近乎完美可达性：采用深度为1的试探电路的变分量子搜索

Junpeng Zhan[1]*

摘要

Grover搜索算法以其在解决许多重要科学问题时带来的显著加速而闻名。近期提出的变分量子搜索（VQS）算法已在最多26个量子比特的情况下展现出相较于Grover算法的指数级优势。然而，对于更多量子比特的情形，其优势尚未得到证明。我们在此证明，对于任意多于五个量子比特的情形，Grover算法所需的指数级深度电路可由一个多控非门以及单层R_y门，或由两层仅含Hadamard门和非门的电路取代。我们证明，以单层R_y门作为试探态的VQS，在从任意规模的无结构数据集中寻找目标元素时具有近乎完美的可达性；其可达性会随量子比特数增加而呈指数级提升。这里，可达性用于量化给定试探态生成最优量子态的能力。数值研究进一步验证了VQS出色的可达性。证明采用深度为1的试探电路的VQS对于任意数量的量子比特都具有近乎完美的可达性，是证明其对于任意数量的量子比特都比Grover算法具有指数级优势这一关键步骤；后者的证明意义重大，因为这意味着VQS能够高效求解NP完全问题。

1. 引言

量子算法大致可分为两类，具体取决于它们是基于 Grover搜索算法还是量子傅里叶变换[1]。Grover算法[2,3] 可在无结构搜索中实现二次加速，并有许多重要应用[1]。然而，Grover算法[2,3] 所需的量子电路深度会随量子比特数呈指数增长。为解决这一问题，近期提出了一种名为变分量子搜索（VQS）[4] 的算法，它能够通过浅层电路将目标元素的总概率放大到接近 1；该电路深度随量子比特数线性增长。受 GPU内存限制，目前已验证至 26 个量子比特。也就是说，就电路深度而言，VQS 相比 Grover算法具有指数级优势。

VQS 是一种变分量子算法（VQA）[5–7]。从不同方面研究变分量子算法的工作很多，例如可训练性[8–11]、表达能力[12–14]和可达性[15,16]。然而，尚未见到针对 VQS 的类似分析，但这类分析对于验证 VQS 在任意量子比特数下的优势至关重要，因为这关系到 VQS 能否高效求解 NP 完全问题[17]。

可达性讨论的是，带有参数的 VQA 试探态表示能够最小化给定目标函数[8]的量子态的能力。QAOA[15,16,18] 和变分Grover搜索[16,19] 的可达性已有研究。本文聚焦于无结构搜索问题中 VQS 的可达性，并证明 Grover算法中指数深度的电路可以替换为由一层 $R_y(\theta)$ 门构成的电路，或一个双层电路

[1] 可再生能源工程系，阿尔弗雷德大学，美国纽约州阿尔弗雷德。*电子邮箱：zhanj@alfred.ed u

该双层电路由 Hadamard 门和 X 门构成。此外，我们的数值研究验证了 VQS 在求解无结构搜索问题方面的有效性，其试探态为由一层 $R_y(\theta)$ 门构成的电路。本文其余部分包括方法、结果和结论章节。

2. 方法

本文要解决的问题是在无结构数据集D中找到目标元素，其中只有一个目标元素和(2^n-1)个非目标元素。我们使用 n 个量子比特和 n 个 Hadamard门来生成所有 2^n 个元素的均匀叠加态，如图 1a 中最左侧红色虚线左边所示。本节其余部分将提出三种解决该问题的方法：1）基于 HX 层的量子电路（见图 1a、b），由算法 1（第 2.1 节）生成；2）基于 R_y 层的量子电路（见图 1c、d），由算法2（第 2.2 节）生成；3）将 R_y 层作为试探态的变分量子求解器（第 2.4 节）。第 2.4 节还讨论变分量子求解器的可达性。第 2.3 节阐明算法 1 和算法 2 的可扩展性。

