# Quantum Annealing

some slides originate
from Scott Pakin (LANL)

# Quantum Architectures

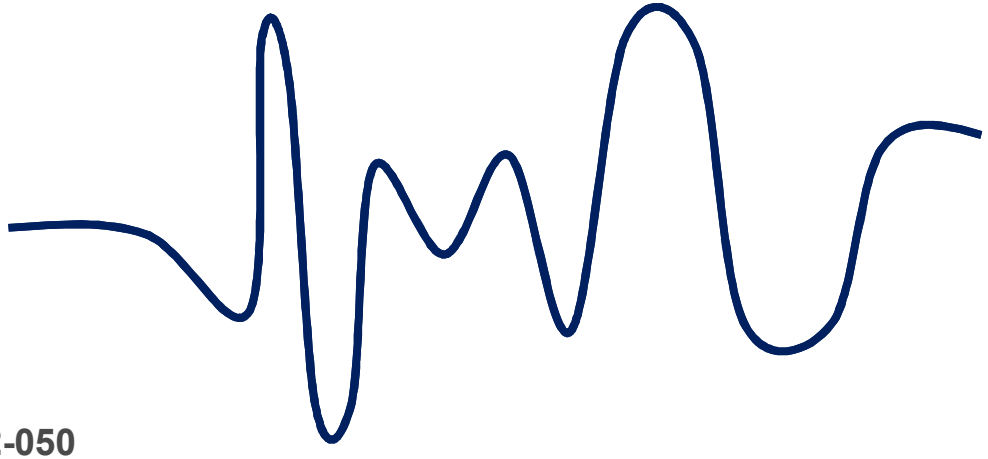
1. Quantum annealer (D-Wave)
   - Specialized: optimization problems → find lowest energy level
   - Uses tunneling and entanglement
   - Better than classical? → unknown, maybe significant speedup
2. Approximate quantum [gate] computer (IBM Q, Regetti, IonQ…)
   - More general: optimization, quantum chemistry, machine learning
   - Superposition, entanglement
   - Better than classical? → likely, sign. speedup for more problems
3. Fault-tolerant quantum computer (in some years from now)
   - Deals w/ errors (noise) algorithmically
   - Most general: crypto, search, and any of the above ones
   - Need 1000 physical qubits per virtual ("error-free") qubit
   - Better than classical? → proved theoretically

# Outline

- Performance potential of quantum computing
- Quantum annealing
- Case study: D-Wave quantum annealers
- How to program a quantum annealer
- Example: Map coloring

# Simulated Annealing

- Classical (and classic) optimization approach
- Find the coordinates of the minimum value in an energy landscape
- Conceptual approach
  - Drop a bunch of rubber balls on the landscape, evaluating the function wherever they hit
  - Hope that one of the balls will bounce and roll downhill to the global minimum
- Challenge: Commonly get stuck in a local minimum



# Quantum Mechanics to the Rescue

- Consider adding a time-dependent transverse field to a 2-local Ising Hamiltonian:

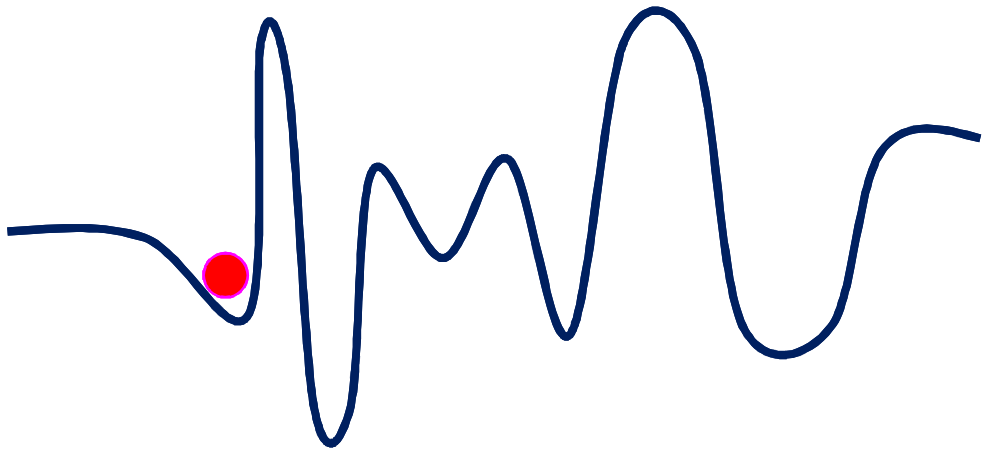
$\mathcal{H}_0$ (classical part)

