arXiv:2304.02358v1 [quant-ph] 5 Apr 2023

量子电路概率的可视化用于量子程序合成的计算作用量估计

Bao Gia Bach[1,6], Akash Kundu[2,3,6], Tamal Acharya[4,6], 和 Aritra Sarkar[5,6,✉]

[1] *Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Viet Nam*
[2] *Joint Doctoral School, Silesian University of Technology, Gliwice, Poland*
[3] *Institute of Theoretical and Applied Informatics, Polish Academy of Sciences Gliwice Poland*
[4] *Independent Researcher Bengaluru Indie*
[5] *Quantum Machine Learning research group Quantum Computing division QuTech The Netherlands*
[6] *Quantum Intelligence research team Department of Quantum & Computer Engineering Delft University of Technology, The Netherlands*
✉ *a.sarkar-3@tudelft.nl*

摘要

本研究将算法概率中的概念应用于布尔与量子组合逻辑电路。文中首先以教程的形式介绍状态及状态复杂度的各种概念，随后定义计算的电路模型中的状态概率。接着，比较经典与量子门集，并选取若干具有代表性的门集。文中列举并可视化了这些门集在受空间和时间限制的条件下的可达性与可表达性，并从计算资源、普适性和量子行为等方面研究这些结果。文章还指出，研究电路概率有助于几何量子机器学习、新型量子算法合成和量子人工通用智能等应用。

关键词： 基于门的量子计算、算法概率、电路复杂度、可达性、可表达性

1 引言

量子计算已达到一定的技术成熟度，量子处理器平台虽仍有限，但已逐渐开放供实验使用。这一快速进展促使研究人员着手探索利用量子算法解决各种真实世界的工业与科学应用问题 [1] 。这些算法的逻辑表述随后经过量子计算栈 [2] 中的系统抽象层处理，转化为适用于基于门的量子计算设备的低层量子电路。所谓的 NISQ（含噪声中等规模量子）时代 [3, 4] ，体现了当前量子处理器在相干时间、门错误和量子比特连接性方面的局限。为此，研究人员也从另一端展开探索 [5], ，制定设计策略并寻找这些有限计算能力的应用场景，以实现计算优势。为了更好地利用这些有限的设备，必须了解量子逻辑与物理资源之间的关系。这正是本文开展这项研究的动机。

量子计算处于量子物理与计算机科学的交汇处，促成了这两个领域之间丰富的概念交流。与本研究关注点直接相关的是：（i）物理定律为计算提供了根本性的界限；（ii）计算则从信息论角度解释了许多物理现象。前者最初是在热力学极限的背景下得到探索的 [6], ，由此推动了可逆计算的发展 [7, 8], ，最终界定了量子计算的极限 [9, 10]。后者的相关研究则属于数字物理学范畴。其中一些开创性贡献包括元胞自动机 [11] 、构造子理论 [12], 、量子力学的信息公理 [13, 14] 、计算的平稳作用原理 [15, 16], 以及张量网络等。本研究所呈现的工作具有基础性，并有望应用于这一协同关系的两个方向。我们重点通过实证展示现有理论思想对量子计算的影响。具体而言，我们将算法信息论 [17] （理论计算机科学与人工智能 [18]的一个子领域）中的概念引入基于门的量子计算 [19]。

在这项工作中，我们考虑对量子电路空间（以 QASM 代码表示）进行枚举。这只是给定维度的希尔伯特空间上可定义的不可数无穷多个量子过程中的一个极小样本。所选过程子集取决于：（i）量子处理器的原生门集；（ii）最大电路宽度（量子比特总数）；以及（iii）根据退相干时间设定的电路深度上限。我们研究这些电路如何将经典输入空间映射到经典输出（通过

Z轴测量）。我们的形式体系假设 QASM/电路采用离散通用门集编码。以这种方式构造的量子程序空间可以枚举，因此在形式上是可数无穷的。然而，在大多数实际实现中（具有有限的电路深度），有意义的计算集合是有限的。即使通用门集包含任意旋转角度，在实际量子计算机实现中，有限数量的控制信号配置也会有效地将原生门集离散化。虽然给定通用门集后，任意酉变换都可以分解到任意精度（例如，$\{Rx(\theta_x)\ Ry(\theta_y)\ CX\}$），但在资源受限的情况下（例如，系统退相干前 QASM 代码的行数受限），程序空间无法映射到任何量子过程。这限制了我们探索那些能够用量子门高效描述、并可在量子计算机上模拟的功能性物理过程。

自 20 世纪 90 年代提出首批量子算法以来，新算法的发现 [20]一直稳步推进。然而，量子算法设计涉及量子力学现象（例如叠加、纠缠），这些现象与人类的直觉经验相悖。因此，以数学形式体系进行推理一直是开发更先进量子逻辑的门槛，也成为量子加速计算更广泛普及的瓶颈。已有一些方案尝试通过遗传编程 [21] 和电路综合 [22]来解决这些问题。本文沿着这一研究方向，采用算法信息论的原则性方法 [23] 继续推进。量子程序综合框架需要理解量子程序的空间及其相关资源，以实现程序搜索/归纳。正如本文所述，受资源限制的量子电路及其相应的经典信息处理能力图景，为此奠定了基础。

本文其余部分安排如下。第 2 节介绍如何将状态表示为符号，以及对这些符号进行变换的方式，并说明这对其统计复杂度和算法复杂度的影响。第 3 节讨论如何从门集中构造布尔电路和量子电路时涉及的细微问题。第 4 节介绍我们使用多种门集枚举状态复杂度的实现，并对结果进行可视化和分析。第 5 节通过讨论这项研究的多种应用，对本文作出总结。

2 状态与复杂性

系统的状态定义了其可观察到的行为。这些状态可以通过多种方式编码。一种常见的编码方式是为每个独特且可区分的状态分配一个符号。对同一系统的后续观测（或由多个此类系统组成的更大系统的观测）用该字母表中的一个字符串表示。只有一种状态的系统并不有趣，因为其中实际上什么也不会发生。稍微有趣一些的系统中，一个非常简单的例子是硬币，它有两种状态——正面和反面。硬币可以是公平的，也可以是有偏的；可以连续抛掷多次，也可以同时、依次或有条件地抛掷多枚硬币。一系列（或并行进行的）抛硬币结果可以用布尔字符串表示。给定由长度为 n 的所有布尔字符串构成的集合，表示为 $\{0,\ 1\}^{\otimes n}$，每个字符串出现的概率都可能相同，即 $1/2^n$。在物理系统中，如果每种输出出现的可能性相同，那么均匀分布就能对该系统进行建模；例如，通信线路可能需要以相同的概率传输每种编码。由此可知，公平硬币抛掷 8 次得到 1111111011 的概率和意外程度，与得到某个由 5 个 1 和 5 个 0 组成的特定排列（例如 0101100101）相同。

如果一切都同等可能，彼此之间也毫无关联，那世界将会是一片极其无趣的灰色。幸好事实并非如此。我们能感知周围的结构。为何会如此，很难回答；但一种可能的解释是，我们的人类生物学／技术感知工具能力有限。因此，我们感知到的不是完整谱系中的底层随机性，而是涌现的统计结构。这些结构为我们提供了另外两种表征状态的方式。我们可以从这两个角度研究状态复杂度：统计和算法。

2.1 熵的统计涌现

第一项改进基于放宽“所有状态都同等可能”的标准。我们发现，由于对排列的微观状态缺乏了解，我们似乎更倾向于将 0101100101 视为一系列公平抛硬币的结果中更容易接受的一种。我们关注这样一种模式：包含 9 个 1 的所有可能状态共有 10 种，而包含 5 个 1 和 5 个 0 的状态则有 $10*9*8*7$ 种，这与统计热力学中的熵类似。熵较高的状态数量更多，而宇宙趋向于预期中熵较高的状态，这正是我们产生时间感知的原因。在信息论中，给定一个离散随机变量 X，它取值于字母表 $\mathcal{X}$，并按照 $p:\mathcal{X}\to[0,1]$ 分布，则从该集合中抽样得到的变量的香农熵 [24] 定义为 $H(X) = -\sum_{x\in\mathcal{X}} p(x)\log\ p(x)$。这种相似性表示统计信息的平均水平，也就是变量可能结果所固有的意外程度或不确定性；当概率分布为均匀分布时，该值达到最大。将有偏的感知集合最优地编码为信息的方法，是霍夫曼编码等码字的基础。编码的设计旨在

根据概念的比特长度进行调整，让最常用的概念编码更简洁。为了取得平衡，较少使用的概念所需的编码长度会超过其原本长度。例如，概率 $p(00) = 0.4$，$p(01)$ =0。05，$p(10) = 0.2$，$p(11) = 0.35$，最佳编码为 $00 :\to 0$，$01 :\to 111$，$10 :\to 110$，$11 :\to 10$。请注意，只有在有偏的概率分布保持不变时，这些码字才优于原始形式（而这种分布反过来也可能是感知涌现宏观状态所造成的结果）。如果所有字符串的出现概率都相同，这些编码方案的代价反而更高。

我们在语言的语义中也采用类似的做法。例如，与其为每一种体型庞大、像卡车一样、具有一组特定的 x，y，z等我们无法区分的特征的大型动物都创造一个新词，不如把它们统称为“大象”；而对于我们能够区分的动物，例如你的宠物猫，则会给它们起特别的名字。你的朋友或许无法分辨你的猫和另一只猫，而在训练有素的驯象师眼中，每头大象都能被单独辨认出来——这体现了语言的主观性。类似地，日常使用某个物种的学名或个体的完整遗传密码会很繁琐，但这对生物分类或医疗却很有用。因此，与霍夫曼编码大致相同，语言中的词语源于人们以不同方式忽略细节、聚焦于特定涌现语义。语言所提供的压缩构成了理解的基础，这与宏观状态（保留某些对称性／特征的微观状态）促成涌现物理定律的方式大致相同。

2.2 普适性的算法涌现

第二种改进是放宽所有状态都必须“互不相关”的标准。为所有待考察的感知采用唯一编码固然不错，但如果能找到这些符号之间的关系，我们就能更好地压缩和理解。例如，可以通过逆来关联比特串，或将整数按顺序排列在数轴上。为关系分配符号，只是尝试通过忽略那些现在可以用其他状态及关系的符号构成的特定句法复合体来描述的状态符号，从而减少码字数量。为了映射到每一种感知，使用这些编码的总长度未必比原始的二进制编码或霍夫曼编码更短。因此，何时这些关系能带来益处而非给编码增加额外复杂性，同样取决于具体情况。目标并非为各种感知提供资源效率最高的方式，而是针对有偏分布或某个感知子集使用更少的符号。通用性与效率之间的这种权衡，正是 MetaGolfScript [25] 这类小众语言被禁止参加代码高尔夫竞赛，或精简指令集计算机与复杂指令集计算机两种体系结构并存的原因。