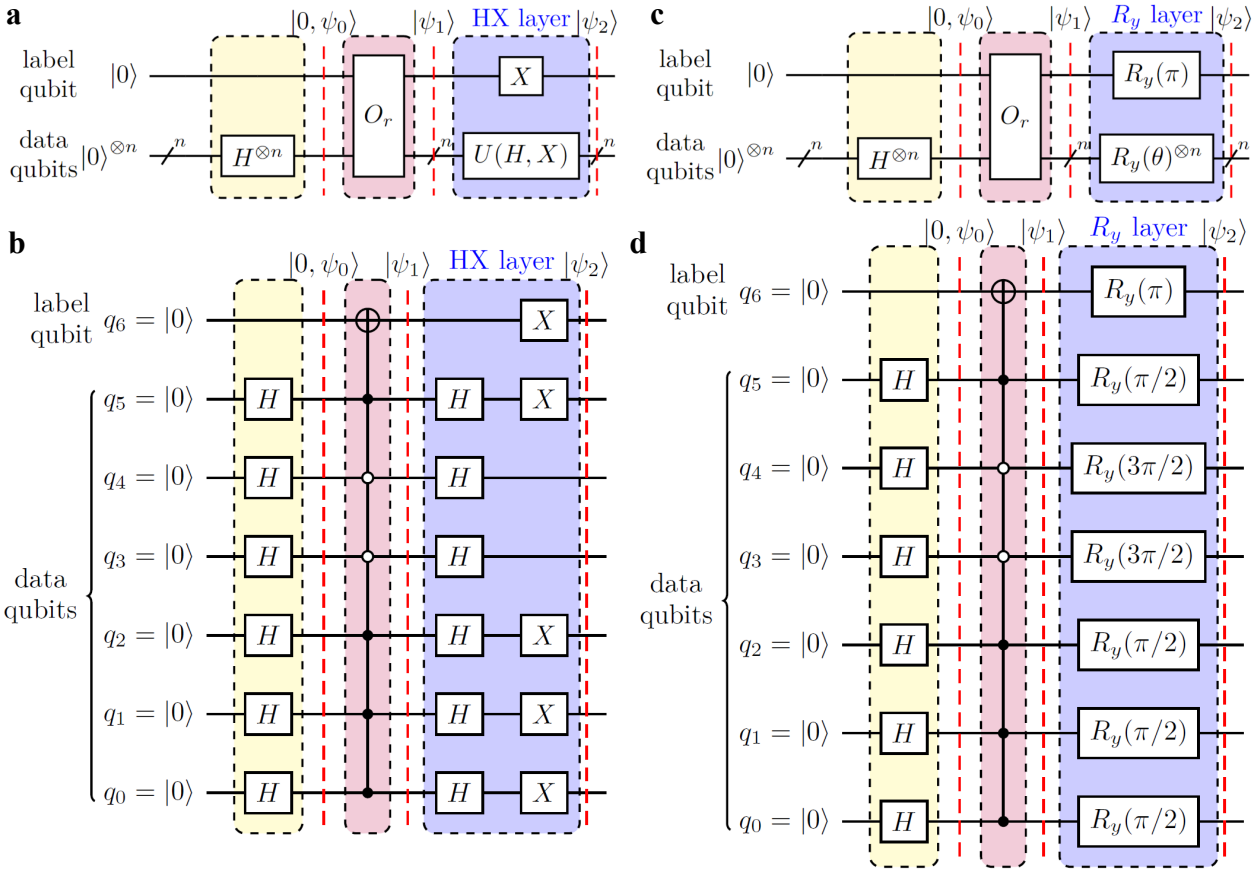

图 1. 用于生成数据集 D 并将其中唯一目标元素的概率放大到接近 1 的量子电路。**a,c,**分别为使用 HX层和 R_y 层处理n量子比特数据的紧凑形式电路（图 1a 和 1c 中的蓝色模块），其中***HX***层由 Hadamard 门和X门组成，而 R_y 层由 $R_y(\theta)$ 门组成。**b,d**为n=6

分别使用 HX 层和 R_y 层。黄色模块（不包括标签量子比特）生成一个量子态，该量子态是所有元素（一个目标元素和 (2^n-1) 个非目标元素）的均匀叠加态，即所有元素的初始概率相同。红色模块（预言机）在标签量子比特上为目标元素提供标签 $|1\rangle$，并为所有非目标元素提供 $|0\rangle$。蓝色模块将目标元素的概率放大到接近 1。标签量子比特是最高位（最高有效位）。在子图 a 中，$U(H,X)$ 表示由 Hadamard 门和 X 门组成的电路。在子图 b、d 中，$n=6$ 且目标元素的索引为 39（二进制形式为 100111）。

2.1 设计基于 HX 层的电路

本节分为四个部分。第一部分详细介绍用于生成 HX 层（图 1a、b 中的蓝色模块）的算法，即算法 1。第二部分给出图 1a、b 所示基于 HX 层的电路中的三个关键量子态的向量形式。第三部分阐述 HX 层的设计目标。最后一部分证明，对于任意数量的量子比特，算法 1 始终能够实现 HX 层的设计目标。

算法 1。本文提出算法 1 来构造所谓的 HX 层，即仅由 Hadamard 门和 X 门组成的双层电路。HX 层与预言机（图 1 中的红色模块）结合后，具有与 Grover搜索算法相同的功能，即将目标元素的概率放大至接近 1。

算　　算法 1 | 用于生成 HX 层的伪代码（图 1a 中的蓝色模块）　　.

输入：量子比特数n以及目标元素的十进制形式索引k，$\forall k \in [0, 2^n-1]$

输出：HX层中的量子门（图 1a 中的蓝色模块）。1 将k转换为二进制形式 $b_{n-1}b_{n-2}\cdots b_2b_1b_0$。2 在标签量子比特（最高有效位量子比特）上添加一个X门。3 令 m=n4 **当** $m \geq 1$5　　$b_{m-1}=1$ **如果** 在量子比特 q_{m-1} 上添加一个哈达玛门，随后 6 添加一个X门。7 **否则**8　　在量子比特 q_{m-1} 上添加一个哈达玛门。9　　$m \leftarrow m-1$

三个量子态的向量形式。本小节详细介绍三个量子态的向量形式，即 $|0,\psi_0$、$|\psi_1$ 和 $|\psi_2$，它们分别由图 1a 中的三条红色虚线表示。

量子态 $|0,\psi_0$ 可以写为 $|0\rangle \bigotimes(|0\rangle+|1\rangle)^{\otimes n}$，其向量形式为：

$$|0,\psi_0\rangle = [\underbrace{\alpha_0^{\mathrm{b}}, \alpha_1^{\mathrm{b}}, \cdots, \alpha_{k-1}^{\mathrm{b}}, \alpha_k^{\mathrm{g}}, \alpha_{k+1}^{\mathrm{b}}, \cdots, \alpha_{N-1}^{\mathrm{b}}}_{\text{1st half: } N \text{ elements}}, \underbrace{0,0,\cdots,0}_{\text{2nd half: } N \text{ elements}}]^T \quad (1)$$

其中，$N=2^n$，上标 b 和 g 分别表示坏元素和目标元素，下标0~N−1表示向量中元素的索引。请注意，本文其余部分的索引均从 0 开始计数。例如，α_k^{g} 表示第k个元素是目标元素。

$|\psi_1\rangle$与$|0,\psi_0\rangle$之间的关系可表示为：

$$|\psi_1\rangle = O_r|0,\psi_0\rangle = [\underbrace{\alpha_0^{\mathrm{b}},\alpha_1^{\mathrm{b}},\cdots,\alpha_{k-1}^{\mathrm{b}},0,\alpha_{k+1}^{\mathrm{b}},\cdots,\alpha_{N-1}^{\mathrm{b}}}_{\text{1st half: } N \text{ elements}},\underbrace{0,\cdots,0,\alpha_{N+k}^{\mathrm{g}},0,\cdots,0}_{\text{2nd half: } N \text{ elements}}]^T \quad (2)$$