$$\mathcal{H}(t) = -\underbrace{\sum_{i=0}^{N-2}\sum_{j=i+1}^{N-1} J_{i,j}\sigma_i^z\sigma_j^z}_{\text{Longitudinal interactions}} - \underbrace{\sum_{i=0}^{N-1} h_i\sigma_i^z}_{\text{Longitudinal field}} - \underbrace{\Gamma(t)\sum_{i=0}^{N-1}\sigma_i^x}_{\text{Transverse field}}$$

- Implication of adiabatic theorem:
  Let's gradually decrease amplitude of transverse field, Γ(t), from a very large value to 0 → should drive system into ground state of $H_0$
- The real benefit: quantum tunneling

# Quantum Tunneling

- Introduced by the Γ(t) (transverse) term
- Enables jumping from one classical state (eigenstate of $H_0$) to another
  - Decreases likelihood of getting stuck in a local minimum
- Unlike simulated annealing, width of energy barrier is important, but height is not



# Time Evolution

- If purely adiabatic and sufficiently slow, system remains in ground state as it moves from initial, "generic" Hamiltonian to problem Hamiltonian
- D-Wave's initial state
  - Ground state (not degenerate):
    |+>|+>|+>...|+>
  - 1st excited state $\binom{N}{1}$-way degenerate:
    |->|+>|+>... |+>,|+>|->|+>... |+>, |+>|+>|->... |+>,..., |+>|+>|+>... |->
  - 2nd excited state $\binom{N}{2}$-way degenerate: :
    |->|->|+>... |+>,|->|+>|->... |+>, |+>|->|->... |+>,..., |+>|+>|+>... |->
  - etc.

# A Brief Aside

- What we just saw is adiabatic quantum optimization
  - Optimization problem is to find the $\sigma_i^z \in \{-1,+1\}$ that minimize $H_0$
- A more powerful variation is adiabatic quantum computing

$$\mathcal{H}_{ZZXX} = \sum_{i=0}^{N-2} \sum_{j=i+1}^{N-1} J_{i,j} \sigma_i^z \sigma_j^z + \sum_{i=0}^{N-1} h_i \sigma_i^z + \sum_{i=0}^{N-2} \sum_{j=i+1}^{N-1} K_{i,j} \sigma_i^x \sigma_j^x + \sum_{i=0}^{N-1} \Delta_i \sigma_i^x$$

  - "[A]diabatic quantum computation (error free) is equivalent to the quantum circuit model (error free). So adiabatic quantum computers (error free) are quantum computers (error free) in the most traditional sense." — Dave Bacon, 27Feb2007
- Let's consider only adiabatic quantum optimization for now
  - That's all that's been built to date at large scale
  - Gate model follows later → smaller scale

