Extended Church-Turing Thesis

- The belief that TMs formalize our intuitive notion of an efficient algorithm is:
  - The "extended" Church-Turing Thesis: everything we can compute in time $t(n)$ on a physical computer can be computed on a Turing Machine in time $t(n)^{O(1)}$ (polynomial slowdown).
  - Quantum computation challenges this belief.

A different model

- Infinite tape of a Turing Machine is an idealized model of computer.
- Real computer is a Finite Automaton (!)
  - $n$ bits of memory
  - $2^n$ states

Model of deterministic computation

\[
\begin{pmatrix}
1 & 0 & 0 & 0 & \cdots & 0 \\
0 & 1 & 0 & 0 & \cdots & 0 \\
0 & 0 & 1 & 0 & \cdots & 0 \\
0 & 0 & 0 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 1
\end{pmatrix}
\]

\[
\begin{pmatrix}
0 & 1 & 0 & 0 & \cdots & 0 \\
0 & 0 & 1 & 0 & \cdots & 0 \\
0 & 0 & 0 & 1 & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 1
\end{pmatrix}
\]

\[
\begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0
\end{pmatrix}
\]
Model of randomized computation

- possible states at time t: \[ \sum_i p_i = 1 \quad p_i \in \mathbb{R}^+ \]
- stochastic matrix
- sum in each column = 1

\[
\begin{pmatrix}
p_0 \\
p_1 \\
p_2 \\
p_3 \\
p_{2^{n-1}}
\end{pmatrix}
\]

state at time t

\[
\begin{pmatrix}
0 & 1/4 & 0 & 1/4 \\
0 & 1/4 & 1/4 & 0 \\
0 & 1/4 & 0 & 0 \\
1/2 & 1/2 & 0 & 0 \\
1/2 & 0 & 0 & 0
\end{pmatrix}
\]

state at time t+1

\[
\begin{pmatrix}
p_0 \\
p_1 \\
p_2 \\
p_3 \\
p_{2^{n-1}}
\end{pmatrix}
\Rightarrow
\begin{pmatrix}
0 \\
1 \\
0 \\
0 \\
0
\end{pmatrix}
\]

see i\textsuperscript{th} basic state with probability \( p_i \)

Model of quantum computation

- possible states at time t: \[ \sum_i |c_i|^2 = 1 \quad c_i \in \mathbb{C} \]
- unitary matrix
- preserves \( L_2 \) norm

\[
\begin{pmatrix}
c_0 \\
c_1 \\
c_2 \\
c_3 \\
c_{2^{n-1}}
\end{pmatrix}
\]

state at time t

\[
\begin{pmatrix}
1/\sqrt{2} & 1/\sqrt{2} \\
1/\sqrt{2} & -1/\sqrt{2}
\end{pmatrix}
\]

state at time t+1

\[
\begin{pmatrix}
c_0 \\
c_1 \\
c_2 \\
c_3 \\
c_{2^{n-1}}
\end{pmatrix}
\Rightarrow
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
0
\end{pmatrix}
\]

see i\textsuperscript{th} basic state with probability \( |c_i|^2 \)

One quantum register

- register with n qubits; shorthand for basic states

\[
|0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad \text{shorthand for general state}
\]

\[
|\psi\rangle = \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ \cdots \end{pmatrix} = \sum_i c_i |i\rangle
\]

Two quantum registers

- registers with n, m qubits: shorthand for \( 2^{n+m} \) basic states:

\[
|0\rangle|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix},
|0\rangle|1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}
\]

\[
|1\rangle|0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix},
|1\rangle|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}
\]
Two quantum registers

|\psi\rangle = |d\rangle = \left( \begin{array}{c} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{2^n-1} \end{array} \right) \otimes \left( \begin{array}{c} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{2^n-1} \end{array} \right) = \sum_{i,j} c_i d_j |i\rangle |j\rangle.

