers) 的、具有有限线路深度的温启动 QAOA，要给出可证明的性能保证，目前仍是一个悬而未决的问题。

2.2.2 Warm-start QAOA

Warm-start QAOA 的思路是先用经典算法计算出一个近似解, 然后利用该解来热启动 QAOA, 从而进一步改进已得解。这些方法继承了用于热启动的经典算法的近似比例, 并且在特定实例上显著优于标准 QAOA。用于热启动的经典算法有多种选择, 先前的讨论集中于 $m = 1, 2, 3$ 的松弛算法 2, 下面进行简单回顾。

$m = 1$ case: WS-QAOA 基于 $m = 1$ 的经典松弛算法, 若 $c_j \in [0, 1]$ 是通过求解 $m = 1$ 的问题 2, 输出的最优 $n = |V|$ 维向量 $\mathbf{c} = (c_1, \dots, c_n)$, 则依据这个结果, WS-QAOA 采用如下混合哈密顿量

$$\hat{B}^{WS}(\{c_j\}) = \sum_j (-\cos \theta_j X_j - \sin \theta_j Z_j) = \tilde{R}_y(\pi - \vec{\theta}) \hat{B} \tilde{R}_y(\vec{\theta} - \pi). \quad (7)$$

其中 $R_y(\vec{\theta} - \pi) = \prod_j e^{-i(\theta_j - \pi)Y}$, 参量 θ_j 采用以下带有正则化参数 ϵ 的策略确定

$$\theta_j = \begin{cases} 2 \arcsin \sqrt{c_j} - \pi/2, & c_j \in [\epsilon, 1 - \epsilon], \\ 2 \arcsin \sqrt{\epsilon} - \pi/2, & c_j \leq \epsilon, \\ 2 \arcsin \sqrt{1 - \epsilon} - \pi/2, & c_j \geq 1 - \epsilon. \end{cases} \quad (8)$$

ϵ 的设定是为了避免在 $c_j = 0$ 或 1 时初始态精确为 $|0\rangle$ 或 $|1\rangle$, 从而导致 $\hat{B}^{WS}(\{c_j\})$ 中仅包含 Z 项。初始态要对应地选择 $\hat{B}(\{c_j\})$ 的基态 $|\psi_0^{\text{ws}}\rangle = \tilde{R}_y(\pi - \vec{\theta}) |-\rangle^{\otimes n}$, 利用 $|-\rangle = \tilde{R}_y(-\pi/2) |0\rangle$, 则输入态满足:

$$|\psi_0^{\text{ws}}\rangle = \tilde{R}_y(\pi/2 - \theta_j) |0\rangle = \cos\left(\frac{\pi/2 - \theta}{2}\right) \mathbb{I} - i \sin\left(\frac{\pi/2 - \theta}{2}\right) Y |0\rangle \quad (9)$$

$$= \begin{pmatrix} c_i & -\sqrt{1 - c_i^2} \\ \sqrt{1 - c_i^2} & c_i \end{pmatrix} |0\rangle = c_i |0\rangle + \sqrt{1 - c_i^2} |1\rangle \quad (10)$$

可见这个初态依据经典松弛算法的结果, 把量子态置于偏向 $|0\rangle$ 或者 $|1\rangle$ 。 p 层的 WS-QAOA 最终输出如下 $(\vec{\gamma}, \vec{\beta})$ 参数化的量子态, 并对参数 $(\vec{\gamma}, \vec{\beta})$ 做梯度下降

$$|\psi_f^{\text{ws}}\rangle = \prod_{k=1}^p e^{-i\beta_k \hat{B}^{WS}(\{c_j\})} e^{-i\gamma_k \hat{C}_G} |\psi_0^{\text{ws}}\rangle \quad (11)$$

$$= \prod_{l=1}^p \tilde{R}_y(\pi - \vec{\theta}) e^{-i\beta_l \hat{B}} \tilde{R}_y(\vec{\theta} - \pi) e^{-i\gamma_l \hat{C}_G} \tilde{R}_y(\vec{\theta} - \pi/2) |0\rangle^{\otimes n}. \quad (12)$$

$m = 2, 3$ case: QAOA-warmest QAOA-warmest 接受 $m = 2, 3$ 的松弛问题 2 的解, 其混合哈密顿量定义为:

$$\hat{B}^{\text{warm}}(\{c_j\}) = - \sum_j (c_{xj} X_j + c_{yj} Y_j + c_{zj} Z_j), \quad (13)$$