然而，令人惊讶的是，我们发现有些关系无处不在，仅凭对关系本身及某个起始感知的编码，就能将其映射到所有感知（通常甚至来自一个无限集合）。例如，后继函数（即加 1）用 1 表示，数字 0 用 0 表示。这样，任何自然数都可以通过嵌套后继函数来表示，例如，1110 表示 3。给定 0，后继函数在自然数集合上具有普遍性；而对所有素数集合进行乘法运算，也能生成自然数。对于目标输出集合而言，这样的输入与变换集合称为普遍的。有些（表示状态和关系的）符号功能强大，能够借助有限个符号和一些组合规则来表示无限多个状态；例如，任何由 d 个符号构成并采用位置记数法的 d进制记数系统都能表示任意整数。

从工程角度来看，拥有这样一组通用的状态和变换，有助于用有限数量的构建块驾驭无限多种可能性。例如，键盘由字母表构成，而不是为词典中的每个词都配备一个按键。这种改进有两点微妙之处。(i) 首先，由于我们现在有有限种构建块类型来构造无限多个状态，用这些构建块进行描述的长度可能是无限的。如果我们限制可使用的构建块数量，那么根据组合论，这就限定了我们能够描述的状态数量。需要更长描述的状态便无法表示。(ii) 其次，原有状态代表着可观察的行为，而我们为减少符号集而引入的这些新伪状态和变换，不一定各自具有直观的独立含义。例如，比特翻转算子可以对应于切换开关的操作；但其他构建块可能没有这样的含义，比如字母表中的字母本身并不具有语义。

现在，我们已经拥有一套符号集，其中包括：（i）某些已观测状态和伪状态的集合；（ii）一套丰富的（通用）变换，用于描述其他已观测到的感知。顺便一提，必须指出，借助信道-状态对偶，变换可以在更高维度上表示为状态 [26]，，也就是将动态表示为静态。现在，我们能否根据用于描述物理现象（来自所有宏观感知的集合）的各种符号的使用概率，选择一种最优编码方案？问题在于，我们事先并不知道这些模块将以何种方式使用，也就是说，不知道整体的分布。想象一下经营一家乐高工厂，并决定为满足客户需求而生产各种模块的数量。由于通常缺乏其他信息，基础集合往往采用标准二进制编码（即假设所有模块和初始感知都具有相同的使用概率），例如 ASCII码。这称为计算机中指令集架构的操作码编码

。在这种情形下（通用计算），从另一个角度研究问题或许会有帮助，也就是考察表示某种特定感知需要哪些资源。资源通常分为两类：（i）以周期（时间）和内存（空间）衡量的计算成本；（ii）使用该语言描述这种感知所需的长度。

计算成本是计算复杂度领域的研究对象。问题（以及由此产生的解决方案，即基于符号的一系列指令）会根据时间和空间随问题规模增长的变化规律，被划分为不同的类别 [27]。一些常见类别包括多项式时间（P）、非确定性多项式时间（NP）和有界误差量子多项式时间（BQP）。

描述长度量化了感知对象的柯尔莫哥洛夫复杂度 [28]，也称算法熵。其定义为 $K_U(X) = \min_p\{\ell(p) : U(p) = x\}$，其中 ℓ表示通用图灵机 U 在所用编码下的程序 p的长度，该程序会输出 x。尽管它取决于构建块的选择及其编码方式，但这种依赖仅体现为一个加性常数项（称为不变性定理），该常数项等于转换为另一种语言或自动机所需的交叉编译器长度。因此，无论字符串所属的集合如何，都可以用柯尔莫哥洛夫复杂度来量化单个字符串的复杂度。不过，精确求出其值是不可计算的。有许多方法可以从上界方向逼近它（下半可计算），例如使用压缩算法、最小描述长度和块分解法。

到目前为止，我们回顾了三种不同的状态复杂度概念：
1. 统计复杂度：状态集合上的香农熵（给定其概率分布） 2. 计算复杂度：生成该状态的程序在时空上的扩展行为（给定一种语言） 3. 算法复杂度：生成该状态的程序长度（给定一种语言）

在本研究中，我们关注的则是状态的电路复杂度。电路复杂度与算法复杂度有关 [29], ，而算法复杂度又与统计 [30]和计算复杂度 [31]有关。计算复杂度通常研究渐近扩展行为，并给出下界。尽管电路族具有特定的复杂度类层级（例如， AC^i、 TC^i、 NC^i），但这与本研究关系不大。我们将关注规模有界的电路（在空间和时间上均有界）。类似地，研究表明，期望柯尔莫哥洛夫复杂度与香农熵 [30], 相对应，不过这种关系对本文并无直接的重要性。 [29] 在某些考量下，可以证明柯尔莫哥洛夫复杂度与电路复杂度非常相似 [29]。另一种类似的关系是，电路复杂度较低的函数，其真值表的柯尔莫哥洛夫复杂度也较低。文献 [15, 16] 曾从拉格朗日作用量的角度提出联系电路复杂度、算法复杂度和统计复杂度的计数论证。我们的研究是这一相当小众的领域中又一步尝试，旨在从不同视角理解观测到的状态。

需要注意的是，大多数关于算法信息论的研究都是在通用自动机的背景下进行的，例如图灵机、lambda演算、细胞自动机等。描述的大小取决于符号对变换的表达能力。到目前为止我们所描述的，即作为两个状态之间关系的变换，通常是电路语言中的情况。用更抽象的逻辑框架编写的程序允许更强大的原语，例如一阶或高阶逻辑中的全称量词和存在量词。通常，通用计算模型需要递归可枚举语言。在乔姆斯基层次结构中，图灵机比线性有界自动机更强大，后者又比下推自动机更强大，而下推自动机又比有限状态机（FSM）更强大。参见 [32] 以比较这些经典和量子计算模型。然而，对于功能较弱的自动机和语言模型，可以推导出相应的算法复杂度概念 [33] 。这一点很重要，因为用图灵完备语言编写的程序最终会通过计算堆栈的各层进行翻译，并由逻辑电路执行。然而，这些逻辑电路是顺序逻辑（允许存储单元）和组合逻辑的组合，可用于模拟FSM。纯组合逻辑（不要与组合逻辑混淆，后者是通用的）的能力甚至低于FSM。前者无循环且无状态，因此是基于输入的输出状态的直接表示。需要注意的是，即使电路纯粹由组合逻辑元件组成，程序执行在经典和量子处理器中通常也是时钟控制的，以防止竞争条件。因此，即使不跟踪和访问中间状态，也可以在此设置中定义时间和空间资源。通过借用算法信息论中的概念（如在函数式程序上定义的），在本工作中，我们研究布尔/量子组合逻辑的电路复杂度对状态复杂度的影响。

3 电路概览

在了解了这些复杂性度量之后，现在让我们先来探讨布尔电路的全貌。量子电路模型受到布尔电路模型的启发，也是对它的推广，因此，从经典模型入手，再将其推广到相应的量子形式，是很自然的做法。

3.1 状态的电路概率

与组合布尔逻辑电路相比，算法信息通常用于研究经典函数（例如 λ-演算）。我们打算研究后者。考虑由 n 位比特串构成的空间。给定一组构成布尔电路的量子门，我们会发现，并非所有输出出现的概率都相同。这是因为，尽管每个{电路的输入} 对只有一个输出，但同一输出可以由多个电路以多种方式生成。事实上，我们可以通过恒等操作或连续执行两次非门这类冗余操作，任意增大电路规模。

由于程序数量众多，为了比较两个字符串，我们关注的是每个电路被生成的概率，而不是寻找输出该字符串的最短电路。这与字符串的算法概率 [34] 概念类似，其定义为 $M(X) = \sum_{p:U(p)=x*} 2^{-\ell(p)}$，其中通用自动机 U 上的前缀自由程序 p 以二进制编码。对该项贡献最大的是最短程序（即柯尔莫哥洛夫复杂度）。复杂度与概率之间的这种联系可以表述为：具有较短程序的字符串有许多其他生成方式，因此由随机编程的通用自动机生成的概率更高。需要注意的是，为生成算法概率而给程序分配均匀随机分布，或在整个字符串集合上采用均匀分布，并没有充分的依据。在 [35] 第 4.5 节中，其中一位作者提出了更具物理动机、并收敛到构造子的“嵌套算法概率”。在本研究中，我们先采用均匀分布，之后再将实现推广为允许使用任意先验分布。为区别通常意义上通用自动机上的字符串算法概率 $M_U(X)$ 与我们所讨论的基于具有不同时空复杂度的等效电路分布的输出字符串概率，我们将自己的算法概率表述记为 $M_{circ}(X)$。

在原始设定中，$M_U(X)$ 是不可计算的，因为它需要运行每一个可能的程序，而其中存在不会终止的程序。然而，它是下半可计算的，并且可以在给定运行时间界限的情况下进行近似。一种近似方法见于 [36]：运行特定枚举中的每一台图灵机，并直接使用在限定运行时间内停机的图灵机的输出分布。对于布尔电路/量子电路，运行时间界限是预先确定的，不存在停机问题。因此，对于状态 s，$M_{circ}(s)$ 可以通过生成目标状态的集合的基数与从初始状态 s_0 出发的电路总数之比来近似，如下所示：

$$M_{circ}(s) \approx \frac{|C \in \{\texttt{gateset}, \texttt{maxspace}, \texttt{maxtime}\}, s \leftarrow C(s_0)|}{|C \in \{\texttt{gateset}, \texttt{maxspace}, \texttt{maxtime}\}|} \tag{1}$$

通过将概率与复杂度之间的关系 [37] 扩展到电路，可以利用编码定理估计量子电路复杂度，即 $K_{circ}(s) = -\log M_{circ}(s)$。

3.2 布尔门集合

在算法概率的布尔电路形式中，我们将考察由 n 位组成的字符串，以及所有可能输入下，所有可能的布尔电路生成每个比特字符串的概率。主要限制（即输出并非同样均匀随机）源于一个事实：并非所有布尔函数都有对应的原语（单时间步量子门）。我们通常使用通用门集，将任何其他布尔函数编译为由该门集中的更多量子门组成的电路。因此，在实际实现中，我们需要为电路的经验枚举选择一个门集。

给定 v 个输入变量，其符号集大小为 s，这些输入共有 s^v 种可能的组合。If v 存在一个输出变量，其符号集大小为 d，可能函数的总数 [38] 为 d^s.

