Quantum Algorithms: Phase Estimation and Factoring

John Watrous
Department of Computer Science
University of Calgary

Integer factoring

The **integer factoring problem** is as follows:

Input: a composite integer N.

Output: any two integers $a, b \in \{2, \mathrm{K}\ , N-1\}$ such that:

$$a\,b = N$$

For example: if

N=15

then

a=3, b=5

is a correct output.

Integer factoring

The **integer factoring problem** is as follows:

Input: a composite integer N.

Output: any two integers $a, b \in \{2, \ldots, N-1\}$ such that:

$$a\,b = N$$

For example: if

N= 156,203,777,432,828,093

then

a= 18,005,557,777 , b= 8,675,309

is a correct output.

Integer factoring

The **integer factoring problem** is **hard** for classical computers (as far as we know).

- no classical polynomial time algorithm is known (polynomial means in the number of digits).

- RSA Laboratories will give $1750 to the first person that factors this (200-digit) number:

**27,997,833,911,221,327,870,829,467,638,722,601,621,070,446,786,955,
428,537,560,009,929,326,128,400,107,609,345,671,052,955,360,856,061,
822,351,910,951,365,788,637,105,954,482,006,576,775,098,580,557,613,
579,098,734,950,144,178,863,178,946,295,187,237,869,221,823,983**

Integer factoring

In 1994, **Peter Shor** (AT&T Labs - Research) discovered a polynomial-time quantum algorithm for factoring integers.

In this talk: how quantum computers can factor integers.

The description will be somewhat different from Shor's description, but is equivalent in principle. See [Kitaev, 1995], [Cleve, Ekert, Macchiavello & Mosca, 1998].

Two Main Steps

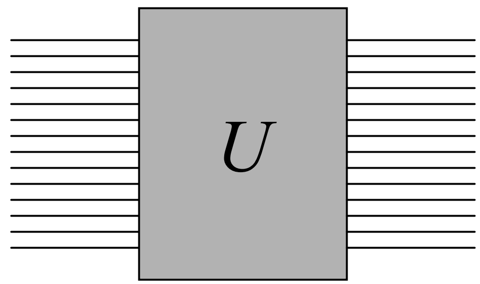
- Phase estimation.

- Reduction of factoring to phase estimation (via order-finding).

Phase Estimation

Suppose we are given a quantum circuit acting on n qubits:

$|\psi_j\rangle$



$e^{2\pi i\theta_j}|\psi_j\rangle$

Since U is unitary, it must have orthonormal eigenvectors

$N = 2^n$

$$|\psi_1\rangle, |\psi_2\rangle, \ldots, |\psi_N\rangle$$

with corresponding eigenvalues of the form

$$\lambda_1 = e^{2\pi i\theta_1},\ \lambda_2 = e^{2\pi i\theta_2}, \ldots, \lambda_N = e^{2\pi i\theta_N}$$

Phase Estimation Problem

Given:

quantum circuit U

an eigenvector $|\psi\rangle$ of U

Goal:

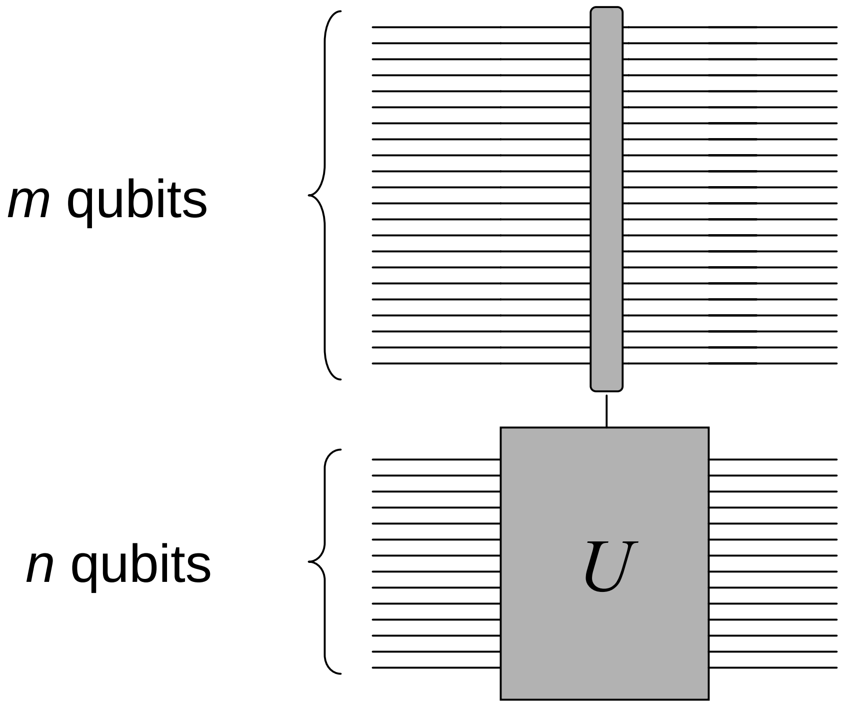
compute (or approximate) θ, where

$$U|\psi\rangle = e^{2\pi i \theta}|\psi\rangle$$

In general we do not know how to solve this problem efficiently…

Phase Estimation

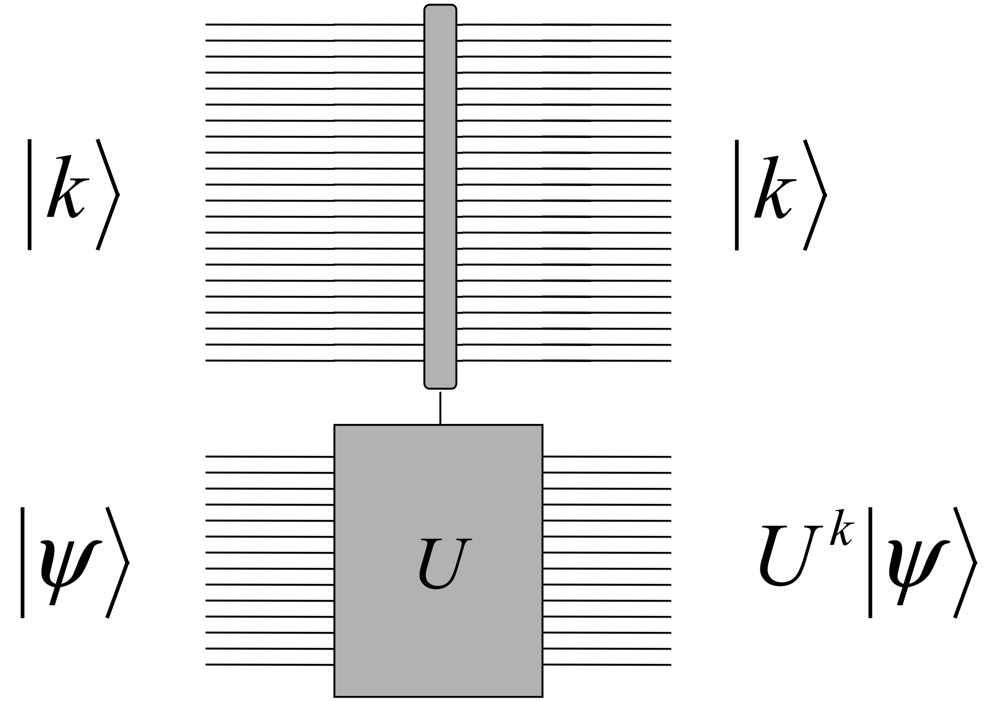
However, the problem can be solved efficiently if instead of a circuit for U we have a circuit as follows:



