# 量子计算
## —算法篇

# Quantum Computing

网址：www.qubits.top

作者：Calvin Tang

邮箱：179209347@qq.com

# 介绍

**教程简介：**
- 面向对象：量子计算初学者
- 依赖课程：线性代数，量子力学（非必需）

**知乎专栏：**
https://www.zhihu.com/column/c_1501138176371011584

**Github & Gitee 地址：**
https://github.com/mymagicpower/qubits
https://gitee.com/mymagicpower/qubits



Calvin, QQ: 179209347 Mail: 179209347@qq.com

# HHL量子算法背景概述

**HHL量子算法解决的问题：**

线性方程组问题可定义为：给定矩阵 $A \in C^{n\times n}$ 和向量 $b \in C^n$，找到 $x \in C^n$ 满足 $Ax=b$ 。
而HHL量子算法所要解决的问题就是**求解线性方程**。

**HHL量子算法介绍：**

HHL量子算法（以三位创始人Aram Harrow，Avinathan Hassidim和Seth Lloyd命名）是一个用量子计算机解决线性问题（求解线性方程组）$Ax = b$ 最优解的算法，广泛的被应用于许多量子机器学习算法中（如支持向量机SVM，主成分分析PCA等）。

- HHL算法相对于经典算法有着指数级的加速
- 但经典算法可以返回精确解，而HHL算法只能返回近似解

Calvin, QQ: 179209347 Mail: 179209347@qq.com

# HHL量子算法背景概述

**对输入 $A$ 和 $b$ 的要求：**

- 首先要求 $n \times n$ 的矩阵 $A$ 是一个**厄米矩阵**（Hermitian Matrix），即自共轭矩阵（即A的共轭转置矩阵等于它本身 $A^\dagger = A$）
- 其次输入b是一个单位向量。(当 $A$ 不是**厄米矩阵**时，原算法作者在文中也给出了构造**厄米矩阵**的方法)

令 $C = \begin{bmatrix} 0 & A \\ A^\dagger & 0 \end{bmatrix}$ 求解 $C\,y = \begin{bmatrix} b \\ 0 \end{bmatrix}$ 可得 $y = \begin{bmatrix} 0 \\ x \end{bmatrix}$

Calvin, QQ: 179209347 Mail: 179209347@qq.com

# 实对称矩阵 $A^T = A$

实对称矩阵是所有元素均为实数的对称矩阵，矩阵的转置与自身相同：

$A^T = A$ $\quad A^T$ 的意思是转置

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

N 阶实对称方阵具有以下重要性质：

性质 1：所有特征值均为实数
性质 2：所有特征向量均为实向量
性质 3：具有 n 个互不相同的特征值
性质 4：具有 n 个线性无关的特征向量
性质 5：不同特征值对应的特征向量之间是正交的

# 厄米矩阵 $A^\dagger = A$

希尔伯特空间中的厄米矩阵：　$A^\dagger = A$　　$A^\dagger$ 的意思是转置共轭

$A^\dagger = A = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$ **实向量空间**中的矩阵表示：

$$\left[\begin{array}{cc|cc} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ \hline 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right]$$

$A^T = A = \begin{bmatrix} 0 & i \\ i & 0 \end{bmatrix}$ **实向量空间**中的矩阵表示：

$$\left[\begin{array}{cc|cc} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ \hline 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right]$$

