量子计算设计工具的基础：数组、决策图、张量网络与 ZX 演算

（特邀论文）

Robert Wille[1],[2*], Lukas Burgholzer[3*], Stefan Hillmich[3*], Thomas Grurl[3],Alexander Ploier[3],Tom Peham[3] [1] 设计自动化教席，慕尼黑工业大学，德国 [2] 哈根贝格软件能力中心（SCCH）有限公司，奥地利 [3] 集成电路研究所，约翰内斯·开普勒林茨大学，奥地利 [*]通讯作者：robert.wille@tum.de, lukas.burgholzer@jku.at，stefan.hillmich@jku.at https://www.cda.cit.tum.de/research/quantum/

arXiv:2301.04147v1 [quant-ph] 10 Jan 2023

摘要——量子计算机有望高效解决一些经典计算机永远无法解决的重要问题。然而，要实现这些前景，就需要开发一套全自动化的量子软件栈。这涉及许多复杂任务，包括量子电路的经典模拟、将其编译到特定设备上，以及验证待执行的电路和获得的结果。所有这些任务都极具挑战性，需要高效的数据结构来应对其内在复杂性。从相当直接的数组，到受设计自动化领域启发的决策图，再到张量网络和 ZX 演算，人们提出了多种互补的方法。本文深入探究当今工具的“内部机制”，展示这些方法如何应用于量子电路的模拟、编译和验证等任务。

I. 引言

我们正处于一个全新计算时代的黎明，量子计算机将逐步应用于密码学 [1],化学 [2],医学 [3],物理学 [4], 金融 [5], 和机器学习 [6]等实际领域。在许多情况下，人们认为，量子计算能够高效解决经典计算机无法处理的问题。除了不断发现新的潜在应用外，现有量子计算机的能力也在迅速提升，例如，IBM雄心勃勃的路线图计划到 2023 [7] 年将量子技术扩展至 1000 多个量子比特。

随着量子比特数量增加、相干时间延长，以及操作速度加快、保真度提高，越来越大规模的量子电路可以在实际设备上可靠运行。计算能力的提升也带来了对相应软件解决方案和工具的需求，以帮助用户和开发者充分利用现有硬件。与经典电路和系统的设计类似，要在实际设备上实现概念性的量子算法，需要完成许多复杂的设计任务。其中一些最重要的任务包括：

- 经典模拟：在经典计算机上模拟量子电路的执行，是开发和测试新应用与用例过程中一项极其重要的任务。除了成本较低之外，它还能详细展示量子电路执行期间的量子态，而这些信息在实际量子计算机上运行电路时无法获得。 [8]–[13]
- 编译：与经典电路和系统类似，量子电路最初是在相当高的抽象层次上描述的，需要编译为符合目标设备所有约束条件的表示形式（例如，门集受限和/或连通性受限） [14]–[18]。
- 验证：由于编译会显著改变量子电路的结构，因此确保生成的电路仍能实现原本预期的功能至关重要。为此，人们采用验证（更准确地说，等价性检查）方法来保证其等价性 [17], [19]–[25]。

无论是由于量子态和操作的底层表示具有固有的指数级规模，还是由于自由度数量庞大，这些设计任务都构成了计算上的难题。因此，我们需要高效的数据结构和方法来应对这些挑战。在本文中，我们简要概述了过去提出的各种互补数据结构，并简要讨论了每种结构如何被用于高效解决上述设计任务。我们希望借此帮助感兴趣的读者对现有的不同方法形成直观认识，并提供必要的指引，使其能够深入探索范围广泛的可行方法和解决方案。

本文其余部分安排如下：第二节回顾量子计算的基础知识，并说明如何以直观但效率不高的方式，将量子态和运算表示为一维和二维数组。第三节介绍决策图：在许多情况下，通过利用底层表示中的冗余，决策图能够以更紧凑的方式表示量子态和功能。第四节介绍张量网络的基础知识。张量网络不利用底层表示中的冗余，而是利用某些量子态和算法的拓扑结构。第五节展示 ZX 演算如何通过配备一组强大重写规则的量子电路图形表示法，实现对量子计算的图形化推理。最后，第六节总结全文。

