量子计算：简介、应用与前景

潘屹

摘要

量子计算是基于量子力学理论的全新计算方式，相比起传统计算机的计算模式，量子计算在应对一些特定问题时能带来较大的效率提升。笔者在计算机科学的伟大思想这门课程（CS1950@SJTU）中了解到了基础的量子计算知识，对这一领域产生好奇与兴趣，在课后阅读了量子计算相关材料、书籍，学习量子计算课程（CS294-2@UCBerkeley）的课堂笔记；对量子计算的发展、理论与应用有了初步的认知。本文对当前量子计算的基础理论进行科普性概述，介绍了一些简单的量子算法与应用，并展望量子计算的前景。

目录

1 引入与简介

Alan Turing 在 1936 年发表的论文中提出了图灵机（Turing Machine）的模型 [1]，宣告了现代计算机科学的诞生。Turing 证明了存在一台通用图灵机，即任何可以在个人电脑上执行的算法，都可以在这台图灵机上完成，这个论断被称为 Church-Turing 命题。随后，von Neumann 设计出了这样的理论模型，用实际元件实现了通用图灵机的全部功能，在随后的几十年里，个人计算机的发展也一直沿用 von Neumann 架构，其发展速度遵从 1965 年 Gordon Moore 所概括的 Moore 定律，即集成电路中单位面积的晶体管数量，以及与之相对应的，计算机计算速度，大约每两年增长一倍 [2]。

自 Moore 定律提出以来，经典计算机硬件发展速度都近似地遵从于该定律；但进入 21 世纪以来，Moore 定律的有效性逐渐下降，许多研究人员认为其将在 21 世纪的前 20 年终结，著名芯片企业，Nvidia 公司的首席执行官 Jensen Huang 就于日前宣称，Moore 定律已死 [3]。其中的重要原因在于传统半导体原件在栅极线宽较小时，可能会产生量子隧穿等效应 [4]。

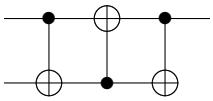
解决 Moore 定律最终失效的一个可能方案是采用不同的计算模式，量子计算就是其中一种。量子计算是一种运用量子力学现象，例如量子叠加、量子干涉与量子纠缠进行计算的方式。量子计算始于 1980 年，物理学家 Paul Benioff 提出了图灵机的量子计算模型 [5]。随后，Feymann 指出，在经典计算机上有效地模拟量子系统的演化似乎是不可能的，量子计算机可能可以模拟经典计算机无法做到的事情 [6]，并引入了早期版本的量子电路符号 [7]。1985 年，David Desutsch 提出，能否用量子力学原理推导出更强的 Church-Turing 命题，并引导出了现代量子计算机的概念 [8]。他用举了一个简单的例子（见 3.1 节），表明量子计算机的计算能力确实超过了传统计算机。

在随后的十年里，对量子算法的研究不断涌现新的成果。1994 年，Peter Shor 提出了一种新的量子算法，可以有效地解决大整数的质因数分解问题 [9]，这在传统计算机上被认为是不可解（难以在多项式时间内解决）的；Shor 算法的出现一度让基于质因数分解的 RSA 加密算法的安全性受到威胁 [10]。1996 年，Lov Grover 证明了，在非结构化搜索空间进行搜索的问题也可以通过量子计算机加速 [11]，这种搜索方法的广泛适用性引起了人们对 Grover 算法的相当关注。

由此可以看出，量子计算与量子算法可以对许多运用传统计算机难以解决的问题进行加速。如今，量子计算领域依旧处于蓬勃发展之中。在下面的章节中，量子计算的相关基础理论将会被介绍。

1.1 量子电路

交换两个量子比特的电路可以被表示为



这是因为容易验证

$$\begin{aligned} |a,b\rangle &\to |a,a\oplus b\rangle \\ &\to |a\oplus(a\oplus b),a\oplus b\rangle = |b,a\oplus b\rangle \\ &\to |b,(a\oplus b)\oplus b\rangle = |b,a\rangle. \end{aligned} \tag{1}$$