- 对于 1 输入布尔代数，即当 $v = 1$、$s = 2$、$d = 2$时，函数总数为 $f = 2^{2^1} = 4$。这些函数是{0、1、A、$\overline{A}$}。

- 对于双输入布尔代数，也就是说，当 $v = 2$、$s = 2$、$d = 2$时，函数总数为 $f = 2^{2^2} = 16$。这些函数记作 {0，1，A，B，$\overline{A}$，$\overline{B}$，$A \bullet B$，$\overline{A \bullet B}$，$A + B$，$\overline{A + B}$，$\overline{A + B}$，$A + \overline{B}$，$\overline{A} + B$，$\overline{A} \bullet \overline{B}$，$A \oplus B$，$\overline{A \oplus B}$} 一组功能完备的逻辑连接词或布尔运算符，可通过将集合中的元素组合成布尔表达式，表示所有可能的真值表。这些集合也可以表示任意布尔 SAT 或 SAT-3 公式。这类通用 [39] 门集包括 {与非门}、{或非门}、{非门、与门}、{非门、或门}。这些门集彼此相关，可通过以下等价关系表示：

- 非门(A) = 与非门(A,A) = 或非门(A,A)
- 或门(A,B) = 与非门(与非门(A,A),与非门(B,B)) = 或非门(或非门(A,B),或非门(A,B)) = 非门(与门(非门(A),非门(B)))
- 与门(A,B) = 与非门(与非门(A,B),与非门(A,B)) = 或非门(或非门(A,A),或非门(B,B)) = 非门(或门(非门(A),非门(B)))

3.3 量子门集合

描述布尔函数全貌的经典表述，如今可以推广到包含量子门和状态的情形。在量子逻辑中，有一种作用于 3 个输入的单门，对经典计算具有通用性，

即 CCX 门（也称托福利门）。它可以通过 CCX(A,B,1) = (A,B,NAND(A,B)) 模拟与非门。经典计算通常是不可逆的，因此无法从输出恢复输入。量子逻辑基于幺正演化，因此具有可逆性。此外，量子计算还支持量子叠加和纠缠，而这两者并非可逆计算所必然具备的。CCX 门可以通过模拟扇出门（或复制门）CCX(A,1,0) = (A,1,A)，来模拟整个可逆计算过程。因此，{与非门、扇出门} 和 {CCX} 都构成可逆计算的通用门集。CSWAP 门（也称弗雷德金门）是可逆逻辑的另一种通用门。

将可逆逻辑推广到量子逻辑只需增加一个量子门，即 H 门（哈达玛）。原则上，由 {CCX、H} 组成的实数门集在计算意义上是完备的 [40]。然而，若要在强普适性意义下实现普适性，就需要辅助量子比特来编码实数归一化因子，并借助复代数将 [41] 分解为任意量子幺正门。此外，值得注意的是，若不假设 $|0\rangle$ 和 $|1\rangle$ 两种状态均可用，就无法模拟非门（即量子中的 X 门）的作用。由于我们将从量子比特初始化为全零态的情形开始枚举量子程序，因此需要将门集扩展为 {X、H、CCX}，以便生成所有二进制串作为输出。

算法概率原理在量子情形下也应成立，即：对所有可能函数和所有可能输入态的均匀分布，并不意味着希尔伯特空间中所有可能输出态也呈均匀分布。Nielsen 的几何量子计算（GQC）方法 [42] 表明，寻找最优量子电路本质上等同于在特定的弯曲黎曼几何中寻找两点之间的最短路径。然而，由于需要考虑所有可能的输入态和所有可能的幺正映射，因此无法通过实证方式将其可视化。研究程序合成的图景，需要针对目标量子处理器（或量子编译器）的原生门集对这个空间进行离散化。此外，量子环境中的可能函数或过程数量（即使仅涉及一个量子比特）也不可数无限。因此，在量子控制领域，选择通用门集显得尤为重要。

从某种意义上说，这很简单。已有研究表明，如果能够每次对少量变量反复施加某个哈密顿量，通常就能影响任意数量变量所需的任意幺正时间演化。因此，几乎任何具有两个或更多输入的量子逻辑门在计算上都是普适的 [43]，也就是说，可以将该门的多个副本连接起来，实现任意所需的逻辑电路，并对一组量子变量执行任意所需的幺正变换。我们将这种相较于其经典对应物更丰富的对应形式称为 [11],量子普适性的普遍性（UQU）。

要表示这组更丰富的量子幺正算符，我们需要门集中多少种量子门，又需要多少个？如果提供的是一族参数化量子算符，那么只需少数几种此类算符即可。量子香农分解（QSD） [44]为使用参数化门族 {RY(θ)、RZ(θ)、CX}精确分解量子幺正算符提供了理论下界，并保证其渐近最优性。它可以递归地应用于更大的量子电路，其中 CX 的数量按 $O(4^n)$增长。

GQC、UQU 和 QSD 都依赖于任意的、表达能力强的量子门集合。由于量子设备在制造和控制时，是为了执行来自预定义词典的操作，因此这种做法并不实用。与经典情形相比，使用有限的算子集合存在一个微妙的差异。在经典情形下，d^s 可以由通用门集 G中的一系列量子门精确表示；而在量子情形下，目标是用 G 中的一系列量子门近似所有可能的酉操作，并对近似精度设定界限。可以把这理解为用特定进制的数字表示所有实数。当然，用有限数量的构建块来驾驭这个可数无穷的空间，需要有所取舍。量子 Kolmogorov 复杂度（QKC） [45] 衡量描述一个量子态所需的信息量。对于任何以描述纯量子态所需的经典比特数来衡量量子 Kolmogorov 复杂度的定义，都存在一个纯 n 量子比特态，需要指数级数量的比特来描述。

尽管如此，Solovay-Kitaev 定理（SKT） [46] 使得一种高效的经典算法能够将任意单量子比特门编译为由固定有限集合中的量子门组成的序列。该算法使用通用门集 [47]（例如，{H、T、CX}），运行时间为 $O(log(1/\epsilon))$，并输出一个包含 $O(log(1/\epsilon))$ 个量子门的序列，其近似所需量子门的精度在 $\epsilon > 0$ 以内。该算法还可推广至多量子比特门以及来自 $SU(d)$ 的量子门。

回顾来看，目前尚无已知的基础性理由能够解释为何 GQC、UQU、QSD、QKC 和 SKT 会在自然界中以这种方式呈现。然而，最终它使我们能够通过任意选取一小组构建块，充分解析并探索广阔的希尔伯特空间。下一节将正式阐述我们的枚举流程，并介绍结果及其分析。

4 实现

我们首先介绍经典情形的实现。问题表述如下：给定 (i) n 个比特，$b_i \in \{b_0、b_1、.....、b_{n-1}\} = B$，(ii) 每个比特的初始状态 $s_0(b_i)$（通常设为 0），(iii) 一组量子门

$g \in G$ （不一定是通用的），以及 (iv) QASM 代码的行数 L；求在每个量子门于每个 $l \in L$ 以概率 $\frac{1}{|G|}$ 应用时最终状态的分布。

在量子情形中，门集现在定义为一组酉量子门，而由一组 n 量子比特 Q 构成的初始状态定义为 $s_0(Q) := \sum_{j\in\{0,2^n-1\}} \alpha_j |j$，使得 $\alpha_j \in \mathbb{C}$ 和 $|j$ 是 Z 基下n维希尔伯特空间的本征态。

4.1 门集

我们考虑以下门集：

1. {CCX} - 若提供 $|0$ 和 $|1$ 的初始状态，该集合对于经典计算和可逆逻辑是通用的。若不知道如何从一种初始状态构造另一种状态，就无法实际提供所有初始状态。由于所有基于量子门的量子算法都从全|0状态开始，并通过量子门制备所需的初始状态，因此我们不会在枚举中考虑此集合。2. {X、CCX} - 从全|0状态开始时，该集合对于经典计算和可逆逻辑是通用的。3. {X、H、CCX} - 在编码和辅助量子比特假设下，该集合对于量子逻辑是弱通用的。尽管这种编码具有通用性，但可能无法保留量子计算在计算资源复杂度方面的优势（也就是说，就像经典计算也可以通过{与非门、扇出}对所有量子计算进行编码一样）。因此，我们不会在量子情形的枚举中考虑此集合。4. {H、S、CX} - Clifford 群对量子纠错很有用。不过，它并不通用，而且可以用经典逻辑高效模拟 [48]。该集合上的变换空间对纠错码进行了编码，因此值得映射。5. {H、T} - 该集合对于单量子比特量子逻辑是通用的。不过，接下来我们将考虑通过在集合中添加一个双量子比特量子门，将其推广到多量子比特的情形。6. {H、T、CX} - 该集合对于量子逻辑是通用的。7. {P(pi/4)、RX(pi/2)、CX} - IBM 原生门集用于构造此门集。以下关系说明了它与前述通用门集之间的联系：T = P(pi/4)，X = RX(pi/2)，以及 H = $e^{i\pi/2}$XRz(pi/2)X = $e^{i\pi/2}$XTTTTX。我们还将考虑设备连通性等附加约束，以便将此技术应用于真实的量子处理器。

因此，在我们的实验中，我们映射以下门集对应的最终状态的算法概率：(i){X门、CCX}，(ii) {H、S、CX}，(iii) {H、T、CX}，以及 (iv) {P(pi/4)、RX(pi/2)、CX}。

4.2 评估指标

我们希望评估每个门集的以下指标：

- 表达能力：指通过使用数量不受限制的量子门，希尔伯特空间可被编码的程度。它不按概率加权，因为这是门集编码能力的一项特征。如果某个最终状态可以通过从初始状态出发并应用门集中的一系列量子门得到，我们就为该状态赋值 1。
- 可达性：指一种有界的可表达性形式。量子门序列的长度必须等于或短于指定上限。这对应于物理实现，而非门集的能力，并刻画了处理器的计算复杂度，进而刻画其退相干时间。

可表达性主要被映射以理解可达性界限是否被低估/高估。随着电路长度 L 的值逐渐增加，任何通用门集都将填充可表达性标准中的完整状态景观，从而保持不变。正是在这个极限，即完全可表达性的第一个实例处，可达性得到最好的理解。我们感兴趣的另一个实例是 L的无限极限，及其对可达性分布的影响。

这些实验 过程以及结果的比较研究在以下 g 节中。

4.3 枚举过程

我们通过构建所有可能的 QASM 程序来构建实验。对于门集中的每个门，$g_i \in G$，目标量子比特数 $q(g_i)$ 是已知的。此后，给定 n 量子比特，应用该门的所有可能排列 $\mathcal{P}$ 被枚举在一个列表中，即 $\mathcal{P}^n_{q\ (g_i)}$。QASM 每行的总可能选项是$\sum_G \mathcal{P}^n_{q\ (g_i)}$，因此，对于 L 行代码长度的总可能 QASM 程序是：

$$\left[\sum_G \mathcal{P}^n_{q(g_i)}\right]^L \tag{2}$$

我们的实现可在 github.com/Advanced-Research-Centre/QCircScape 获取

例如，考虑门集 G = {X，CCX}，其中 n = 4 和 L = 3。 q(X) = 1 和 q(CCX) = 3。因此，$\mathcal{P}_q^4$ (X) = 4，以及 $\mathcal{P}_q^4$ (CCX) = 24。请注意，即使交换托福利门两个控制端的分配会产生相同的效果，这也是该门的一种对称性，而对于三量子比特酉算子通常并非如此。因此，这些情况下的描述编号（程序 ID）被视为不同的计算路径。可以看出，这两种托福利门选项在面对各个量子比特特有的噪声特性以及其他控制约束时，表现会大相径庭。对于 QASM，每一行共有 28 种选项，因此长度为 3 时，程序总数为 28^3 = 21952。对于一个小规模示例而言，这已经是一个需要模拟的大量量子电路，也让我们得以预见程序空间会有多大。

