Introduction To The Physics Of D-Wave and Comparison To Gate Model

North Carolina State University
Joel M. Gottlieb
March 20, 2018

Gate Model vs. Quantum Annealing

- "What is the logical difference between the qubits on the D-Wave machines and the others on the circuit based machines (IBM Quantum Experience)? I'm not so much interested in the hardware, but in why D-Wave goes up to 2K qubits but the others are struggling to get to 50 or so. I'm assuming it has something to do with D-Wave computers being annealing machines, but I'd like a little more depth of understanding about what that really means."

- Customer question, 2017

Overview

- *Brief Review* Quantum Computers
- **Gate Model Quantum Computers**
- **Quantum Annealing**
- **D-Wave Physics**
- **Conclusions**

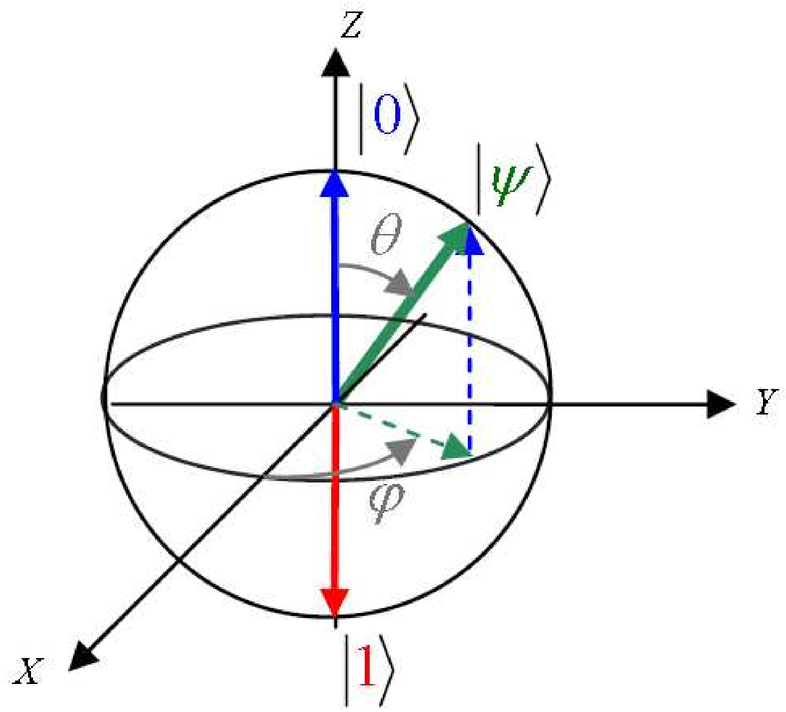
Quantum Computing Overview

- Classical computers are composed of
  - Logic
  - Registers
  - Memory
  - Disk
  - Networks

- Quantum computers are composed of
  - Quantum bits ≡ qubits
  - Qubit registers
  - State preparation mechanism
  - Unitary transformations
  - Measurement operation

A single qubit inhabits the Bloch Sphere

- A qubit is a two-state quantum-mechanical system
- A qubit can be represented as a linear combination of basis states:
- Collections of qubits live in Hilbert space



Qubit registers & wavefunctions

0	0	0	0	α_0
0	0	0	1	α_1
0	0	1	0	α_2
0	0	1	1	α_3
0	1	0	0	α_4
0	1	0	1	α_5
0	1	1	0	α_6
0	1	1	1	α_7

1	0	0	0	α_8
1	0	0	1	α_9
1	0	1	0	α_{10}
1	0	1	1	α_{11}
1	1	0	0	α_{12}
1	1	0	1	α_{13}
1	1	1	0	α_{14}
1	1	1	1	α_{15}

4-qubit register : each α_i is a complex amplitude
The collection of all α_i is called a wavefunction

Overview

- ***Brief Review* Quantum Computers**
- Gate Model Quantum Computers
- **Quantum Annealing**
- **D-Wave Physics**
- **Conclusions**

Gate Model Quantum Computing

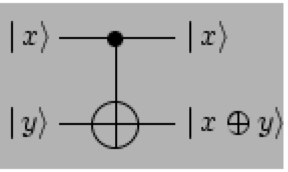
- AKA circuit model
- Idea: replace classical gates with quantum equivalents
- Classical gates become unitary operators on Hilbert space
- A quantum register is described by a wavefunction
- Gate model algorithms proceed as follows:
    1. Initialize the wavefunction
    2. Apply a sequence of gates (i.e. unitary operators)
    3. Measure the final wavefunction of the system
- It is almost always the case that the final measurement collapses the wavefunction

Many Gates

- Quantum physics puts restrictions on types of gates that can be used in quantum computer
- Must be reversible
- Classical NOT gate is reversible, but AND, OR and NAND are not
- Matrices must be unitary
- Pauli X, Y and Z
- Hadamard and many more
- Build up calculations using gates