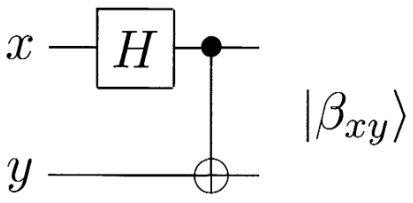
考虑在阿达码门后面跟着一个受控非门, 并对四个二维基态进行计算, 我们会得到

$$\begin{aligned} |\beta_{0,0}\rangle &= \frac{|00\rangle+|11\rangle}{\sqrt{2}} \\ |\beta_{0,0}\rangle &= \frac{|01\rangle+|10\rangle}{\sqrt{2}} \\ |\beta_{0,0}\rangle &= \frac{|00\rangle-|11\rangle}{\sqrt{2}} \\ |\beta_{0,0}\rangle &= \frac{|00\rangle-|10\rangle}{\sqrt{2}} \end{aligned} \tag{2}$$

容易发现

$$|\beta_{x,y}\rangle = \frac{|0,y\rangle+(-1)^x|1,\overline{y}\rangle}{\sqrt{2}}. \tag{3}$$

其量子电路为



2 基础量子算法

量子计算理论以量子物理领域的数学物理方法、记号与公式为重要基础；本节将对相关内容进行简单介绍。

2.1 量子比特与量子比特门

量子比特（qubit）是量子计算和量子信息的基本概念。在传统计算机中，经典比特以 0 和 1 两种状态存在。而量子比特也有相对应的两种形态，在 Dirac 表示法中记作 $|0\rangle, |1\rangle$，任何一个两态的量子系统都可以实现这一点，例如在氢原子中 $|0\rangle, |1\rangle$ 可以代表基态和第一激发态，在质子自旋中可以表示任意方向的 $+\frac{1}{2}, -\frac{1}{2}$ 分量。与经典比特不同的是，量子比特除了 $|0\rangle$ 和 $|1\rangle$ 态，还可以处于叠加态（superposition）；这是 $|0\rangle, |1\rangle$ 两态的一个线性组合，可以记为

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{或列向量形式} \quad |\psi\rangle = \begin{bmatrix}\alpha\\ \beta\end{bmatrix}, \tag{4}$$

其中 $\alpha, \beta \in \mathbb{R}$ （实际上取值域为 $\mathbb{C}^2$，方便起见，这里首先以实数系作为研究对象）且 $|\alpha|^2 + |\beta|^2 = 1$。

在经典比特意义下，n 个比特可以表示 2^n 种不同的状态，在量子比特意义下也是同理，以 $n = 2$ 的情况为例，经典比特有四种情况 $00, 01, 10, 11$，而量子比特则用四个维度 $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ 来表示，可以直接写作两个单量子比特的张量积，例如

$$|\psi_1\rangle = \begin{bmatrix}\alpha_1\\ \beta_1\end{bmatrix}, |\psi_2\rangle = \begin{bmatrix}\alpha_2\\ \beta_2\end{bmatrix} \tag{5}$$

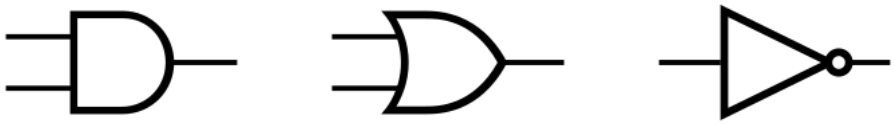
的叠加即为

$$|\psi_1\psi_2\rangle = |\psi_1\rangle \otimes |\psi_2\rangle = \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \alpha_2\beta_1|10\rangle + \beta_1\beta_2|11\rangle = \begin{bmatrix}\alpha_1\alpha_2\\ \alpha_1\beta_2\\ \beta_1\alpha_2\\ \beta_1\beta_2\end{bmatrix}. \tag{6}$$

特殊地，长度为 n 的全零量子比特表示为 $0^{\otimes n}$，全一量子比特表示为 $1^{\otimes n}$。

在经典计算机中，逻辑电路由一系列逻辑门与电路元件构成，图 2.1 即为熟知的与或非逻辑门的表示，它们之间相互嵌套组合，构成了经典电子计算机的电路体系。

而在量子电路中的计算则由一系列逻辑门、测量与赋值操作构成。但是，不同于传统电路是用金属线所连接以传递电压信号或电流信号；在量子线路中，线路是由时间所连接，亦即量子比特的状态随着时间自然演化，一直到遇上逻辑门而被操作。另一方面，经典计算机的大多数基本逻辑门（除了非门）都是不可逆的，例如，对于与门，我们不可能从输出信息的每一位恢复到两个输入信息；而量子计算中的每一步操作则必须由酉