其中, $c_{xj} = \sin \theta_j \cos \phi_j, c_{yj} = \sin \theta_j \sin \phi_j, c_{zj} = \cos \theta_j$, 它的角度参数 $(\vec{\theta}, \vec{\phi})$ 编码了输入的温启动态, 可通过经典的 BM-MC $_\kappa$ 算法获得。输入的初始态应当为 $\hat{B}^{\text{warm}}(\{c_j\})$ 的基态

$$|\psi_0^{\text{warm}}\rangle = \bigotimes_j \left(\cos \frac{\theta_j}{2} |0\rangle + e^{i\phi_j} \sin \frac{\theta_j}{2} |1\rangle \right). \quad (14)$$

举例考虑 $\kappa = 2$ 的情况, 此时在球面上, 则 $\phi_j = 0$, 将 θ_j 替换为 $\pi/2 - \theta_j$, 则有 $c_{xj} = \cos \theta_j, c_{zj} = \sin \theta_j$ 则每个 \mathbf{x}_j 可表示为单位圆上的点 $\mathbf{x}_j = (\cos \theta_j, \sin \theta_j)$, 在通过前文提到的 BM-MC $_2$ 算法 2, 求解获得最优的 $\vec{\theta}_{\text{opt}}$ 后, 随机选择 $\vec{\theta}_{\text{opt}}$ 中的一个分量, 然后将所有角度平移该值得到 θ_j 。定义 $\theta_j = \alpha_j - \pi/2$, 则混合哈密顿量和初始态为:

$$H_M^{\text{warm}} = \sum_j (\cos \alpha_j X_j - \sin \alpha_j Z_j) = \tilde{R}_y(\vec{\alpha}) \hat{B} \tilde{R}_y(-\vec{\alpha}), \quad (15)$$

$$|\psi_0^{\text{warm}}\rangle = \bigotimes_j \left(\sin \left(\frac{\alpha_j}{2} + \frac{\pi}{4} \right) |0\rangle_j - \cos \left(\frac{\alpha_j}{2} + \frac{\pi}{4} \right) |1\rangle_j \right) \quad (16)$$

$$= \tilde{R}_y(\vec{\alpha} - \frac{\pi}{2}) |0\rangle^{\otimes n}. \quad (17)$$

设定 $\vec{\delta} = \vec{\alpha} - \pi/2 = \vec{\theta}$, 则 p 层的 QAOA-warmest 最终输出如下 $(\vec{\gamma}, \vec{\beta})$ 参数化的量子态, 并对参数 $(\vec{\gamma}, \vec{\beta})$ 做梯度下降:

$$|\psi_f^{\text{warm}}\rangle = \prod_{l=1}^p \tilde{R}_y(\vec{\alpha}) e^{-i\beta_l \hat{B}} \tilde{R}_y(-\vec{\alpha}) e^{-i\gamma_l \hat{C}_G} \tilde{R}_y(\vec{\delta}) |0\rangle^{\otimes n}. \quad (18)$$

2.2.3 Adaptive bias field QAOA

ab-QAOA 的思路是除了 QAOA 的参数 $\gamma, \beta \in [0, 2\pi]^p$ 外, 还引入变量 $\mathbf{h} = \{h_j\} \in [0, 2\pi]^n$, 在优化的每一步, 通过计算 $\langle Z_j \rangle$, 算法评估每个量子比特 j 是更倾向于处于 $|0\rangle$ 态还是 $|1\rangle$ 态。偏置 h_j 会根据这个倾向动态地加强这个趋势。即如果第 j 个比特贴近 $|1\rangle$, 那么 h_j 会减小, 使得 $e^{iB(\alpha)\beta}$ 把它往 $|1\rangle$ 的方向上推。为了实现这个目的, 可以将原算法中的 \hat{B} 替换为含参的

$$\hat{B}(\{h_j\}) := \sum_j \frac{X_j - h_j Z_j}{\sqrt{1 + h_j^2}} = \tilde{R}_y(\vec{\alpha}) \hat{B} \tilde{R}_y(-\vec{\alpha}) \quad (19)$$