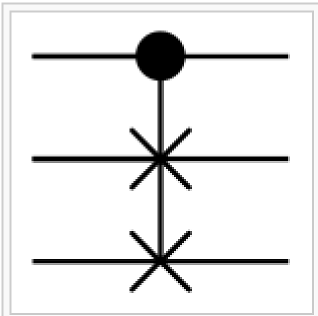
Quantum Circuit Unitary Building Blocks

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$



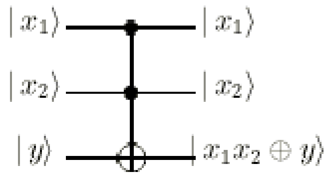
CNOT gate

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



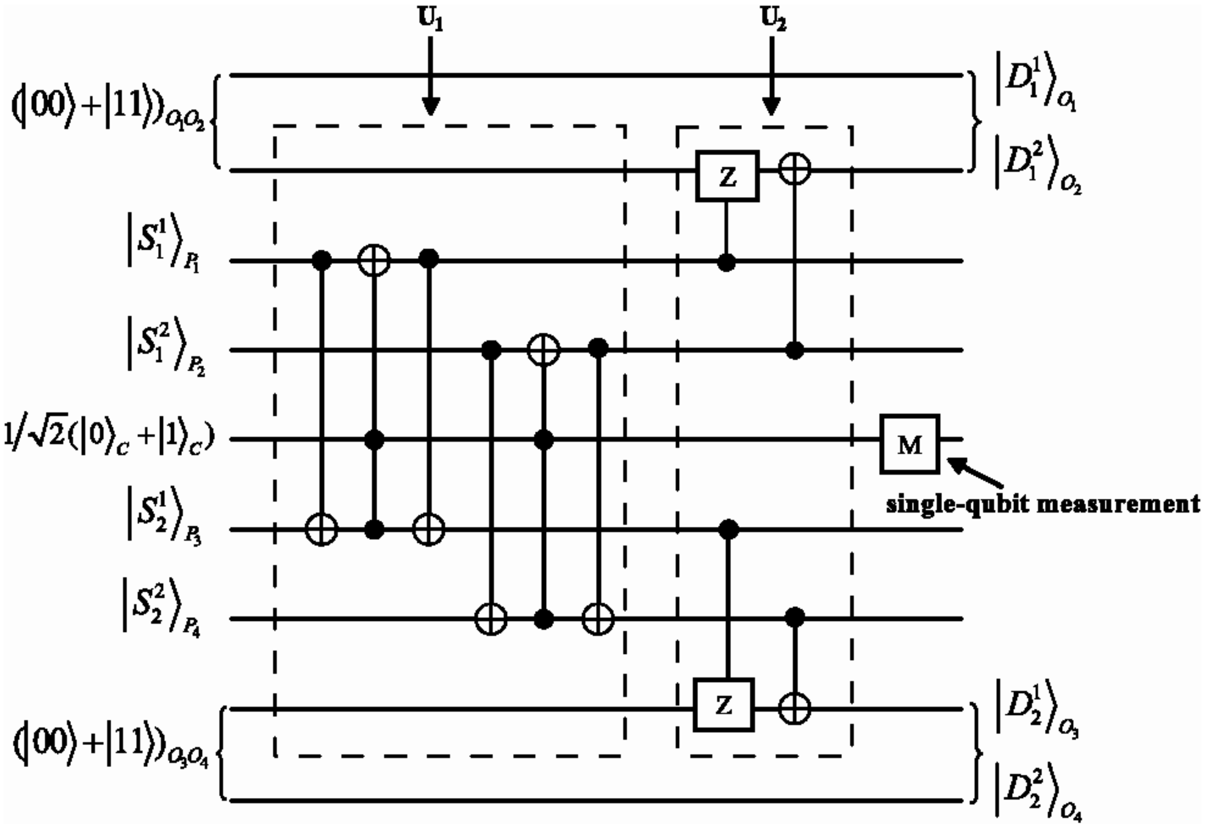
Fredkin 3-bit gate

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Toffoli 3-bit gate

Sample Quantum Circuit



Gate Model Quantum Computing – March 2018

- Ion Q (University of Maryland) – 5 qubits – trapped ions
- IBM – 5 qubit on Web; 50 qubits in lab
- Rigetti – Forest API released
- Intel – January 2018 - "Tangle Lake" 49 qubits, announced
- March 5, 2018: Google announces Bristlecone, 72 qubit processor, 1% error rate on readout, 0.1% error rate for one-qubit gates, and 0.6% for two-qubit gates
- Limited-connectivity – nearest-neighbor interactions in Google processor, for example

