LOCKHEED MARTIN

Near-Term Applications of Quantum Annealing

Steve Adachi, Ph.D.
Lockheed Martin

International Workshop on Quantum Annealing
and its Applications in Science and Industry (QuAASI'16)
Jülich Supercomputing Centre
27 July 2016

LM Research Partners in Quantum Information Science
(partial list)

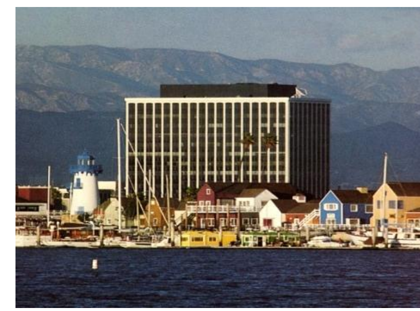
USC-Lockheed Martin Quantum Computation Center

- (May 2011) D-Wave Systems announced sale of first 128-qubit D-Wave One™ to Lockheed Martin.

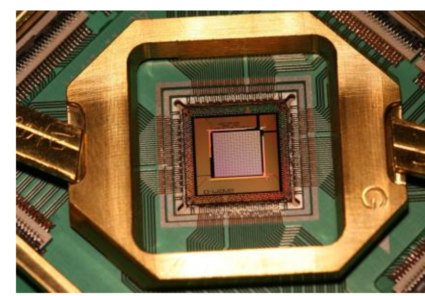
D:Wave
The Quantum Computing Company™

- (Oct 2011) USC-Lockheed Martin Quantum Computing Center unveiled at USC Information Sciences Institute, Marina del Rey, CA.

USC Viterbi
School of Engineering
Information Sciences Institute



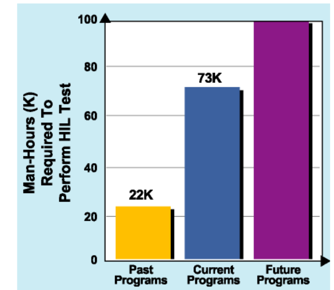
- (Mar 2013) System upgraded to 512-qubit D-Wave Two™ ("Vesuvius") chip.

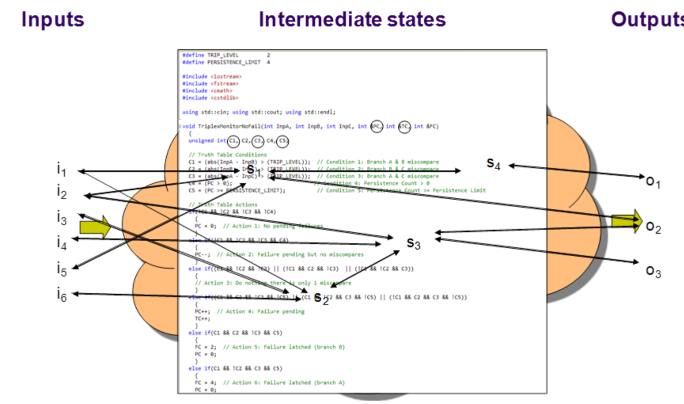


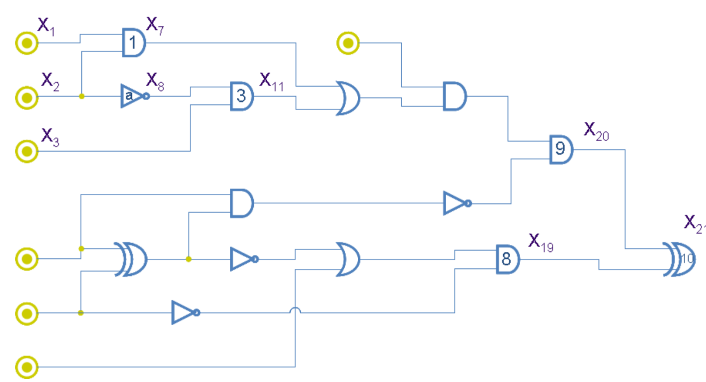
- (Mar 2016) System upgraded to 1152-qubit D-Wave 2X™ ("Washington") chip.