# Annealing Time

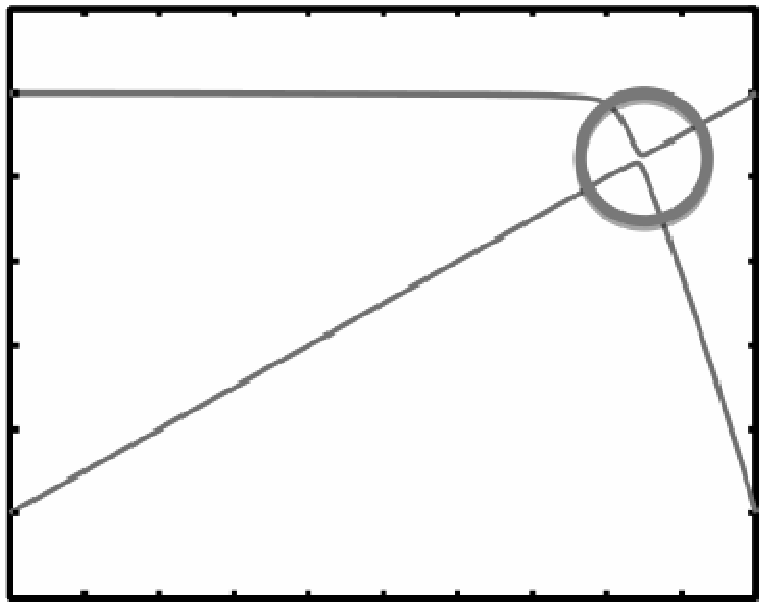
- From a few slides back:
  Let's gradually decrease amplitude of transverse field, $\Gamma(t)$, from a very large value to 0 → should drive system into ground state of $H_0$
- What does "gradually" mean? → (Explanation from Farhi & Gutmann)
  - H(t) encodes our problem
  - Want to evolve system according to Schrödinger, $i\frac{d}{dt}|\psi\rangle=H(t)\,|\psi\rangle$
  - Given that H(t) has one eigenvalue $E\neq 0$ and rest 0, find eigenvector $|w\rangle$ with eigenvector E
  - Assume we're given an orthonormal basis $\{|a\rangle\}$ with a=1,…,N and that $|w\rangle$ is one of those N basis vectors
  - Let $|s\rangle=\frac{1}{\sqrt{N}}\sum^{N}_{a=1}|a\rangle$
  - We consider Hamiltonian $H=E|w\rangle\langle w|+E|s\rangle\langle s|$ (i.e., problem+driver)
  - Let $x=\langle s|w\rangle$
  - Then, omitting a lot of math, we wind up with the probability at time t of finding the state $|w\rangle$ being $Pr(t)=\sin^2(Ext)+x^2\cos^2(Ext)$
  - To find state $|w\rangle$ with (near) certainty
    we need to run for time $t_m=\frac{\pi}{2Ex}$

# Determining the Annealing Time

- Unfortunately, we don't generally know how long we need to run (i.e., we can't quickly compute $t_m$)
- Function of minimum gap b/w two smallest eigenvalues at any point during the Hamiltonian's time evolution
- Gap can get quite small
- Grover's search (right)
  - Find an n-bit number such that

$$H_p |z\rangle = \begin{cases} |z\rangle \text{ if } z \neq w \\ 0 \quad \text{if } z = w \end{cases}$$

    for some black-box Hamiltonian $H_p$
  - Here, $g_{min} \simeq 2^{1-\frac{1}{2}}$ for n bits
  - Implication: Solution time is $O(2^n)$—no better than classical brute force




*Two lowest eigenvalues for a Grover search, 12 bits*

Image credit: Farhi, Goldstone, Gutmann, and Sipser (2000)

# Annealing Time: Discussion

The bad

- Very difficult to analyze an algorithm's computational complexity
  - Need to know the gap between the ground state and first excited state, which can be costly to compute
  - In contrast, circuit-model algorithms tend to be more straightforward to analyze
- Unknown if quantum annealing can outperform classical
  - If gap always shrinks exponentially then no
  - (Known that in adiabatic quantum computing the gap shrinks polynomially)

# Annealing Time: Discussion (cont.)

The good

- Constants do matter
  - If the gap is such that a correct answer is expected only once every million anneals, and an anneal takes 5μs, that's still only 5s to get a correct answer—may be good enough
  - On current systems, the gap scaling may be less of a problem than the number of available qubits
- We may be able to (classically) patch the output to get to the ground state
  - Hill climbing or other such approaches may help get quickly from a near-groundstate solution into the ground state
- We may not even need the exact ground state
- For many optimization problems, "good and fast" may be preferable to "perfect but slow"

# Outline

- Performance potential of quantum computing
- Quantum annealing
- Case study: D-Wave quantum annealers
- How to program a quantum annealer
- Example: Map coloring

# D-Wave's Hamiltonian

- Problem Hamiltonian (longitudinal field):

$$\mathcal{H}_P = \sum_{i=0}^{N-2} \sum_{j=i+1}^{N-1} J_{i,j} \sigma_i^z \sigma_j^z + \sum_{i=0}^{N-1} h_i \sigma_i^z$$