没有没有共轭条件，其等价的实数矩阵并不对称！

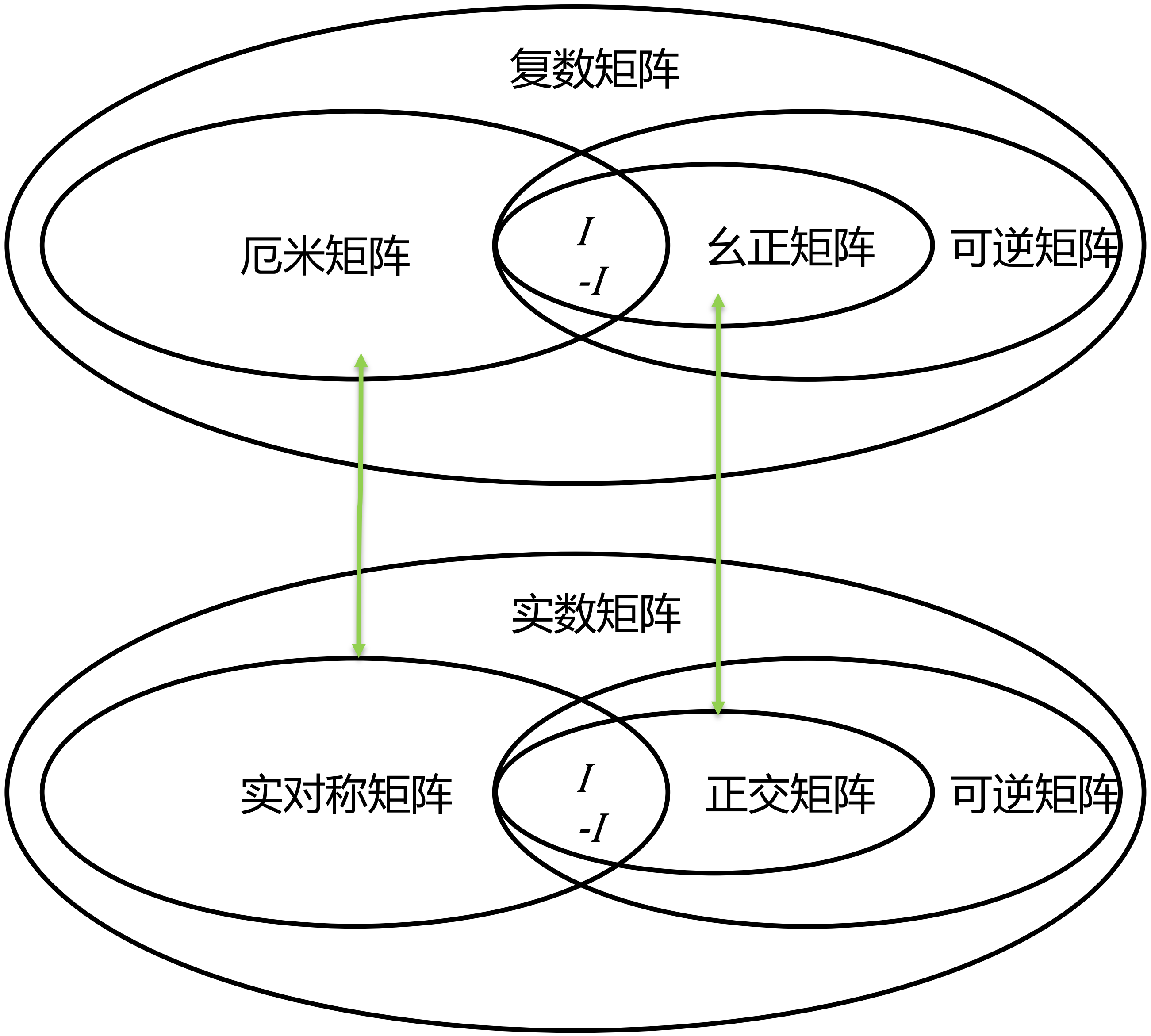
结论：1、厄米矩阵 $A^\dagger$ 其等价的实数矩阵是实对称矩阵。

2、所以厄米矩阵具有实对称矩阵同样性质。

3、唯独性质 2不满足，即所有特征向量均为实向量不成立，有可能是复向量。

# 矩阵类型对比

通过这张图，我们可以清晰的看到，
实向量空间与希尔伯特空间之间的
对应关系：
厄米矩阵对应实对称矩阵。



# 特征分解

**特征分解**（Eigen decomposition），**又称谱分解**（Spectral decomposition）是将矩阵分解为由其特征值 $\{\lambda_i\}$ 和特征向量 $\{v_i\}$ 表示的矩阵 $V = [\, v_1 \;\; v_2 \;\; \ldots \;\; v_n \,]$ 之积的方法：

$$A = V \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} V^{-1}$$

需要注意只有对可对角化矩阵才可以施以特征分解。特征值的集合 $\{\lambda_i\}$，也称为“谱”（Spectrum）。

因为**厄米矩阵（**表达自伴算子的矩阵是**厄米矩阵）**属于正规矩阵，根据正规矩阵的性质可知，其可以对角化。假设 $A$ 是一个复数域正规矩阵，它的特征值为 $\{\lambda_i\}$，标准正交基为 $\{|e_i\rangle\}$，那么 $A$ 可以分解为：

$$A = \sum_{i=1}^{n} \lambda_i |e_i\rangle\langle e_i|$$

# HHL算法原理

以下内容中将默认A为**厄米矩阵**（自共轭矩阵），所以可以进行谱分解。
将向量 $b$，$x$ 分别归一化后采用编码到振幅上的方式映射到量子态 $|b\rangle, |x\rangle$，原问题转换为 $A|x\rangle = |b\rangle$.

对矩阵 $A$ 进行谱分解有：

$$A = \sum_{j=0}^{N-1} \lambda_i |\mu_j\rangle\langle\mu_j|, \quad \lambda_j \in R$$

其中 $\lambda_j$，$\mu_j$ 为矩阵 $A$ 特征值及相应的特征向量，将 $|b\rangle$ 以特征向量 $|\mu_j\rangle$ 为基，线性组合表示得到：

$$|b\rangle = \sum_{j=0}^{N-1} b_j |\mu_j\rangle, \quad b_j \in C$$

于是原方程组的解 $|x\rangle$ 为：

$$|x\rangle = A^{-1}|b\rangle = \sum_{j=0}^{N-1} b_j A^{-1} |\mu_j\rangle = \sum_{j=0}^{N-1} \frac{b_i}{\lambda_j} |\mu_j\rangle$$

$$A|\mu_j\rangle = \lambda_j|\mu_j\rangle$$
$$\rightarrow A^{-1}A|\mu_j\rangle = A^{-1}\lambda_j|\mu_j\rangle$$
$$\rightarrow |\mu_j\rangle = A^{-1}\lambda_j|\mu_j\rangle$$
$$\rightarrow A^{-1}|\mu_j\rangle = \frac{1}{\lambda_j}|\mu_j\rangle$$