Software Verification & Validation (V&V)

- **Purpose**
  - To reduce V&V cost by 40% and critical path length by 50% for existing systems
- **Goal**
  - To develop enabling system-level V&V processes and tools that will generate probabilistic measures of correctness for large-scale cyber-physical systems
- **Approach**
  - Employ the D-Wave Two™ to demonstrate the utility of the process on a representative cyber-physical system







Value of Studying Near-Term Applications

- **Enables business stakeholders to better understand future potential of quantum computing technology**
  - Near-term “waypoints” may help maintain support for overall field
- **Helps to identify and prioritize future hardware improvements:**
  - Qubit connectivity
  - Control precision
  - Processor noise
  - Error correction
  - Non-stoquastic Hamiltonians
  - Etc
- **Advances understanding of how different problem types scale:**
  - Qubit resource requirements
  - Performance

Pragmatic Expectations for Quantum Computing

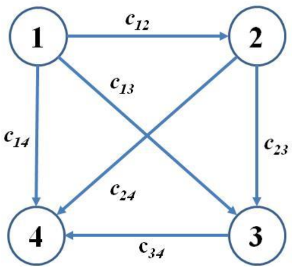
- **Don't expect Quantum Computing to replace HPC**
  - Envision a future hybrid quantum-classical computing architecture
  - Quantum computing complements HPC
  - Quantum "co-processor" for solving computationally complex sub-problems
- **Exponential speedup not required**
  - Better exponential scaling can have tremendous practical benefit
- **Quantum vs. classical benchmarking is a win-win**

Near-Term Applications Lessons Learned

- **Focus on applications that "fit" well on the D-Wave hardware**
  - Relatively low QUBO modeling overhead
  - Relatively low embedding overhead
  - Problems where "nearly optimal is good enough"
- **Tricks for reduction to quadratic order and minimizing qubit resource requirements**
  - "Gadgets"
  - Variable fixing (e.g. roof duality)
  - etc
- **Tricks for mitigating control errors and processor noise**
  - Gauge transformations
  - Parameter setting
  - etc

Example: Traveling Salesman Problem

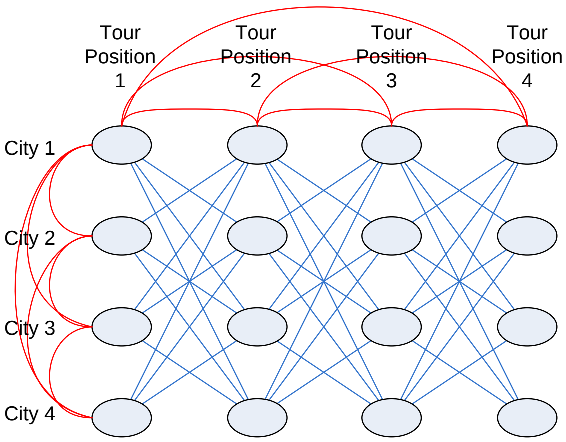
QUBO modeling overhead



Problem graph w 4 cities
size of feasible set = 24

$$\mathbf{Q} = \sum_j \sum_{(uv)\in E} W_{uv} x_{u,j} x_{v,j+1} + C_1 \sum_j \sum_{(uv)\notin E} x_{u,j} x_{v,j+1}$$
$$+ C_2 \sum_j \left(\sum_i x_{ij} - 1 \right)^2 + C_3 \sum_i \left(\sum_j x_{ij} - 1 \right)^2$$