> **Note:**
> This is a *classical* 2-local Ising Hamiltonian

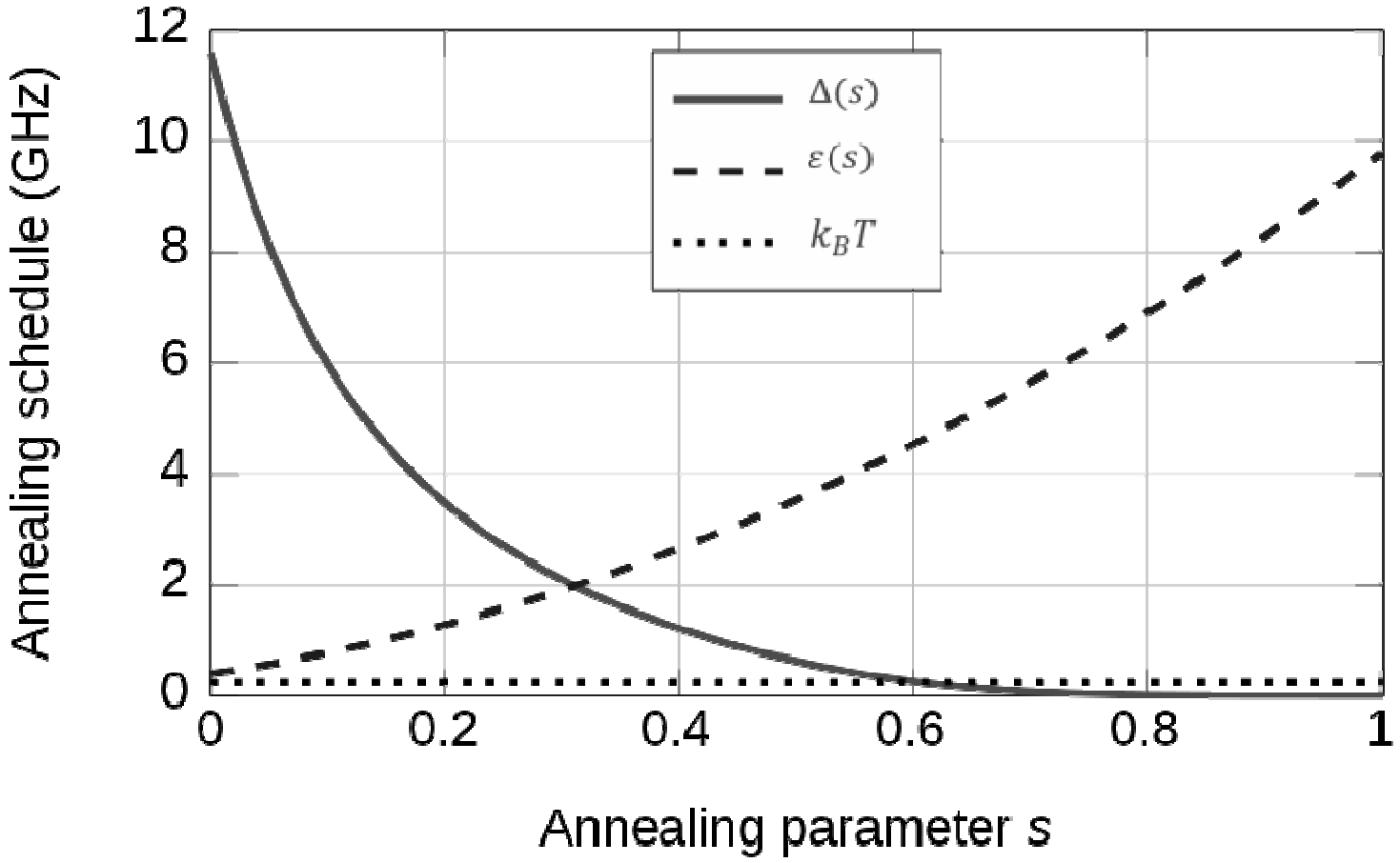
  - programmer specifies $J_{i,j}$ and $h_i$, system solves for $\sigma_i^z$
  - $\sigma_i^z \in \{-1,+1\}$
  - Nominally, $J_{i,j} \in \mathbb{R}$ and $h_i \in \mathbb{R}$, but hardware limits to a small set of distinguishable values in ranges $J_{i,j} \in [-1,+1]$ and $h_i \in [-2,+2]$