$$c_m U : \left|k\right\rangle\left|\varphi\right\rangle \; \alpha \; \left|k\right\rangle U^k \left|\varphi\right\rangle$$

Phase Estimation

What does this circuit do?

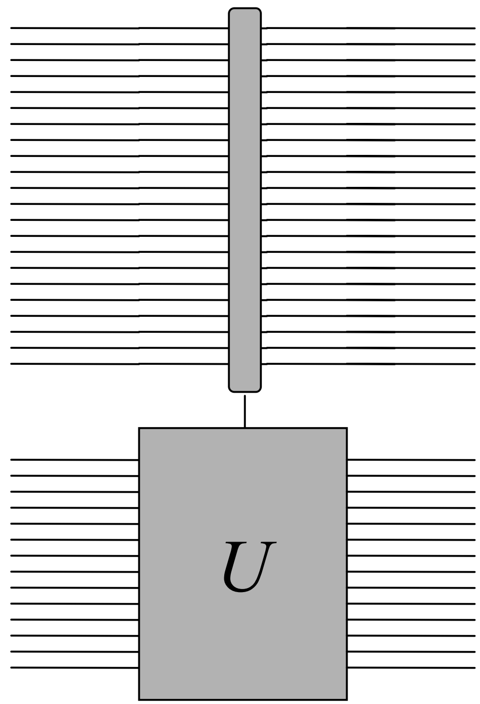


$$c_m U : |k\rangle|\varphi\rangle \ \alpha \ |k\rangle U^k|\varphi\rangle$$

Phase Estimation

What does this circuit do?