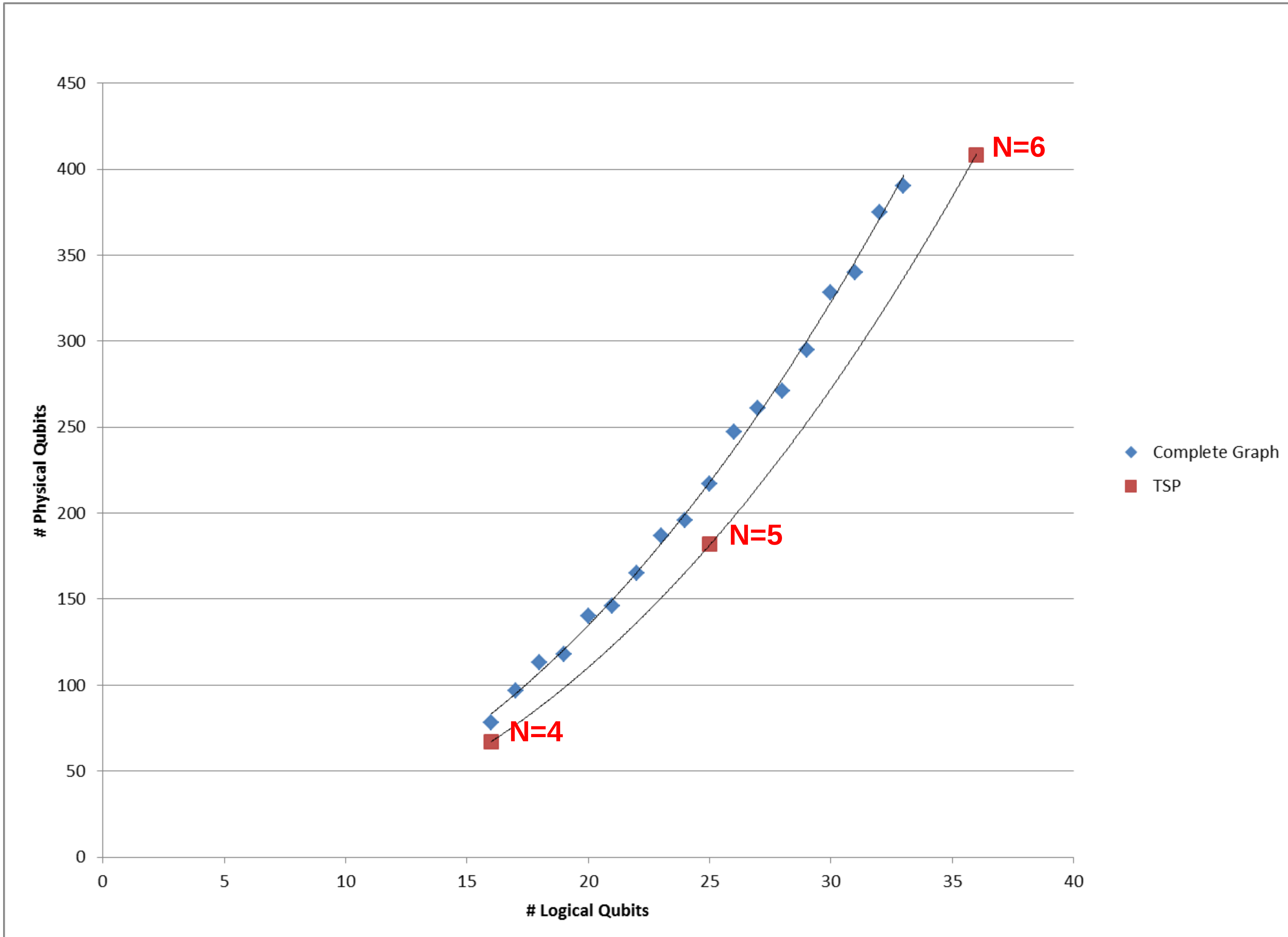
standard QUBO formulation



QUBO graph w 16 vertices
size of QUBO domain = 2^{16} = 65,536

Example: Traveling Salesman Problem

Embedding Overhead



Examples of Near-Term Applications

- **Optimization example**
  - Identifying Codes on deBruijn Graphs
- **Machine Learning example**
  - Quantum-assisted training of Deep Neural Networks
- **These examples are illustrative of problem types that “fit” relatively well on the machine, and common tricks for solving these problems using the D-Wave hardware**

Identifying Code Problem on deBruijn Graphs

Acknowledgments

This work was done in collaboration with V. Horan (V. Goliber) and S. Bak, Air Force Research Laboratory, Rome NY.

- Horan, V., Adachi, S., Bak, S. (2016) A comparison of approaches for finding minimum identifying codes on graphs. Quantum Information Processing 15(5) 1827-1848.

Work was performed under Air Force Cooperative Research and Development Agreement 14-RI-CRADA-02. Approved for public release; distribution unlimited: 88ABW-2015-2163.