求解量子态 $|x\rangle$，可以转换为由量子态 $|b\rangle$ 构造。

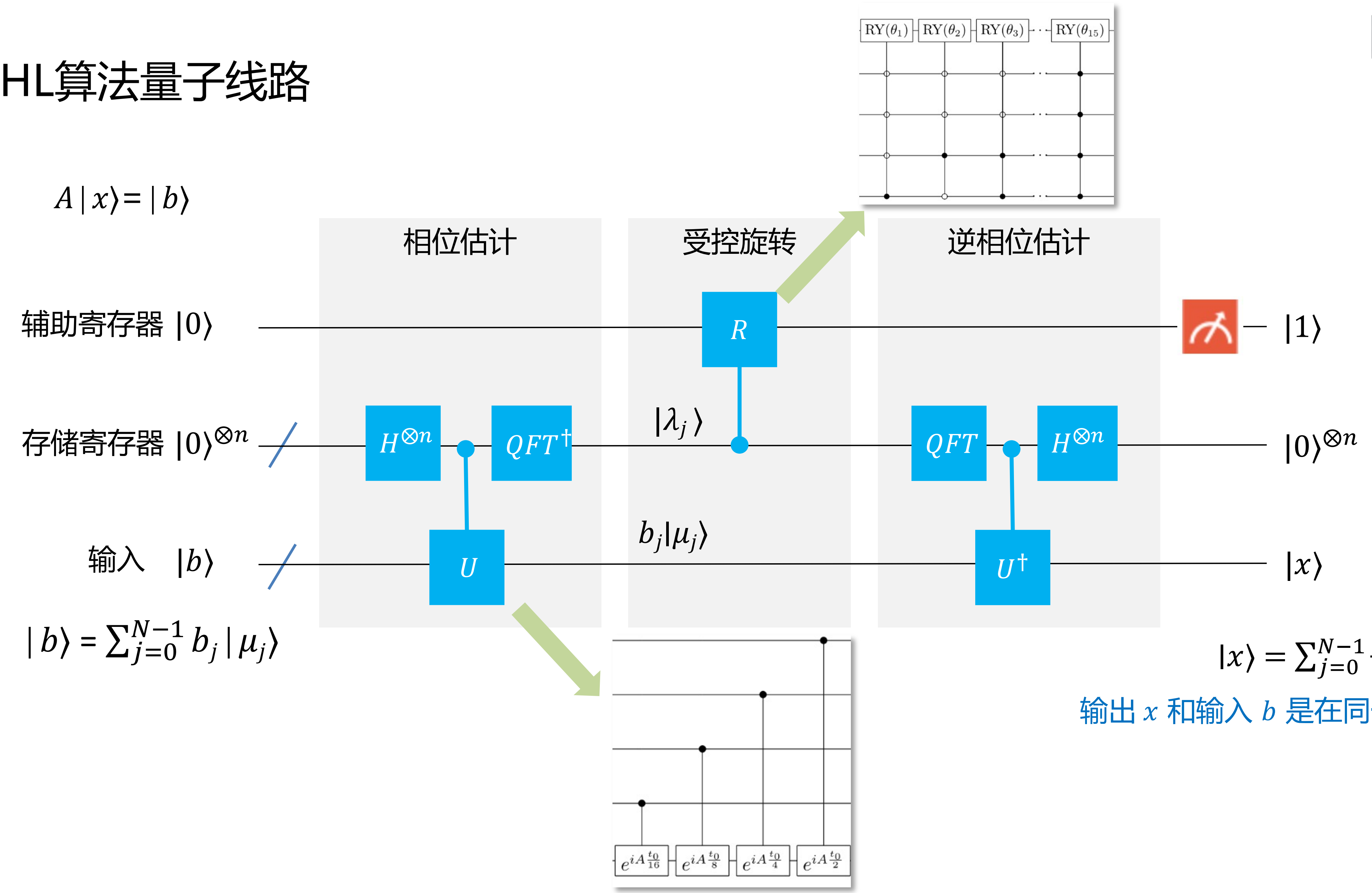
# HHL算法原理 - 三个核心步骤

HHL算法主要包含了三大步骤，并需要使用三个寄存器：

- **相位估计，**将矩阵 $A$ 的整数形式特征值全部转移到存储比特的基向量中。
- **受控旋转，**利用受控旋转门将 $\lambda_j$ 将特征值从存储比特的基向量转移到振幅上，其中 c 是可调参数。
- **逆相位估计，**对特征存储比特及右端项比特进行逆相位估计，将存储比特振幅上的特征值合并到右端项比特上，当辅助比特测量得到特定状态时，在右端项比特上可得到解的量子态。