$|k\rangle$ $\qquad$ $|k\rangle$

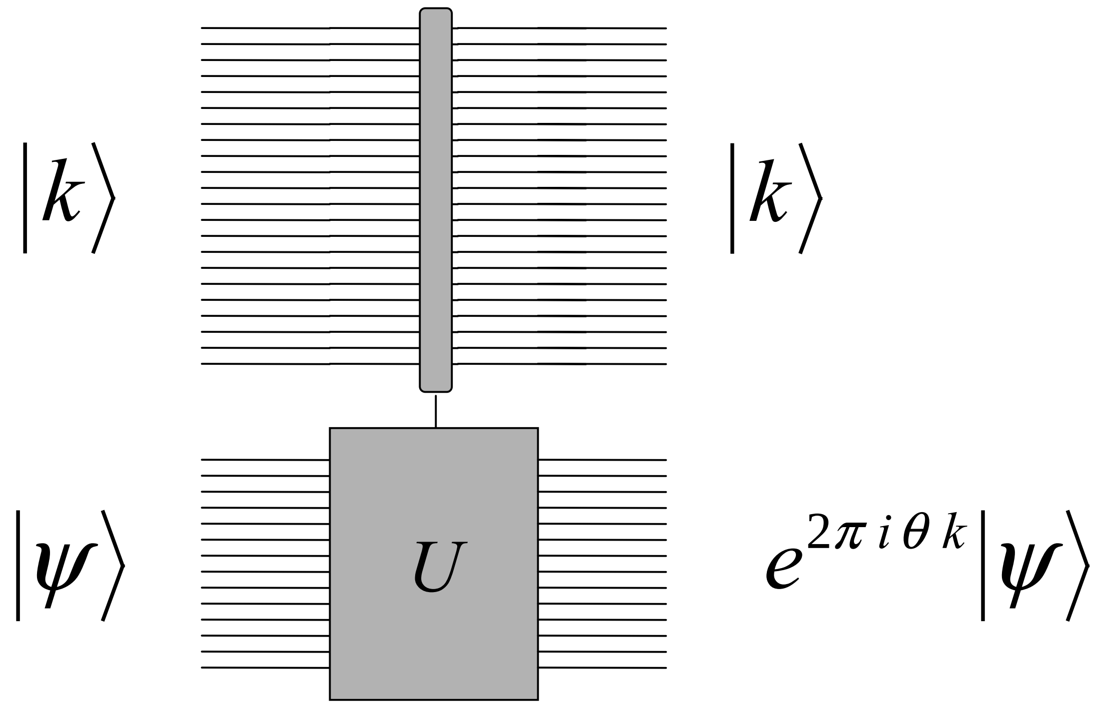


$|\psi\rangle$ $\qquad$ $\left(e^{2\pi i\theta}\right)^k|\psi\rangle$

$$c_m U : |k\rangle|\varphi\rangle \;\alpha\; |k\rangle U^k|\varphi\rangle$$

Phase Estimation

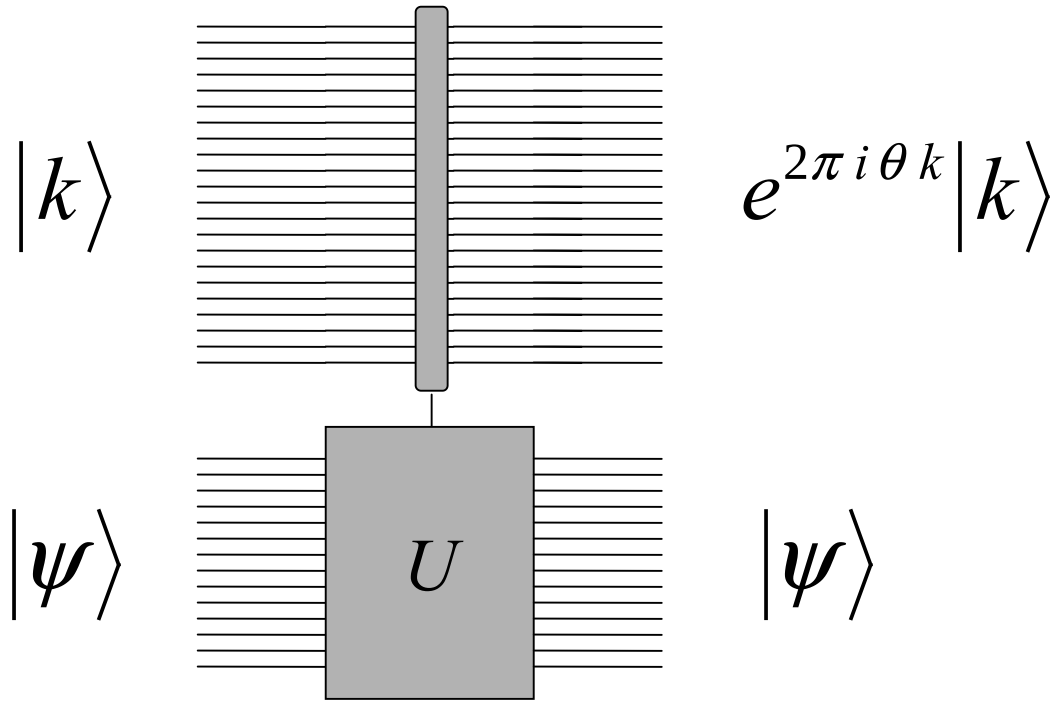
What does this circuit do?



$$c_m U : \ |k\rangle|\varphi\rangle \ \alpha \ |k\rangle U^k|\varphi\rangle$$

Phase Estimation

What does this circuit do?



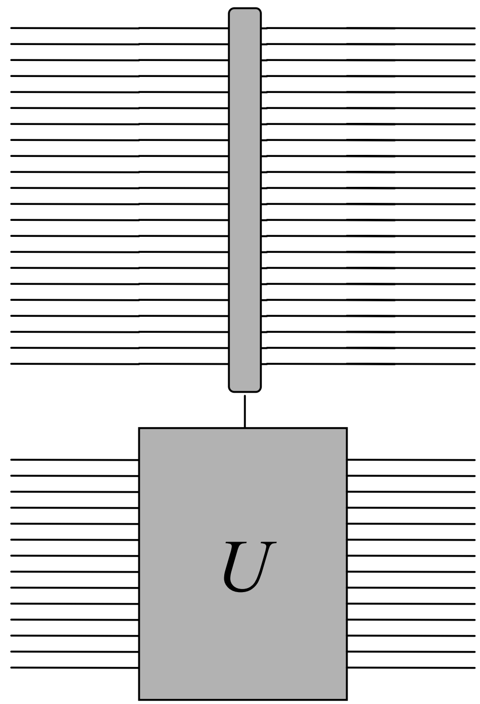
$$c_m U : |k\rangle|\varphi\rangle \;\; \alpha \;\; |k\rangle U^k|\varphi\rangle$$

Phase Estimation

What does this circuit do?

$$\sum_{k=0}^{2^m-1} |k\rangle$$

$$|\psi\rangle$$



$$\sum_{k=0}^{2^m-1} e^{2\pi i \theta k} |k\rangle$$

$$|\psi\rangle$$

$$c_m U : |k\rangle |\varphi\rangle \ \alpha \ |k\rangle U^k |\varphi\rangle$$

Phase Estimation

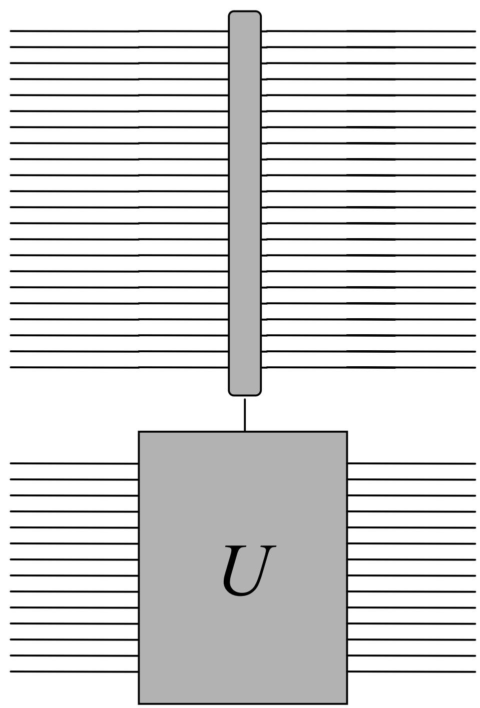
Simple case:

$$\theta = \frac{j}{2^m} \qquad \text{for} \qquad j \in \{0, \ldots, 2^m - 1\}$$

easy to create

need to compute *j* from this

$$\frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} |k\rangle$$



$$\frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} e^{2\pi i \frac{jk}{2^m}} |k\rangle$$

$$|\psi\rangle \qquad |\psi\rangle$$

Phase Estimation

Want some transformation T that acts as follows:

$$T: \frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} e^{2\pi i \frac{jk}{2^m}} |k\rangle \quad \alpha \quad |j\rangle$$

Equivalently:

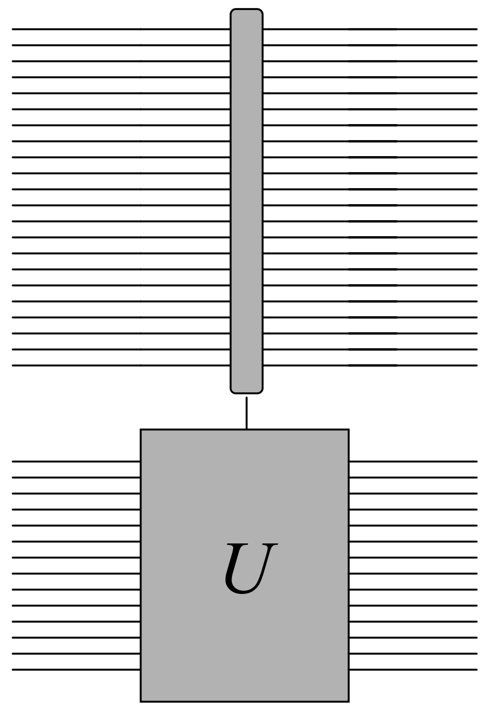
$$T^{-1}: |j\rangle \quad \alpha \quad \frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} e^{2\pi i \frac{jk}{2^m}} |k\rangle$$