S. Adachi was supported by Internal Research and Development funding from Lockheed Martin.

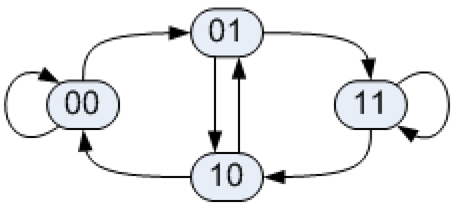
Identifying Code Problem: Informal Example

- Want to install smoke detectors in a house.
- Nodes in the graph represent rooms.
- A smoke detector can detect a fire in the same room or in an adjacent room.
  - If there is an edge from node A to node B, then a smoke detector placed in room A can detect a fire in room B.
- Want to install enough smoke detectors so that:
  - If a fire occurs in any room, it will be detected
  - If a fire occurs in one room, it can be uniquely determined which room the fire is in, by knowing which smoke detectors went off
- Questions:
  - What is the minimum number of smoke detectors that need to be installed?
  - How many different ways are there to place this number of smoke detectors?

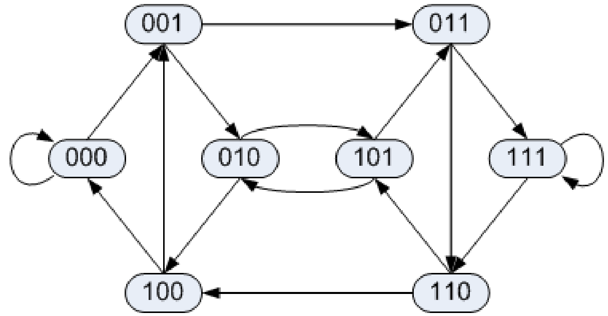
Examples of Identifying Codes (1)

