QUANTUM COMPUTING VIA DISCRETE QUANTUM WALKS AND QUANTUM GAMES

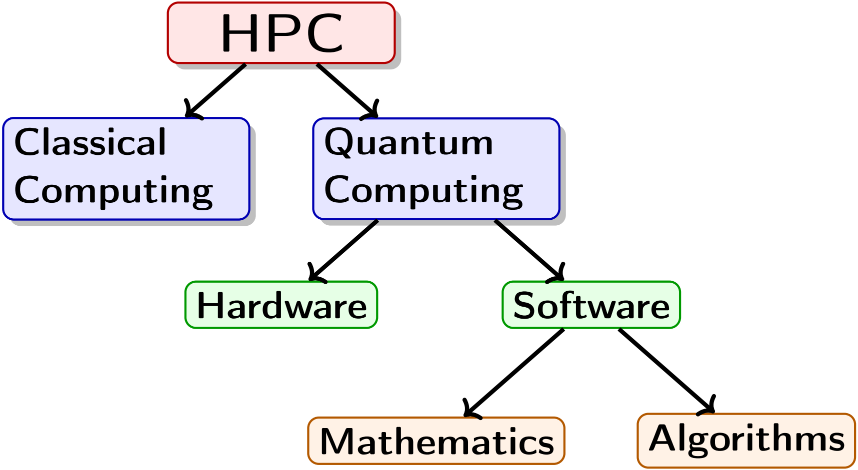
HPCSE 2026
High Performance Computing
in Science and Engineering

Michael Mc Gettrick

Ollscoil na Gaillimhe
University of Galway

Outline

1. Introduction
2. Quantum Walks
3. Alternating Quantum Walks
4. Quantum Walks with memory
5. Quantum (Evolutionary) Game Theory



HPC

Classical Computing

Quantum Computing

Hardware

Software

Mathematics

Algorithms

Any fundamental advantages to quantum computing?

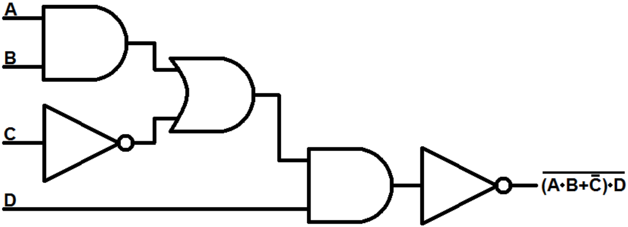
Ultimate miniaturization A qubit can be a fundamental particle (e.g. a photon). A (classical) bit needs about 50,000 particles.

Ultimate energy conservation In principle, no energy loss, since all quantum operations are reversible. In classical computing, every bit erasure causes energy loss of 3 x 10^{-21} Joules (Landauer limit).

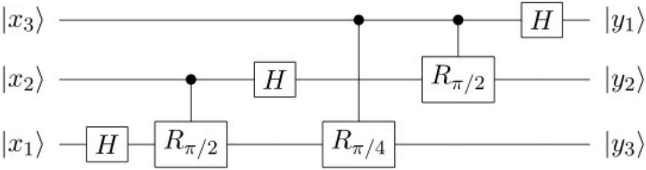
***Some* clever algorithms**
- The Grover algorithm is polynomially faster at finding an item in an unordered list ($O(\sqrt{n})$ versus $O(n)$).

What about circuits?

CLASSICAL:



QUANTUM:



	(CLASSICAL) COMPUTER SCIENCE	QUANTUM COMPUTING
WHAT DO WE WANT TO DO?	Construct (Efficient) Algorithms	Construct (Efficient) Algorithms
PHYSICS USED	None? Solid State Physics?	Quantum Mechanics
MATHEMATICS USED	Logic, Boolean Algebra	Unitary Matrices, Probability Theory
PROCEDURE	Input, iterate, output	Input, iterate, output, *measure*
PROBABILITY	L1 norm	L2 norm

- **Feynman, R.P. Quantum mechanical computers. Found Phys 16, 507–531 (1986)**
  "...we are going to be even more ridiculous later and consider bits written on one atom instead of the present 10^{11} atoms. Such nonsense is very entertaining to professors like me. I hope you will find it interesting and entertaining also."
- **Childs, A. M. (2009). Universal Computation by Quantum Walk. Physical Review Letters, 102(18), 180501.** "...quantum walk can be regarded as a universal computational primitive, with any desired quantum computation encoded entirely in some underlying graph."
- **Lovett, N. B. et al. (2010). Universal quantum computation using the discrete-time quantum walk. Physical Review A, 81(4), 042330.** "...both discrete and continuous-time quantum walks are computational primitives."