II. 数组

在量子计算中，向量和矩阵通常被认为是表示量子对象最直观的数据结构。这些结构可以直接用数组实现，也可用于设计自动化任务。本文将介绍这一数据结构，并简要介绍量子计算。感兴趣的读者可在 [26] 中找到深入介绍。

与经典比特类似，量子比特 （量子比特）可以处于 0 或 1 态。这些称为计算基态，并使用狄拉克符号写作 $|0\rangle$ 和 $|1\rangle$。此外，它们还可以处于这两个基态的（几乎）任意线性组合（即叠加态）。更准确地说，量子比特的量子态 $|\psi\rangle$ 由 $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$ 给出，其中 α_0, $\alpha_1 \in \mathbb{C}$ 满足 $|\alpha_0|^2 + |\alpha_1|^2 = 1$。两个因子$\alpha_0$ 和 α_1 是振幅，表示量子比特与两个基态各自的关联程度。量子比特测量得到 0 和 1 的概率分别为 $|\alpha_0|^2$ 和$|\alpha_1|^2$。量子比特中的各个振幅从根本上说是不可观测的，而测量是从量子比特中提取信息的唯一方式。

单个量子比特的概念可以推广，用以描述由多个量子比特组成的态，通常称为量子寄存器。一个 n 量子比特寄存器可以取 2^n 个基态，其振幅为 α_0, α_1, $\ldots \alpha_{2^n-1}$，且必须满足归一化约束 $\sum_{i\in\{0,1\}^n} |\alpha_i|^2 = 1$。量子态通常简写为仅包含振幅的状态向量，例如，$\begin{bmatrix}\alpha_{00} & \alpha_{01} & \alpha_{10} & \alpha_{11}\end{bmatrix}^{\mathrm{T}}$ 表示两个量子比特的状态向量。

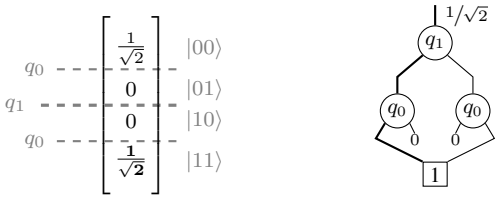
量子态可以通过量子操作进行操控。量子操作本质上是可逆的，并由酉矩阵描述。它们通过矩阵向量乘法作用于量子态。重要的单量子比特操作包括非 $= \left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]$ 操作，它会将量子比特的状态取反；以及阿达马操作 $\mathrm{H} = {}^1\!/\!_{\sqrt{2}}\left[\begin{smallmatrix}1 & 1\\ 1 & -1\end{smallmatrix}\right]$，它能将量子比特从基态变换为叠加态。此外还有多量子比特操作。最重要的双量子比特操作是受控非操作（受控非门）：当且仅当控制量子比特处于状态 $|1\rangle$ 时，它才会将目标量子比特的状态取反。

示例 1. 考虑由两个量子比特组成的量子寄存器 $|\psi\rangle$，其处于量子态 ${}^1\!/\!_{\sqrt{2}} \cdot \begin{bmatrix}1 & 0 & 1 & 0\end{bmatrix}^{\mathrm{T}}$。对第一个量子比特施加控制、以第二个量子比特为目标的受控非门 操作，得到输出量子态 $|\psi'\rangle$，其由下式确定

$$\underbrace{\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}}_{\text{CNOT}} \cdot \underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\1\\0\end{bmatrix}}_{|\psi\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\1\end{bmatrix}}_{|\psi'\rangle}.$$

测量 $|\psi'\rangle$ （也称为贝尔态）会使量子态坍缩，并以概率 $|{}^1\!/\!_{\sqrt{2}}|^2 = {}^1\!/\!_2$分别返回 $|00\rangle$ 或 $|11\rangle$。