通过应用所有可能的情形，我们得到一个大小为 n 的数组，它表示从某一特定状态到另一状态的可用转移数量。测量基（此处视为默认的 Z 基）对本研究至关重要。若考虑大小为 n的比特字符串的所有可能初始状态（Z 基状态制备），我们会得到一个 $n\times n$ 矩阵。探索其他初始状态有助于我们理解量子门对比特值的不对称性（例如，带反转控制量子比特的托福利门的一般化形式有 4 种： ¯C¯CX ¯CCX，C¯CX，CCX）。

在经典情形下（例如，对于 {X、CCX}），这对应于使用集合中的量子门按指定长度排列后，这两个状态之间计算路径数量的统计。对于量子情形，该统计对应于计算路径在测量时坍缩到目标状态的概率之和。将矩阵除以程序总数，即可得到以初始状态为条件、对应于每一行状态的固定长度算法概率。这个归一化的 $n \times n$ 矩阵就是可达性景观。所有非零值都对应于至少通过一条路径可达的状态（即，至少存在一个程序可以将状态转换为该状态）。由此得到布尔可表达性矩阵。

4.4 结果

为开始枚举，我们首先绘制不同门集下程序数量（即公式 2）随量子比特数和电路深度的增长趋势。我们注意到，该趋势与量子门的描述方式无关
在门集中，唯一重要的信息是集合中的每个门作用于多少个目标量子比特。这使我们选定的门集分为两类：（i）包含一个单量子比特门和一个三量子比特门；（ii）包含两个单量子比特门和一个双量子比特门。结果如图 1 所示。我们发现，三量子比特门所产生的排列增长速度远快于另一类门集。

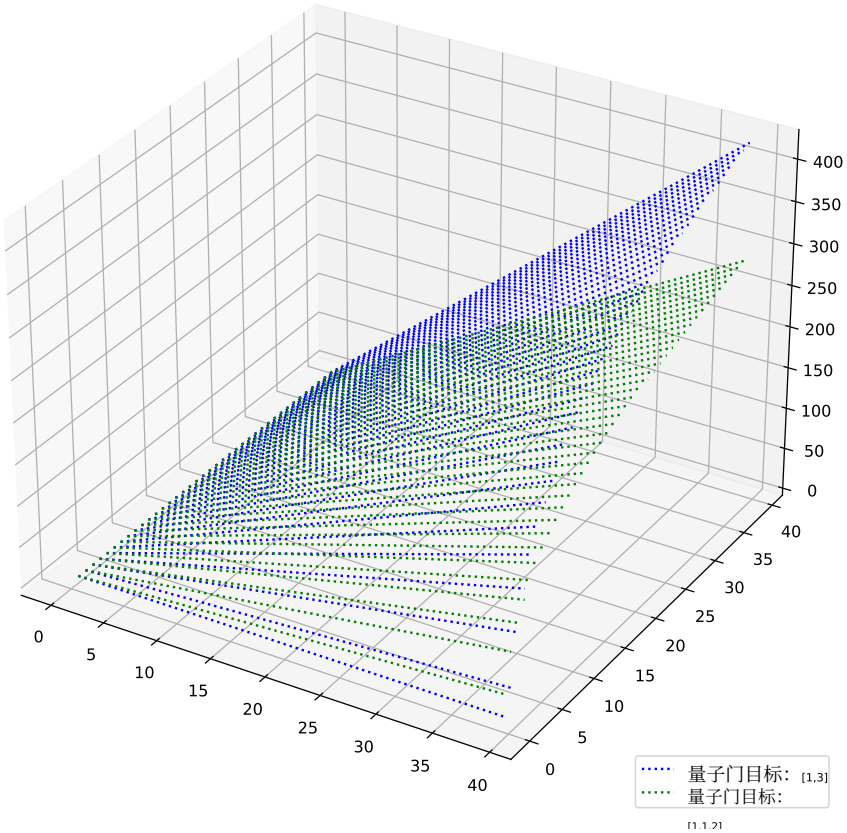


图 1：两类门集下程序数量随量子比特数和电路深度的增长情况：(i)$[1,3]$ 量子比特：{X，CCX}，(ii) $[1,1,2]$ 量子比特：{H，S，CX}，{H，T，CX}，{P(π/4)，RX(π/2)，CX}

以下图示展示了 4 个量子比特在电路深度从 0 增至 4 次操作时，各门集的可表达性（上排）和可达性（下排）。我们考虑的门集包括 {X，CCX}（图2）、{H，S，CX}（图3）、{H，T，CX}（图4）以及 {P(pi/4)，RX(pi/2)，CX}（图5）。

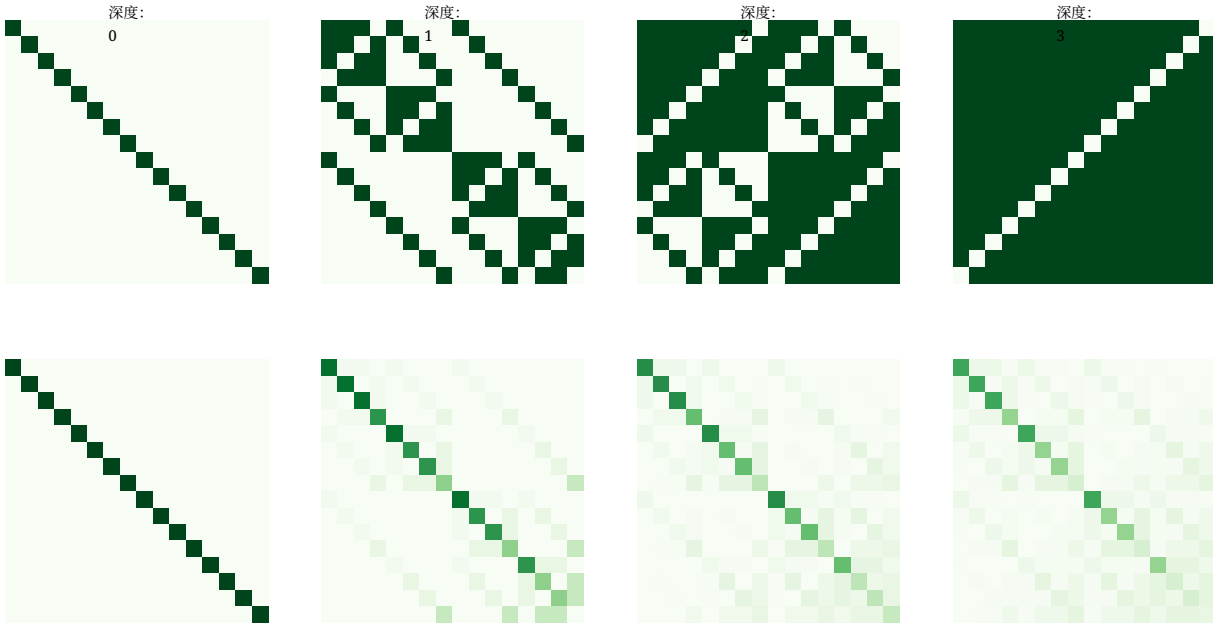


图2：可表达性 能力和可达性：使用 {X，CCX} 门集时，4个量子比特的电路深度为 0 至 3 .

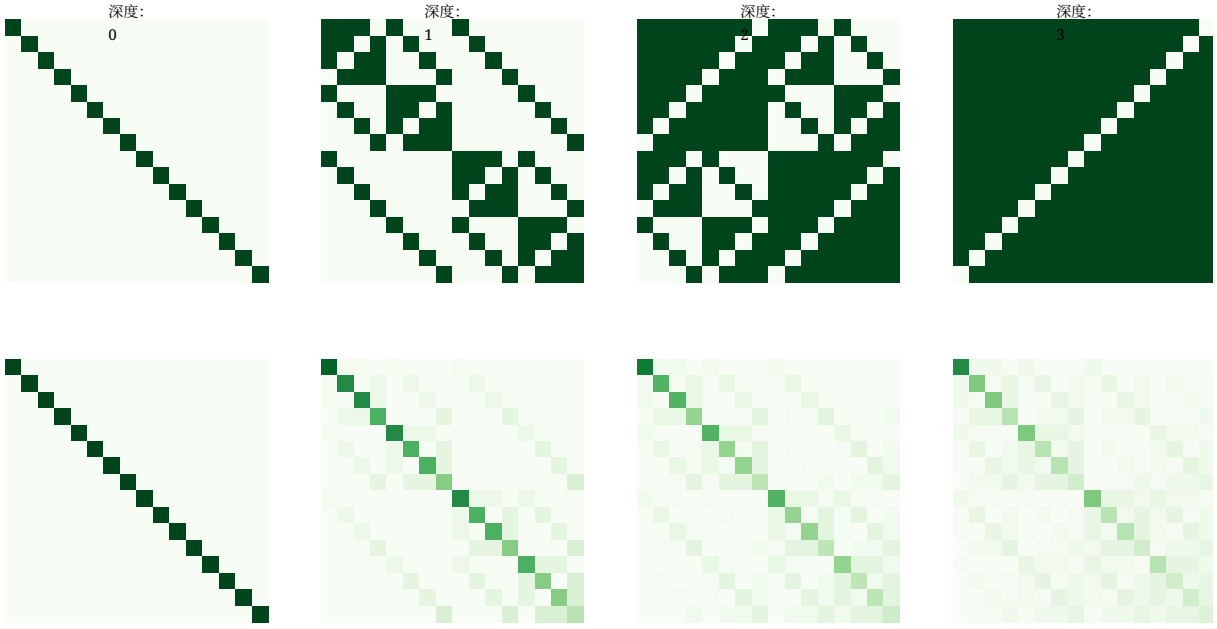


图3：可表 示性和可达性：门集 {H，S，CX} 在 4个量子比特上、深度为 0 到 3 时的情况 .