Challenges for Gate Model

- You have to know your physics to program it
- You can run Shor's algorithm and Grover's algorithm
- Error rates are very important
- Gates have a cost in time – must take into account the system coherence time

Overview

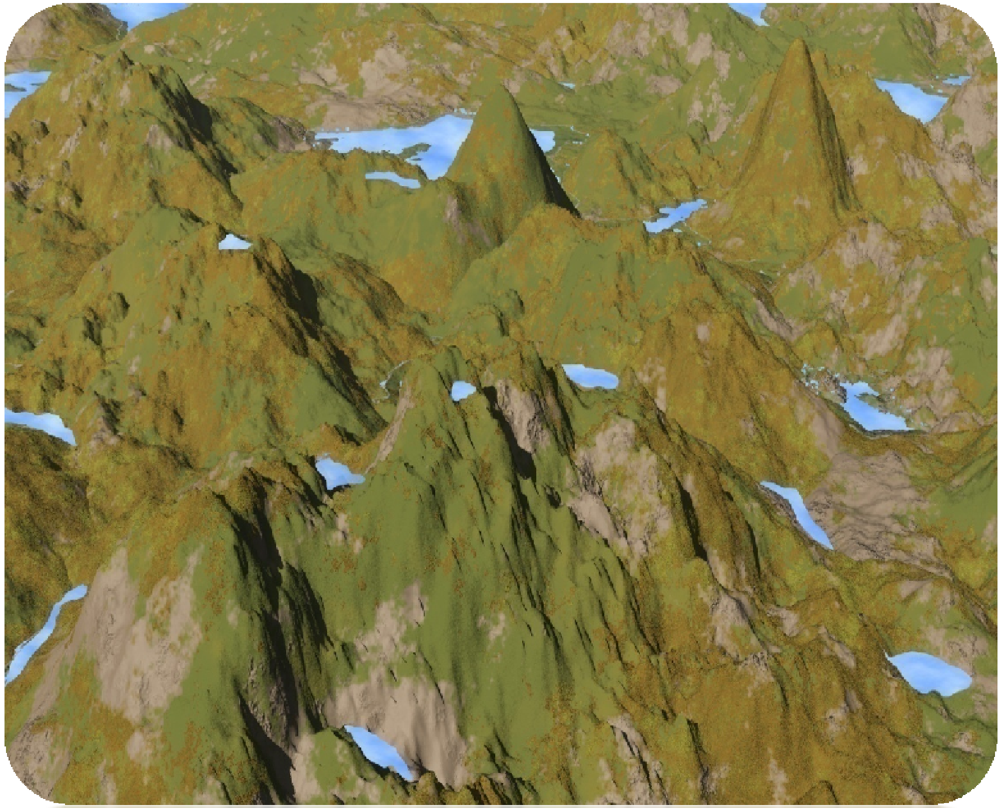
- *Brief Review* Quantum Computers
- **Gate Model Quantum Computers**
- Quantum Annealing
- **D-Wave Physics**
- **Conclusions**

Quantum Annealing History

- Kadowaki & Nishimori, “Quantum Annealing in the Transverse Ising Model” (1998)
- E. Farhi, J. Goldstone, S. Gutmann, M. Sipser, Quantum Computation by Adiabatic Evolution (2000)
- E. Farhi et. al, A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem (2001)
- W. M. Kaminsky, S. Lloyd and T. P. Orlando, Scalable Superconducting Architecture for Adiabatic Quantum Computation (2004)