STEPS IN A 1D (CLASSICAL) RANDOM WALK

1. Toss a (Boolean) coin.
2. Move left (heads) or right (tails).
3. Repeat from first step.

ASYMPTOTICALLY GAUSSIAN PROBABILITY DISTRIBUTION

$$P(k,n) \approx 2e^{-k^2/2n}/\sqrt{2\pi n}$$

where $P(k,n)$ is the probability of finding the walker at position k after n steps and $k+n$ is even.

DIFFUSIVE SPREAD The standard deviation of the probability distribution after n steps, $\sigma(n)$, is proportional to $\sqrt{n}$.

STEPS IN A QUANTUM (RANDOM) WALK

1. Chose a qubit state from the Bloch Sphere

$$\begin{pmatrix} \cos(\theta/2) \\ e^{i\phi}\sin(\theta/2) \end{pmatrix} \tag{1}$$

2. Operate on the qubit state with the Hadamard matrix.

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

3. The first (second) component of the resulting 2-vector dictates the amplitude with which we move right (left), respectively.
4. Repeat from step 2.

$$|U0\rangle \xrightarrow{H}$$

$$\frac{1}{\sqrt{2}}(|U0\rangle + |D0\rangle) \longrightarrow \frac{1}{\sqrt{2}}(|U{-1}\rangle + |D1\rangle) \xrightarrow{H}$$

$$\frac{1}{2}(|U{-1}\rangle + |D{-1}\rangle + |U1\rangle - |D1\rangle) \longrightarrow \frac{1}{2}(|U{-2}\rangle + |D0\rangle + |U0\rangle - |D2\rangle)$$

$$\frac{1}{2\sqrt{2}}(|U{-2}\rangle + |D{-2}\rangle + |U0\rangle - |D0\rangle + |U0\rangle + |D0\rangle - |U2\rangle + |D2\rangle) \longrightarrow$$

$$\frac{1}{2\sqrt{2}}(|U{-3}\rangle + |D{-1}\rangle + \boxed{|U{-1}\rangle} - \boxed{|D1\rangle} + \boxed{|U{-1}\rangle} + \boxed{|D1\rangle} - |U1\rangle + |$$

where in the last line we see constructive interference (terms 3 and 5) and destructive interference (terms 4 and 6).

PROBABILITY PEAKS AT $\pm n/\sqrt{2}$ with a flat probability distribution in a central plateau around the origin.

BALLISTIC SPREAD The standard deviation of the probability distribution after n steps, $\sigma(n)$, is proportional to n.

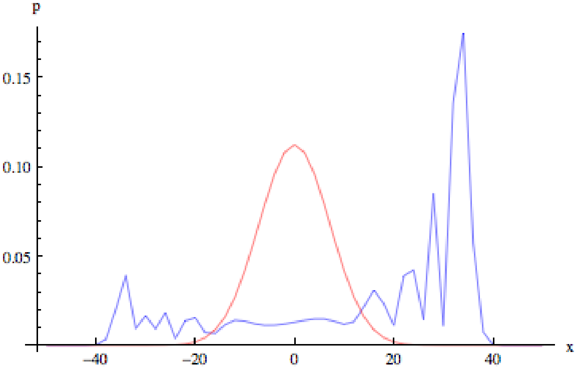


Figure 1: Probability distribution resulting from one-dimensional discrete time random walks. The quantum walk created using the Hadamard coin is plotted (blue) vs a classical walk (red) after 50 time steps. (SOURCE: Wikipedia)

Alternating Quantum Walk

An alternating quantum walk uses a two-state (Boolean) quantum system (qubit) to perform a walk in four (4) different directions: In the simplest model we walk on the square lattice Z^2.

STEPS IN AN ALTERNATING QUANTUM WALK

1. Start with an (arbitrary) qubit state.
2. Operate on the qubit state with the (Hadamard) matrix.
3. The first (second) component of the resulting 2-vector dictates the amplitude with which we move right (left), respectively.
4. Operate on the qubit state with the (Hadamard) matrix.
5. The first (second) component of the resulting 2-vector dictates the amplitude with which we move up (down), respectively.
6. Repeat from step 2.