# HHL算法量子线路

$$A|x\rangle = |b\rangle$$




$$|b\rangle = \sum_{j=0}^{N-1} b_j |\mu_j\rangle$$

$$|x\rangle = \sum_{j=0}^{N-1} \frac{b_j}{\lambda_j} |\mu_j\rangle$$

输出 $x$ 和输入 $b$ 是在同一个寄存器中

Calvin，QQ: 179209347 Mail: 179209347@qq.com

# HHL算法过程 – 初态制备（1/4）

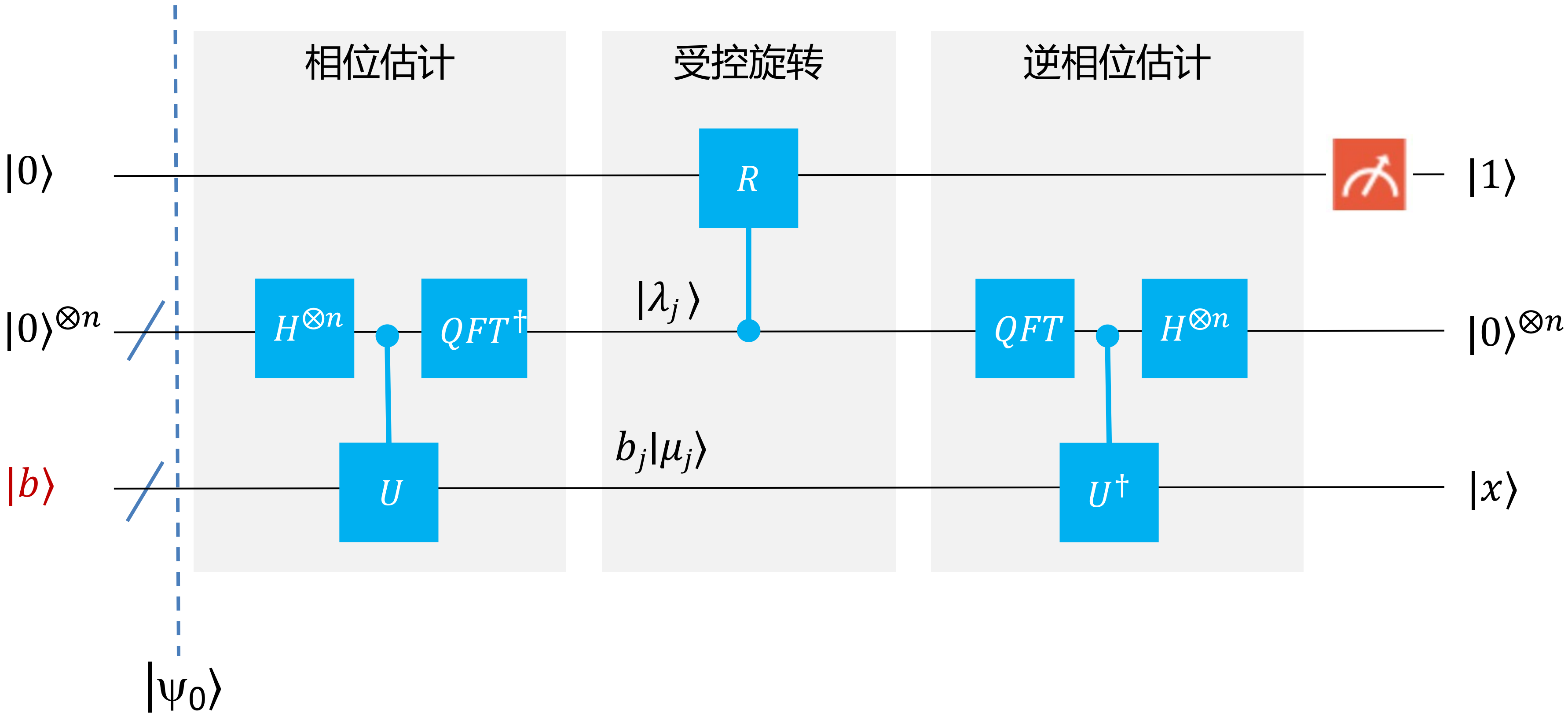
**初态制备：**

$|b\rangle = \sum_{j=0}^{N-1} b_j \, |j\rangle$

其中：

$b = ( b_0 \, , \ldots b_{N-1} )$

$\sum_{j=0}^{N-1} |b_j|^2 = 1$



# HHL算法过程 - 相位估计（2/4）

## 相位估计（QPE）：通过QPE提取特征值

为了将矩阵 $A$ 的特征值提取到解量子态的振幅，首先需要完成特征值的提取。 $QPE$ 量子线路可以用于特征值提取。

对 $|0\rangle^{\otimes n}|b\rangle$ 进行一次 $QPE$ 操作，得到：

$$QPE(|0\rangle^{\otimes n}|b\rangle) = \sum_{j=0}^{N-1} b_j \, |\tilde{\lambda}_j\rangle|\mu_j\rangle$$

其中 $\tilde{\lambda}_j$ 是对应特征值 $\lambda_j$ 的近似整数。 于是矩阵A的特征值信息存入到了基向量 $|\tilde{\lambda}_j\rangle$ 中。