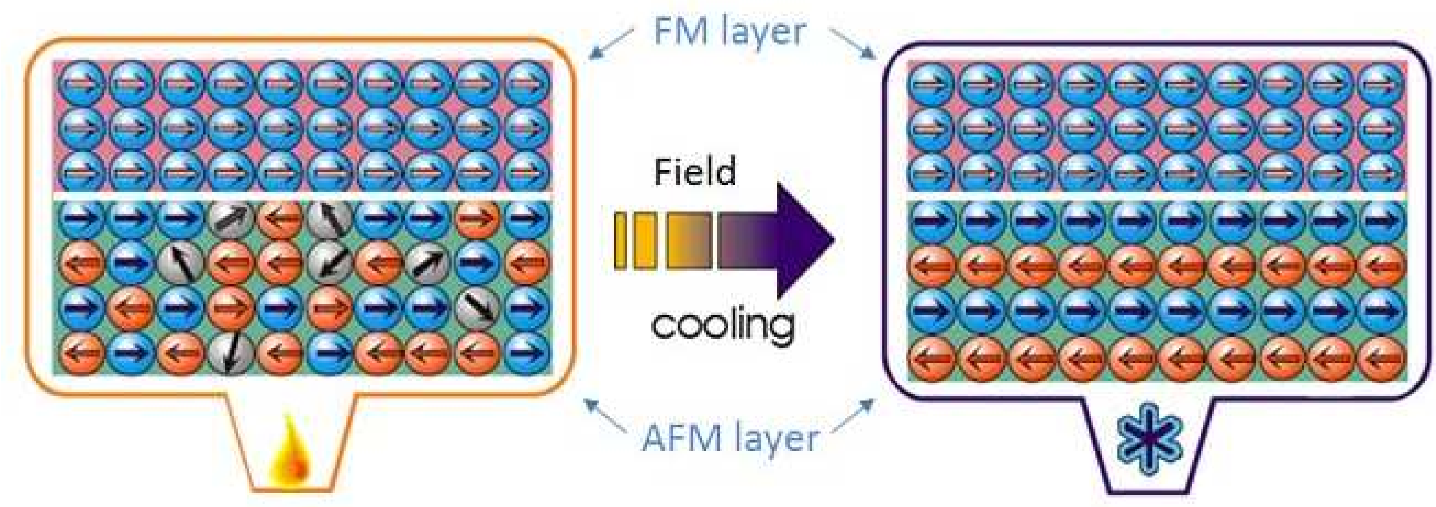
Landscape metaphor

- Space of solutions defines an energy landscape & best solution is lowest valley
- Classical algorithms can only **walk over** this landscape
- Quantum annealing uses **quantum effects** to go



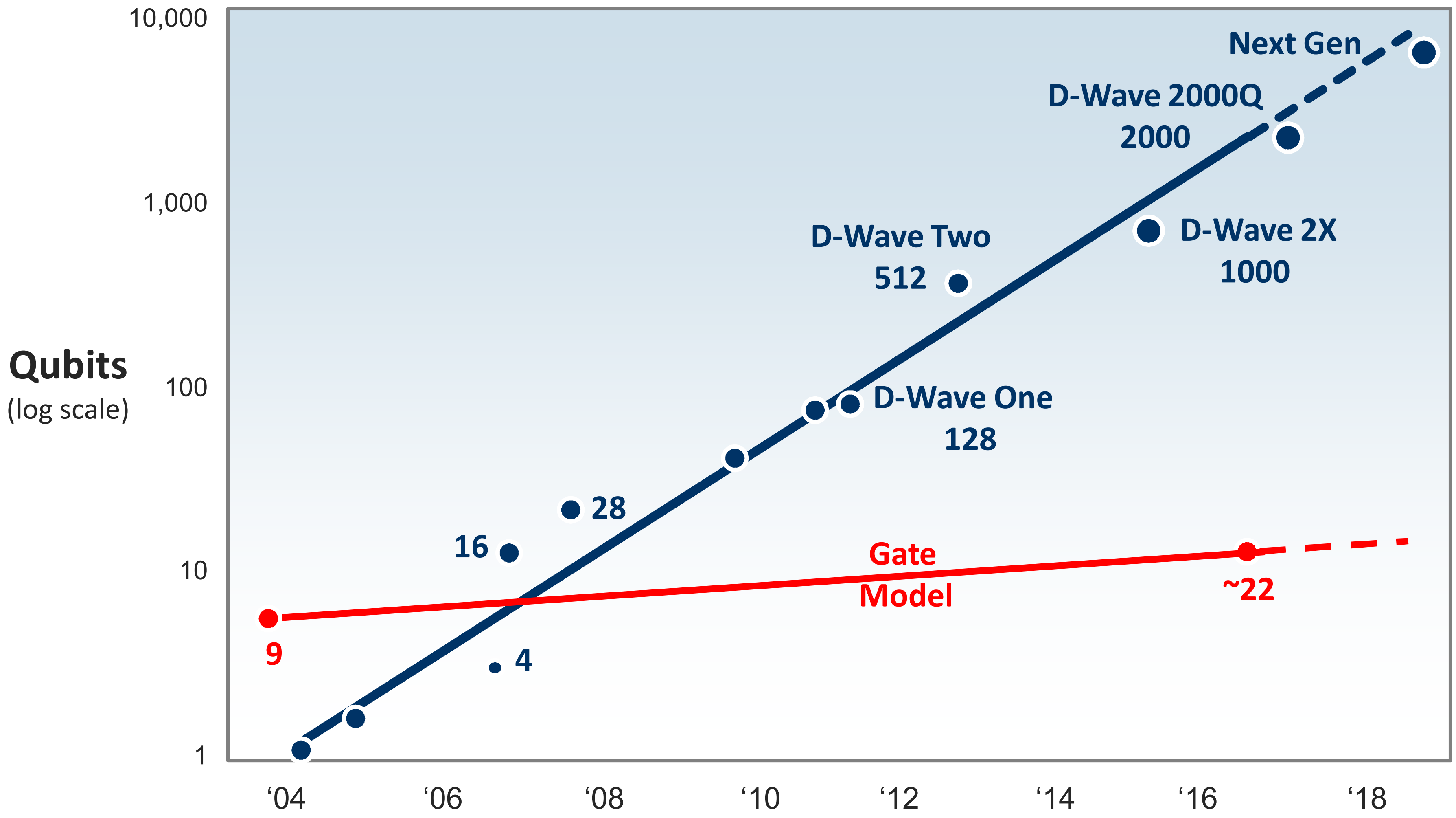
Annealing perspective

Thermal effects help systems explore complicated energy landscapes and find stable minima.