THESE TWO MODELS COINCIDE!

model **(a)**

Standard 2D quantum walk with

- Initial state $(1, -1, -1, 1)/2$.
- Grover coin operator

$$G = \frac{1}{2}\begin{pmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix}$$

model **(b)**

Alternating quantum walk with

- Initial state $(1, i)/\sqrt{2}$.
- Hadamard coin operator

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Quantum Walk with “memory”

This is a quantum version of the classical notion of a *Higher Order Markov Chain*, where the position to which we move depends not just on our current position, but also on previous positions.

Definition

A Boolean Function of arity p is a function

$$f : \{0, 1\}^p \rightarrow \{0, 1\}$$

This allows us to define a quantum walk with p-step memory as follows:

STEPS IN A QUANTUM WALK WITH MEMORY p

1. Fix a particular boolean function f of arity p.
2. Start with an (arbitrary) qubit state.
3. Operate on the qubit state with the (Hadamard) matrix.
4. Identify the symbols $0 \equiv L$ and $1 \equiv R$, where L stands for “left” and R for “right”. The p-step history/memory is given by an element $a \in M = \{R, L\}^p$. The first component of the qubit dictates the amplitude with which we move in the direction $f(a)$ (and the second component is the amplitude for the direction opposite to $f(a)$).
5. Repeat from step 3.

For a quantum walk with memory $p = 1$, choose

$$f : \{0,1\} \to \{0,1\}, \qquad f(0) = 0, \qquad f(1) = 1$$

Since in this case $f(a) = a$, and a (“left” or “right”) corresponds to the direction, we can say the first qubit component corresponds to maintaining the same direction of motion (“transmission”) while the second component corresponds to reversing the direction (“reflection”).

For a quantum walk with memory $p = 1$, choose

$$f : \{0,1\} \to \{0,1\}, \qquad f(0) = 0, \qquad f(1) = 0$$

Since in this case $f(a) = 0$, for all a, the first/second qubit component corresponds to moving left/right (respectively) *irrespective of the previous step*. In fact - this is the same as a quantum walk without memory

Properties of the quantum walk with 1-step memory include:

FEATURES FROM BOTH CLASSICAL AND QUANTUM WALKS
There is a central peak in the probability distribution (reminiscent of the classical case) but also peaks at $n/\sqrt{2}$ (reminiscent of the quantum case).

LOCALIZATION AT THE ORIGIN By this we mean a non-zero asymptotic probability the walker is found at the origin. We have proven this asymptotic probability is greater than 0.5 (Konno and Machida calculate it to be exactly $2 - \sqrt{2}$).

What about quantum walks with long memory ($p > 1$)?

CONJECTURE

We believe that as p gets larger, the probability distribution of a quantum walk with p-step memory tends to that of a classical random walk.

For $p = 1$, our states are effectively pairs of positions (at current time step and previous time step): They are the edges of the graph. So, a $p = 1$ walk on a graph G corresponds to a $p = 0$ walk on the Line Graph of G (we denote by $\mathcal{L}(\mathcal{G})$).

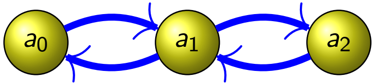
Definition

For a Graph G with vertex set V and edge set $E \subset V \otimes V$, the *Line Graph* of G, denoted by $\mathcal{L}(\mathcal{G})$, is the graph with vertex set E and edge set

$$\{((p,q),(q,r)) | p,q,r \in V, (p,q),(q,r) \in E\}$$

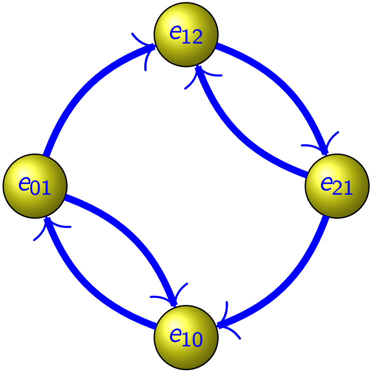
Consider a quantum walk with memory $p = 1$ on the cycle C_n. This corresponds to a quantum walk with memory $p = 0$ on the line graph $\mathcal{L}(C_n)$, which turns out to be the Cayley graph of the Dihedral group D_3.