(a) 逻辑与门 (b) 逻辑或门 (c) 逻辑非门

图 2.1: 与或非门（ANSI 及 IEEE 标准）

变换（unitary transformation）来刻画 [12]，变换矩阵 U 满足 $U^\dagger U = I$。下面给出几个基本单量子逻辑门的代数形式。

定义矩阵 X 表示非门

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \mapsto \begin{bmatrix} \beta \\ \alpha \end{bmatrix}. \tag{7}$$

显然地，这表示将单量子比特的两位相互交换。这与传统非门是相似的。

定义矩阵 Z, U 表示相位翻转和旋转

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \qquad \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \mapsto \begin{bmatrix} \alpha \\ -\beta \end{bmatrix}, \tag{8}$$

$$U = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \qquad \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \mapsto \begin{bmatrix} \cos(\alpha+\theta) \\ \sin(\alpha+\theta) \end{bmatrix}. \tag{9}$$

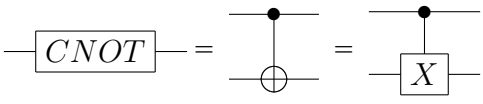
定义矩阵 H 表示 Hadamard 门

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{10}$$

考虑将对单量子比特的操作拓展到多量子比特, 我们从一种简单的情况出发。现在有两个量子比特，但是仅对其中的一个，即目标量子比特进行操作，另一个作为控制量子比特，决定是否对目标进行否运算, 于是容易得到这一操作的酉矩阵

$$U_{\text{CNOT}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{11}$$

当且仅当控制量子比特为 1 时, 目标量子比特通过非门. 可以被表示为 $|A, B\rangle \to |A, B \oplus A\rangle$。因此也被形象地表示为



理论研究证明，任何多量子比特逻辑门可以由受控非门和单量子门组成 []，因此，受控非门具有通用性，这也与经典电路中与非门（XAND）的通用性相对应。

思考：与线性代数知识的关联

基向量，投影与测量

在学习量子计算相关入门知识时，笔者发现其与本科一年级所学线性代数知识的高度关联性，这样就能更方便地从向量空间的意义上去理解量子比特了。例如，在常规使用的量子比特基 $|0\rangle, |1\rangle$ 下，我们有一个态向量

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle.$$

在此前的章节中，我们看到了 $|0\rangle, |1\rangle$ 在现实世界中具有的一些物理意义；那么，式子中的系数 α, β 又对应了什么含义呢？首先，考虑这个态向量在这组正交基上的投影，这也就是量子力学中的测量过程：

$$\overrightarrow{\psi_0} = (\overrightarrow{0})^{\mathrm{T}}\overrightarrow{\psi}\,\overrightarrow{0} \Rightarrow |\psi\rangle^{\dagger}|0\rangle|0\rangle = \langle\psi||0\rangle|0\rangle = \langle\psi|0\rangle|0\rangle = \alpha|0\rangle$$

其中，左式是在线性代数中熟知的投影运算形式 $\boldsymbol{p} = \boldsymbol{u}^T\boldsymbol{v}\boldsymbol{u}$，右式是在量子力学中用 Dirac 记号刻画的更加优美的形式，其中将 $|\psi\rangle^{\dagger}$ 改写为 $\langle\psi|$，进而将 $|\psi\rangle \cdot |0\rangle$ 缩写为 $\langle\psi|0\rangle$，再考虑投影向量长度的平方

$$\alpha\langle 0| \cdot \alpha|0\rangle = \alpha^2.$$

而在 $|1\rangle$ 上的分量长度平方同理则为 β^2，这样一来，约束条件 $\alpha^2 + \beta^2 = 1$ 的意义就逐渐清晰：态向量始终在单位圆上运动，被测量时，其落在 $|0\rangle, |1\rangle$ 上的概率之和恰好为 1.

线性变换

对于一些更加抽象的情况，例如一组新的标准正交基

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

可以从基变换的角度进行理解。根据基坐标变换的公式，变基矩阵为

$$M = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{12}$$

这是一个标准正交阵（当然，在 $\mathbb{C}^n$ 中应为酉阵，但线性代数课程尚未涉及），便有了

$$M^{-1} = M = M^{\mathrm{T}}.$$

这样的规律还可以推广到更加一般的基变换，例如在随后的公式中进行的复向量基变换等，对线性代数基础知识的掌握要求较高。

同时，容易观察到，式 12 与式 10 的形式是相同的，并且有

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} = \begin{bmatrix} \cos(\frac{\pi}{4} - \alpha) \\ \sin(\frac{\pi}{4} - \alpha) \end{bmatrix}$$