参考来源：https://pyqpanda-toturial.readthedocs.io/zh/latest/HHL.html

Calvin，QQ: 179209347 Mail: 179209347@qq.com

# HHL算法过程 - 相位估计（2/4）

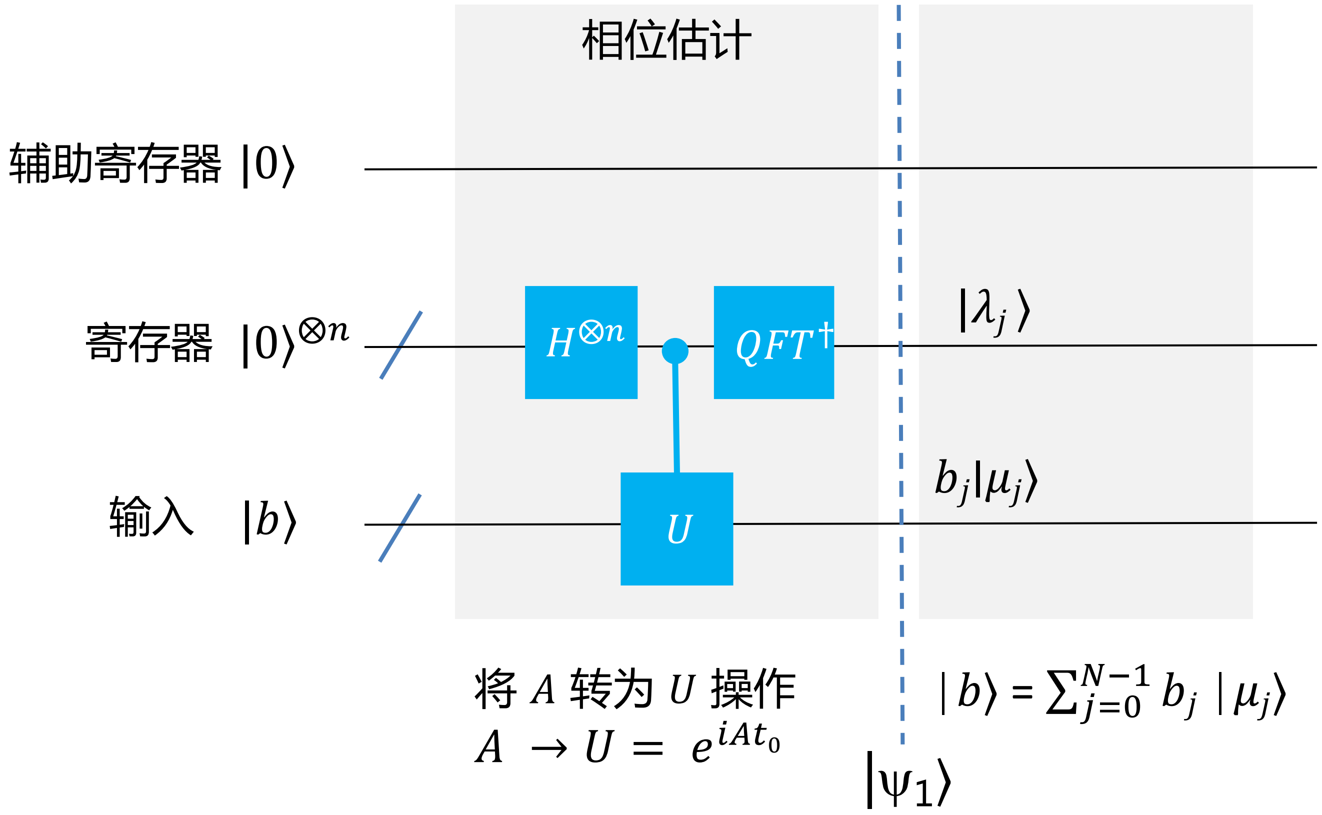
## 相位估计（QPE）：通过QPE提取特征值

- 中间的寄存器会存储一系列的特征值 $\lambda_j$（存储在基态 $|\lambda_j\rangle$ 中）
- 而底部寄存器存储的输入 $|b\rangle$ 会在 $A$ 的特征空间上进行分解，表示为：

$$|b\rangle = \sum_{j=0}^{N-1} b_j\,|\mu_j\rangle$$

$|\mu_j\rangle$ 是 $A$ 的特征向量，$\lambda_j$ 为对应特征值。

输入 $b$ 存放在底部寄存器中，输入 $A$ 作为相位估计中酉算子的一个组成部分。




$$|\psi_1\rangle = \sum_{j=0}^{N-1} |0\rangle|\lambda_j\rangle b_j|\mu_j\rangle$$