其中 $\alpha_j = \arcsin(h_j / \sqrt{1 + h_j^2})$ 。为了保证算法的收敛性, 修改输入态为混合哈密顿量 $\hat{B}(\{h_j\})$ 的基态 $|\psi_0^{\text{ab}}\rangle = \tilde{R}_y(\vec{\alpha}) |-\rangle^{\otimes n}$, 最终 p 层的 ab-QAOA 输出如下 $(\vec{h}, \vec{\gamma}, \vec{\beta})$ 参数化的量子态

$$|\psi_f^{\text{ab}}\rangle = \prod_{k=1}^p e^{-i\beta_k \hat{B}(\{h_j\})} e^{-i\gamma_k \hat{C}_G} |\psi_0^{\text{ab}}(\{h_j\})\rangle. \quad (20)$$

一个合理的变量 \mathbf{h} 的更新格式为 $h_j = (1 - l)h_j + l \langle \psi_f^{\text{ab}} | Z_j | \psi_f^{\text{ab}} \rangle$, 其中 l 为学习率, 期望值 $\langle Z_j \rangle$ 可以通过测量得到。为了防止贫瘠高原问题, 有的算法会通过更新 $(\vec{\gamma}, \vec{\beta})$ 的傅里叶变换, 这总结于算法 3 [Yu+22]。

Algorithm 3 ab-QAOA in level p

- 1: Initialize p 元集合 $\{u_l\}$ 和 $\{v_l\}$ 用于构造 $\{\gamma_k\}$ 和 $\{\beta_k\}$;
- 2: 初始化局部场参数 $\{h_j\}$;
- 3: 设置学习率 η ;
- 4: **while** 未收敛 **do**
- 5: 通过傅里叶变换设置:

$$\gamma_k = \sum_{l=1}^p u_l \sin \left[\left(\frac{l-1}{2} \right) \left(\frac{k-\frac{1}{2}}{p} \right) \pi \right], \beta_k = \sum_{l=1}^p v_l \cos \left[\left(\frac{l-1}{2} \right) \left(\frac{k-\frac{1}{2}}{p} \right) \pi \right]$$
- 6: 构造混合哈密顿量 $H_M^{\text{ab}}(\{h_j\}) = \sum_{j=1}^n (X_j - h_j Z_j)$
- 7: 准备初始态 $|\psi_0^{\text{ab}}\rangle$, 即 H_M^{ab} 的基态;
- 8: 计算最终态 $|\psi_f^{\text{ab}}\rangle = \prod_{k=1}^p e^{-i\beta_k H_M^{\text{ab}}(\{h_j\})} e^{-i\gamma_k \hat{C}_G} |\psi_0^{\text{ab}}\rangle$
- 9: 测量梯度与偏差 $\frac{\partial \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle}{\partial u_l}, \frac{\partial \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle}{\partial v_l}, \delta h_j = h_j - \langle \psi_f^{\text{ab}} | Z_j | \psi_f^{\text{ab}} \rangle$
- 10: 使用 Adam 优化器更新 $\begin{cases} u_l, v_l \leftarrow \text{AdamStep}(u_l, v_l) \\ h_j \leftarrow h_j - \eta \delta h_j \end{cases}$
- 11: 计算当前能量 $E(\{u_l\}, \{v_l\}, \{h_j\}) = \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle$
- 12: **end while**
- 13: 输出最终能量 $E_f(\{u_l\}, \{v_l\}, \{h_j\})$ 并测量 $|\psi_f^{\text{ab}}\rangle$

2.2.4 WS-ab-QAOA

WS-ab QAOA 在 ab-QAOA 的基础上引入了初始热化的可能性, 算法把 $|0\rangle^{\otimes n}$ 的输入态更改为 GW 算法求得的近似解 $|\mathbf{s}\rangle$ 。同样地, 为了保证算法从 $\hat{B}(\alpha)$ 的基态开始, 需要对应更改初始化矩阵。注意到通过设定 $\alpha_j^{(\mathbf{s})} = \begin{cases} \pi/2 & \text{if } s_j = 0 \\ -\pi/2 & \text{if } s_j = 1 \end{cases}$, 有恒等式 $|\mathbf{s}\rangle = e^{i\left(\frac{\pi}{4}-\frac{\alpha_j^{(\mathbf{s})}}{2}\right)Y_j} |0\rangle^{\otimes n}$,