其中，预言机O_r 由$C^n(X)$实现，它是一个受n个量子比特控制的X门，如图 1b 所示。如式（2）所示，$C^n(X)$将目标元素的索引从k变为N+k。

HX层（图 1a 中的蓝色模块）可表示为

$$X \otimes U(H,X) = \begin{bmatrix} \mathbf{0} & U(H,X) \\ U(H,X) & \mathbf{0} \end{bmatrix} \quad (3)$$

其中，$U(H,X)$是一个N×N矩阵。为便于表述，我们将式（3）中的矩阵称为***HX***层矩阵。那么，$|\psi_2\rangle$与$|\psi_1\rangle$之间的关系可表示为：

$$\begin{aligned} |\psi_2\rangle &= \begin{bmatrix} \mathbf{0} & U(H,X) \\ U(H,X) & \mathbf{0} \end{bmatrix} |\psi_1\rangle \\ &= \begin{bmatrix} \mathbf{0} & U(H,X) \\ U(H,X) & \mathbf{0} \end{bmatrix} [\underbrace{\alpha_0^{\mathrm{b}},\alpha_1^{\mathrm{b}},\cdots,\alpha_{k-1}^{\mathrm{b}},0,\alpha_{k+1}^{\mathrm{b}},\cdots,\alpha_{N-1}^{\mathrm{b}}}_{\text{1st half: } N \text{ elements}},\underbrace{0,\cdots,0,\alpha_{N+k}^{\mathrm{g}},0,\cdots,0}_{\text{2nd half: } N \text{ elements}}]^T \end{aligned} \quad (4)$$

为便于分析，我们将$|\psi_2\rangle$写成向量形式：

$$|\psi_2\rangle = [\beta_0,\beta_1,\cdots,\beta_{N-1},\beta_N,\cdots,\beta_{2N-1}]^T \quad (5)$$

其中

$$\sum_{i=0}^{2N-1}|\beta_i|^2 = 1 \quad (6)$$

HX层的设计目标。 设计$U(H,X)$的**目标**是让其第k 行中的每个元素都满足$1/\sqrt{N}$，这样β_{N+k}，即$|\psi_2$的第（$N+k$）个元素，就可以计算为：

$$\beta_{N+k} = \left(\sum_{i=0}^{k-1}\alpha_i^{\mathrm{b}} + \sum_{i=k+1}^{N-1}\alpha_i^{\mathrm{b}}\right)/\sqrt{N} \quad (7)$$

其中，右侧由公式 (4) 得出，即将HX层矩阵的第（$N+k$）行与$|\psi_1$的向量形式相乘，并乘以整数$k \in [0,2^n-1]$。

从这里开始，我们假设所有元素的初始幅度相同。也就是说，$\alpha_i^{\mathrm{b}} = \alpha_k^{\mathrm{g}} = \alpha_{N+k}^{\mathrm{g}} = 1/\sqrt{N},\ \forall i \in [0,N-1]$。然后，（7）可以改写为：

$$\beta_{N+k} = \left(1/\sqrt{N}\right)(N-1)/\sqrt{N} = 1-1/N = 1-1/2^n \quad (8)$$

获得目标元素的概率等于 $\beta_{N+k}^2 =$ （$1-1/2^n$）^{2}, , 对于 n，其分别等于 0.25、0.5625、0.7656、0.8789、0.9386、0.9690、0.9844、0.9922 和 0.9961。也就是说，当 n 分别大于 5 和 7 时，找到目标元素的概率分别大于 0.95 和 0.99。

HX层的全 1 行的位置索引。 在本小节的其余部分，我们将回答这样一个问题：为什么算法 1 生成的 HX层能够实现上一小节中规定的设计目标。

张量积 $\begin{bmatrix} a_n, & a_n \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_{n-1}, & a_{n-1} \\ 1, & 0 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} a_2, & a_2 \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_1, & a_1 \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_0, & a_0 \\ 1, & 0 \end{bmatrix}$ 的结果是一个包含 2^{n+1} 个元素的列向量。元素 $a_{n,i_n}a_{n-1,i_{n-1}}\cdots a_{2,i_2}a_{1,i_1}a_{0,i_0}$ 在该列向量中的位置索引等于二进制形式 $i_n i_{n-1}\cdots i_2 i_1 i_0$ 的十进制值，其中 $i_{r_0}\in\{0,1\}$, $\forall\, r \in [0,n]$。为了更好地理解这一点，我们提供两个例子：在与 $\begin{bmatrix} a_3, & a_3 \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_2, & a_2 \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_1, & a_1 \\ 1, & 0 \end{bmatrix} \otimes \begin{bmatrix} a_0, & a_0 \\ 1, & 0 \end{bmatrix}$ 相关的列向量中，元素 $a_{3,0}a_{2,0}a_{1,0}a_{0,}1$ 的位置索引为 1（二进制形式为 0001），而同一向量中 $a_{3,1}a_{2,0}a_{1,0}a_{0,}1$ 的位置索引为 9（二进制形式为 1001）。

我们可以将张量积 $Y_{n-1}\otimes Y_{n-2}\otimes\cdots\otimes Y_2\otimes Y_1\otimes Y_0$ 表示为 $\frac{1}{\sqrt{2^n}}M$，其中 M 表示一个 $2^n\times 2^n$ 矩阵，而 Y_r、$\forall\, r\in[0,n-1]$分别表示 XH 或 H。注意，$X=\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$， $H=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$，$XH=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$。不难验证，$M$只有一行全为 1（即该行中的每个元素都是 1），而其他各行都由 -1 和 1 组成。那么问题就变成了：全 1 行位于何处？下一段将回答这个问题。