Example: \[ \frac{1}{\sqrt{2}} (|0\rangle |0\rangle + |1\rangle |1\rangle) \]

Partial measurement

- General state: \[ |\psi\rangle = \sum_{i,j} a_{i,j} |i\rangle |j\rangle \]
- If measure just 2nd register, see state \[ |j\rangle \]
  in 2nd register with probability \[ \sum_{i} |a_{i,j}|^2 \]
- State collapses to: \[ \alpha \left( \sum_{i} a_{i,j} |i\rangle \right) \otimes |j\rangle \]

EPR “paradox”

\[ \frac{1}{\sqrt{2}} (|0\rangle |0\rangle + |1\rangle |1\rangle) \]

- Register 1 in LA, register 2 sent to NYC
- Measure register 2
  - Probability \( \frac{1}{2} \): see \(|0\rangle |0\rangle \)
  - Probability \( \frac{1}{2} \): see \(|1\rangle |1\rangle \)
- Measure register 1
- Guaranteed to be same as observed in NYC
- Instantaneous “communication”

Quantum complexity

- One measure: complexity of \( f \) = length of shortest sequence of local operations computing \( f \)
- Example local operation:
  - Position \( x = 0010 \)
  - Logical OR
  - Position \( x' = 1010 \)

Quantum complexity

- Analogous notion of “local operation” for quantum systems
- In each step
  - Split qubits into register of 1 or 2, and rest
  - Operate only on small register
- “Efficient” in both settings: \( \# \) local operations polynomial in \( \# \) bits \( n \)
Efficiently quantum computable functions

- For every $f: \{0,1\}^n \to \{0,1\}^m$ that is efficiently computable classically
- the unitary transform $U_f$:
  $$U_f(|i\rangle|j\rangle) = |i\rangle|f(i) \oplus j\rangle$$
- note, when 2nd register = $|0\rangle$
  $$U_f(|i\rangle|0\rangle) = |i\rangle|f(i)\rangle$$

Shor’s factoring algorithm

- well-known: factoring equivalent to order finding
  - input: $y, N$
  - output: smallest $r > 0$ such that $y^r \equiv 1 \text{ mod } N$

Factoring: step 1

- given $y, N$; $f(i) = y^i \text{ mod } N$; have $\sum_i |i\rangle f(i)\rangle$
- in each vector, period = $r$, the order of $y \text{ mod } N$
- offset depends on 2nd register

Factoring: step 2

- measure register 2
- state collapses to:
  $$f(s) = \sum_{y^s \equiv y \text{ mod } N} \langle s | f(s) \rangle$$

Fourier Transform

- $N = 2^n$, $\omega$ such that $\omega^N = 1$; unitary matrix $FT = \begin{pmatrix} \omega^0 & \omega^1 & \cdots & \omega^{N-1} \\ \omega^{N-1} & \omega^{N-2} & \cdots & \omega^0 \\ \vdots & \vdots & \ddots & \vdots \\ \omega^0 & \omega^1 & \cdots & \omega^{N-1} \end{pmatrix}$
- usual FT dimension $n$; this is dimension $N$
- note: $FT|0\rangle = \text{ all ones vector}$

“quantum parallelization” key: period = $r$ (the number we are seeking)
Factoring: step 3

- Apply FT to register 1
- large in positions \( b \) such that \( r \cdot b \) close to \( N \)
- measure register 1
- obtain \( b \)
- determine \( r \) from \( b \)
  (classically, basic number theory

Quantum computation

- if can build quantum computers, they will be capable of factoring in polynomial time
  - big "if"
- do not believe factoring possible in polynomial time classically
  - but factoring in \( P \) if \( P = NP \)
- serious challenge to extended Church-Turing Thesis

The very last slide

- Fill out TQFR surveys!
- Course to consider
  - CS139 (advanced algorithms)
  - CS150 (probability and computation)
  - CS151 (complexity theory)
  - CS153 (current topics in theoretical CS)
- Good luck
  - on final
  - in CS, at Caltech, beyond...
- Thank you!