This is just the **quantum Fourier transform.**

Phase Estimation

Back to simple case: $\theta = \frac{j}{2^m}$

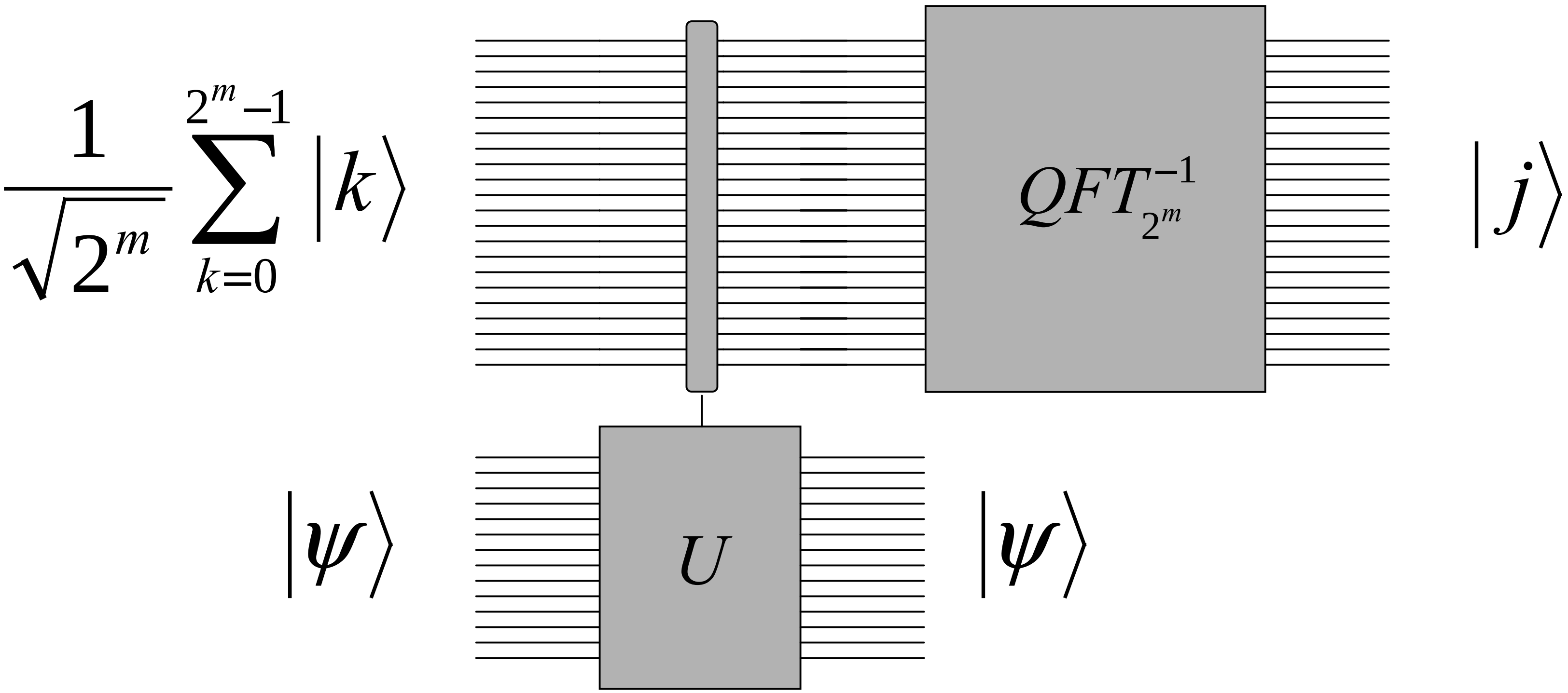
$$\frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} |k\rangle \qquad \frac{1}{\sqrt{2^m}} \sum_{k=0}^{2^m-1} e^{2\pi i \frac{jk}{2^m}} |k\rangle$$