本段将说明如何找到上一段所给张量积中的全 1 行；反过来，对于任意给定的数 k，又该如何构造一个量子电路，使与该电路相关联的矩阵的第 k 行为全 1 行。由于每个 Y_r 恰好有一行包含 $[1\ \ 1]/\sqrt{2}$，全 1 行由每个Y_r的 $[1\ \ 1]/\sqrt{2}$ 行的张量积生成。换句话说，如果涉及 $[1\ \ -1]/\sqrt{2}$ 行，所得的张量积结果就不会全为 1。令 $j_r=0$ 表示 $[1\ \ 1]/\sqrt{2}$ 位于 Y_r 的第 0 行（即 $Y_r=H$），令$j_r=1$ 表示 $[1\ \ 1]/\sqrt{2}$ 位于 Y_r 的第 1 行（即 $Y_r=XH$）。根据前前一段，我们可以知道，全 1 行位于第 k 行，其中 k 是其二进制形式 $j_{n-1}j_{n-2}\cdots j_2j_1j_0$对应的十进制值。反过来，对于任意给定的 $k\in[0,2^n-1]$，如果我们想生成一个表示张量积 $Y_{n-1}\otimes Y_{n-2}\otimes\cdots\otimes Y_2\otimes Y_1\otimes Y_0$的矩阵，使矩阵第 k 行的每个元素都为 1（忽略每个 Y_r的系数 $1/\sqrt{2}$），就可以将 k转换为二进制形式 $j_{n-1}j_{n-2}\cdots j_2j_1j_0$；若 $j_r=1$，则令 Y_r 为 XH；若 $j_r=0$，则令 Y_r 为 H。这正是算法 1所做的。为了更好地理解这一点，接下来的三段将给出 k=5、8 和 39 的三个示例。

对于k=5，其二进制形式为 $j_2j_1j_0$ =101。于是我们有 $Y_2\otimes Y_1\otimes Y_0$。将 Y_2 和 Y_0 替换为XH，如 $j_2=j_0=1$ 所示，并将 Y_1 替换为H，如 $j_1=0$ 所示，即可得到：

$$XH\otimes H\otimes XH=\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\otimes\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\otimes\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

$$= \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \end{bmatrix} \quad (9)$$

这表明第 5 行是唯一一行全为 1 的行（忽略系数 $\frac{1}{2\sqrt{2}}$）。需要注意的是，在本文中，行索引始终从 0 开始计数，即第一行是第 0 行。

对于k=8，其二进制形式为 1000。于是我们有

$$XH \otimes H \otimes H \otimes H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \otimes \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (10)$$

我们可以很容易地验证，式 (10) 中所示的张量积只有一行全为 1（忽略系数 1⁄4），位于第 8 行。

对于k=39，其二进制形式为 100111。于是，我们可以使用 $XH \otimes H \otimes H \otimes XH \otimes XH \otimes XH$ 得到一个矩阵，其中唯一一行全为 1，位于第 39 行，如图 1b 所示。

总之，通过使用 $C^n(X)$ 门和一个 HX层，我们可以将目标元素的概率从 $1/2^n$ 提高到大于 0.95 和 0.99 的值（分别对应n大于 5 和 7 的情况），其中我们已经知道目标元素的位置索引。

2.2 设计基于 R_y 层的电路

本节分为四个部分。第一部分详细介绍用于生成R_y 层的算法（算法2），该层即图 1c、d中的蓝色模块。第二部分给出图 1c、d所示的基于R_y层的电路中量子态$|\psi_2$的向量形式。第三部分解释了R_y 层在解决寻找目标元素的问题时等效于HX层。

算法2。这里我们设计一个R_y 层（图 1c 中的蓝色模块）来替代 HX层（图 1a 中的蓝色模块），即用单层 R_y 门替代由 X、H 和 XH门组成的两层。已知 $R_y(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix}$，我们有 $R_y\left(\frac{\pi}{2}\right) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$，这与 XH 完全等价；$R_y(\pi) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ 尽管存在相位差，仍可用于替代 X 门；$R_y(3\pi/2) = \frac{1}{\sqrt{2}}\begin{pmatrix} -1 & -1 \\ 1 & -1 \end{pmatrix}$ 尽管存在相位差，仍可用于替代 H 门。相位差的影响可以忽略，原因将在本节后文说明。算法2详细介绍了如何生成R_y 层。

算法 2 | 生成 R_y 层的伪代码（图 1c 中的蓝色模块）。

输入：量子比特数量 n，以及目标元素的十进制形式的位置索引 k，$\forall k \in [0, 2^n - 1]$ 输出：R_y 层中的量子门（图 1c 中的蓝色模块）。1 将 k 转换为二进制形式 $b_{n-1}b_{n-2}\cdots b_2 b_1 b_0$。2 在标签量子比特（最高有效位量子比特）上添加一个 $R_y(\pi)$ 门。3 令 m=n4 当 $m \geq 1$5 $b_{m-1} = 1$ 时，在量子比特 q_{m-1} 上添加一个 $R_y(\pi/2)$ 门。6 7 否则8 在量子比特 q_{m-1} 上添加一个 $R_y(3\pi/2)$ 门。9 $m \leftarrow m-1$

量子态的向量形式 $|\psi_2$。 图 1c 中的量子态 $|0,\psi_0$ 和 $|\psi_1$ 与图 1a 中的相同。因此，这里仅给出图 1c 中的量子态 $|\psi_2$。由算法2得到的R_y 层（如图 1c、d 中的蓝色模块所示）可表示为

$$R_y(\pi) \otimes R_y(\theta)^{\otimes n} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \otimes R_y(\theta)^{\otimes n} = \begin{bmatrix} \mathbf{0} & -R_y(\theta)^{\otimes n} \\ R_y(\theta)^{\otimes n} & \mathbf{0} \end{bmatrix} \tag{11}$$