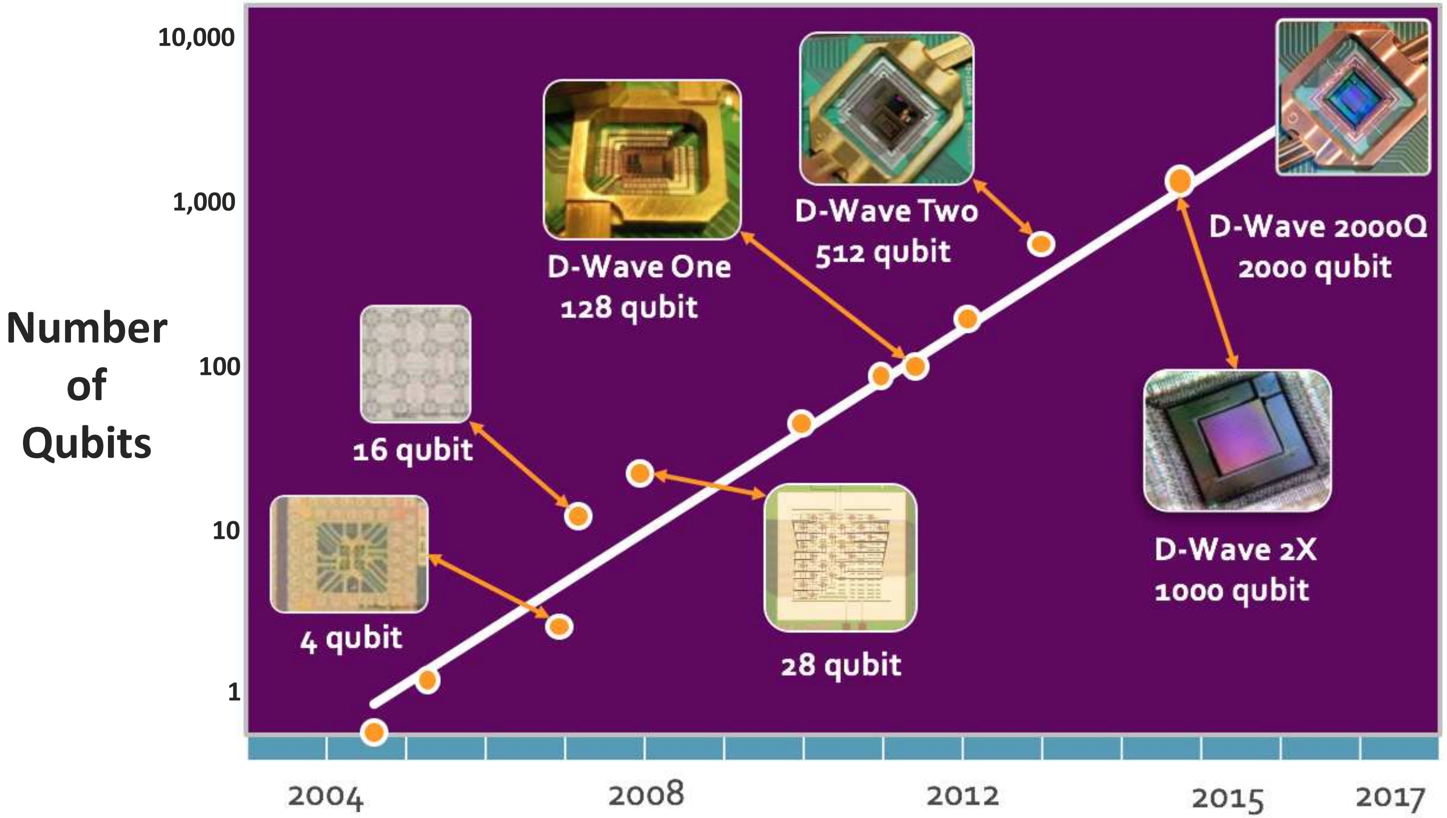
Quantum effects can do the same.