则 WS-ab-QAOA 线路在给定某个经典算法的近似解 \mathbf{s} 后, 输出如下 $(\vec{\alpha}, \vec{\gamma}, \vec{\beta})$ 参数化的量子态

$$|\vec{\alpha}, \vec{\gamma}, \vec{\beta}, G\rangle := \left(e^{-i\beta_p \hat{B}(\alpha)} e^{-i\gamma_p \hat{C}_G} \right) \times \cdots \times \left(e^{-i\beta_1 \hat{B}(\alpha)} e^{-i\gamma_1 \hat{C}_G} \right) \times \left(\prod_j e^{-i\frac{\alpha_j - \alpha_j^{(\mathbf{s})}}{2} Y_j} \right) |\mathbf{s}\rangle \quad (21)$$

$$= \prod_{l=1}^p \tilde{R}_y(\vec{\alpha}) e^{-i\beta_l H_M^{\mathbf{s}}} \tilde{R}_y(-\vec{\alpha}) e^{-i\gamma_l H_C} \tilde{R}_y(\vec{\delta}) |\mathbf{s}\rangle, \quad (22)$$

其中 $\gamma_i, \beta_i \in [0, 2\pi)$ 是执行梯度下降法的 p 维向量, $\delta = \alpha - \alpha^{(\mathbf{s})}$ 是涉及偏置场参数和描述初始热化的 n 维向量。它们和 ab-QAOA 的表达式 19一致:

$$\hat{B}(\alpha) := \sum_j \frac{X_j - h_j Z_j}{\sqrt{1 + h_j^2}}, \quad \alpha_j = \arcsin\left(\frac{h_j}{\sqrt{1 + h_j^2}}\right). \quad (23)$$

特别地，若取 $s = 0$ 作为输入解，则退化为上文的 ab-QAOA 态：

$$\left| \vec{\alpha}, \vec{\gamma}, \vec{\beta}, G \right\rangle := \left(e^{-i\beta_p \hat{B}(\alpha)} e^{-i\gamma_p \hat{C}_G} \right) \times \cdots \times \left(e^{-i\beta_1 \hat{B}(\alpha)} e^{-i\gamma_1 \hat{C}_G} \right) \times \left(\prod_j e^{-i\frac{\alpha_j}{2} Y_j} \right) e^{i\frac{\pi}{4} Y} |0\rangle^{\otimes n} \quad (24)$$

WS-ab QAOA 是否会带来优势呢？利用下面的引理可以证明 $p = 1$ 的 WS-ab-QAOA 的效果 [Yu+25]

Lemma 1. 对 Level-1 WS-ab-QAOA，当图是 r 正则图且 r 为奇数时，若参数满足 $\beta_1 = \pi/2, \gamma_1 = \pi$ ，以及 $\alpha_j = \begin{cases} -\pi/6 & \text{if } s_j = 0 \\ \pi/6 & \text{if } s_j = 1 \end{cases}$ ，则生成的态将正好是 $|s\rangle$

在算法 4 执行过程中，先通过梯度下降最小化 $\langle \hat{C}_G \rangle$ 来更新 α, β 。而变量 \mathbf{h} （或者等价地， α ）不参与梯度下降，它的更新格式仿照 Adam 算法写为：

$$h_j^{(t)} \leftarrow \text{AdamStep}(h_j^{(t-1)}) \Leftrightarrow \begin{cases} g_{h_j}^{(t)} = h_j^{(t)} + \frac{\sqrt{3}}{3} \langle Z_j \rangle^{(t)}, \\ m_{h_j}^{(t)} = b_1 m_{h_j}^{(t-1)} + (1 - b_1) g_{h_j}^{(t)}, \\ v_{h_j}^{(t)} = b_2 v_{h_j}^{(t-1)} + (1 - b_2) g_{h_j}^{(t)2}, \\ h_j^{(t)} = h_j^{(t-1)} - \ell_t \frac{m_{h_j}^{(t)}}{\sqrt{v_{h_j}^{(t)} + \epsilon}}, \\ \ell_t = \ell_0 \frac{\sqrt{1 - b_2^t}}{(1 - b_1^t)} \end{cases} \quad (25)$$