为便于表述，我们将公式 (11) 给出的矩阵称为 $\boldsymbol{R_y}$**-**层矩阵。

与公式 (4) 类似，根据公式 (11)，图 1c 中的量子态 $|\psi_2$ 和 $|\psi_1$ 具有如下关系：

$$\begin{aligned} |\psi_2\rangle &= \begin{bmatrix} \mathbf{0} & -R_y(\theta)^{\otimes n} \\ R_y(\theta)^{\otimes n} & \mathbf{0} \end{bmatrix} |\psi_1\rangle \\ &= \begin{bmatrix} \mathbf{0} & -R_y(\theta)^{\otimes n} \\ R_y(\theta)^{\otimes n} & \mathbf{0} \end{bmatrix} [\underbrace{\alpha_0^{\mathrm{b}}, \alpha_1^{\mathrm{b}}, \cdots, \alpha_{k-1}^{\mathrm{b}}, 0, \alpha_{k+1}^{\mathrm{b}}, \cdots, \alpha_{N-1}^{\mathrm{b}}}_{\text{1st half: } N \text{ elements}}, \underbrace{0, \cdots, 0, \alpha_{N+k}^{\mathrm{g}}, 0, \cdots, 0}_{\text{2nd half: } N \text{ elements}}]^T \end{aligned} \tag{12}$$

其中，$R_y(\theta)^{\otimes n}$ 是一个 $N \times N$ 矩阵。

层与 HX 层之间的关系R_y 。本小节将说明，在从无结构数据集中寻找目标元素的问题中，R_y 层等价于 HX 层。

与 (4)–(7) 中的分析类似，公式 (5) 中定义的 β_{N+k} 的计算仅涉及 R_y-层矩阵的左下部分，即 $R_y(\theta)^{\otimes n}$，而不涉及其右上部分，即 $-R_y(\theta)^{\otimes n}$。因此，$R_y(\pi)$ 与 X 之间的相位差不会影响 β_{N+k} 的值。换言之，我们可以用 $R_y(\pi)$ 替换 X 门，而不会影响找到目标元素的概率。

我们可以将 $R_y\ \theta$ $Y_{n-1}\otimes\cdots\otimes Y_2\otimes Y_1\otimes Y_0$ $Y_r\ \forall r\in$ $[0,n-1]$ $R_y\ \pi/2$ $R_y\ 3\pi/2$ $R_y\ \theta\ ^{\otimes n}$ 1 ( )$^{\otimes n}$ 写成张量积，其中，，分别表示 ( ) 或 ( )。于是，当 $R_y(3\pi/2)$ 门的数量为偶数（奇数）时，( ) 将只有一行全-（全-负-1）行，其中全-负-1 行中的每个元素均为 −1，忽略系数 $1/\sqrt{2^n}$。当 $R_y(3\pi/2)$ 门的数量为偶数时，式 (7) 和式 (8) 中给出的计算对于图 1c 同样有效。当 $R_y(3\pi/2)$ 门的数量为奇数时，第 $N+k$ 行中的每个元素均为 −1，忽略系数 $1/\sqrt{2^n}$，于是有

$$\beta_{N+k}=-\left(\sum_{i=0}^{k-1}\alpha_i^{\mathrm{b}}+\sum_{i=k+1}^{N-1}\alpha_i^{\mathrm{b}}\right)/\sqrt{N} \tag{13}$$

$$\beta_{N+k}=-\left(1/\sqrt{N}\right)(N-1)/\sqrt{N}=-1+1/N=-1+1/2^n \tag{14}$$

由于概率是模的平方 β_{N+k}，公式（8）和公式（14）给出的目标元素的模会得到相同的概率。因此，无论 $R_y(3\pi/2)$ 门的数量是奇数还是偶数，图 1c 的输出 $|\psi_2\rangle$ 中找到目标元素的概率，在 n 分别大于 5 和 7 时，都会高于 0.95 和 0.99。这与上一节对图 1a 的分析相同（见公式（8）下方的段落）。因此，我们可以用算法 2 生成的 R_y 层替代算法 1 生成的 HX 层。

2.3 算法 1 和算法 2 的可扩展性

在上文对图 1a 和图 1c 的分析中，没有对量子比特的数量施加限制或作出假设。也就是说，第 2.1 节和第 2.2 节的分析适用于任意数量的量子比特。具体而言，对于任意 n 值，算法 1（算法 2）均可生成一个量子电路，其对应的矩阵形式仅在第k 行有一行全为 1（全为 1 或全为 -1），$k\in[0,2^n-1]$，其中n为量子比特的数量。

在算法 1 和算法 2 中，目标元素的位置索引 k 必须预先已知。这意味着，如果位置索引未知，它们就无法用于在数据集中寻找目标元素。然而，上述分析已经证明，当 n 大于 5 时，HX层（由算法 1 生成）或 R_y 层（由算法2生成）与一个 $C^n(X)$ 门（图 1a、c 中的红色和蓝色模块）结合，可以将目标元素的概率从 $1/2^n$ 提高到接近 1。需要注意的是，对于相同的任务，Grover搜索算法需要一个电路深度随量子比特数量呈指数增长的量子电路，这体现了算法 1 和算法 2 生成的电路在电路深度方面的指数级优势。

2.4 变分量子搜索及其可达性

本节首先简要介绍变分量子求解器，随后详细说明变分量子求解器的可达性定义。变分量子求解器[4] 是一种变分量子算法，涉及经典计算机与量子计算机之间的交互。在经典部分，使用优化器根据目标函数 $f(\boldsymbol{\theta})$ 更新试探态的参数 $\boldsymbol{\theta}$：

$$f(\boldsymbol{\theta})=-0.5\langle\psi_1|\psi_2\rangle+0.5\langle\psi_1|Z\otimes I^{\otimes n}|\psi_2\rangle \tag{15}$$

其中，ψ_1 和ψ_2 分别表示试探态前后的量子态，如图 1a、c 所示；Z 和 I 分别表示泡利Z和单位矩阵，n 为量子比特的数量。目标函数中的第一项和第二项可分别通过对基于哈达玛测试的两个量子电路进行测量得到，详见论文[4]。论文[4] 已验证了两种浅层