上述概念可以通过一种直接的方式实现：向量和矩阵分别用一维数组和二维数组表示。虽然这种表示具有巨大的并发执行潜力，但其内存占用也非常庞大，因为随着所考虑的量子比特数量增加，相关数组的规模会呈指数级增长。因此，



（a）向量　　（b）决策图

图 1. 贝尔态的不同表示

这些内存需求限制了基于数组的模拟方法，使其只能用于规模较小或中等的量子计算（目前的实际极限低于 50 个量子比特 [27]）。

III. 决策图

决策图的总体思路 [28], [29] 是发现并利用所涉及的量子态和操作中的冗余。更准确地说，考虑一个由 n 个量子比特 $q_{n-1}, \ldots, q_1, q_0$组成的量子寄存器，其中$q_{n-1}$ 表示最高有效位量子比特。对应态矢量的前 2^{n-1} 个条目表示 q_{n-1} 为 $|0\rangle$ 时各基态的振幅，其余 2^{n-1} 个条目则表示 q_{n-1} 为 $|1\rangle$时的振幅。在决策图中，这用一个标记为 q_{n-1} 的节点表示，该节点连接到两个标记为 q_{n-2}的后继节点，分别表示零后继和一后继。此过程递归重复，直至剩下大小为 1 的子向量（即单个复数），再将它们连接到终端节点。

在这一分解过程中，相同的子向量由同一个节点表示，从而缩小决策图的整体规模。此外，无需为所有振幅分别设置不同的终端节点，而是使用边权重存储振幅的公因子。将态矢量编码为决策图后，可以通过将对应路径上的边权重相乘来还原特定振幅。为提高决策图的可读性，通常会在可视化中省略值为 1 的边权重；而入边权重为零的节点则显示为零桩，以表明整个子部分均为零。

示例 2. 图 *1* 展示了量子寄存器 $|\psi'\rangle$ 的向量表示和决策图表示。图 *1a* 中态矢量的注释说明了如何构建相应的决策图。要从决策图中还原特定振幅，需要将对应路径上的边权重相乘。例如，还原量子态 $|00\rangle$的振幅（图中的粗线）需要将根边 $({}^1\!/\!_{\sqrt{2}})$ 的边权重与 q_1 *(1)* 的右边以及 q_0 *(1)* 的边权重相乘，即 ${}^1\!/\!_{\sqrt{2}} \cdot 1 \cdot 1 = {}^1\!/\!_{\sqrt{2}}$。

矩阵的决策图表示采用与向量类似的方式构建，通过递归地将矩阵分解为四个部分，而不是将向量分成两半。与底层的向量和矩阵一样，决策图支持乘法和加法，因此可用于不同的设计自动化任务，例如量子电路模拟（例如 [9]）或

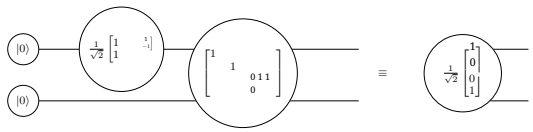


图 2. 用于创建贝尔态的量子电路的张量网络表示

等价性检查（例如 [20]）。
https://iic.jku.at/eda/research/quantum dd/tool/ [30] 提供了一款基于 Web 的可视化工具，可帮助建立对决策图的直观理解。

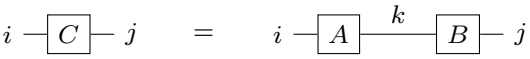
IV. 张量网络

张量网络可以利用量子电路拓扑结构中的冗余，帮助降低基于数组的模拟的复杂度 [31], [32]。要将量子电路转换为张量网络，每个对象，无论是量子态还是操作，都会表示为一个由复数组成的多维数组，即张量，并根据底层量子电路连接到其他张量。从这样的网络中提取有用信息，通常需要将张量两两缩并，直到只剩下一个张量。