2.3 QAOA 算法的模拟方法

如果对于和较浅的深度 p 的 QAOA 问题，Pauli 算符的传播始终在经典可有效模拟的范围内，那么可以利用张量网络或者 Clifford perturbation theory 在经典计算机上计算能量均值和梯度，进而完成大部分参数优化，最后只需要量子计算机制备最终态并采样。在这种情况下，我们可以模拟 QAOA 算法。为简单起见，考虑无权重的 u3r 图 $G = (E, V)$ ，设定 $p = 1$ 的 WS-ab-QAOA 算法，此时均值 $\langle Z_j \rangle$ 和 $\langle Z_j Z_k \rangle$ 均可高效计算，且仅涉及到一个步长的局域图，在三正则无向图中，仅有三种步长为 1 的局域结构，如图 2。计算可得均值

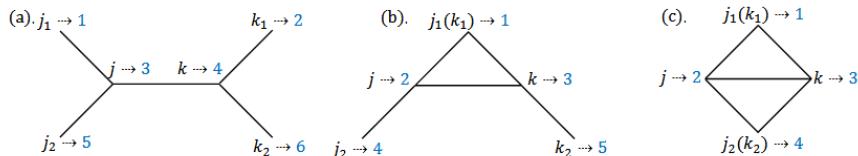


图 2：三正则无向图的所有三种步长为 1 的局域结构 [Yu+25]

Algorithm 4 WS-ab QAOA in level p

- 1: Initialize $\{\gamma_k\}$ 和 $\{\beta_k\}$ 和局部场参数 $\{h_j\}$;
- 2: 设置学习率 η ;
- 3: while 未收敛 do
 - 4: 构造混合哈密顿量 $H_M^{\text{ab}}(\{h_j\}) = \sum_j \frac{X_j - h_j Z_j}{\sqrt{1+h_j^2}}$
 - 5: 准备初始态 $|s\rangle$, 并计算 $\alpha_j^{(s)} = \begin{cases} \pi/2 & \text{if } s_j = 0 \\ -\pi/2 & \text{if } s_j = 1 \end{cases}$;
 - 6: 计算最终态 $|\psi_f^{\text{ab}}\rangle = \prod_{k=1}^p e^{-i\beta_k H_M^{\text{ab}}(\{h_j\})} e^{-i\gamma_k \hat{C}_G} |\psi_0^{\text{ab}}\rangle$
 - 7: 测量梯度与偏差 $\frac{\partial \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle}{\partial \gamma_k}, \frac{\partial \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle}{\partial \beta_k}$
 - 8: 使用 Adam 优化器更新 $\begin{cases} u_l, v_l \leftarrow \text{AdamStep}(u_l, v_l) \\ h_j \leftarrow \text{AdamStep}(h_j) \end{cases}$
 - 9: 计算当前能量 $E(\{u_l\}, \{v_l\}, \{h_j\}) = \langle \psi_f^{\text{ab}} | \hat{C}_G | \psi_f^{\text{ab}} \rangle$
- 10: end while
- 11: 输出最终能量 $E_f(\{u_l\}, \{v_l\}, \{h_j\})$ 并测量 $|\psi_f^{\text{ab}}\rangle$

满足如下公式 [Yu+25]:

$$\begin{aligned} \langle \hat{C}_G \rangle &= \sum_{(j,k) \in E} (C_j^Z C_k^Z E_{Z_j Z_k} + C_j^Z C_k^Y E_{Z_j Y_k} + C_j^Y C_k^Z E_{Y_j Z_k} \\ &\quad + C_j^Z C_k^X E_{Z_j X_k} + C_j^X C_k^Z E_{X_j Z_k} + C_j^Y C_k^Y E_{Y_j Y_k} \\ &\quad + C_j^X C_k^X E_{X_j X_k} + C_j^Y C_k^X E_{Y_j X_k} + C_j^X C_k^Y E_{X_j Y_k} - \frac{1}{2}) \end{aligned} \quad (26)$$

$$\langle Z_j \rangle = C_j^Z E_{Z_j} + C_j^Y E_{Y_j} + C_j^X E_{X_j}, \quad (27)$$