且有效的变分量子求解器试探态。然而，上述分析表明，仅使用单个R_y层作为试探态的变分量子求解器，就足以在未知数据集中找到唯一的目标元素；这种试探态比论文[4]，中使用的两种试探态更浅。与算法 1 和算法 2 不同，变分量子求解器无需预先知道目标元素的位置索引k。为了量化试探态生成能够使给定目标函数最小化的量子态的能力，我们将在本节余下部分为变分量子求解器定义可达性。

式 (15) 给出的目标函数可改写为

$$f(\boldsymbol{\theta}) = \langle\psi_1|O|\psi_2\rangle \tag{16}$$

其中

$$O = 0.5\left(-I^{\otimes n+1} + Z \otimes I^{\otimes n}\right) = \mathrm{diag}([\underbrace{0,0,\cdots,0}_{N\text{ elements}}, \underbrace{-1,-1,\cdots,-1}_{N\text{ elements}}]) \tag{17}$$

其中，diag(w) 表示 主对角线元素为向量 w 中各元素的方形对角矩阵 .

然后，我们可以为变分量子求解器定义一个可达性[15,16]：

$$\Re = \frac{\min_{\theta}\langle\psi_1|O|\phi(\theta)\rangle - \min_{|\varphi\rangle\in\mathcal{H}}\langle\psi_1|O|\varphi\rangle}{-\min_{|\varphi\rangle\in\mathcal{H}}\langle\psi_1|O|\varphi\rangle} \tag{18}$$

其中，$|\psi_1$ 表示试探态之前的量子态（见图 1c）；$\min\theta$ 项（为便于表达，用来指代 $\min_{\theta}\psi_1|O|\phi(\theta)$）是变分量子求解器的试探态所生成的所有可达量子态中的最小值；$\min\theta$ 项中的 $\phi(\theta)$ 表示图 1c 中的 $|\psi_2$；而 minH 项（为便于表达，用来指代 $\min_{|\varphi\rangle\in\mathcal{H}}$ $\psi_1|O|\varphi$）表示希尔伯特空间 H 中所有量子态 $|\varphi$ 的最小值。

本段讨论式（18）中分子的可能取值范围。注意，在本文中，$|\psi_1$ 只有一个非零元素，该元素等于 $1/\sqrt{N}$，且位于其后半部分；$x|O|y$ 等于 $-|x$ 的后半部分与 $|y$ 的后半部分的张量积，其中量子态的后半部分指其向量形式中后半部分的元素。因此我们可以得知，在式（18）的分子中，$\psi_1|O|\phi(\theta)$ 的取值范围是 $[-1/\sqrt{N}, 1/\sqrt{N}]$ 的子集，而 $\psi_1|O|\varphi$ 的取值范围是 $[-1/\sqrt{N}, 1/\sqrt{N}]$。因此，$\psi_1|O|\phi(\theta) - \psi_1|O|\varphi$ 的取值范围是 $[-2/\sqrt{N}, 2/\sqrt{N}]$ 的子集；当 n 较大时，该范围极小，因为上下界都以指数速度趋近于零。

鉴于定义良好的度量不应始终处于零附近极小的范围内，因此不适合使用式（18）中的分子来衡量可达性，因为当量子比特数量较大时，该分子会极小，并且随着量子比特数量增加而指数级趋近于零，无论 θ 的取值如何。为解决这一问题，我们采用分母中的 minH 项，具体说明见下一段。

根据论文[4]，中的分析，minH 项的最小值，即 $-1/\sqrt{N}$，在 $|\varphi$ 等于计算基态 $|N+k$ 时取得，该计算基态可表示为一个向量

该向量仅在索引 $N+k$ 处有一个元素为 1，其余元素均为 0，其中 $N = 2^n$。由于将范围 $[-2/\sqrt{N}, 2/\sqrt{N}]$ 除以 minH 项后得到的范围为 $[-2, 2]$，因此我们将 minH 项用作分母，使式（18）中定义的 R 的范围成为 $[0, 2]$ 的子集，而不是前文提到的极小范围。$[0, 2]$ 的范围更适合作为可达性度量，因为它不局限于零附近的极小范围，这也是我们在可达性的定义中引入分母的原因。

现在，我们解释为何要在每一项中加入绝对值符号，如公式（19）所示。根据公式（8）和（14），β_{N+k} 等于$1 - 1/N$ 或$-1 + 1/N$ 时，找到目标元素的概率相同；也就是说，这两种解同样好，其中$N = 2^n$。然而，将β_{N+k} 的这两个值代入$|\psi_2$，以替换$|\phi(\theta)$，会得到不同的$\psi_1|O|\phi(\theta)$值，进而导致不同的可达性，但它们应对应相同的可达性。为了解决这一问题，我们使用绝对值项，即$|\psi_1|O|\phi(\theta)|$ 和$|\psi_1|O|\varphi|$，分别替换$\psi_1|O|\phi(\theta)$和$\psi_1|O|\varphi$，使符号不同的两个振幅对应相同的可达性。

本段讨论了公式（19）所给出的一个更好的可达性定义。注意，当 $\psi_1|O|\varphi$ 等于 $1/\sqrt{N}$ 或 $-1/\sqrt{N}$ 时，$\max_{|\varphi\rangle\in\mathcal{H}} |\psi_1|O|\varphi|$ 达到最大值，这对应于以 1 的概率找到目标元素。另一方面，$\min_{|\varphi\rangle\in\mathcal{H}} |\psi_1|O|\varphi|$ 达到
当 $\psi_1|O|\varphi$ 等于 0 时达到最小值，这对应于以 0 的概率找到目标元素。因此，使用绝对值项时，我们采用“max”而非“min”，以便能够以较高的概率找到目标元素。根据上一段的讨论，如果使用（18）作为可达性的定义，符号不同的两个振幅会导致不同的可达性值；使用公式（19）则可以解决这一问题。因此，我们建议使用公式（19）而非（18）作为可达性 的定义。

$$\Re = \frac{\max_{\theta}|\langle\psi_1|O|\phi(\theta)\rangle| - \max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|}{-\max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|} \tag{19}$$