$$|\psi\rangle \qquad |\psi\rangle$$

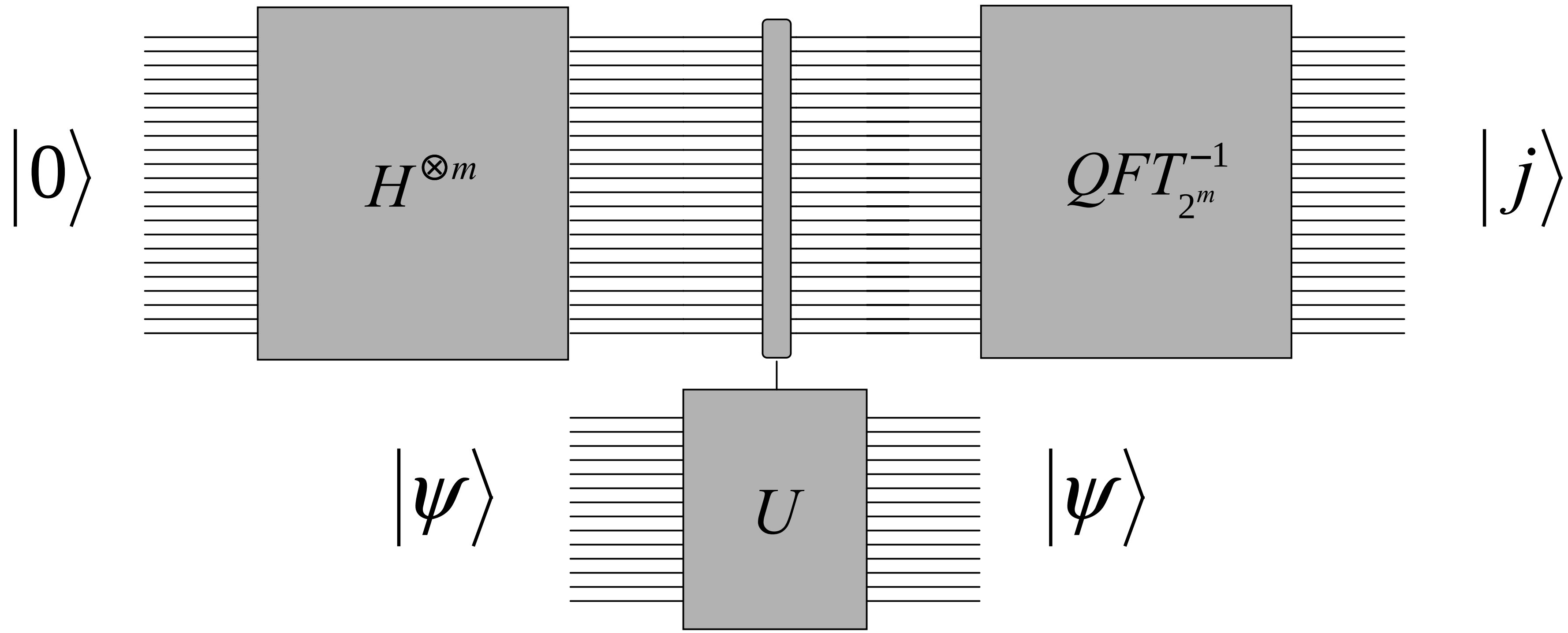
Phase Estimation

Back to simple case: $\theta = \dfrac{j}{2^m}$



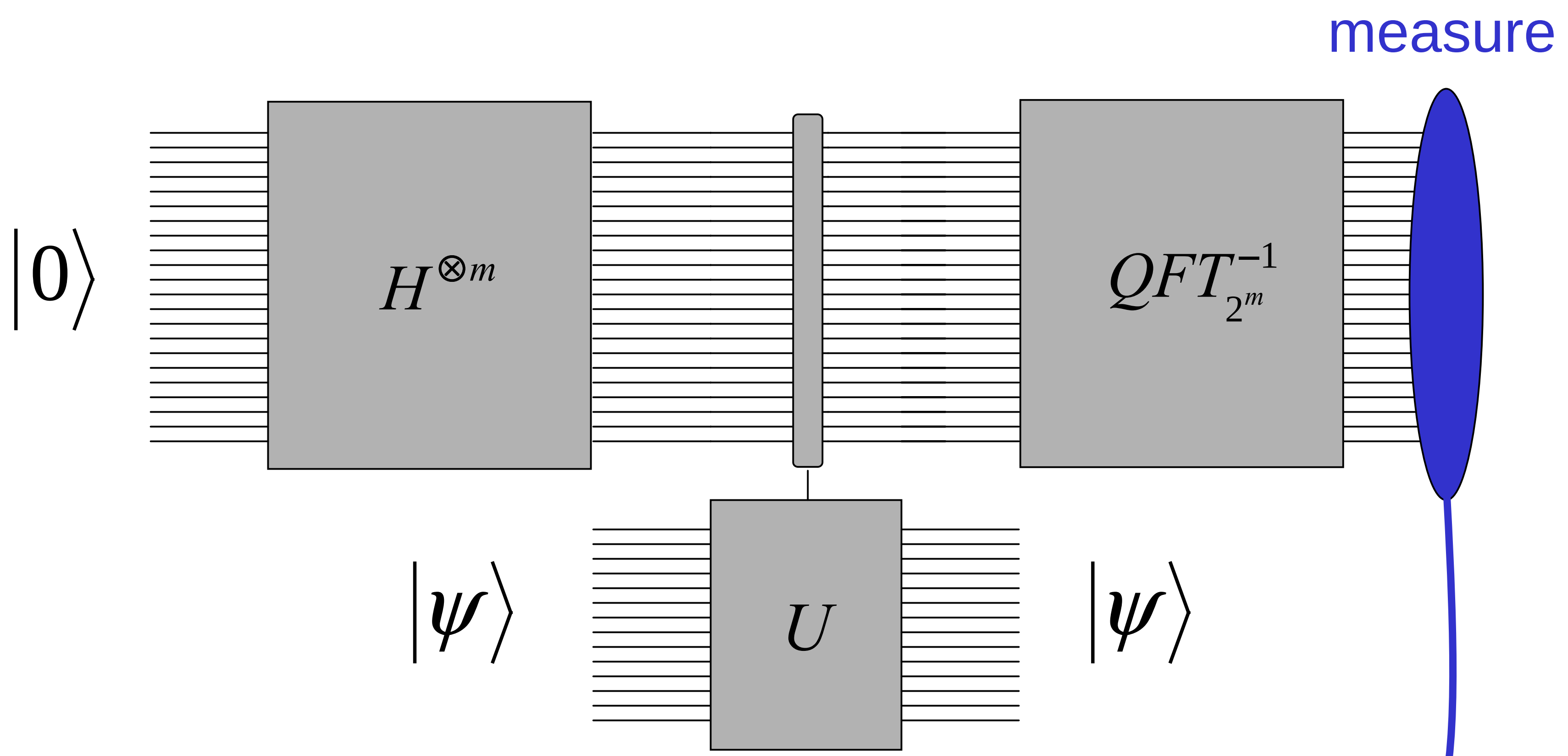
Phase Estimation

Back to simple case: $\theta = \frac{j}{2^m}$



Phase Estimation

General case: θ is arbitrary.