Annealing is far more scalable



D::WAVE The Quantum Computing Company™

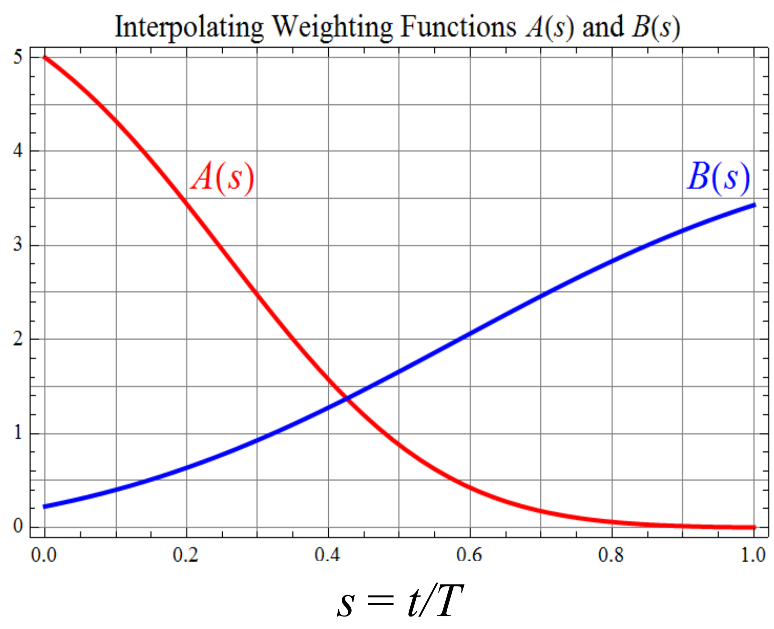
D-Wave Product Generations



Adiabatic Quantum Optimization perspective

A system will remain in the ground state of all the instantaneous Hamiltonians passed through provided the change is made sufficiently slowly (i.e., **“adiabatically”**)

- $H(s) = A(s)\, H_{\text{initial}} + B(s)\, H_{\text{final}}$
- where $s = t/T$



Quantum Enhanced Optimization

Quantum Hamiltonian is an operator on Hilbert space:

$$\mathcal{H}(s) = A(s) \sum_i \sigma_i^x + B(s) \left[\sum_i a_i \, \sigma_i^z + \sum_{i<j} b_{ij} \, \sigma_i^z \sigma_j^z \right]$$

Corresponding classical optimization problem:

$$\mathrm{Obj}(a_i, b_{ij}; q_i) = \sum_i a_i q_i + \sum_{i<j} b_{ij} q_i q_j$$

D-Wave The Quantum Computing Company™

Review: Programming Model

QUBIT	q_i	Quantum bit which participates in annealing cycle and settles into one of two possible final states: $\{0,1\}$
COUPLER	$q_i q_j$	Physical device that allows one **qubit** to influence another **qubit**
WEIGHT	a_i	Real-valued constant associated with each **qubit**, which influences the **qubit's** tendency to collapse into its two possible final states; controlled by the programmer
STRENGTH	b_{ij}	Real-valued constant associated with each **coupler**, which controls the influence exerted by one **qubit** on another; controlled by the programmer
OBJECTIVE	Obj	Real-valued function which is minimized during the annealing cycle

$$Obj(a_i, b_{ij}; q_i) = \sum_i a_i q_i + \sum_{ij} b_{ij} q_i q_j$$

The system samples from the q_i that minimize the objective

D-Wave The Quantum Computing Company™

Programming the D-Wave QPU

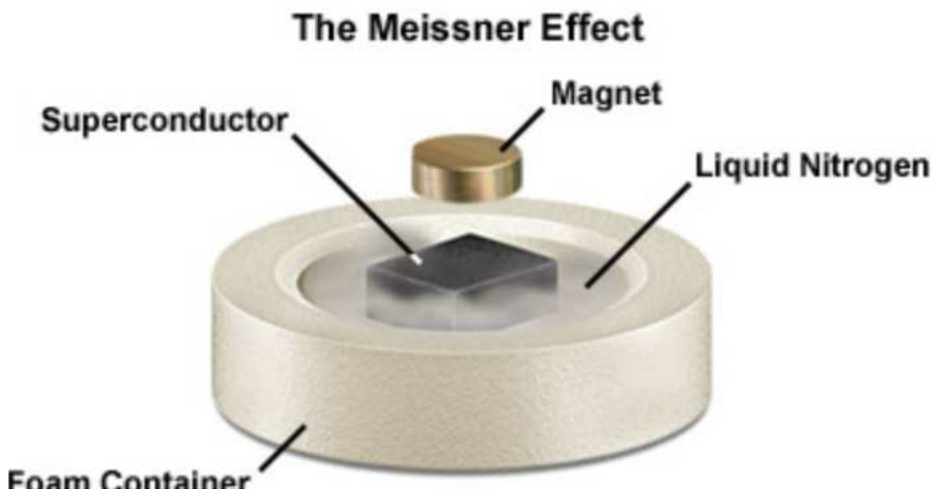
- You don’t have to know your quantum physics to program it
- If you can turn your problem into a QUBO, you can explore it using the QPU
- It can’t run Shor’s algorithm or Grover’s algorithm, which are designed for gate model quantum computers