代表了将一个 $\mathbb{R}^2$ 中的向量对 $\theta = \frac{\pi}{8}$ 进行对称，从而将 $|0\rangle$ 映射为 $|+\rangle$，$|1\rangle$ 映射为 $|-\rangle$，这样就能制备出等概率落在 $|0\rangle, |1\rangle$ 上的叠加态。

而在从单量子比特到多量子比特的拓展中，

正如在线性代数中对线性映射进行线性变换一样，在量子力学中也可以对量子比特门进行变换，比如通过给单量子比特门 H 进行张量积 $H^{\otimes n}$，就可以制备 n 量子比特的等概率叠加态

$$H^{\otimes n}(|0\rangle^{\otimes n}) = |+\rangle^{\otimes n}.$$

2.2 Deutsch-Jozsa 算法

Desutsch-Jozsa 算法可由一个简单的游戏进行引入：Alice 从 $0 - 2^n - 1$ 中选一个数 x 并将其传送给 Bob。Bob 计算出某个函数 $f(x)$ 的值，可以为 0 或 1，并将它传回给 Alice。已知该函数只有可能有两种情况：要么 $f(x)$ 对于所有的 x 均为常数，要么 $f(x)$ 恰好对于一半的 x 取 0，一半的取 1。Alice 怎样能够最快地判断 $f(x)$ 的类型?

形式化地，已知函数 $f : \{0,1\}^{\otimes n} \to \{0,1\}$ 一定是下列两种极端形式的一种：

1. Constant：$f(x) \equiv 0$ or $f(x) \equiv 1$;
2. Balanced：$f(x) = 0$ for half of $\{0,1\}^{\otimes n}$, and $f(x) = 1$ for the other half

问如何用最少的查询次数确定 f 属于二者中的哪一种。

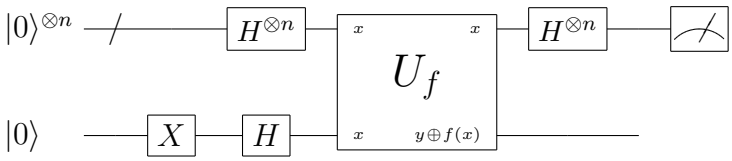


图 2.2: Deustch-Jozsa 算法的量子电路

考虑图 2.2 所示量子电路，输入 U_f 的初态为 $|+\rangle^{\otimes n}|-\rangle$。考虑初态的前 n 位，设其在某一状态下为 x，那么

$$U_f|x\rangle|-\rangle = |x\rangle| - \otimes f(x)\rangle = (-1)^{f(x)}|x\rangle|-\rangle.$$

从而

$$U_f|+\rangle^{\otimes n}|-\rangle = \sum_{x\in\{0,1\}^{\otimes n}} \frac{(-1)^{f(x)}}{\sqrt{2^n}}|x\rangle|-\rangle.$$

另外，容易验证，对于单量子比特门 $H|x\rangle = \sum_{z\in\{0,1\}}(-1)^{xz}|z\rangle/\sqrt{2}$，将这一结果推广到 n 个量子比特上，就有了

$$H^{\otimes n}|x\rangle = \sum_{z\in\{0,1\}^{\otimes n}} \frac{(-1)^{x\cdot z}}{\sqrt{2^n}}|z\rangle.$$

从而

$$H\left(\sum_{x\in\{0,1\}^{\otimes n}}\frac{(-1)^{f(x)}}{\sqrt{2^n}}|x\rangle\right)=\sum_{x,z\in\{0,1\}^{\otimes n}}\frac{(-1)^{x\cdot z+f(x)}}{2^n}|z\rangle\to|\psi\rangle_{\text{measure}}$$

于是考虑测量结果在 $\langle 0^{\otimes n}|$ 上的分量，即 z 只取 $|0\rangle^{\otimes n}$ 时表达式的值

$$\langle 0^{\otimes n}|\psi\rangle=\sum_{x\in\{0,1\}^{\otimes n}}\frac{(-1)^{f(x)}}{2^n}$$

因此，当 f 为常值函数时，测量结果必为 $|0\rangle^{\otimes n}$；当 f 为平衡函数时，测量结果不可能出现 $|0\rangle^{\otimes n}$.