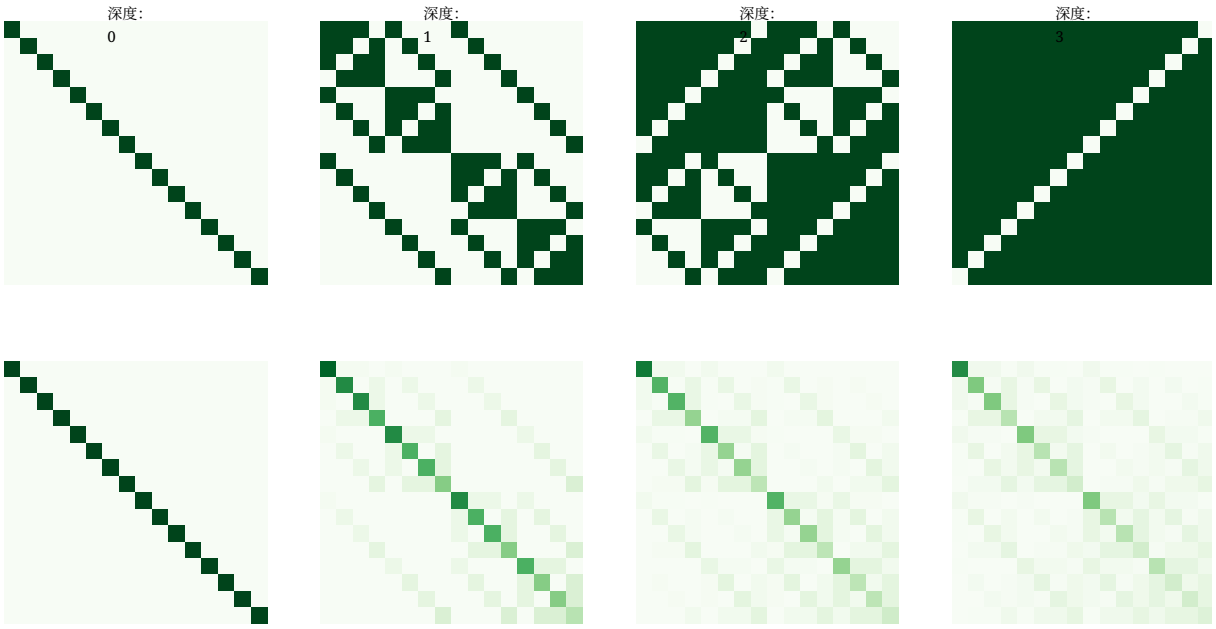


图 4：可表　　达性和门集 {H，T，CX} 在 4个量子比特上、从电路深度 0 到 3 时的可达性　　.

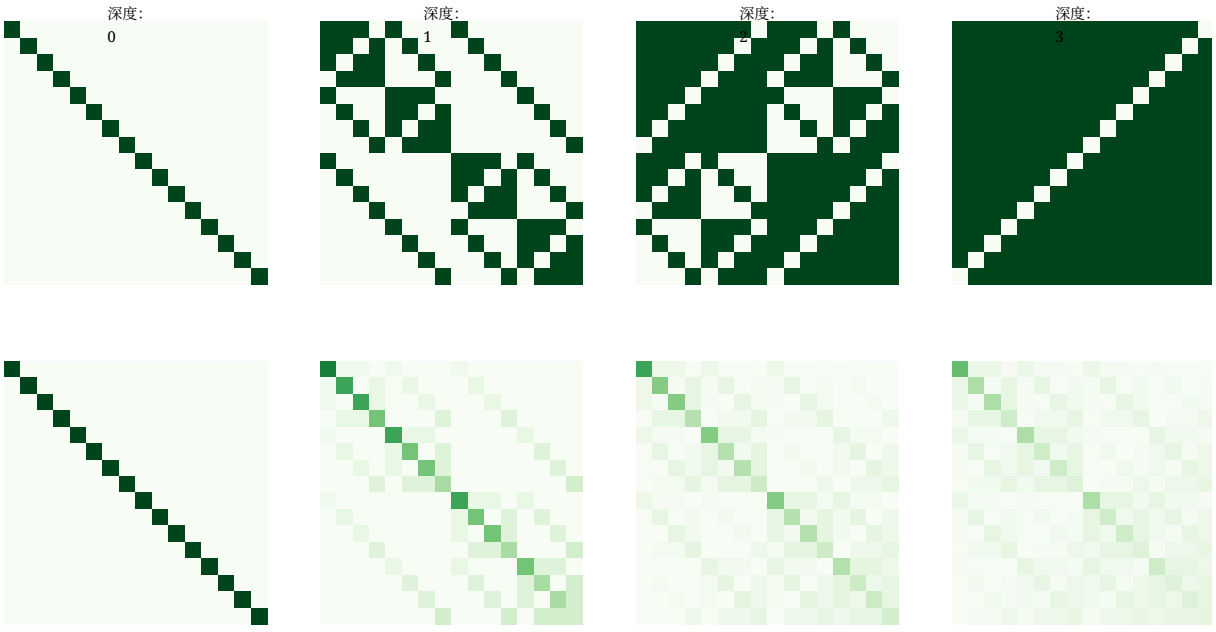


图5：门集 {P(π/4), RX(π/2),CX} 在4个量子比特上的可表达性和可达性，电路深度从0到3。

4.5 分析与讨论

考虑 maxspace = 4 和经典门集 {X、CCX}。每个时间步有 28 种不同的可能性。时间步 0、1、2、3 的 Z基态可达性统计如下：

$$R_0^{\{\mathtt{X},\ \mathtt{CCX}\}} =$$

$$\begin{array}{c}
1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1\,0\\
0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,1
\end{array}$$

$$R_1^{\{\mathtt{X},\ \mathtt{CCX}\}} =$$

$$\begin{array}{c}
24\,1\,1\,0\quad 1\,0\,0\,0\quad 1\,0\,0\,0\,0\,0\,0\,0\\
1\,24\,0\quad 1\quad 0\quad 1\,0\,0\,0\quad 1\,0\,0\,0\,0\,0\,0\\
1\,0\,24\,1\,0\,0\quad 1\,0\,0\,0\quad 1\,0\,0\,0\,0\,0\\
0\,1\,1\,20\,0\,0\,0\,3\,0\,0\,0\,3\,0\,0\,0\,0\,3\,0\,0\,0\,0\,0\quad 1\,0\,0\,0\\
1\,0\,0\,0\,24\,1\,1\,0\,0\,0\,0\,0\\
0\,1\,0\,0\,1\,20\,0\,3\,0\,0\,0\,0\,0\,3\,0\,0\\
0\,0\,1\,0\,1\,0\,20\,3\,0\,0\,0\,0\,0\,3\,0\\
0\,0\,0\,3\,0\,3\,3\,12\,0\,0\,0\,0\,0\,0\,7\\
1\quad 0\quad 0\quad 0\quad 0\quad 0\quad 0\quad 0\,24\,1\quad 1\quad 0\quad 1\quad 0\quad 0\quad 0\\
0\,1\,0\,0\,0\,0\,0\,0\,1\,20\,0\,3\,0\,3\,0\,0\\
0\,0\,1\,0\,0\,0\,0\,0\,1\,0\,20\,3\,0\,0\,3\,0\\
0\,0\,0\,3\,0\,0\,0\,0\,0\,3\,3\,12\,0\,0\,0\,7\\
0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,20\,3\,3\,0\\
0\,0\,0\,0\,0\,3\,0\,0\,0\,3\,0\,0\,3\,12\,0\,7\\
0\,0\,0\,0\,0\,0\,3\,0\,0\,0\,3\,0\,3\,0\,12\,7\\
0\,0\,0\,0\,0\,0\,0\,7\,0\,0\,0\,7\,0\,7\,7\,0
\end{array}$$

$$R_2^{\{\mathtt{X},\ \mathtt{CCX}\}} =$$

$$\begin{array}{c}
580\,48\,48\quad 2\quad 48\quad 2\quad 2\quad 0\quad 48\quad 2\quad 2\quad 0\quad 2\quad 0\quad 0\quad 0\\
48\,580\,2\quad 44\quad 2\quad 44\quad 0\quad 6\quad 2\quad 44\quad 0\quad 6\quad 0\quad 6\quad 0\quad 0\\
48\quad 2\,580\,44\quad 2\quad 0\quad 44\quad 6\quad 2\quad 0\quad 44\quad 6\quad 0\quad 0\quad 6\quad 0\\
2\quad 44\,44\,420\,0\quad 10\,10\,96\quad 0\quad 10\,10\,96\quad 0\quad 0\quad 0\quad 42\\
48\quad 2\quad 2\quad 0\,580\,44\,44\quad 6\quad 2\quad 0\quad 0\quad 0\quad 44\quad 6\quad 6\quad 0\\
2\quad 44\quad 0\quad 10\,44\,420\,10\,96\quad 0\quad 10\quad 0\quad 0\quad 10\,96\quad 0\quad 42\\
2\quad 0\quad 44\,10\,44\,10\,420\,96\quad 0\quad 0\quad 10\quad 0\quad 10\quad 0\quad 96\,42\\
0\quad 6\quad 6\quad 96\quad 6\quad 96\,96\,220\,0\quad 0\quad 0\quad 58\quad 0\quad 58\,58\,84\\
48\quad 2\quad 2\quad 0\quad 2\quad 0\quad 0\quad 0\,580\,44\,44\quad 6\quad 44\quad 6\quad 6\quad 0\\
2\quad 44\quad 0\quad 10\quad 0\quad 10\quad 0\quad 0\quad 44\,420\,10\,96\,10\,96\quad 0\quad 42\\
2\quad 0\quad 44\,10\quad 0\quad 0\quad 10\quad 0\quad 44\,10\,420\,96\,10\quad 0\quad 96\,42\\
0\quad 6\quad 6\quad 96\quad 0\quad 0\quad 0\quad 58\quad 6\quad 96\,96\,220\,0\quad 58\,58\,84\\
2\quad 0\quad 0\quad 0\quad 44\,10\,10\quad 0\quad 44\,10\,10\quad 0\,420\,96\,96\,42\\
0\quad 6\quad 0\quad 0\quad 6\quad 96\quad 0\quad 58\quad 6\quad 96\quad 0\quad 58\,96\,220\,58\,84\\
0\quad 0\quad 6\quad 0\quad 6\quad 0\quad 96\,58\quad 6\quad 0\quad 96\,58\,96\,58\,220\,84\\
0\quad 0\quad 0\quad 42\quad 0\quad 42\,42\,84\quad 0\quad 42\,42\,84\,42\,84\,84\,196
\end{array}$$