Calvin，QQ: 179209347 Mail: 179209347@qq.com

# HHL算法过程 - 受控旋转（3/4）

## 受控旋转：通过受控旋转转移特征值

相位估计的第二阶段把存储在概率幅上的值提取到了基态上。而在HHL算法用到了类似的技巧（不过是反向的），即通过受控旋转操作，**将基态中的值提取到了概率幅上。**

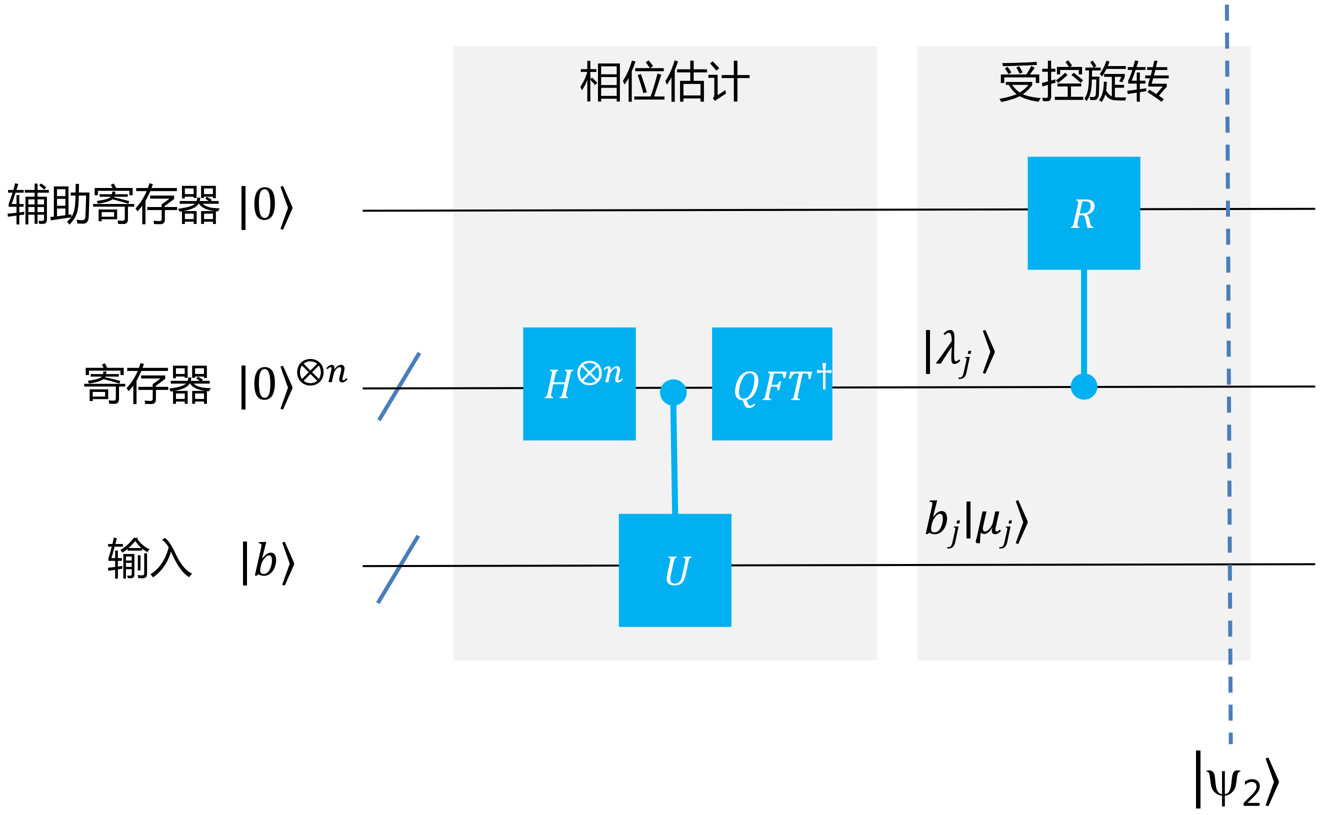
以 $|\lambda_j\rangle$ 作为控制比特对辅助量子比特进行旋转，实现了将基态值的倒数按比例提取到了对应基态的概率幅上。

$$\sqrt{1-\frac{C^2}{\lambda_j{}^2}}\ |0\rangle+\frac{C}{\lambda_j}|1\rangle$$

$C$为归一化系数

令 $f(\lambda_j)= C/\lambda_j$，将辅助量子比特由基态 $|0\rangle$ 映射到 $|0\rangle$ 和 $|1\rangle$ 的叠加态上，同时将函数值 $f(\lambda_j)$ 提取到了基态 $|1\rangle$的概率幅上，如下所示：

$$R\ |0\rangle=\sqrt{1-f(\lambda_j)^2}\ |0\rangle+f(\lambda_j)|1\rangle$$




$$|\psi_2\rangle=\sum_{j=0}^{N-1}\left(\sqrt{1-\frac{C^2}{\lambda_j{}^2}}\ |0\rangle+\frac{C}{\lambda_j}|1\rangle\right)|\lambda_j\rangle b_j|\mu_j\rangle$$