Overview

- ***Brief Review*** **Quantum Computers**
- **Gate Model Quantum Computers**
- **Quantum Annealing**
- D-Wave Physics
- **Conclusions**

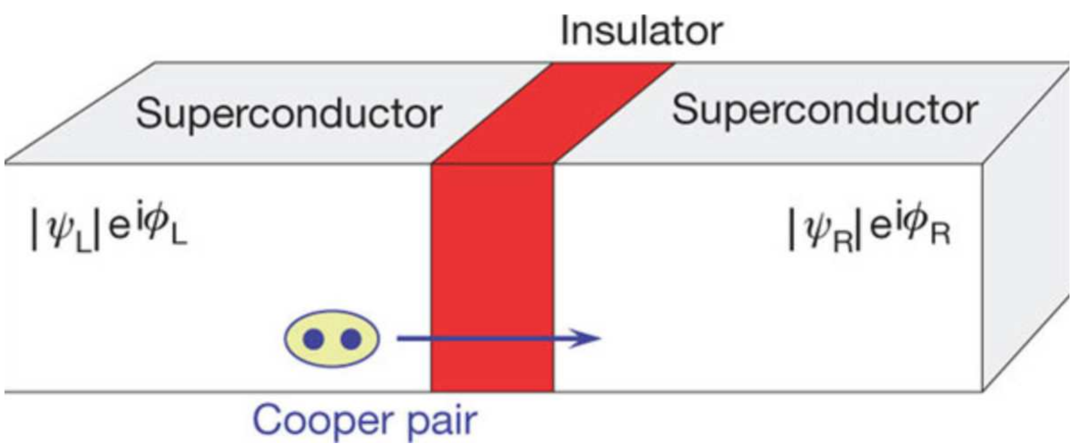
Superconductivity

- If you cool many metals and alloys to a very low temperature (the *critical temperature*), the material goes from a normal state (with resistance) to a *superconducting* state (with no resistance)
- Electrons in the metal form *Cooper pairs*, which can move w/o interacting with the ions of the lattice – thus no resistance
- However, there is a *critical current* – maximum current
- Superconductor expels magnetic fields (Meissner Effect)



Josephson Junction

- Josephson effect: Two superconductors coupled by thin non-superconducting barrier (insulator or metal)
- There is current/voltage across the weak link/junction
- Explained by tunneling of superconducting Cooper pairs
- Brian D. Josephson, 1962 (Nobel Prize, 1973)
- There is both DC Josephson effect, and AC Josephson effect

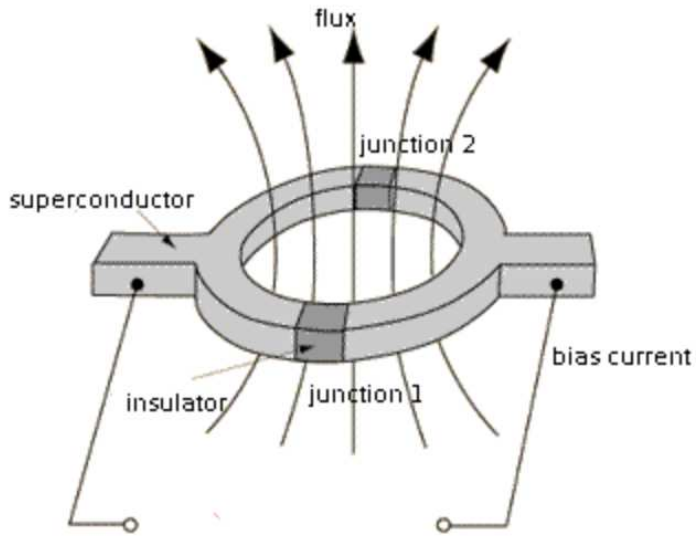


D-Wave The Quantum Computing Company™

DC SQUID

- SQUID = Superconducting Quantum Interference Device
- Superconducting loops contains two Josephson junctions
- The voltage you measure across the device is very strongly correlated to the total magnetic field
- Quantized magnetic flux through loop
- Can detect miniscule flux/voltage