其中参数 $\vec{\delta}$ 由表达式 (21) 确定, $\begin{cases} C_j^Z = (1 - 2 \cos^2 \alpha_j \sin^2 \beta_1), \\ C_j^Y = \cos \alpha_j \sin 2\beta_1, \\ C_j^X = -\sin 2\alpha_j \sin^2 \beta_1, \end{cases}$ 且

$$\begin{cases} E_{O_j O_k} = \langle s | \tilde{R}_y(-\vec{\delta}) e^{i\gamma_1 \hat{C}_G} O_j O_k e^{-i\gamma_1 \hat{C}_G} \tilde{R}_y(\vec{\delta}) | s \rangle \\ E_{O_j} = \langle s | \tilde{R}_y(-\vec{\delta}) e^{i\gamma_1 \hat{C}_G} O_j e^{-i\gamma_1 \hat{C}_G} \tilde{R}_y(\vec{\delta}) | s \rangle \end{cases} \quad (28)$$

通过在不同的局域结构下计算 $E_{O_j O_k}, E_{O_j}$, 可以求得均值, 进而实施 Adam 更新与简单的梯度下降法。

3 研究内容和研究目标

本项目旨在基于 MindSpore Quantum 量子计算框架，对近期在 arXiv:2503.20048v1 [Yu+25] 中提出的热启动自适应偏置量子近似优化算法（WS-ab-QAOA）进行复现与拓展。具体开发内容如下：

算法复现与实现：

- 在 MindSpore Quantum 框架内，完整实现 WS-ab-QAOA 算法的核心逻辑，包括：
 - 用于热启动的经典解生成模块（可集成或调用 GW 算法实现）。
 - 将经典解编码为量子初始态的旋转门序列。
 - 包含自适应偏置场的混合哈密顿量 $\hat{B}(\alpha)$ 的构建。
 - 代价哈密顿量 \hat{C}_G 的定义（针对最大割问题）。
 - QAOA 的参数化量子线路（ansatz）构建。
 - 自适应偏置场 h_j 的迭代更新机制。
 - 变分参数 $\{\gamma_k, \beta_k\}$ （或其傅里叶分量）的优化循环。
- 确保算法能够应用于解决最大割（MaxCut）问题，支持从标准图表示（如邻接矩阵或边列表）自动生成代价哈密顿量。

性能验证与对比：

- 选择论文中使用的图实例（u3r 图）或标准的最大割问题测试集进行算法性能验证。
- 复现论文中的关键性能指标，如在不同规模图上找到的割值（近似比）。
- 将实现的 WS-ab-QAOA 算法与论文中提及的对比算法进行性能比较，这些对比算法应至少包括：(i). 标准 QAOA,(ii). 仅使用热启动的 WS-QAOA,(iii). Goemans-Williamson (GW) 经典算法（包括其通过多次采样改进解的策略）。
- 目标是验证并展现 WS-ab-QAOA 相对于这些对比算法，在求解最大割问题时，能够在近似解质量、收敛速度或达到同等解质量所需的计算资源（如量子线路深度、迭代次数、经典采样次数）方面取得显著的性能提升。

拓展与分析：

- 探究不同热启动策略（如 SG3 等经典算法 [MKB24] 生成的初始解）对 WS-ab-QAOA 性能的影响。
- 分析自适应偏置场的更新规则、学习率等超参数对算法收敛性和最终解质量的敏感度。

- 尝试将算法应用于不同类型或更大规模的图实例，评估其可扩展性。
- 基于 MindSpore Quantum 的自动微分功能，优化梯度计算和参数更新过程，提升训练效率。

代码与文档：

- 提交结构清晰、注释良好、可复现的 MindSpore Quantum 代码。
- 撰写详细的项目报告，包括算法原理、实现细节、实验设置、结果分析、性能对比以及遇到的问题与解决方案。