R 的值越小，可达性越好。当 $\mathrm{R} = 0$ 时，变分量子求解器具有完美可达性。使用分母和绝对值项，使我们对可达性的定义有别于论文[15,16]中提出的定义。

对于图 1c 所示的电路，其中 R_y 层由算法2生成，输出态 $|\psi_2$ 的第 $(N+k)$ 个元素要么等于 $1 - 1/N$，要么等于 $-1 + 1/N$，分别如公式（8）和（14）所示。若在公式（19）中将 $\max_{\theta} |\psi_1|O|\phi(\theta)|$ 替换为 $|\psi_1|O|\psi_2|$，则有

$$\frac{|\langle\psi_1|O|\psi_2\rangle| - \max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|}{-\max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|} = \frac{\frac{1-1/N}{\sqrt{N}} - \frac{1}{\sqrt{N}}}{-\frac{1}{\sqrt{N}}} = \frac{1}{N} = \frac{1}{2^n} \tag{20}$$

由于 $|\psi_2$ 是由算法2生成的，因此它是具有算法2中指定的 θ 的 $|\phi(\theta)$。于是我们知道 $\max_{\theta} |\psi_1|O|\phi(\theta)| \geq |\psi_1|O|\psi_2|$。因此，我们有

$$\Re = \frac{\max_{\theta}|\langle\psi_1|O|\phi(\theta)\rangle| - \max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|}{-\max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|} \leq \frac{|\langle\psi_1|O|\psi_2\rangle| - \max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|}{-\max_{|\varphi\rangle\in\mathcal{H}}|\langle\psi_1|O|\varphi\rangle|} = \frac{1}{2^n} \quad (21)$$

其中 $\leq$ 两侧分别来自公式（19）和公式（20）。公式（20）给出了一个具体电路，即使用由算法2生成的 R_y 层作为变分量子求解器的试探态，其可达性值等于 $1/2^n$；当 n 较大时，这是一个极小的值，并且随着 n 的增加而指数级收敛至零。也就是说，该电路已经实现了近乎完美的可达性。

总之，变分量子求解器至少具有近乎完美的可达性，并且量子比特越多，变分量子求解器的可达性就越好。

3. 结果

本文所述结果由量子模拟器通过 PennyLane 的设备[20]获得。关于 2、8 和 14 量子比特（20 和 26 量子比特）输入量子态的结果，分别使用 Intel i5-6500 CPU（A40x4 48 GB GPU）上的 PennyLane default.qubit（lightning.gpu）设备获得。试探态中 $\boldsymbol{\theta}$ 的初始值从 0 到 2π之间的均匀分布中随机采样。与论文[4], 相同，变分量子求解器采用两个迭代过程终止准则。第一个准则是迭代次数达到指定数量（本文设为 300 次）。第二个准则是连续指定次数（本文设为 5 次）发生小变化事件；小变化事件定义为：连续两次迭代中目标函数相对变化量的绝对值小于指定的小值（本文设为 1×10^{-4} ）。满足任一准则时（以先满足者为准），变分量子求解器的迭代过程即告终止。

在此，我们通过数值验证，仅含一层R_y（θ）门的变分量子求解器（VQS）能够高效地从无结构数据集中找到目标元素，而无需事先知道目标元素的位置索引k；这与 VQS 的可训练性有关。我们使用试探态为 R_y 层的 VQS，从包含 2、8、14、20 和 26 个量子比特的数据集中寻找目标元素。

结果如图2所示，这表明对于n=8分别为 14、20 和 26 的情况，在 100 次运行中的大多数运行里，变分量子求解器确实能够找到目标元素，因为放大后的概率非常接近 1；而在某些运行中（四种情况下分别有 22、16、16 和 16 次），放大后的概率接近 0。与论文[4], 中展示的结果相比，尽管在试探态中使用R_y 层的变分量子求解器不如论文[4], 中展示的使用两种试探态的变分量子求解器稳定，但除 2 量子比特的情况外，前者仍能在大多数运行中将概率放大至接近 1。2 量子比特情况下表现相对较差并不值得担忧，实际上，正如下文所述，这进一步验证了我们的分析。当n等于 2，且我们使用R_y 层作为变分量子求解器的试探态时，根据公式（8），找到目标元素的概率为$(1 - 1/2^2)^2 = 0.5625$，大致处于 2 量子比特情况箱线图结果的中间位置（图2a中最左侧的结果）。换言之，图2a中的结果验证了可达性分析所得出的结论，即量子比特数量越多，变分量子求解器的可达性越好。

使用带有R_y层的变分量子求解器之所以重要，是因为前文已证明，R_y层与$C^n(X)$结合，可以将目标元素的概率从$1/2^n$放大至接近1，适用于任何大于5的量子比特数量。也就是说，对于任何大于5的量子比特数量，使用R_y层的变分量子求解器都能保证可达性；这意味着我们可以将变分量子求解器扩展到任意多的量子比特，同时将电路深度保持为2（即一层R_y和一层$C^n(X)$），相较于Grover算法具有指数级优势。