例 3. 设 A、B、C 是 $\mathbb{C}^{N\times N}$中的矩阵。此外，矩阵乘积 $C = AB$ 定义为

$$C_{i,j} = \sum_{k=0}^{N-1} A_{i,k} B_{k,j},$$

其中 i、$j = 0$、.....、$N-1$。那么，这对应于对秩为2 的张量 $A = [A_{i,k}]$ 和 $B = [B_{k,j}]$ 沿共享指标 k进行缩并。可以方便地用图形表示为

as: 

所有张量的缩并顺序称为缩并方案。此类方案的主要目标是在计算过程中控制中间张量及其缩并指标的维数（也称为键维数）——事实证明，这是一项 NP困难问题 [33]。因此，人们开发了大量方法来高效确定合适的缩并方案 [34]。

例 4. 再来看看图 *1a* 中的贝尔态。图 *2* 展示了如何将其表示为张量网络。每个张量都用一个包含其实际数据的“气泡”表示。这种表示所需的内存与量子比特和量子门的总数呈线性关系（与基于数组的方法所需的指数级表示形成对比）。另一方面，最终的态矢量大小仍为 2^n，其中 n 表示系统中的量子比特数量。

如示例所示，使用张量网络计算完整的输出态矢量通常难以实现。为降低这种复杂度，人们提出了不同的专用张量网络类型，通过将整体量子态分解为较小的张量来施加特定结构（参见 [35]及其中的参考文献）。

例如，这种方法用于经典量子电路模拟，其中通常需要确定一个标量，例如某个可观测量的期望值或某个单独的



(a) Bell电路　(b) 贝尔态　(c) 类图示

图 3. 贝尔态的 ZX图

振幅。基于张量网络的方法通过固定电路张量网络的输出指标来实现这一点，也就是在电路末端添加“气泡”。对该网络进行收缩后，会得到一个单一的秩零张量——一个标量。只要能控制中间张量的大小和键维数，就能非常高效地完成这一操作。

V. ZX演算

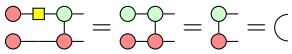
ZX演算 [36], [37] 是一种用于表示量子电路的图形表示法，并配备了一套强大的重写规则，可用于对量子计算进行图形推理。ZX图由相互连接的彩色节点（称为蜘蛛）组成。每个蜘蛛可以是绿色的（Z蜘蛛）或红色的（X蜘蛛），并且可以带有标量相位。没有输入的蜘蛛称为态，而没有输出的蜘蛛称为效应。

ZX图的一个重要概念是只需关注连通性范式，它表达了这样一个事实：如果只需弯折线就能将两个ZX图相互转换，那么这两个图就被视为相等。

示例 5。考虑图 *3a* 中的 *Bell*电路。它与该*ZX*图等价，因为只需交叉或解开线就能将两者相互转换。这里，Hadamard盒是*ZX*图 

的简写，表示  *Hadamard*变换。

要了解这个电路如何作用于$= |0\rangle$ 态，我们可以将它们代入*ZX*图，并使用*ZX*演算进行化简：



最终得到图 *3b* 所示的贝尔态。

任何量子电路都可以解释为ZX图。不过，ZX图比量子电路更通用，还能表示那些无法被有意义地解释为量子电路的对象。正是这种脱离量子电路形式体系的灵活性，使ZX演算成为处理量子电路时一种优秀的中间语言。

虽然ZX演算的基本公理化是量子信息论的一种强大语言，但很难直接应用于自动推理。其原因在于ZX图缺乏范式，而范式是自动重写的一项重要特性。许多使用ZX演算的自动化方法，其基础是采用仅含Z蜘蛛和带有Hadamard盒的线的ZX图的另一种表示形式，这类ZX图称为图样式ZX图。与Bell电路对应的图样式ZX图如图3c所示。针对这些图样式图，还定义了基于图论简化的额外重写规则，从而能够定义终止重写过程——