- Application of the time-dependent transverse field:

$$\mathcal{H}_S(s) = \frac{\varepsilon(s)}{2} \mathcal{H}_P - \frac{\Delta(s)}{2} \sum_{i=0}^{N-1} \sigma_i^x$$

  - Programmer specifies total annealing time, $T \in [5,2000]$ μs
  - $s = t/T$ (i.e., time normalized to [0, 1])
  - $\varepsilon(s)$ and $\Delta(s)$ are scaling parameters (not previously user-controllable but most recent hardware provides a modicum of control over the shape)

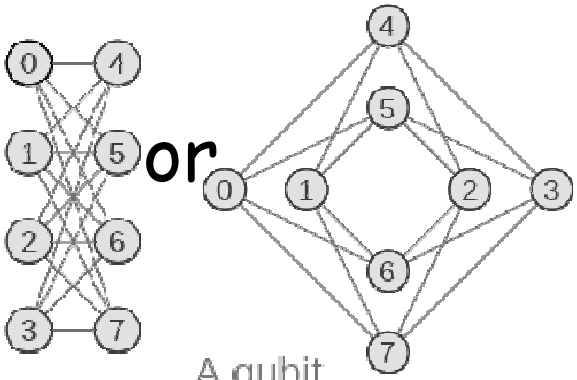
# D-Wave's Annealing Schedule

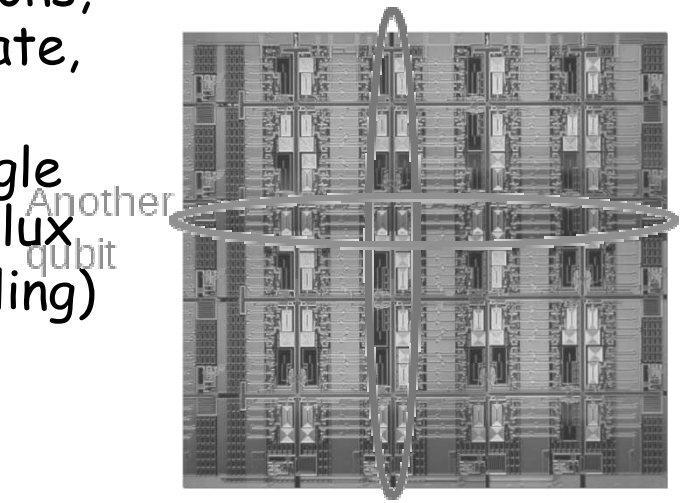


# Building Block: The Unit Cell

- Logical topology
  - 8 qubits arranged in a bipartite graph
- Physical implementation
  - Based on rf-SQUIDs
  - Flux qubits are long loops of superconducting wire interrupted by a set of Josephson junctions (weak links in superconductivity)
  - "Supercurrent" of Cooper pairs of electrons, condensed to a superconducting condensate, flows through the wires
  - Large ensemble of pairs behaves as a single quantum state w/ net positive/negative flux
  - …or a superposition of the two (w/ tunneling)
  - Entanglement introduced at qubit intersections