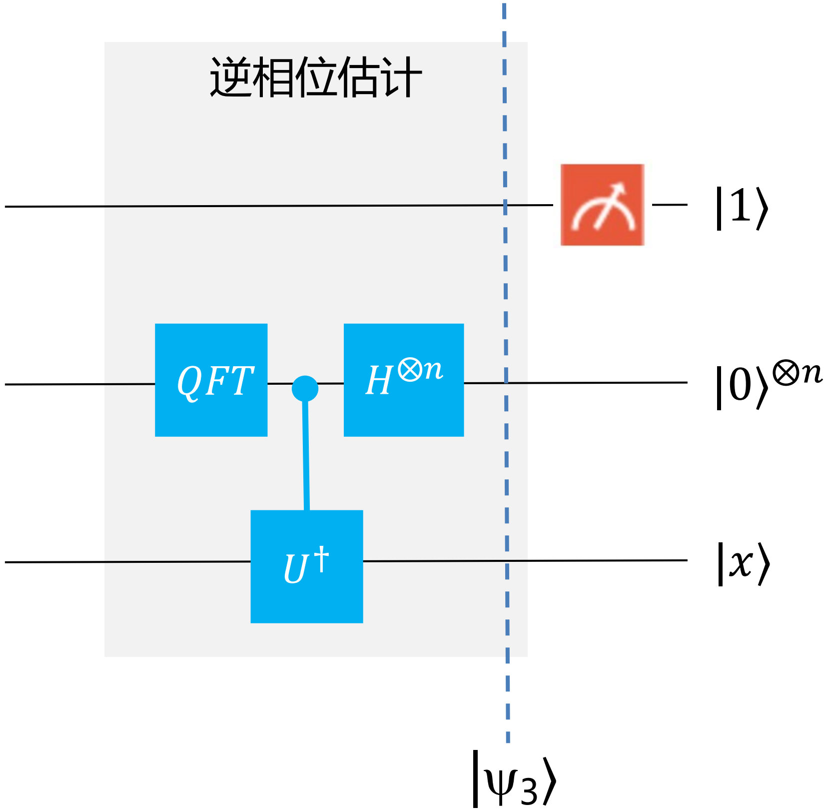
Calvin, QQ: 179209347 Mail: 179209347@qq.com

# HHL算法过程 - 逆相位估计（4/4）

## 逆相位估计：通过逆QPE输出结果量子态

理论上，受控旋转后的量子态已经可以通过测量得到解量子态 $|x\rangle$ 。

但为了避免出现 $|\mu_j\rangle$ 相同但 $|\tilde{\lambda}_j\rangle$ 不同的需要合并的量子态 $\frac{C}{\tilde{\lambda}_j} b_j|1\rangle|\tilde{\lambda}_j\rangle|\mu_j\rangle$ ，应当选择逆 $QPE$ 操作来得到形如 $\frac{C}{\tilde{\lambda}_j} b_j|1\rangle|0\rangle|\mu_j\rangle$ 的结果量子态。对旋转结果进行逆 $QPE$ ，有：




$$
\begin{aligned}
|\psi_3\rangle &= (I \otimes QPE^{\dagger}) \sum_{j=0}^{N-1} \left( \sqrt{1-\frac{C^2}{\tilde{\lambda}_j^{\,2}}}\,|0\rangle + \frac{C}{\tilde{\lambda}_j}|1\rangle \right) |\tilde{\lambda}_j\rangle b_j |\mu_j\rangle \\
&= \sum_{j=0}^{N-1} \left( b_j\sqrt{1-\frac{C^2}{\tilde{\lambda}_j^{\,2}}}\,|0\rangle|0\rangle|\mu_j\rangle + \frac{C}{\tilde{\lambda}_j} b_j|1\rangle|0\rangle|\mu_j\rangle \right)
\end{aligned}
$$