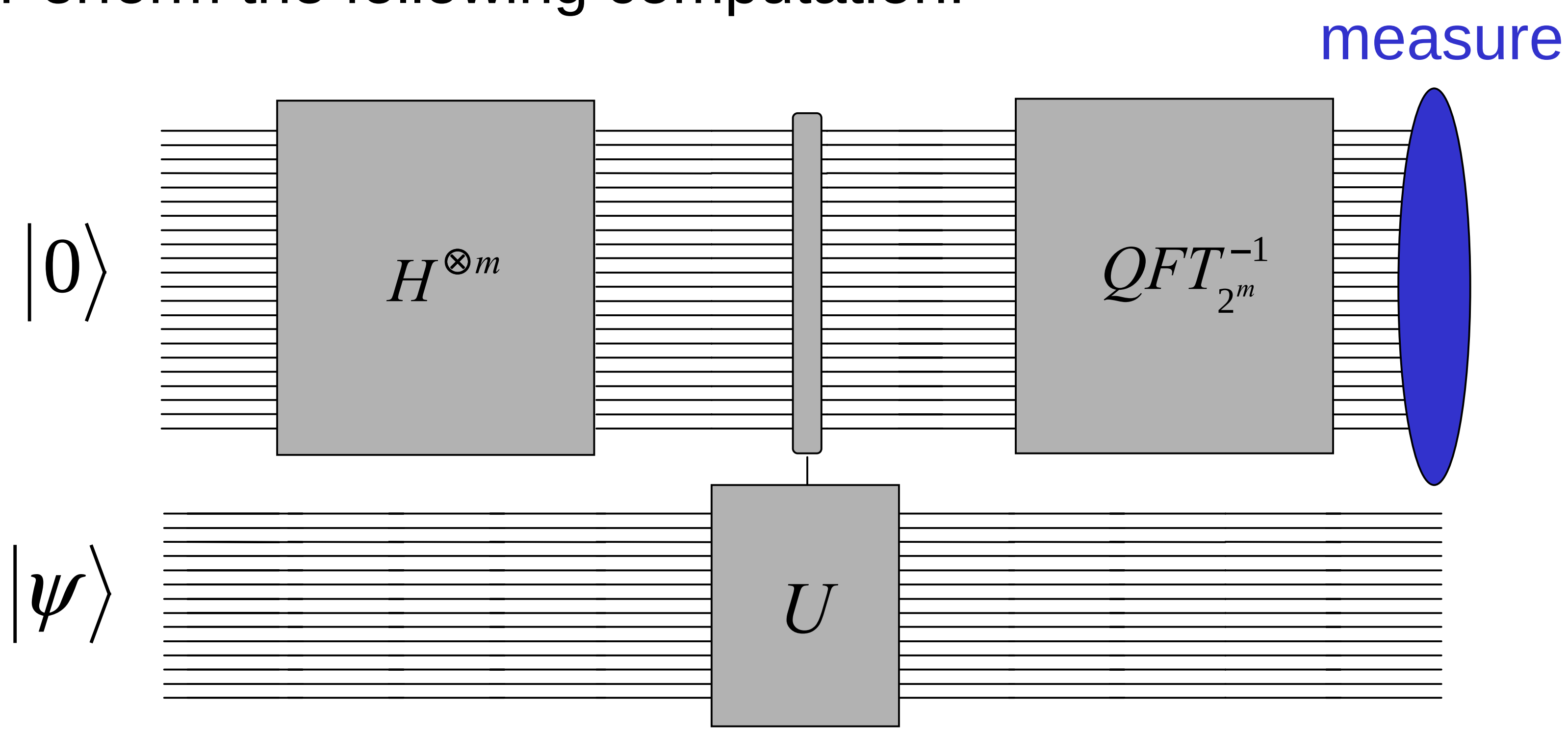


Claim: w.h.p. the result will be some j such that $j/2^m$ is a good approximation to θ.

Summary of Phase Estimation

Have $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$ for $\theta \in [0,1)$

Perform the following computation:



The result is j such that $j/2^m$ is a good approximation to θ with high probability.

Back to Factoring

We want to reduce factoring to phase estimation.

Factoring

$\downarrow$

Order Finding

$\downarrow$

Phase Estimation

Order Finding

Notation:

$$\mathbb{Z}_N = \{0, 1, \ldots, N-1\}$$

$$\mathbb{Z}_N^* = \{a \in \mathbb{Z}_N : \gcd(a, N) = 1\}$$

(addition and multiplication always modulo N)

Order Finding

Given $a \in \mathbb{Z}_N^*$ we define the **order** of a modulo N to be the smallest positive integer r such that

$$a^r \equiv 1 \pmod{N}$$

For example, if $N = 21$ and $a = 2$, then:

$$2^2 \equiv 4, \quad 2^3 \equiv 8, \quad 2^4 \equiv 16, \quad 2^5 \equiv 11, \quad 2^6 \equiv 1$$

so the order of 2 modulo 21 is 6.

Order Finding

The **order finding problem** is:

Given a and N such that $a \in \mathbb{Z}_N^*$

Goal: find the order of a modulo N.

Relevant facts:

- **Factoring is easy** if we have the ability to solve order finding.

- We can solve order finding via **phase estimation**.

Factoring and Order Finding

Suppose we want to factor N.

Assume we have $a \in \mathbb{Z}_N^*$ and we know the order r of a modulo N.

Then

$$a^r \equiv 1 \pmod N$$

$$\Rightarrow \quad a^r - 1 \equiv 0 \pmod N$$

$$\Rightarrow \quad N \text{ divides } a^r - 1$$

Factoring and Order Finding

Suppose we are lucky and r is even. Then

$$a^r - 1 = \left(a^{r/2} - 1\right)\left(a^{r/2} + 1\right)$$

so

$$N \text{ divides } \left(a^{r/2} - 1\right)\left(a^{r/2} + 1\right)$$

Some of the factors of N divide $\left(a^{r/2} - 1\right)$ and some divide $\left(a^{r/2} + 1\right)$

Factoring and Order Finding

$$N \text{ divides } \left(a^{r/2}-1\right)\left(a^{r/2}+1\right)$$

If we are lucky again:

$$\gcd\left(N, a^{r/2}-1\right)$$

will be a **proper** divisor of N.

Fact: if we choose $a \in \mathbb{Z}_N^*$ uniformly, we will be lucky both times with probability at least 1/2.

Factoring and Order Finding

Algorithm to factor N:

<u>Repeat</u>

Choose a random $a \in \mathbb{Z}_N^*$

$\boxed{\text{Compute the order } r \text{ of } a \text{ modulo } N.}$

If r is even, compute

$$d = \gcd\left(N, a^{r/2} - 1\right)$$

<u>Until</u> we find a proper divisor d of N

(or until we get tired).

Order Finding and Phase Estimation

Given N and $a \in \mathbb{Z}_N^*$

Our goal is to find the smallest positive r such that

$$a^r \equiv 1 \pmod{N}$$

Define a transformation M_a as follows:

$$M_a : |x\rangle \alpha \; |a\,x\rangle$$

(we assume $x \in \mathbb{Z}_N$ and arithmetic is modulo N).

Order Finding and Phase Estimation

$$M_a : |x\rangle \,\alpha\, |a\,x\rangle$$

What are the eigenvectors/eigenvalues of M_a?

Here is one eigenvector:

$$|\psi_0\rangle = |1\rangle + |a\rangle + |a^2\rangle + \Lambda + |a^{r-1}\rangle$$

(eigenvalue is 1).

Order Finding and Phase Estimation

Another one: $\omega = e^{2\pi i \frac{1}{r}}$

$$\left|\psi_1\right\rangle = \left|1\right\rangle + \omega^{-1}\left|a\right\rangle + \omega^{-2}\left|a^2\right\rangle + \Lambda + \omega^{-(r-1)}\left|a^{r-1}\right\rangle$$