使终止重写过程的定义成为可能 [38]。

这种自动重写，以及高效处理相位多项式等量子操作的能力，使得许多基于ZX演算的算法被用于解决量子电路设计自动化中的问题，例如编译和优化，以及模拟和验证 [39]–[42]。

VI. 结论

本文简要回顾了数组、决策图、张量网络和ZX演算的基础知识，以及它们在量子计算设计自动化中的应用。这些互补的数据结构各自在内存消耗、性能和概念复杂度之间提供了不同的权衡。针对当前任务选择最合适的数据结构，对于实现高效的量子计算设计自动化工作流程至关重要。我们希望本文能帮助感兴趣的读者直观了解这些数据结构，并提供必要的指引，以便进一步探索丰富多样的可能方法和应用。

致谢

本项目获得欧洲研究理事会（ERC）通过欧盟地平线2020研究与创新计划提供的资助（拨款协议编号：101001318）。本项目是慕尼黑量子谷的一部分；慕尼黑量子谷由巴伐利亚州政府通过巴伐利亚高科技议程Plus提供资金支持。此外，本项目还获得了BMK、BMDW和上奥地利州在COMET计划框架下提供的部分资助（由FFG管理）。

参考文献

[1] P. W. Shor，“量子计算机上用于素因数分解和离散对数的多项式时间算法”，《*SIAM*计算杂志》，1997年。[2] A. Kandala、A. Mezzacapo、等，“面向小分子和量子磁体的硬件高效变分量子本征求解器”，《自然》，第549卷，第7671期，2017年。[3] B. A. Cordier、N. P. D. Sawaya、G. G. Guerreschi和S. K. McWeeney，《量子优势领域中的生物学与医学》，2021年。arXiv: 2112.00760。[4] H.-S. Zhong、H. Wang、等，“利用光子实现量子计算优势”，《科学》，第370卷，第6523期，2020年。[5] D. Herman、C. Googin、等，《金融量子计算综述》，2022年。arXiv: 2201.02773。[6] H.-Y. Huang、R. Kueng、等，《量子多体问题的可证明高效机器学习》，2022年。arXiv: 2106.12627。[7] J. Gambetta，“IBM扩展量子技术的路线图”，*IBM*研究博客，2020年。 [在线]。网址：https://www.ibm.com/blogs/research/2020/09/ibm-quantum-roadmap/。[8] G. F. Viamontes、I. L. Markov和J. P. Hayes，《量子电路模拟》。施普林格，2009年。[9] A. Zulehner和R. Wille，“量子计算的高级模拟”，《*IEEE*集成电路与系统计算机辅助设计汇刊》，2019年。[10] T. Jones、A. Brown、I. Bush和S. C. Benjamin，“QuEST与量子计算机的高性能模拟”，载于《科学报告》，2018年。[11] S. Bravyi和D. Gosset，“改进由克利福德门主导的量子电路经典模拟”，《物理评论快报》，第116卷，第25期，2016年。[12] S. Hillmich、R. Kueng、I. L. Markov和R. Wille，“按需实现精度，尽可能提升效率：基于决策图的量子电路模拟中的近似”，载于欧洲设计、自动化与测试会议，2020年。[13] T. Grurl、J. Fuß和R. Wille，“利用决策图进行噪声感知量子电路模拟”，2022年。[14] A. Botea、A. Kishimoto和R. Marinescu，“论量子电路编译的复杂性”，载于国际组合搜索研讨会，2018年。[15] A. Zulehner、A. Paler和R. Wille，“将量子电路映射到IBM QX 架构的高效方法”，《IEEE集成电路与系统计算机辅助设计汇刊》，2019年。[16] T. H¨aner、D. S. Steiger、K. Svore和M. Troyer，“量子程序编译的软件方法”，《量子科学与技术》，第3卷，第2期，2018年。[17] K. N. Smith和M. A. Thornton，“面向特定技术目标的量子计算编译器与设计工具”，载于国际计算机体系结构研讨会，2019年。