$$R_3^{\{\mathtt{X},\ \mathtt{CCX}\}} =$$

$$\begin{array}{c}
14112\,1738\quad 1738\quad 136\quad 1738\quad 136\quad 136\quad 18\quad 1738\quad 136\quad 136\quad 18\quad 136\quad 18\quad 18\quad 0\\
1738\,14100\quad 140\quad 1498\quad 140\quad 1498\,22\quad 336\quad 140\quad 1498\,22\quad 336\quad 22\quad 336\quad 0\quad 126\\
1738\quad 140\quad 14100\,1498\quad 140\quad 22\,1498\,336\quad 140\quad 22\,1498\,336\quad 22\quad 0\quad 336\quad 126\\
136\quad 1498\,1498\,9064\quad 22\quad 532\,532\,2766\quad 22\quad 532\,532\,2766\quad 0\quad 354\,354\,1344\\
1738\quad 140\quad 140\quad 22\,14100\,1498\,1498\,336\quad 140\quad 22\quad 22\quad 0\quad 1498\,336\,336\quad 126\\
136\quad 1498\quad 22\quad 532\,1498\,9064\,532\,2766\quad 22\quad 532\quad 0\quad 354\,532\,2766\,354\,1344\\
136\quad 22\quad 1498\,532\,1498\,532\,9064\,2766\quad 22\quad 0\quad 532\,354\,532\,354\,2766\,1344\\
18\quad 336\quad 336\,2766\,336\,27662766\,4092\quad 0\quad 354\,354\,1572\,354\,1572\,1572\,2758\\
1738\quad 140\quad 140\quad 22\quad 140\quad 22\quad 22\quad 0\quad 14100\,1498\,1498\,336\,1498\,336\,336\quad 126\\
136\quad 1498\quad 22\quad 532\quad 22\quad 532\quad 0\quad 354\,1498\,9064\,532\,2766\,532\,2766\,354\,1344\\
136\quad 22\quad 1498\,532\quad 22\quad 0\quad 532\,354\,1498\,532\,9064\,2766\,532\,354\,2766\,1344\\
18\quad 336\quad 336\,2766\quad 0\quad 354\,354\,1572\,336\,2766\,2766\,4092\,354\,1572\,1572\,2758\\
136\quad 22\quad 22\quad 0\quad 1498\,532\,532\,354\,1498\,532\,532\,354\,9064\,2766\,2766\,1344\\
18\quad 336\quad 0\quad 354\quad 336\,2766\,354\,1572\,336\,2766\,354\,1572\,2766\,4092\,1572\,2758\\
18\quad 0\quad 336\quad 354\quad 336\quad 354\,2766\,1572\,336\quad 354\,2766\,1572\,2766\,1572\,4092\,2758\\
0\quad 126\quad 126\,1344\,126\,1344\,1344\,2758\,126\,1344\,1344\,2758\,1344\,2758\,2758\,2352
\end{array}$$

图中绘制的可达性矩阵按每行代码所有可能选项的总数（即该行的总和）进行了归一化。例如，在此门集中，R_1对应 28，R_2 对应 784，R_3对应 21952。可表达性矩阵是对 R中非零元素进行二值化表示的矩阵。显然，R_0 只有对角线元素，因为创建不含量子门的电路只有一种方式，而且它们实际上将状态映射到自身。然而，使用 R_1，所有其他 R_l 都可以通过矩阵幂运算直接推导得出：

$$R_l^G = (R_1^G)^l \qquad G = \{\mathtt{X}, \mathtt{CCX}\} \tag{3}$$

从网络理论的角度来看，这似乎显而易见：n跳邻居可以通过对邻接矩阵进行幂运算找到。图论技术可用于研究量子电路 [49], ，但真正令人着迷的是，它无需执行每个电路，就能预测状态的分布。在我们的枚举中，我们为每个量子门赋予了相同的权重。可以在量子程序上训练机器学习模型，以找出现实量子算法中量子门的分布，从而用于研究大多数量子算法会生成哪类状态。

现在让我们来构建一种关于 M_{circ} 的概念。在无前缀程序的算法概率表述中，收敛到半测度是基于这样一个概念：无限和 $\lim_{i\to\inf}\sum_i 2^{-i} = \frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\cdots = 1$。在我们的情形中，我们需要确保后续代码行的贡献以类似的方式递减。由此，对于给定的门集和系统规模，我们得到以下关于 M_{circ} 的方程4。

$$M_{circ}^{G,n} = \lim_{L\to\inf}\sum_{i=1}^{L}\left\{2^{-i} * R_i^G * \left[\sum_{g_i\in G}\mathcal{P}_{q(g_i)}^n\right]^{-i}\right\} \tag{4}$$

接下来，我们分析量子通用门集 {H，T，CX}。现在我们需要注意，

$$R_l^G \neq (R_1^G)^l \qquad G = \{\mathrm{H}, \mathrm{T}, \mathrm{CCX}\} \tag{5}$$

我们得到以下数值：

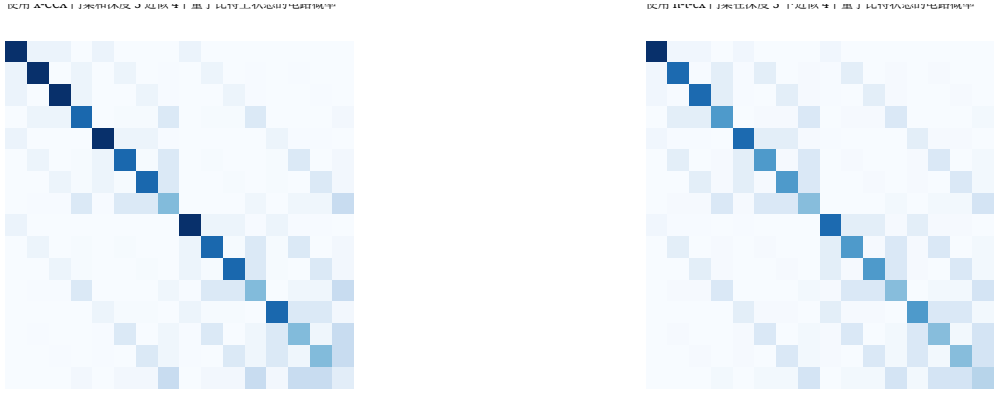
$$R^{\{\mathrm{H},\ \mathrm{T},\ \mathrm{CCX}\}\ 1} =$$

$$R^{\{\mathrm{H},\ \mathrm{T},\ \mathrm{CCX}\}\ 2} =$$

$$(R_1^{\{\mathrm{H},\ \mathrm{T},\ \mathrm{CCX}\}\ 2} =$$

让我们理解其中的原因。如果从状态 $|0$ 开始并施加哈达玛门，我们会得到状态 $\frac{1}{\sqrt{2}}|0 + \frac{1}{\sqrt{2}}|1$。然而，就概率而言，在归一化的可达性矩阵中，这对应于 $0.5\,|0\ + 0.5\,|1$。现在，当再次对该状态施加哈达玛门时，状态会演化回 $|0$，而可达性则重新分配该状态，并保持为 $0.5\,|0\ + 0.5\,|1$。请注意，如果允许在非计算基下进行测量，经典门也会出现类似情况。本质上，这是因为复振幅可以发生相消干涉，而概率不能——这是导致量子加速的核心特征之一 [50] 。

在量子门集场景中， M_{circ} 的定义仍然成立，但已无法在多项式时间内计算，只能像算法概率一样进行近似。我们得到了门集 {X，CCX} 和 {H，T，CX} 的 M_{circ} 近似值，见图 6。



(a) (b)