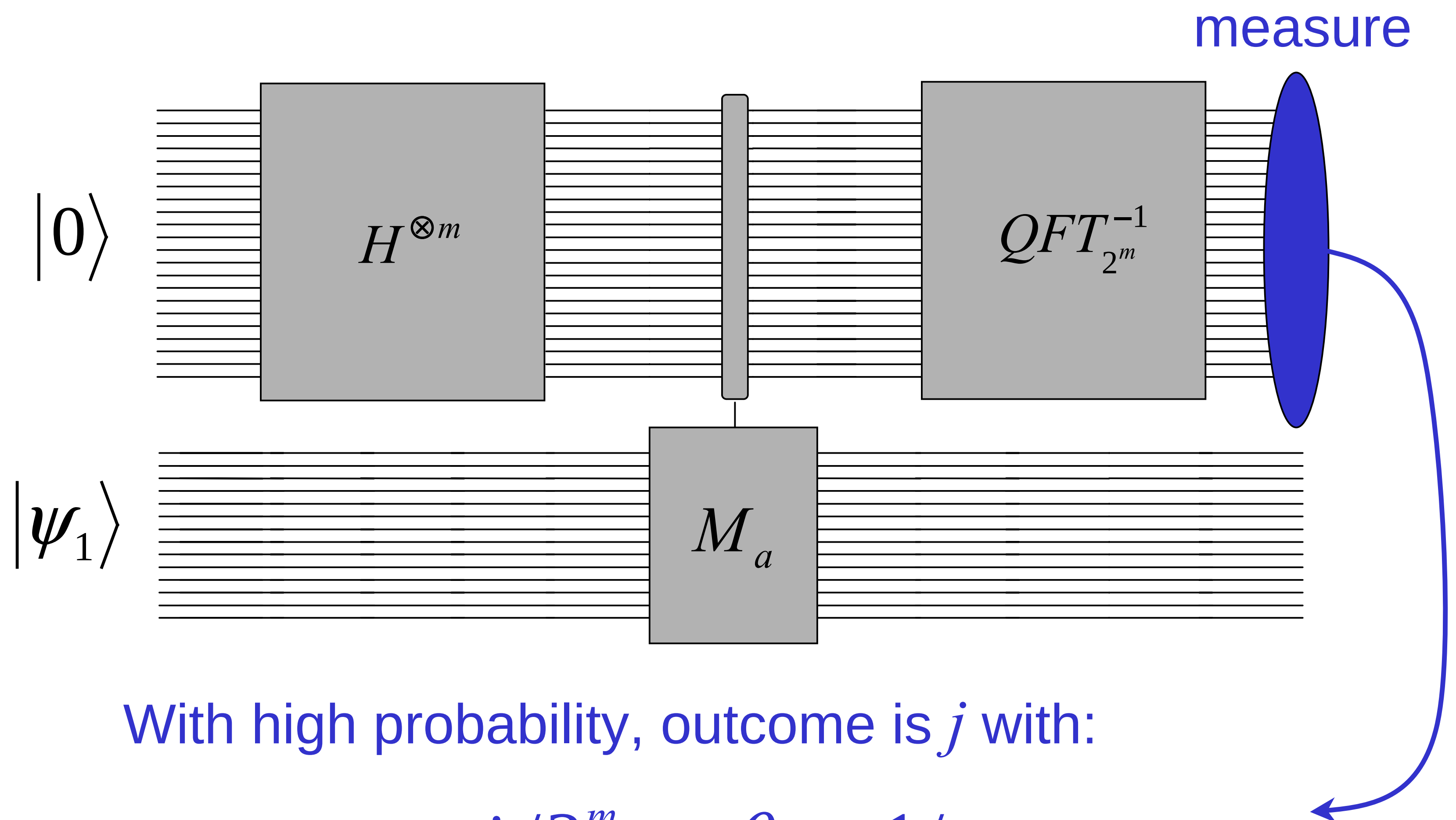
$$\begin{aligned} M_a\left|\psi_1\right\rangle &= \left|a\right\rangle + \omega^{-1}\left|a^2\right\rangle + \omega^{-2}\left|a^3\right\rangle + \Lambda + \omega^{-(r-1)}\left|a^r\right\rangle \\ &= \omega\left(\omega^{-1}\left|a\right\rangle + \omega^{-2}\left|a^2\right\rangle + \Lambda + \omega^{-(r-1)}\left|a^{r-1}\right\rangle + \left|1\right\rangle\right) \\ &= \omega\left|\psi_1\right\rangle \end{aligned}$$

(so the associated eigenvalue is ω).

$$\omega = e^{2\pi i \frac{1}{r}} \Rightarrow \theta = \frac{1}{r}$$

Order Finding and Phase Estimation

Suppose we plug M_a and $|\psi_1\rangle$ into our phase estimation method:



With high probability, outcome is j with:

$$j/2^m \approx \theta = 1/r$$

Controlled Multiply by a

We need to be able to implement a $c_m M_a$ gate for this procedure to work.

$$c_m M_a |k\rangle|x\rangle = |k\rangle M_a^k|x\rangle = |k\rangle|a^k x\rangle$$

This is just modular exponentiation… can be implemented reversibly using

$$O\left((\log m)(\log N)^2\right)$$

gates.

Need Other Eigenvectors

We do not know an easy way to construct $|\psi_1\rangle$.

Instead, what we will do **in effect** is to randomly choose one of the eigenvectors

$$|\psi_0\rangle,\ |\psi_1\rangle,\ |\psi_2\rangle,\ \ldots\ ,\ |\psi_{r-1}\rangle$$