4 项目开发时间计划及预期结果

阶段	时间范围	主要任务与预期结果
第一阶段	07/01-07/15	<p>搭建开发环境: 安装 MindSpore Quantum、依赖库及必要工具。</p> <p>文献与需求分析: 深入研读 arXiv:2503.20048v1 中 WS-ab-QAOA 相关内容，明确算法模块划分。</p> <p>集成经典解生成模块: 实现或调用 Goemans–Williamson (GW) 算法，确保能够从输入图生成近似割的经典解。</p> <p>预期成果: 完成开发环境搭建，掌握 MindSpore Quantum 基本用法；实现 GW 算法接口并能输出符合格式要求的经典解向量。</p>
第二阶段	07/16-07/31	<p>编码量子初始态: 根据经典解结果设计旋转门序列，将经典割向量映射为量子态。</p> <p>构建混合哈密顿量: 按照式(19)实现含偏置场的混合哈密顿量 $\hat{B}(\{h_j\})$。</p> <p>定义代价哈密顿量: 针对最大割问题，用图的邻接矩阵自动生成 \hat{C}_G。</p> <p>搭建 QAOA 参数化线路: 实现包含自适应偏置场的量子线路框架，能够在 MindSpore Quantum 中正确编译并执行。</p> <p>预期成果: WS-ab-QAOA 核心逻辑在框架内可运行，能够根据输入图自动构造相应的参数化量子线路并输出量子态。</p>
第三阶段	08/01-08/20	<p>性能验证（复现论文结果）: 选择文献中使用的无权重 u3r 图或其他标准的 MaxCut 测试集，运行 WS-ab-QAOA 并记录割值、近似比等指标。</p> <p>基准对比实验: 分别运行标准 QAOA、WS-QAOA (仅热启动)、GW 算法 (多次采样优化) 并收集相同指标。</p> <p>数据分析与可视化: 对比各算法在不同规模图上的近似比、收敛速度、量子电路深度与经典采样次数，形成数据表和曲线图。</p> <p>预期成果: 生成完整的实验数据集及分析报告，验证 WS-ab-QAOA 在近似效果、收敛速度等方面相较基准算法的性能提升。</p>
第四阶段	08/21-09/10	<p>拓展实验: 尝试其他经典热启动策略 (如 SG3 等算法 [MKB24])，比较不同热启动对 WS-ab-QAOA 性能的影响。</p> <p>超参数敏感性分析: 分析学习率、偏置场更新规则等对收敛性与最终解质量的影响，绘制敏感度曲线。</p> <p>大规模与不同类型图测试: 将算法应用于更大规模图实例 (如随机稀疏图或具有特殊结构的图)，评估算法可扩展性。</p> <p>梯度计算优化: 利用 MindSpore Quantum 的自动微分功能，加速梯度计算与更新循环，记录优化前后运行时对比。</p> <p>预期成果: 得到不同热启动和超参数设置下的性能对比结论，评估算法在更大规模实例上的可行性；优化后的梯度计算方案显著提高训练效率。</p>
第五阶段	09/11-09/30	<p>代码重构与完善: 根据前期实验结果和反馈，对所有模块进行重构，补充注释、完善接口，并加入单元测试。</p> <p>文档与报告撰写: 编写详细的项目报告，包括算法原理介绍、实现细节、实验设置与结果分析、性能对比图表、遇到的问题与解决方案。</p> <p>代码托管与演示: 整理可复现代码并上传至公共仓库，生成示例脚本与 README，撰写使用说明与快速上手指南。</p> <p>预期成果: 产出结构清晰、注释完备的 MindSpore Quantum 代码库；完成不少于 20 页的中英文对照项目报告；生成可直接运行的示例脚本与演示视频素材。</p>

表 1: 2025 年 07 月 01 日至 09 月 30 日项目开发计划与预期结果

5 参考文献

References

- [FGG14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A quantum approximate optimization algorithm”. In: *arXiv preprint arXiv:1411.4028* (2014).
- [GW95] Michel X Goemans and David P Williamson. “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming”. In: *Journal of the ACM (JACM)* 42.6 (1995), pp. 1115–1145.
- [Mei+17] Song Mei et al. “Solving SDPs for synchronization and MaxCut problems via the Grothendieck inequality”. In: *Conference on learning theory*. PMLR. 2017, pp. 1476–1515.
- [MKB24] Manuel H. Muñoz-Arias, Stefanos Kourtis, and Alexandre Blais. “Low-depth Clifford circuits approximately solve MaxCut”. In: *Physical Review Research* 6.2 (June 2024). ISSN: 2643-1564. DOI: 10.1103/physrevresearch.6.023294. URL: <http://dx.doi.org/10.1103/PhysRevResearch.6.023294>.
- [Yu+22] Yunlong Yu et al. “Quantum approximate optimization algorithm with adaptive bias fields”. In: *Physical Review Research* 4.2 (2022), p. 023249.
- [Yu+25] Yunlong Yu et al. *Warm Start Adaptive-Bias Quantum Approximate Optimization Algorithm*. 2025. arXiv: 2503.20048 [quant-ph]. URL: <https://arxiv.org/abs/2503.20048>.