参考来源：https://pyqpanda-toturial.readthedocs.io/zh/latest/HHL.html

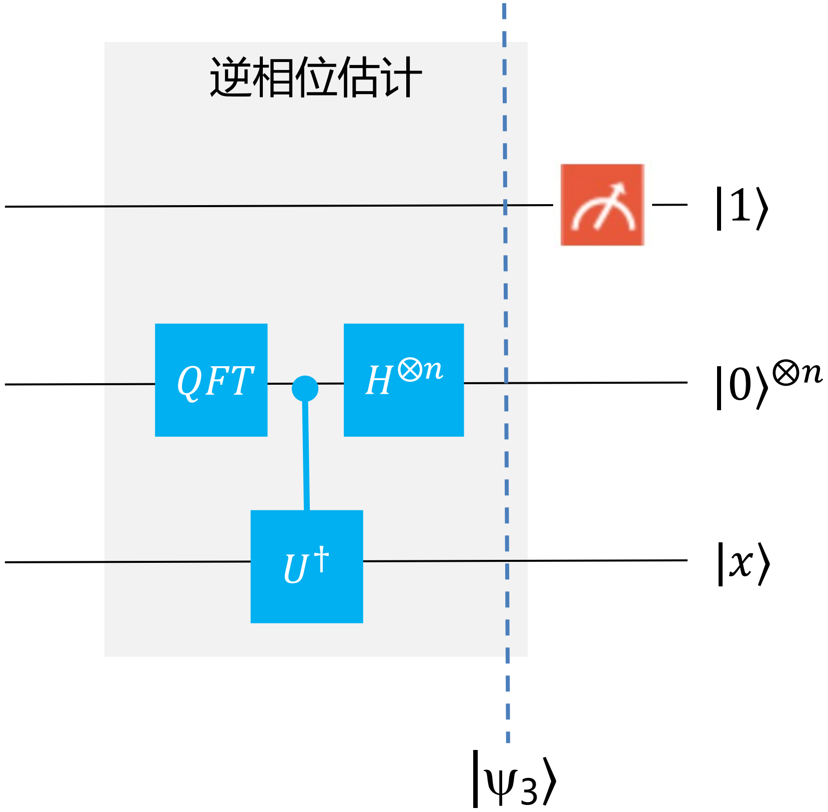
# HHL算法过程 - 逆相位估计（4/4）

## 逆相位估计：通过逆QPE输出结果量子态

事实上即使是这种形式的结果量子态，由于误差的存在，依然无法在第一个和第二个量子寄存器分别为 $|1\rangle, |0\rangle$ 的情况下以概率 1 得到解量子态：

$$|x\rangle = A^{-1}|b\rangle = \sum_{j=0}^{N-1} b_j A^{-1} |\mu_j\rangle = \sum_{j=0}^{N-1} \frac{b_j}{\lambda_j} |\mu_j\rangle$$

HHL算法充分利用了量子相位估计提取特征值信息的功能，巧妙构造了受控旋转门从存储比特的基向量中抓取特征值存入振幅， 最后利用逆相位估计还原存储量子比特，从而得到了振幅含特征值的方程解。



# HHL算法过程 - 逆相位估计（4/4）

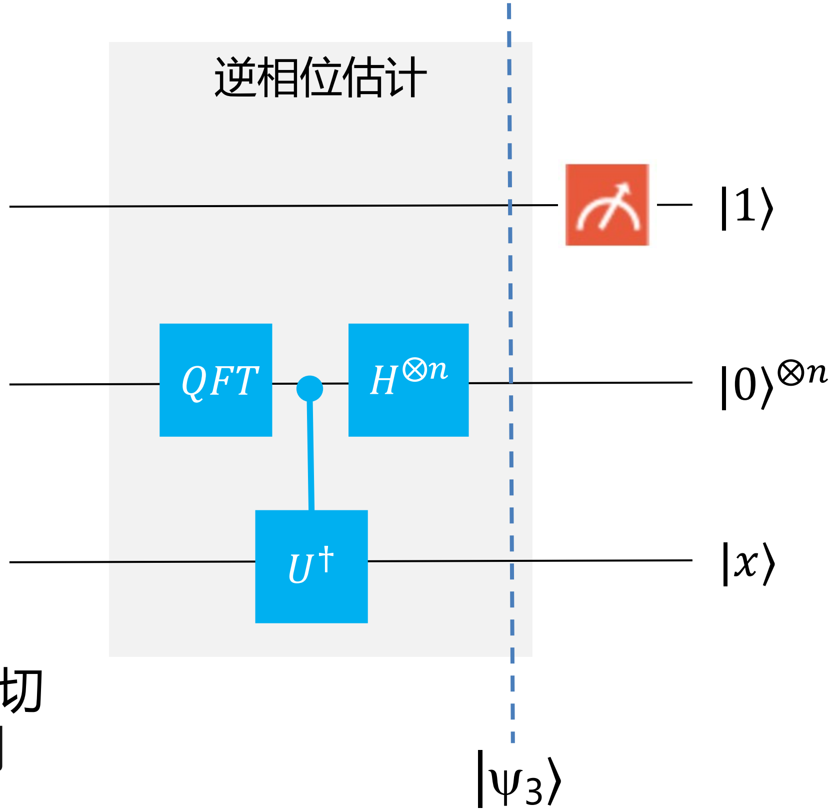
**逆相位估计：**

执行逆相位估计，将 $|\lambda_j\rangle$ 转换为 $|0\rangle^{\otimes n}$ 。

对辅助量子比特进行测量，当测量得到 1 时，原来的寄存器由一系列 $|\lambda_j\rangle$ 的叠加变为 $f(\lambda_j)|1\rangle$ 的叠加。

基态 $|\lambda_j\rangle$ 中的值按照 $f(\lambda_j)$ 的比例被提取到了基态 $|\lambda_j\rangle$ 的概率幅上。此时：

$$|x\rangle = \sum_{j=0}^{N-1} \frac{b_j}{\lambda_j} |\mu_j\rangle$$




* 最后得到的 $|x\rangle$ ，各分量的振幅并没有显式地得到，我们并不能读出 $x$ 的确切值是什么。不过我们能够得到一个关于 $x$ 的整体特性。比如目的是求某个观测量 M 在 $|x\rangle$ 上的期望 $\langle x|M|x\rangle$ 。对于 $x$ 的确切值并不是非要提取出来的应用，HHL 算法还是相当高效的。

Qubits
qubits.top

Thank

You