图 6：使用两种门集（a）{X, CCX} 和（b）{H, T, CX}，对 $L = 3$ 4个量子比特状态的电路概率进行近似

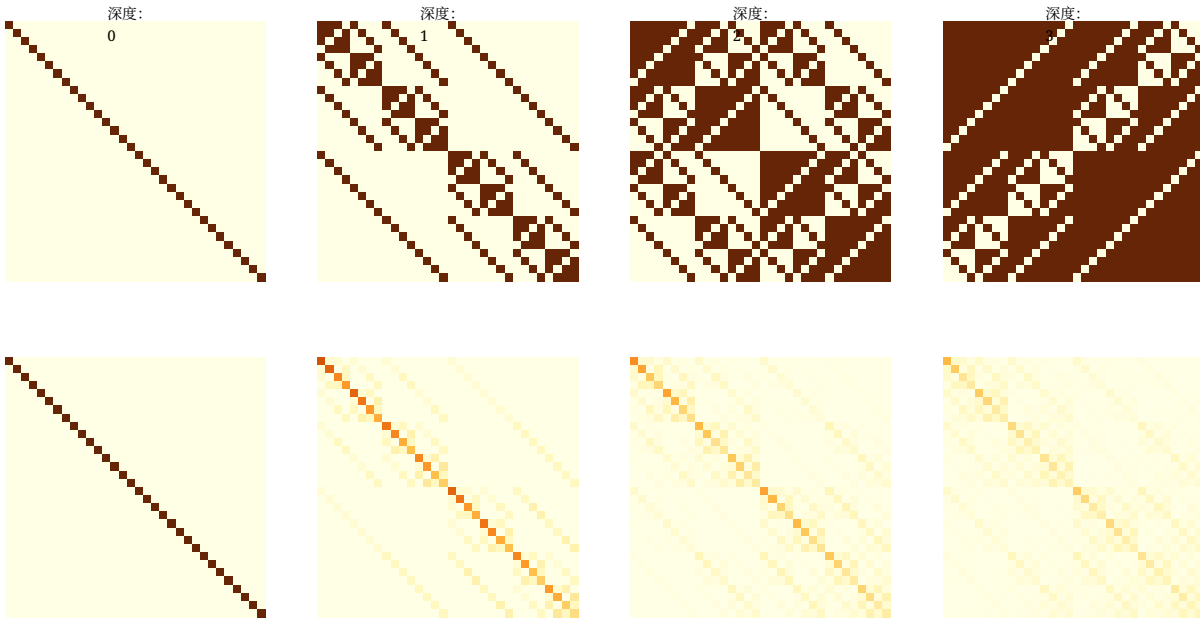


图 7：在 IBM T 拓扑上，5 个量子比特、0 至 3 的电路深度下，门集 {P(π/4)，RX(π/2)，CX} 的可表达性和可达性。

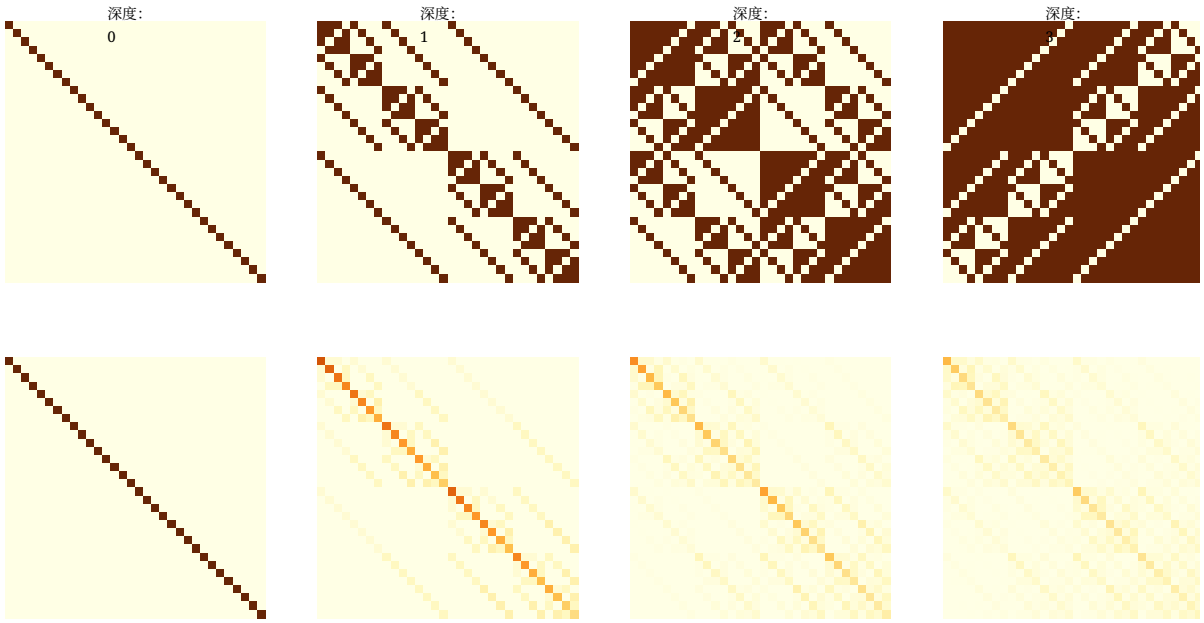


图 8：在 IBM L 拓扑上，5 个量子比特、0 至 3 的电路深度下，门集 {P(π/4)，RX(π/2)，CX} 的可表达性和可达性。

利用这些可达性矩阵，我们现在可以计算两种设备量子比特连接拓扑的 M_{circ} 。随后可以对两者的电路概率进行比较，从而了解哪种设备在可达性分析方面表现更好。图 9 所示的结果表明，对于某些变换，L 型拓扑更优，概率更高，因此电路复杂度更低（即差值图中间的红色部分）；而对于其他变换，T 型拓扑更优（即 M_{circ} 中 T 型的数值更高的绿色部分）。

表达能力图呈现出分形结构，实质上体现了子系统的迹，如下所示：

$$E_i^n = \begin{bmatrix} A_i^n & B_i^n \\ B_i^n & A_i^n \end{bmatrix} = \begin{bmatrix} B_{i+1}^n & A_{i-1}^n \\ A_{i-1}^n & B_{i+1}^n \end{bmatrix}, \quad A_i^n = E_i^{n-1}, \quad B_i^n = E_{i-1}^{n-1} \tag{6}$$

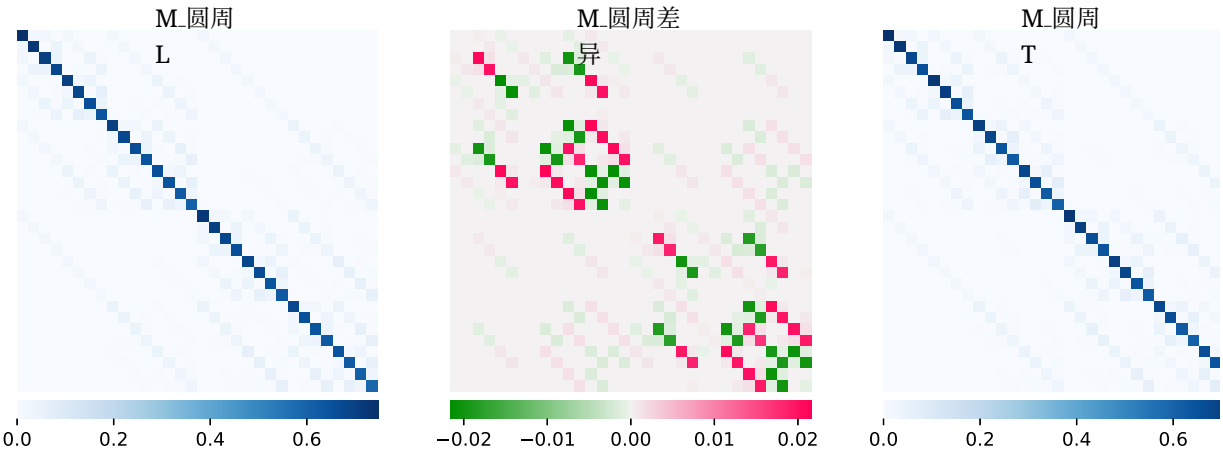


图 9： M_{circ} 及其在 5 个量子比特、0 到 3 的电路深度下，IBM L 拓扑和 T 拓扑上使用门集 {P(π/4)，RX(π/2)，CX} 的比较。

另一个重要发现是，可达性/可表达性分析与门集是否具有强普适性无关（例如 {H，T，CX}），也与门集不具有强普适性无关（例如 {H，S，CX}）。这是因为我们关注的是是否存在一条路径可以在两个状态之间进行映射，而不考虑我们在多大程度上能够控制系统并引导它沿着特定路径演化。举例来说，仅由 {H} 或 {X门} 组成的门集，在深度足够时也能在任意一对状态之间进行映射（弱普适），但显然无法近似表示任何经典函数（如 {与非门}），更不用说所有函数了。在这一框架中，为了定义普适性，我们将函数定义为状态之间的一组变换（例如，和项表达式、概率质量函数或酉矩阵）。通用门集在深度足够时能够近似表示任意此类变换。关于通用门集的进一步讨论将作为本研究的扩展留待后续工作。

5 应用

这项研究的应用主要有两个方面。一方面，它通过枚举来探索希尔伯特空间的特性。以可视化地图呈现研究结果中的结构，有助于直观地理解量子计算的能力。这是对经典逻辑 [51] 和算法信息 [52]领域类似项目的延伸。从更务实的角度看，这项研究适用于多种应用场景。本文最后将简要介绍这项研究与这些应用场景的联系。

5.1 几何量子机器学习

量子过程的整体图景对于量子信息的基础理论和实际应用都具有重要意义。在基础理论方面，量子复杂度理论 [53, 54], 量子资源理论 [55],范畴量子力学 [56] 和量子形式逻辑 [57] 都依赖于这一图景的性质。向实际应用的过渡，则由量子计算的几何表述 [42, 58]推动。近年来，这也成为几何深度学习量子化 [59]的基础。这些研究是在数学函数或量子场 [60]的形式体系中开展的。电路复杂度的研究要少得多，它能够连接算法复杂度与计算复杂度。通过从量子逻辑电路的统计/算法复杂度几何这一视角出发，我们希望在不久的将来借助量子计算框架，使这些结果变得具体可感。另一方面，针对特定应用场景的两个量子状态/过程之间的操作距离度量，也可以借鉴这些理论技术。

5.2 新颖量子算法综合

量子算法设计目前需要谨慎操控量子信息，利用量子力学现象（如叠加、纠缠、干涉）来获得计算优势。这通常与人类的现象学经验相悖，因此需要大量训练，且往往依赖偶然的灵感时刻 [61]。尽管发现新算法是一个蓬勃发展的研究领域 [20]，以数学形式化推理仍是量子加速计算得到更广泛应用的障碍。已有一些自动化量子编程的方案 [21, 22, 23] 被提出，旨在解决这些问题

。为进一步拓展量子算法的适用范围，可以将新颖性搜索 [62]和大型语言模型 [63] 的技术融入这些自动化引擎。正如本研究所展示的，对这一图景进行刻画，可极大助益于在量子过程空间中开展开放式搜索。

5.3 量子人工通用智能

在发展通用智能的较为严谨的方法中，有一种积极的表述方式，即所罗门诺夫归纳智能理论 [34], ，称为通用人工智能 [18]。AIXI 和 KSA 等通用强化学习模型，能够基于与压缩过去观测相对应的最短程序，并最大化给定的奖励函数，进行理性决策或建模环境动态。这些模型既通过采用量子算法（例如 AIXI-q 模型 [64]）实现了量子化，也被应用于量子环境（例如 QKSA 模型 [65]）。

智能的另一个关键方面 [66] 是理解因果关系。因果推断的量子加速 [67, 68, 69] 可受益于对因果预言机概率分布的了解；因果预言机是嵌入问题特定属性的一类量子过程。除了因果推断之外，类似技术还可应用于其他统计关系学习任务，例如概率逻辑网络 [70]和量子变分算法。

通用分布与因果推断都与量子程序的空间密切相关。正如本研究所示，这一空间又取决于特定门集的选择。因此，在通用门集空间中寻求新颖性，能够对特定应用算法的量子程序合成进行元优化。在当前研究中，我们正利用本文奠定的基础，探索自动化量子操作理论的二阶控制论这一方向。

致谢

本项目在 QIntern 2021 项目“量子基础的强化学习智能体”框架下启动。B.G.B.、T.A. 和 A.S. 谨向该项目的组织者及 QWorld 协会致谢。A.K. 的部分研究经费来自波兰国家科学中心（NCN），资助协议编号为 2019/33/B/ST6/02011。A.S. 的研究经费来自荷兰研究理事会（NWO）通过“QuTech 第三部分：基于应用的研究”项目（项目编号：601.QT.001 Part III-C - NISQ）提供。

作者贡献

概念化，A.S.；方法论，A.S.、A.K. 和 B.G.B.；软件，B.G.B. 和 A.K.；撰写–原始草稿准备，A.S.、A.K. 和 B.G.B.；可视化，A.K. 和 T.A.；监督，A.S.。所有作者均已阅读并同意手稿的发表版本。