2.3 量子傅里叶变换

在离散傅里叶变换（Discrete Fourier Transform，DFT）中，我们熟知求多项式 $f(x)$ 进行点值 (ω_N^k,y_k) 求值的方法

$$y_k=\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}x_j\omega_N^{jk}.$$

在经典计算机中，可以通过分治加速的方式将这一过程（快速傅里叶变换，Fast Fourier Transform）的时间复杂度优化为 $O(n\log n)$[13]。在量子计算机中，我们同样希望进行相同的变换，使得 $N-1$ 维量子态 $|X\rangle=\sum_{j=0}^{N-1}x_j|j\rangle$，经过变换得到 $|Y\rangle=\sum_{k=0}^{N-1}y_k|k\rangle$，其中 y_k 与 x_k 的关系满足上式，$|j\rangle,|k\rangle$ 表示其二进制分解所得结果的张量积。代入可得

$$|Y\rangle=\sum_{k=0}^{N-1}y_k|k\rangle=\sum_{k=0}^{N-1}\left(\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}x_j\omega_N^{jk}|k\rangle\right)=\sum_{j=0}^{N-1}x_j\left(\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}\omega_N^{jk}|k\rangle\right)$$

对比 $|X\rangle,|Y\rangle$ 的形式，发现变换的实质就是一次基变换

$$|j\rangle\to\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}w_N^{jk}|k\rangle$$

两组标准正交基的变换可改写为酉变换

$$U_{\text{QFT}}=\frac{1}{\sqrt{N}}\begin{bmatrix}1&1&1&\cdots&1\\1&\omega_N&\omega_N^2&\cdots&\omega_N^{N-1}\\1&\omega_N^2&\omega_N^4&\cdots&\omega_N^{2(N-1)}\\1&\omega_N^3&\omega_N^6&\cdots&\omega_N^{3(N-1)}\\\vdots&\vdots&\vdots&\ddots&\vdots\\1&\omega_N^{N-1}&\omega_N^{(N-1)\times 2}&\cdots&\omega_N^{(N-1)(N-1)}\end{bmatrix}$$

更形象化地，考虑直接对 $\omega_{2^n}^{jk} = \exp(\frac{2\pi jk}{2^n})$ 指数中的 $\frac{k}{2^n}$ 进行二进制小数分解，也即对 k 进行二进制分解，得到

$$
\begin{aligned}
|j\rangle \to \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w_N^{jk} |k\rangle &= 2^{-\frac{n}{2}} \sum_{l\in[n], k_l \in \{0,1\}} \exp\left(2\pi j \left(\sum_{l\in[n]} k_l 2^{-l}\right)\right) |k_1, \cdots, k_n\rangle \\
&= 2^{-\frac{n}{2}} \bigotimes_{l\in[n]} \left(\sum_{k_l\in[0,1]} \exp(2\pi j k_l 2^{-l}) |k_l\rangle\right) \\
&= 2^{-\frac{n}{2}} \bigotimes_{l\in[n]} \left(|0\rangle + \exp(2\pi j 2^{-l}) |1\rangle\right) \\
&= 2^{-\frac{n}{2}} \bigotimes_{l\in[n]} \left(|0\rangle + \exp(2\pi \{j \gg l\}) |1\rangle\right)
\end{aligned}
$$

这里的 $\{j \gg l\}$ 表示 j 按位右移 l 位并取其小数部分。

2.4

参考文献

[1] Alan Mathison Turing et al. On computable numbers, with an application to the entscheidungsproblem. *J. of Math*, 58(345-363):5, 1936. 1

[2] Gordon E Moore et al. Cramming more components onto integrated circuits, 1965. 1

[3] Wallace Witkowski. ”moore’s law’s dead,” nvidia ceo jensen huang says in justifying gaming-card price hike, 9 2022. 1

[4] Suhas Kumar. Fundamental limits to moore’s law. *arXiv preprint arXiv:1511.05956*, 2015. 1

[5] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of statistical physics*, 22(5):563–591, 1980. 1

[6] Richard P Feynman. Simulating physics with computers, 1981. *International Journal of Theoretical Physics*, 21(6/7), 1981. 1

[7] Richard P Feynman. Quantum mechanical computers. *Found. Phys.*, 16(6):507–532, 1986. 1

[8] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985. 1

[9] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994. 1

[10] David Mermin. Breaking rsa encryption with a quantum computer: Shor’ s factoring algorithm. *Lecture notes on Quantum computation*, pages 481–681, 2006. 1

[11] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996. 1

[12] John Von Neumann. *Mathematical foundations of quantum mechanics: New edition*. Princeton university press, 2018. 2.1

[13] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965. 2.3