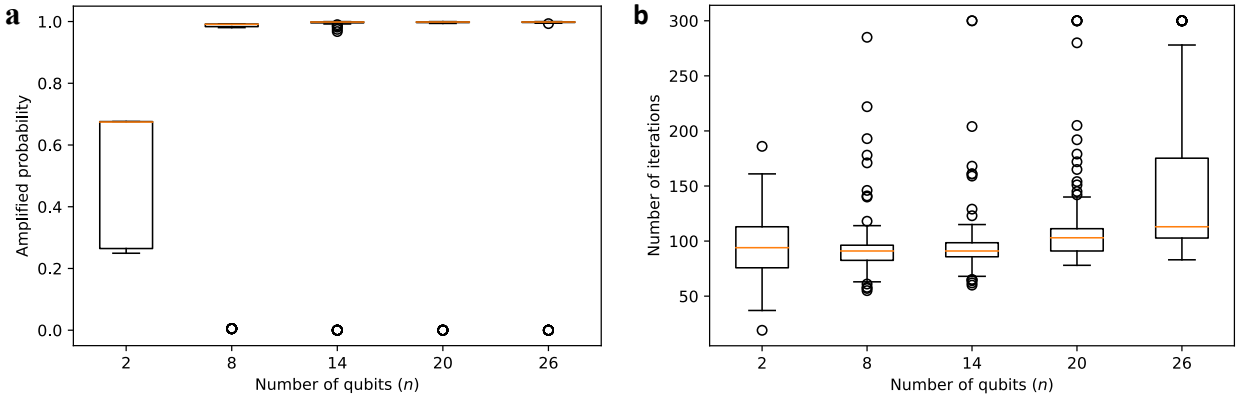

图2. **变分量子求解器（VQS）100次运行的箱线图结果**，使用R_y层作为n量子比特输入量子态的试探态。**a**，目标元素放大后的概率。**b**，满足终止准则时所用的迭代次数。

4. 结论

我们提出了两种算法，分别用于构造深度为2（HX层）和深度为1（R_y层）的电路。如果事先已知目标元素的位置索引，这两种电路都可替代Grover算法所需的指数深度电路。我们证明，对于任意多于5个量子比特的情形，以R_y层作为试探态的变分量子求解器（VQS）具有近乎完美的可达性，且其可达性会随量子比特数量增加而呈指数级提升；数值实验进一步验证了这一结论。在这些实验中，我们使用以R_y层作为试探态的VQS，在由8、14、20和26个量子比特表示的无结构数据集中搜索唯一的目标解，并在100次独立运行中的78至84次里，以接近1的概率成功找到目标元素。实验还表明，VQS在多达26个量子比特的情况下具有良好的可训练性。未来我们将进一步研究VQS在更多量子比特下的可训练性。

参考文献

1. Nielsen, M. A. & Chuang, I. L. 量子计算与量子信息：十周年纪念版。剑桥大学出版社（剑桥大学出版社，2011年）。doi:10.1017/CBO9780511976667。2. Grover, L. K. 量子力学有助于在大海捞针中寻找目标。《物理评论快报》 **79**，(1997年)。3. Grover, L. K. 一种用于数据库搜索的快速量子力学算法。载于*ACM*计算理论年度研讨会论文集第F129452卷（1996年）。

4. Zhan, J. 具有指数级加速的变分量子搜索。（2022年）。5. Cerezo, M. 等。变分量子算法。《自然·物理评论》第3卷，625–644。预印本：https://doi.org/10.1038/s42254-021-00348-9（2021年）。6. McClean, J. R., Romero, J., Babbush, R. & Aspuru-Guzik, A. 变分混合量子-经典算法理论。*New J Phys* **18**，(2016年)。7. Peruzzo, A. 等。光子量子处理器上的变分本征值求解器。*Nat Commun***5**，(2014年)。8. Bharti, K. 等。含噪中等规模量子（NISQ）算法。*arXiv:2101.08448*预印本（2021年）。9. Bittel, L. & Kliesch, M. 训练变分量子算法是NP难问题。《物理评论快报》 **127**，120502（2021年）。10. McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. 量子神经网络训练景观中的贫瘠高原。*Nat Commun* **9**，(2018年)。11. Cerezo, M., Sone, A., Volkoff, T., Cincio, L. & Coles, P. J. 浅层参数化量子电路中依赖代价函数的贫瘠高原。*Nat Commun* **12**，(2021 年)。12. Liao, Y. & Zhan, J. 增强量子神经网络表达能力的策略。（2022年）。13. Haug, T., Bharti, K. & Kim, M. S. 参数化量子电路的容量与量子几何。《*PRX*量子》 **2**，(2021年)。14. Sim, S., Johnson, P. D. & Aspuru-Guzik, A. 混合量子-经典算法中参数化量子电路的表达能力与纠缠能力。*Adv Quantum Technol* **2**，1900070（2019年）。15. Akshay, V., Philathong, H., Zacharov, I. & Biamonte, J. 图问题量子近似优化中的可达性缺口。《*Quantum*》 **5**，(2020年)。16. Akshay, V., Philathong, H., Morales, M. E. S. & Biamonte, J. D. 量子近似优化中的可达性缺口。《物理评论快报》 **124**，090504（2020年）。17. Zhan, J. NP完全顶点着色问题的量子可行性标记。（2023年）doi:10.48550/arxiv.2301.01589。18. Bharti, K. 等。含噪中等规模量子算法。*Rev Mod Phys* **94**，015004（2022年）。19. Morales, M. E. S., Tlyachev, T. & Biamonte, J. 通过变分学习Grover量子搜索算法。*Phys Rev A (Coll Park)* **98**，(2018年)。20. Bergholm, V. 等。PennyLane：混合量子-经典计算的自动微分。预印本：https://doi.org/10.48550/ARXIV.1811.04968（2018年）。

致谢

本研究部分得到了 NSF ERI 计划的支持，项目编号为 2138702。本研究通过先进网络基础设施协调生态系统：服务与支持（ACCESS）计划的分配额度 CIS220136，使用了美国国家超级计算应用中心的 Delta 系统；该计划由美国国家科学基金会的资助项目 #2138259、#2138286、#2138307、#2137603 和 #2138296 提供支持。我们感谢 IBM Quantum 服务为本研究提供支持。本文所表达的观点仅代表作者立场，并不反映 IBM 或 IBM Quantum 团队的官方政策或立场。