参考文献

[1] Koen Bertels、Aritra Sarkar 和 Imran Ashraf。量子计算——从 NISQ 到 PISQ。 *IEEE Micro*，41(5):24–32，2021。[2] Koen Bertels、Aritra Sarkar、Thomas Hubregtsen、M Serrao、Abid A Mouedenne、Amitabh Yadav、A Krol 和 Imran Ashraf。量子计算机架构：迈向全栈量子加速器。载于 *2020 Design Automation & Test in Europe* ，，第 – 页。 *Conference & Exhibition (DATE)* 1 6 2020 IEEE，。[3] John Preskill。NISQ 时代及以后的量子计算。 *Quantum*，2:79，2018。[4] Frank Leymann 和 Johanna Barzen。NISQ 时代基于量子门的量子算法的残酷现实。 *Quantum Science and Technology* 5 4 044007, 2020 ，( ): 。[5] Yunong Shi、Pranav Gokhale、Prakash Murali、Jonathan M Baker、Casey Duckering、Yongshan Ding、Natalie C Brown、Christopher Chamberland、Ali Javadi-Abhari、Andrew W Cross 等。通过打破抽象实现资源高效的量子计算。 *Proceedings of the IEEE*，108(8):1353–1370，2020。[6] Rolf Landauer。计算过程中的不可逆性与热量产生。 *IBM journal of research and development*，5 3 183( ): –191，1961。[7] Charles H Bennett。计算的逻辑可逆性。 *IBM journal of Research and Development*，17(6):525–532，1973。[8] Edward Fredkin 和 Tommaso Toffoli。保守逻辑。 *International Journal of theoretical physics*，21(3-4):219–253，1982。[9] Seth Lloyd。计算的终极物理极限。 *Nature*，406(6799):1047–1054，2000。[10] Igor L Markov。关于计算基本极限的界限。 *Nature*，512(7513):147–154，2014。[11] Stephen Wolfram 等。 *A new kind of science*，第 5 卷。Wolfram Media，香槟，2002。[12] David Deutsch。构造子理论。 *Synthese*，190(18):4331–4359，2013。[13] Lucien Hardy。从五条合理公理出发构建量子理论。 *arXiv preprint quant-ph/0101012*，2001。[14] Markus P M ¨ *Quantum* 4 301, uller。无律 2020 之律：从观察者状态出发，通过算法信息论通向物理学。，：。[15] Tommaso Toffoli。作用，或计算的可互换性。载于 *Feynman and computation*，第 349–392 页。CRC Press，2018。

[16] Seth Lloyd 和 Olaf Dreyer。普适路径积分。 *Quantum Information Processing*，15(2):959–967，2016。[17] Ray J Solomonoff。归纳推理一般理论的初步报告。Zator Company，马萨诸塞州剑桥，1960。[18] Marcus Hutter。*Universal artificial intelligence: Sequential decisions based on algorithmic probability*。Springer Science & Business Media，2004。[19] David Elieser Deutsch。量子计算网络。*Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 425 1868 73 90, 1989。[20] Stephen Jordan。量子算法大全。[21] Lee Spector。*Automatic Quantum Computer Programming: a genetic programming approach*，第 7 卷。Springer Science & Business Media，2004。[22] Nils Quetschlich、Lukas Burgholzer 和 Robert Wille。迈向实现量子计算解决方案的自动化框架。*arXiv preprint arXiv:2210.14928*，2022。[23] Aritra Sarkar。量子软件工程自动化：为何？是什么？如何做？*arXiv preprint arXiv:2212.00619*，2022。[24] Claude E Shannon。通信的数学理论。 *The Bell system technical journal*，27(3):379–423，1948。[25] MetaGolfScript - Esolang。[26] Min Jiang、Shunlong Luo 和 Shuangshuang Fu。信道-状态对偶。*Physical Review A*，87(2):022310，2013。[27] 复杂度动物园。[28] Andrei Nikolaevich Kolmogorov。定义“信息量”概念的三种方法。*Problemy peredachi informatsii* 1 1 3 11, 1965，():–。[29] Eric Allender。电路复杂度、 Kolmogorov 复杂度及下界前景。载于 *DCFS*，第 7–13 页。Citeseer，2008。[30] Peter Grunwald 和 Paul Vit´anyi。Shannon 信息与 Kolmogorov 复杂度。 *arXiv preprint cs/0410002*，2004。[31] Lance Fortnow。Kolmogorov 复杂度与计算复杂度。*Complexity of Computations and Proofs. Quaderni di Matematica* 13 2004，,。[32] Aritra Sarkar、Zaid Al-Ars 和 Koen Bertels。利用经典自动机叠加构造通用分布的量子电路设计。 *arXiv e-prints*，页码 arXiv–2006，2020。[33] Hector Zenil、Liliana Badillo、Santiago Hern´andez-Orozco 和 Francisco Hern´andez-Quiroz。类似编码定理的行为，以及资源受限算法概率中通用分布的涌现。 *International Journal of Parallel Emergent and Distributed Systems* 34 2 161 180, 2019，():–。[34] Ray J Solomonoff。归纳推理的形式理论。第一部分。 *Information and control*，7(1):1–22，1964。[35] Aritra Sarkar。*Applications of Quantum Computation and Algorithmic Information for Causal Modeling in Genomics and Reinforcement Learning* 2022。博士论文，代尔夫特理工大学，7 月。[36] Fernando Soler-Toscano、Hector Zenil、Jean-Paul Delahaye 和 Nicolas Gauvrit。根据小型图灵机的输出频率分布计算 Kolmogorov 复杂度。 *PloS one*，9(5):e96223，2014。[37] Jean-Paul Delahaye 和 Hector Zenil。短字符串算法复杂度的数值评估：一窥随机性的内在结构。 *Applied Mathematics and Computation*，219(1):63–77，2012。[38] William Wernick。 *Complete sets of logical functions*。纽约大学，1941。[39] Henry Maurice Sheffer。布尔代数的五条独立公设及其在逻辑常项中的应用。*Transactions of the American mathematical society*，14(4):481–488，1913。[40] Yaoyun Shi。Toffoli 门和受控非门只需少许帮助即可实现通用量子计算。*arXiv preprint quant-ph/0205115* 2002，。[41] Dorit Aharonov。Toffoli 门与哈达玛门具有量子通用性的一个简单证明。 *arXiv preprint quant-ph/0301040*，2003。[42] Michael A Nielsen、Mark R Dowling、Mile Gu 和 Andrew C Doherty。作为几何学的量子计算。 *Science*，311 5764 1133( ): –1135，2006。[43] Seth Lloyd。几乎任意量子逻辑门都是通用的。 *Physical review letters*，75(2):346，1995。[44] Vivek V Shende、Stephen S Bullock 和 Igor L Markov。量子逻辑电路综合。载于 *Proceedings of the 2005 Asia and South Pacific Design Automation Conference* 272 275, 2005，第 –页。[45] Caterina E Mora 、 Hans J Briegel 和 Barbara Kraus。量子 Kolmogorov 复杂度及其应用。*International Journal of Quantum Information* 5 05 729 750, 2007，():–。[46] Christopher M Dawson 和 Michael A Nielsen。Solovay–Kitaev 算法。 *arXiv preprint quant-ph/0505030*，2005。[47] Adriano Barenco、Charles H Bennett、Richard Cleve、David P DiVincenzo、Norman Margolus、Peter Shor、Tycho Sleator、John A Smolin 和 Harald Weinfurter。量子计算的基本量子门。 *Physical review A*，52(5):3457，1995。[48] Daniel Gottesman。量子计算机的海森堡表象。 *arXiv preprint quant-ph/9807006*，1998。[49] Medina Bandi´c、Carmen G Almudever 和 Sebastian Feld。基于交互图的量子基准测试分析，以改进量子电路映射技术。 *arXiv preprint arXiv:2212.06640*，2022。[50] Marc-Olivier Renou、David Trillo、Mirjam Weilenmann、Thinh P Le、Armin Tavakoli、Nicolas Gisin、Antonio Ac´ın 和 Miguel Navascu´es。基于实数的量子理论可以通过实验予以证伪。 *Nature*，600(7890):625–629，2021。[51] Lorenz Demey。元逻辑、元语言与逻辑几何。 *Logique et Analyse*，62(248):453–478，2019。[52] Hector Zenil。一件可计算的不可计算艺术作品：它的扩展或可解释软件空间中的宇宙。*arXiv preprint arXiv:2109.08523* 2021，。[53] Adam R Brown 和 Leonard Susskind。量子复杂度的第二定律。 *Physical Review D*，97(8):086015，2018。[54] Jonas Haferkamp、Philippe Faist、Naga BT Kothakonda、Jens Eisert 和 Nicole Yunger Halpern。量子电路复杂度的线性增长。*Nature Physics*，18(5):528–532，2022。[55] Nicole Yunger Halpern、Naga BT Kothakonda、Jonas Haferkamp、Anthony Munson、Jens Eisert 和 Philippe Faist。量子非复杂性的资源理论。 *Physical Review A*，106(6):062417，2022。[56] John H Selby、Carlo Maria Scandolo 和 Bob Coecke。从图示公设重构量子理论。*Quantum*，5:445，2021。[57] Vaughan R Pratt。广义量子力学的线性逻辑。载于 *Proceedings Workshop on Physics and Computation Dallas* 1993，*IEEE Computer Society*。Citeseer，。

[58] Michael A Nielsen、Mark R Dowling、Mile Gu 和 Andrew C Doherty。最优控制、几何与量子计算。*Physical Review A*，73(6):062323，2006。[59] Elija Perrier、Dacheng Tao 和 Chris Ferrie。面向量子电路与控制的量子几何机器学习。*New Journal of Physics* 22 10 103056, 2020 ，( ): 。[60] Ignacio J General。最大熵原理与量子物理。 *Physical Review E*，98(1):012110，2018。[61] Peter W Shor。量子计算的早期岁月。 *arXiv preprint arXiv:2208.09964*，2022。[62] Joel Lehman 和 Kenneth O Stanley。通过寻找新颖性来高效演化程序。载于 *Proceedings of the 12th annual conference on Genetic and evolutionary computation* 837 844, 2010 ，第 – 页。[63] Joel Lehman 、 Jonathan Gordon 、 Shawn Jain 、 Kamal Ndousse 、 Cathy Yeh 和 Kenneth O Stanley。通过大型模型实现演化。 *arXiv preprint arXiv:2206.08896*，2022。[64] Elliot Catt 和 Marcus Hutter。量子计算算法及其在通用预测中应用的简明介绍。 *arXiv preprint arXiv:2005.03137*，2020。[65] Aritra Sarkar、Zaid Al-Ars 和 Koen Bertels。Oksa：量子知识寻求智能体。载于 *15th International Conference AGI 2022 Seattle WA USA August 19–22 2022 Proceedings* 384 393 、 、 、 、 、 、 、 、，第 – 页。Springer，2023。[66] Alexander Lavin 、 Hector Zenil 、 Brooks Paige 、 David Krakauer 、 Justin Gottschlich、Tim Mattson、Anima Anandkumar、Sanjay Choudry、Kamil Rocki、Atılım G¨une¸s Baydin 等。模拟智能：迈向新一代科学方法。*arXiv preprint arXiv:2112.03235*，2021。[67] Aritra Sarkar、Zaid Al-Ars 和 Koen Bertels。利用量子计算估算算法信息，并应用于基因组学。 *Applied Sciences*，11(6):2696，2021。[68] Giulio Chiribella 和 Daniel Ebler。因果关系识别中的量子加速。 *Nature communications*，10 1( ):1472，2019。[69] Tamal Acharya、Akash Kundu 和 Aritra Sarkar。量子加速因果层析：面向应用的电路考量。 *arXiv preprint arXiv:2209.02016*，2022。[70] Peter Wittek 和 Christian Gogolin。量子增强的马尔可夫逻辑网络推理。 *Scientific reports* 7 1 1 8, , ( ): – 。2017