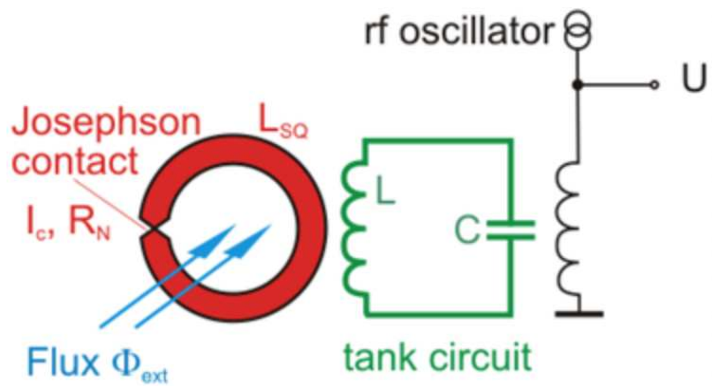
changes



RF (Radio Frequency) SQUID

- Single Josephson junction on superconducting ring (SQUID), with an external LC circuit ("tank circuit")
- An oscillating (rf) current is applied to the external LC circuit, whose voltage changes as an effect of the interaction between the circuit and superconducting ring
- When the tank circuit is excited at or near its resonant frequency, the amplitude of the oscillating voltage across oscillates in response to the magnetic flux

threading the SQUID



D-Wave The Quantum Computing Company™

D-Wave's Goal

- Create adiabatic quantum optimization (AQO) quantum programming unit (QPU) as programmable chip
- Build programmable chip as a grid of flux qubits
- Chip must be fabricated as integrated circuit, and robust against fabrication variations
- Must work well at milli-Kelvin temperatures, without significant interference and thermal effects
- Must be able to represent and solve Ising model problems using voltages and magnetic fields to control chip circuitry
- Must be able to read flux qubits at end of adiabatic evolution when quantum mechanical tunneling is done
- Each problem must be independent of the one before – fully reprogrammable and no hysteresis

An individual flux qubit

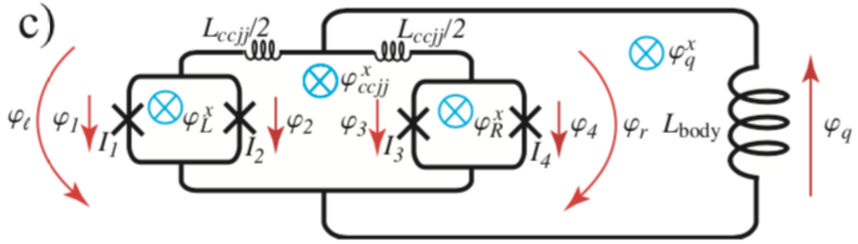
- The single-junction rf-SQUID flux qubit was found to be too limited, and too sensitive to fabrication variations, to be used in D-Wave's quantum annealer
- Compound Josephson Junction (CJJ) rf-SQUID flux qubit was found to require time-dependent flux bias compensation – too difficult to stabilize on the needed timescales
- Compound-Compound Josephson Junction (CCJJ) rf-SQUID flux qubit allows time-independent flux bias compensation signals
- Compensation can be done using either analog control lines/

on-chip programmable flux sources

- Also needed tunable

inductive interqubit couplers

c)



D-Wave The Quantum Computing Company™

The important details to remember

- Build network of **rf SQUID** devices, some are **qubits** and some are **couplers**
- In order to implement an Adiabatic Quantum Optimization algorithm in which the global transverse magnetic field is altered, one has to be able to tune the tunneling energy in situ. This is accomplished by incorporating at least one **dc-SQUID** loop into an **rf-SQUID** body
- Controllable transverse magnetic field term (Pauli x matrix)
- All single spin tunneling energies, proportional to transverse magnetic field) are much larger than energy scales involved in problem of interest – spins relax into ground states
- Then decrease transverse magnetic field until it is smaller than the energy scales of problem of interest
- Flux qubits are read at end of adiabatic evolution when QM tunneling is done

So how does the problem actually run?

- Your h’s and J’s are turned into voltages, currents and magnetic fields (the problem is programmed onto the D-Wave QPU)
- The qubit spins begin in their superposed states
- The qubit spins evolve, exploring problem space
- By the end of the annealing cycle, the system is in the ground state, or a low excited state, of the submitted problem
- The states of the spins are read, optional postprocessing is applied, and delivered back to the user
- This can be done hundreds or thousands of times per second

What is the difference vs. Gate Model?

- D-Wave's physics relies on time-independent signals, whereas gate model cannot
- Gate model approaches require explicit use of excited states as computational states, which then makes this approach particularly sensitive to decoherence.
- Gate model seems to be capable of becoming a universal quantum computer, but that goal is a long way off

Overview

- ***Brief Review*** **Quantum Computers**
- **Gate Model Quantum Computers**
- **Quantum Annealing**
- **D-Wave Physics**
- *Conclusions*

Conclusions

- D-Wave's QPU uses superconducting flux qubits with Josephson junctions, as qubits and couplers
- Gate model QC companies have some of the same challenges, and some different ones
- Algorithms space, thus far, is totally different
- Incredibly competitive arena – stay tuned
- Perhaps I have motivated you to get involved!
- jgottlieb@dwavesys.com