Logical view

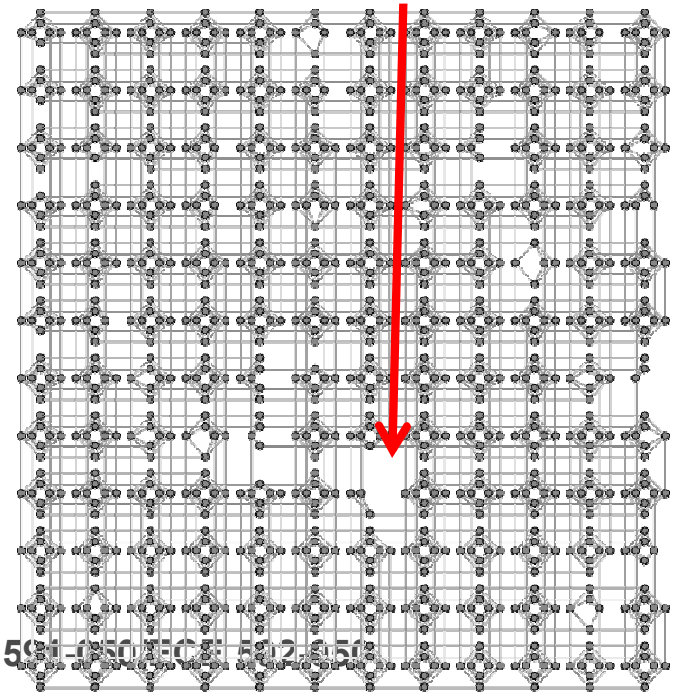


# A Complete Chip

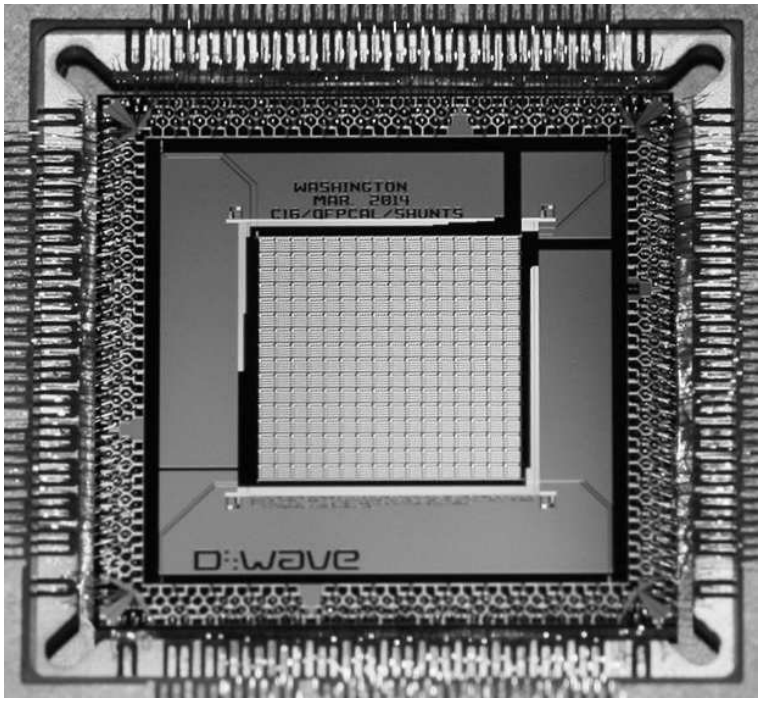
Logical view

- "Chimera graph": 16×16 unit-cell grid
- Qubits 0–3 couple to north/south neighbors; 4–7 to east/west
- Incomplete/defects(not in 2k machine)