$G =$



(1)

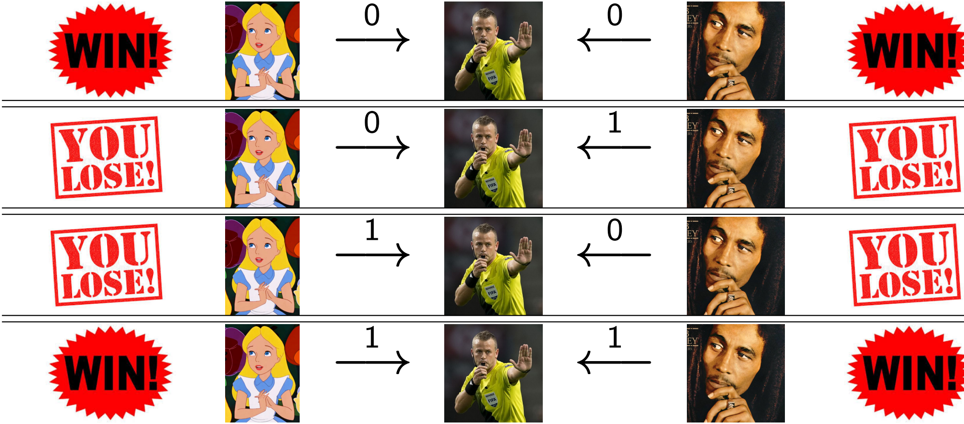
$L(G) =$



(2)

Game theory 101

- Alice picks $a \in \{0, 1\}$
- Bob picks $b \in \{0, 1\}$
- **Both** win iff $a = b$, otherwise both lose.



Bayesian games

A bit like playing a game where you are told the moves / strategies allowed, but not (exactly) the winning criterion!

- Referee sends bits x/y to Alice/Bob, respectively, $x, y \in \{0, 1\}$.
- Alice and Bob reply with bits a and b respectively, $a, b \in \{0, 1\}$.
- **Both** win iff $\boxed{a - b = xy \mod 2}$, otherwise both lose.

The referee chooses (x, y) uniformly at random (so $(0, 0), (0, 1), (1, 0), (1, 1)$ all have probability 0.25).

		BOB	
		0	1
ALICE	0	W	L
	1	L	W

(a) $(x, y) = (0, 0)$

		BOB	
		0	1
ALICE	0	W	L
	1	L	W

(b) $(x, y) = (0, 1)$

		BOB	
		0	1
ALICE	0	W	L
	1	L	W

(c) $(x, y) = (1, 0)$

		BOB	
		0	1
ALICE	0	L	W
	1	W	L

(d) $(x, y) = (1, 1)$

Non - Bayesian game

Alice and Bob can decide in advance on a winning strategy, e.g. to always output 0.

Bayesian game

Alice and Bobs strategy choice is to pick a **function** from $\mathbb{Z}_2$ to $\mathbb{Z}_2$. The four possible strategies are:

	strategy:	f1	f2	f3	f4
Input	0	0	0	1	1
	1	0	1	0	1

or...

- **f1** Play zero no matter what the referee gives you
- **f4** Play one no matter what the referee gives you
- **f2** Whatever the referee gives you, give it back to him/her
- **f3** Flip the bit the referee gives

Bayesian game strategies

With no communication allowed between Alice and Bob, there is no pre-arranged pair of strategies they can agree on that will guarantee they always win (since they do not know what bit they will receive from the referee - and so they do not know for sure which of the four games previously mentioned they are actually playing). The best they can hope for is 0.75 success rate (by for example always playing $a = b = 0$)
Writing $a = f_A(x), b = f_B(y)$, the maximum success probability is given by

$$\text{Prob(success)} = \max_{f_A, f_B : \{0,1\} \to \{0,1\}} \frac{\#\{(x,y) | f_A(x) - f_B(y) = xy \mod 2\}}{4}$$

Bayesian game over $\mathbb{Z}/n\mathbb{Z}$

Our winning condition becomes $\boxed{a - b = xy \mod n}$ with maximum success probability

$$\text{Prob(success)} = \max_{f_A, f_B : \mathbb{Z}_n \to \mathbb{Z}_n} \frac{\#\{(x, y) | f_A(x) - f_B(y) = xy \mod n\}}{n^2} = a_n$$

Alice's and Bob's strategies are now any of the n^n functions $f_A, f_B : \mathbb{Z}_n \to \mathbb{Z}_n$. This gives us the sequence

$$a_n = 1, \frac{3}{4}, \frac{2}{3}, \frac{5}{8}, \frac{12}{25}, \frac{1}{2}, \dots$$

(or, removing the denominator, $1, 3, 6, 10, 12, 18, \dots$). A Brute Force program to enumerate over the strategies is super-exponential:

n	1	2	3	4	5	6	7
Number of strategy combinations (n^{2n})	1	16	729	6×10^4	9×10^6	2×10^9	6×10^{11}

BUT WHY?