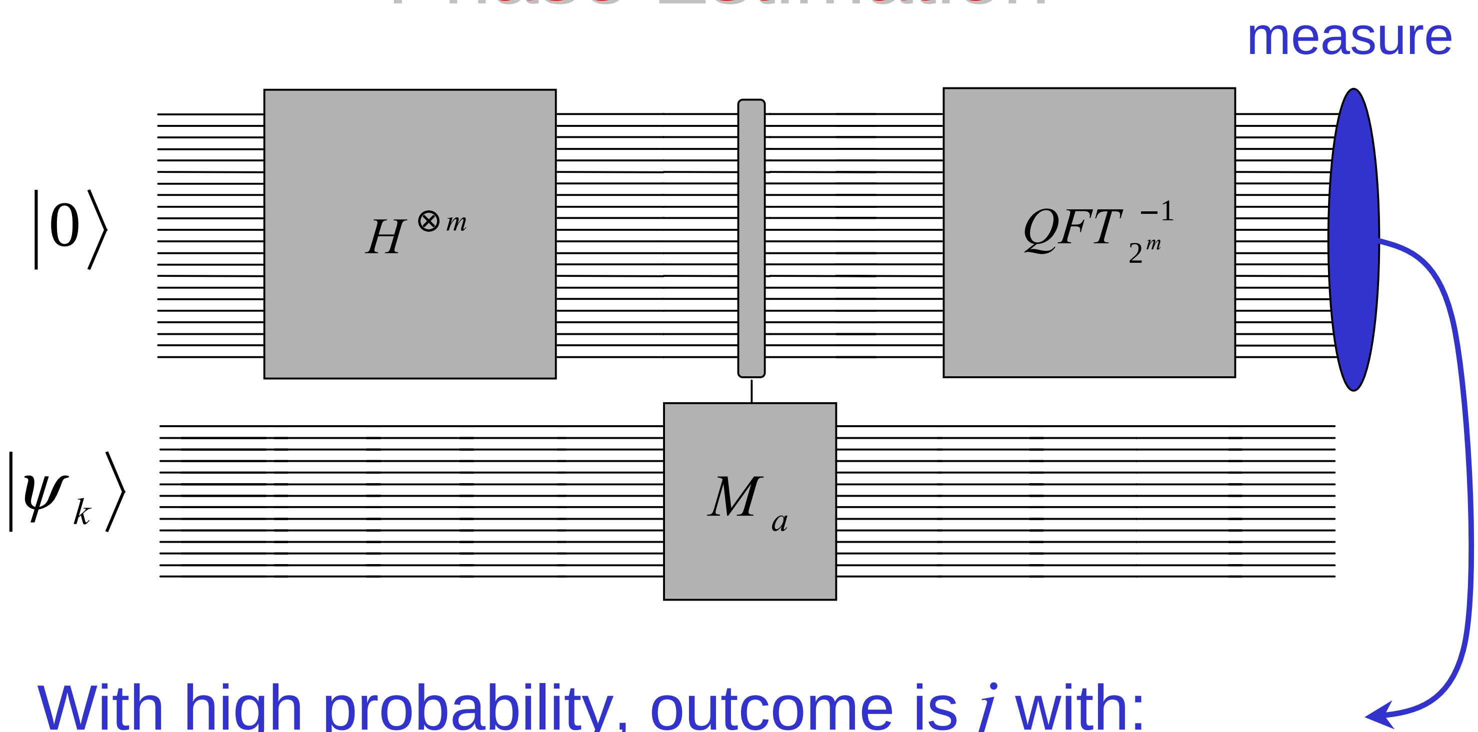
where

$$|\psi_k\rangle = |1\rangle + \omega^{-k}|a\rangle + \omega^{-2k}|a^2\rangle + \cdots + \omega^{-(r-1)k}|a^{r-1}\rangle$$

and the associated eigenvalue is

$$\omega^k = e^{2\pi i \boxed{\frac{k}{r}}}$$

Phase Estimation



With high probability, outcome is j with:

$$j/2^m \approx \theta = k/r$$

With several samples (with different k each time) we can determine r with high probability.

(Use **continued fraction algorithm** for this.)

Remaining Obstacle

Need to (in effect) generate a random eigenvector

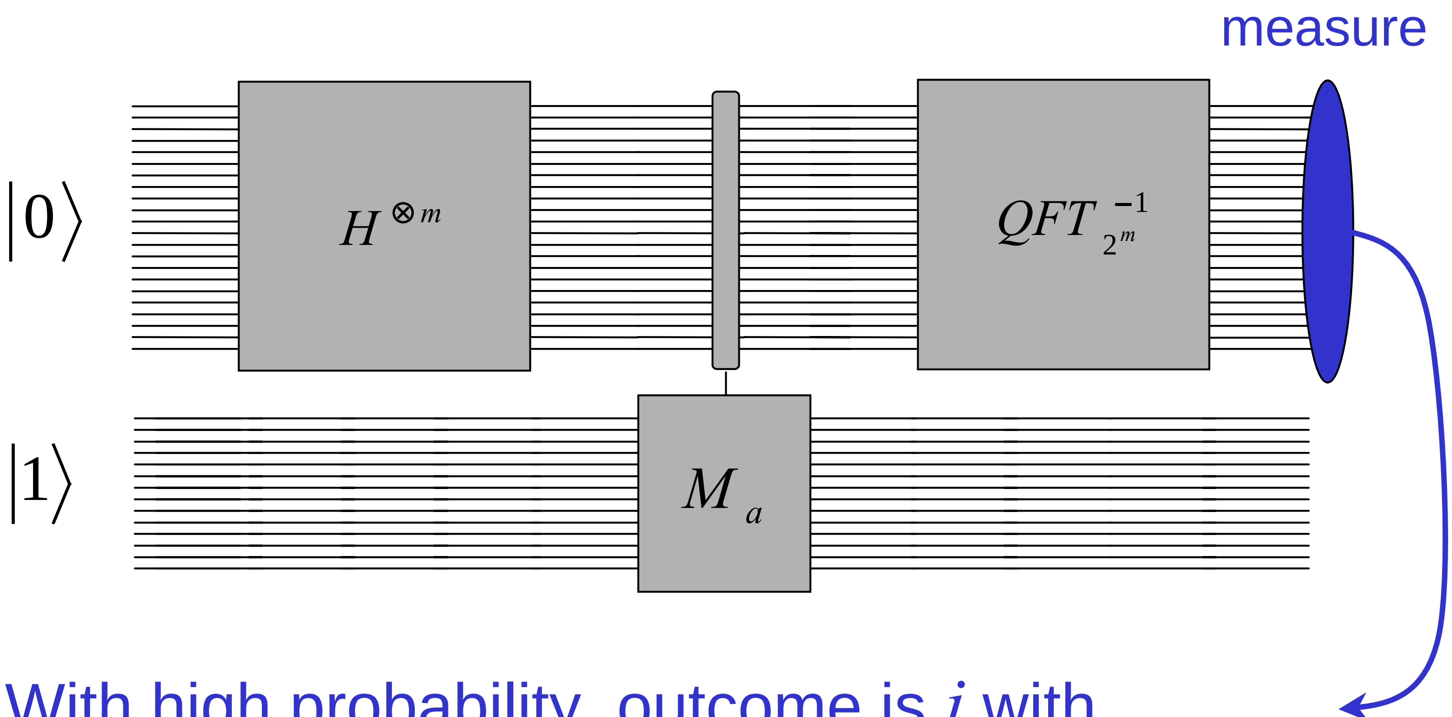
$$|\psi_0\rangle,\ |\psi_1\rangle,\ |\psi_2\rangle,\ \mathrm{K}\ ,\ |\psi_{r-1}\rangle$$

This turns out to be very simple… note that

$$|1\rangle = \frac{1}{\sqrt{r}}\left(|\psi_0\rangle + |\psi_1\rangle + \ \Lambda\ + |\psi_{r-1}\rangle\right)$$

Running the phase estimation procedure with $|1\rangle$ in place of $|\psi_k\rangle$ will be equivalent to randomly choosing an eigenvector $|\psi_k\rangle$.

Final Phase Estimation Procedure



With high probability, outcome is j with

$j/2^m \approx k/r$ for random $k \in \{0, \ldots, r-1\}$

After a constant number of samples, r can be determined with high probability.

Other Problems

Examples of other problems that can be solved in quantum polynomial time (but for which no polynomial-time classical algorithms are known):

- computing discrete logarithms
- generalizations to problems regarding abelian groups: decomposition of abelian groups, extensions to solvable groups, (abelian) **hidden subgroup problem**
- solutions to instances of Pell's equation
- shifted Legendre symbol problem, hidden coset problem