[18] G. Li、Y. Ding和Y. Xie，“解决NISQ 时代量子设备的量子比特映射问题”，载于国际编程语言与操作系统体系结构支持会议，2019年。[19] S. Yamashita和I. L. Markov，“量子电路的快速等价性检查”，载于国际纳米尺度架构研讨会，2010年。[20] L. Burgholzer和R. Wille，“量子电路的高级等价性检查”，《IEEE集成电路与系统计算机辅助设计汇刊》，2021年。[21] G. F. Viamontes、I. L. Markov和J. P. Hayes，“检查量子电路与量子态的等价性”，载于国际计算机辅助设计会议，2007年。[22] P. Niemann、R. Wille和R. Drechsler，“多能级量子系统中的等价性检查”，载于国际可逆计算会议，2014年。[23] S.-A. Wang、C.-Y. Lu、I.-M. Tsai和S.-Y. Kuo，“一种基于XQDD 的量子电路验证方法”，载于《IEICE基础汇刊》，2008年。[24] M. Amy，“迈向通用量子电路的大规模功能验证”，载于国际量子物理与逻辑会议，2019年。[25] X. Hong、X. Zhou等，《一种基于张量网络的量子电路表示决策图》，2020年。arXiv: 2009.02618。[26] M. A. Nielsen和I. L. Chuang，《量子计算与量子信息》。剑桥大学出版社，2010年。[27] H. De Raedt等，“大规模并行量子计算机模拟器：十一年之后”，《计算机物理通讯》，第237卷，2019年。[28] P. Niemann、R. Wille等，“QMDD：高效的量子函数表示与操作”，《IEEE集成电路与系统计算机辅助设计汇刊》，2016年。[29] A. Zulehner、S. Hillmich和R. Wille，“如何高效处理复数值？量子计算决策图的实现”，载于国际计算机辅助设计会议，2019年。[30] R. Wille、L. Burgholzer和M. Artner，“量子计算决策图的可视化”，载于欧洲设计、自动化与测试会议，2021年。[31] M. Fannes、B. Nachtergaele和R. F. Werner，“量子自旋链上的有限相关态”，《数学物理通讯》，第144卷，第3期，1992年。[32] J. C. Bridgeman和C. T. Chubb，“信手涂画与诠释性舞蹈：张量网络入门课程”，《物理学杂志A：数学与理论》，2017年。[33] L. Chi-Chung、P. Sadayappan和R. Wenger，“论通过归约实现并行执行的一类多维循环优化”，《并行处理快报》，1997年。[34] J. Gray和S. Kourtis，“超优化张量网络收缩”，《Quantum》，第5卷，2021年。[35] J. I. Cirac、D. P´erez-Garc´ıa、N. Schuch和F. Verstraete，“矩阵乘积态和投影纠缠对态：概念、对称性与定理”，《现代物理评论》，第93卷，第4期，2021年。[36] J. van de Wetering，《面向量子计算机从业者的ZX演算》，2020年。arXiv: 2012.13966。[37] B. Coecke和A. Kissinger，“描绘量子过程”，载于《图解表示与推理》，2018年。[38] R. Duncan、A. Kissinger、S. Perdrix和J. van de Wetering，“利用ZX演算对量子电路进行图论简化”，《Quantum》，第4卷，2020年。[39] A. Kissinger和J. van de Wetering，“利用ZX演算减少T 计数”，《物理评论A》，2020年。[40] A. Kissinger和J. van de Wetering，“利用ZX演算简化的稳定子分解模拟量子电路”，《量子科学与技术》，2022年。[41] N. de Beaudrap、X. Bian和Q. Wang，“受ZX演算启发的 $\pi/4$-奇偶相位电路化简技术”，《电子理论计算机科学会议录》，第318卷，2020年。[42] A. Cowtan、W. Simmons和R. Duncan，《幺正耦合簇试探态的通用编译策略》，2020年。arXiv: 2007.10515。