The quantum version of our original Bayesian game (over $\mathbb{Z}_2$) is called the "CHSH" game: It uses a shared entangled pair of qubits as a resource to allow Alice and Bob (**without any communication**) to win with probability $\cos^2(\pi/8) \sim 0.85$, about a ten percent improvement on the classical game. Can sharing entangled quantum states give us a better success probability in the Bayesian game over $\mathbb{Z}_n$?

	n	1	2	3	4	5	6	7	8
Maximum win probability	classical	1	0.75	0.66	0.625	0.48	0.5	?	?
	quantum	1	0.85	?	?	?	?	?	?

Input bits: x (Alice) and y (Bob) ■ Output bits: a (Alice) and b (Bob) ■ CHSH winning condition: $a - b = xy \bmod 2$, or, $x \wedge y = a \oplus b$ ■ c stands for $\cos(\pi/8)$ ■ s stands for $\sin(\pi/8)$

①	Input (x, y)	(0,0)	(0,1)	(1,0)	(1,1)
②	Alice's θ choice	0	0	$\pi/4$	$\pi/4$
③	Bob's θ choice	$-\pi/8$	$\pi/8$	$-\pi/8$	$\pi/8$
④	(θ_A, θ_B)	$(0, -\pi/8)$	$(0, \pi/8)$	$(\pi/4, -\pi/8)$	$(\pi/4, \pi/8)$
⑤	Rotation matrices $M = R(\theta_A) \otimes R(\theta_B)$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \begin{pmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & c & s \\ 0 & 0 & -s & c \end{pmatrix}$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} c & -s \\ s & c \end{pmatrix} = \begin{pmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & c & -s \\ 0 & 0 & s & c \end{pmatrix}$	$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \begin{pmatrix} c & s \\ -s & c \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} c & s & c & s \\ -s & c & -s & c \\ c & s & -c & -s \\ -s & c & s & -c \end{pmatrix}$	$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \begin{pmatrix} c & -s \\ s & c \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} c & -s & c & -s \\ s & c & s & c \\ c & -s & -c & s \\ s & c & -s & -c \end{pmatrix}$
⑥	Initial shared entangled state (time $t = 0$), ψ_0	$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$			
⑦	After local rotations (time $t = 1$), $\psi_1 = M\psi_0$	$\frac{M}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} c \\ -s \\ s \\ c \end{pmatrix}$	$\frac{M}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} c \\ s \\ -s \\ c \end{pmatrix}$	$\frac{M}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} c+s \\ c-s \\ c-s \\ -(c+s) \end{pmatrix}$	$\frac{M}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} c-s \\ c+s \\ c+s \\ s-c \end{pmatrix}$
⑧	Probability of output $(a, b) = (0, 0)$	$c^2/2$	$c^2/2$	$(c+s)^2/4$	$(c-s)^2/4$
⑨	Probability of output $(a, b) = (0, 1)$	$s^2/2$	$s^2/2$	$(c-s)^2/4$	$(c+s)^2/4$
⑩	Probability of output $(a, b) = (1, 0)$	$s^2/2$	$s^2/2$	$(c-s)^2/4$	$(c+s)^2/4$
⑪	Probability of output $(a, b) = (1, 1)$	$c^2/2$	$c^2/2$	$(c+s)^2/4$	$(c-s)^2/4$
⑫	Winning values of (a, b)	$(0, 0), (1, 1)$	$(0, 0), (1, 1)$	$(0, 0), (1, 1)$	$(0, 1), (1, 0)$
⑬	Probability of winning	$\boldsymbol{c^2}$	$\boldsymbol{c^2}$	$\boldsymbol{(c+s)^2/2}$	$\boldsymbol{(c+s)^2/2}$
⑭	Overall winning probability	$(\boldsymbol{c^2} + \boldsymbol{c^2} + \boldsymbol{(c+s)^2/2} + \boldsymbol{(c+s)^2/2})/4 = (2c^2 + 1 + 2cs)/4$ $= (2 + \cos(\pi/4) + \sin(\pi/4))/4 = (1 + \sqrt{2})/(2\sqrt{2})$			