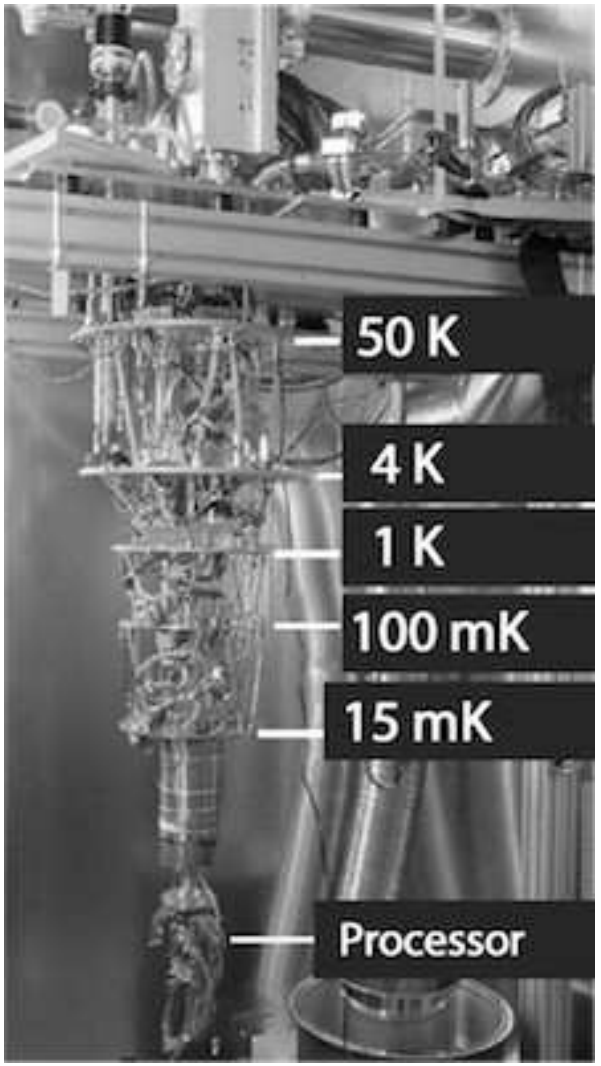
Physical view

- Chip is about the size of a small fingernail
- Can even make out unit cells with the naked eye



# Cooling

- Chip must be kept extremely cold for macroscopic circuit to behave like a two-level (qubit) system
  - Much below superconducting transition temperature (9000 mK for niobium)
- Dilution refrigerator
- Nominally runs at 15 mK
- LANL's D-Wave 2X runs at 10.45 mK
  - That's 0.01°C above absolute zero
  - For comparison, interstellar space is far warmer: 2700 mK



# What You Actually See

- A big, black box
  - 10′×10′×12′ (3m×3m×3.7m)
  - Mostly empty space
  - Radiation shielding, dilution refrigerator, chip + enclosure, cabling, tubing
  - LANL also had to add a concrete slab underneath to reduce vibration
- Support logic
  - Nondescript classical computers
  - Send/receive network requests, communicate with the chip, …

# Deviation from the Theoretical Model

- **No all-to-all connectivity**
  - Each qubit can be directly coupled to at most 6 other qubits
  - Qubits/couplers are absent (irregular, installation-specific)
- **Not running at absolute zero**
- **Not running in a perfect vacuum**
- **No error correction**
- **We can therefore think of our Hamiltonian as being**

$$\mathcal{H}_S(s) = \frac{\varepsilon(s)}{2}\left(\sum_{\langle i,j\rangle} J_{i,j}\sigma_i^z\sigma_j^z + \sum_{\langle i\rangle} h_i\sigma_i^z\right) - \frac{\Delta(s)}{2}\sum_{\langle i\rangle} h_i\,\sigma_i^x + \mathcal{H}_?(s)$$

**where $H_?(s)$ encapsulates the interaction with the environment**
  - i.e., all things we don't know and can't practically measure
  - Nonlinear and varies from run to run
- **takes time to set up a problem and get results back**
  - *Before: reset + programming + post-programming thermalization*
  - *After: readout*
  - these dominate annealing time by many orders of magnitude

# Summary

- Is any quantum computer today faster than a modern classical computer?
  - No, not for any real problem today
  - Read fine print: Google's 108X speedup for a D-Wave-friendly problem vs. non-optimal classical algorithm on single core
- Will quantum computers eventually outperform classical computers?
  - Likely, but not guaranteed
  - For adiabatic quantum optimization, more murky answers…
    - Instead of $O(2^n) \rightarrow O(n^k)$, may see speedup by sign. linear factor
- Gate model hard to program → no std techniques
  - how to represent data, write algorithms? → art of quantum pgm.
- Need methods, tools, collection of algorithms, appl. Areas

# Summary (cont.)

- Some opportunities may arise:
  - Given NP-complete (or NP-hard) problem

| method | write algo | run | solution |
|---|---|---|---|
| Classical (brute force) | Easy | Slow | exact |
| Classical (heuristic) | Hard | Fast | Approximate |
| Quantum annealing | Easy | Fast | Approximate |
| Approx. quantum gate | Hard | Maybe faster | Approximate |
| Fault-tolerant quantum | Tough! | Much faster | Exact |

- FT-quantum computer → "quantum supremacy" $O(2^n) \rightarrow O(n^k)$

- For now: QC are expensive accelerators (other than GPU/FPGA)
  - Any (linear, large factor) speedup is a big win
  - But classical will improve speed as well, watch out!