DIRECTED deBruijn Graphs

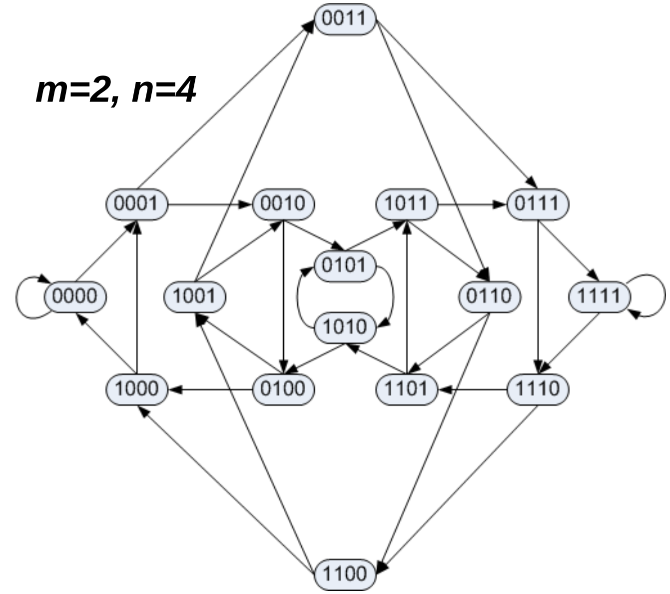
m=2, n=2

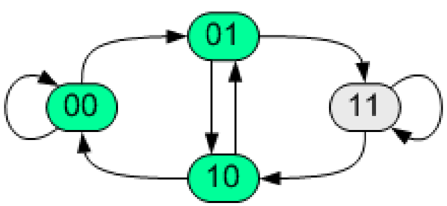


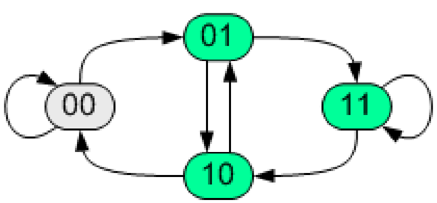
m=2, n=3

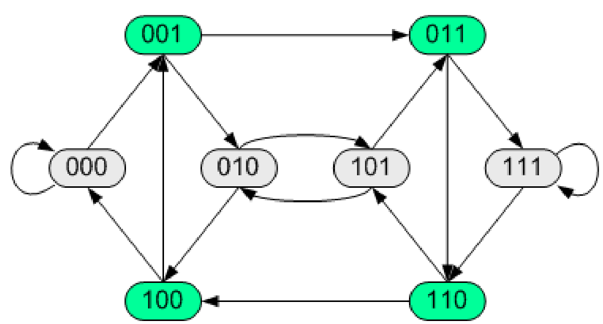


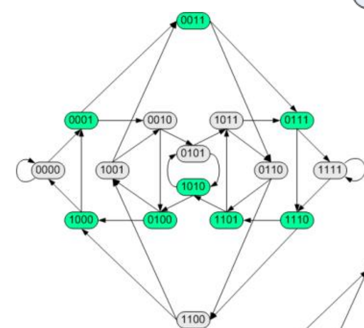
m=2, n=4

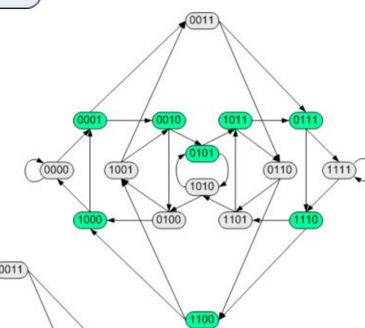


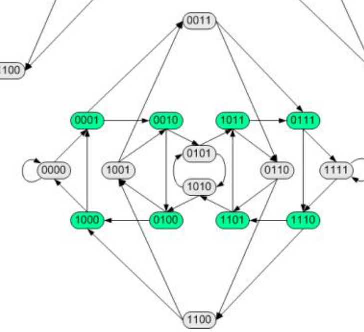








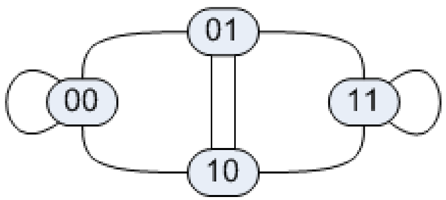




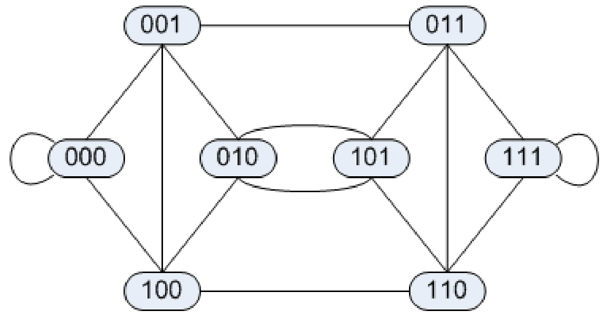
Examples of Identifying Codes (2)

UNDIRECTED deBruijn Graphs

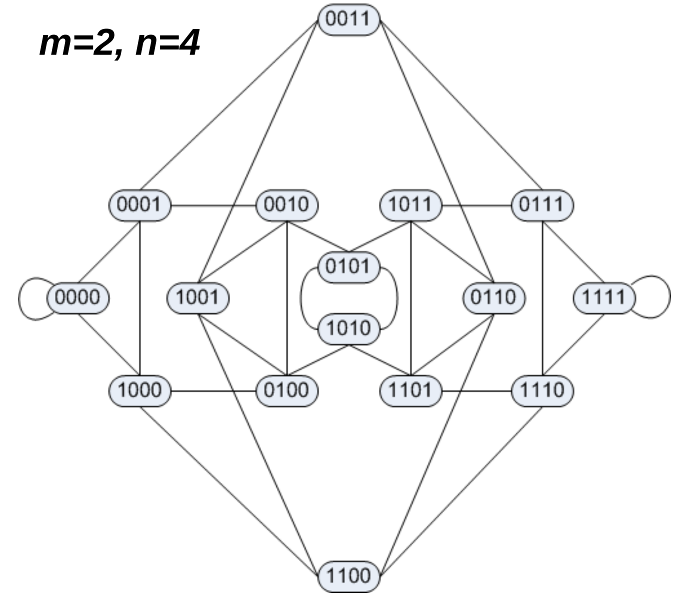
m=2, n=2



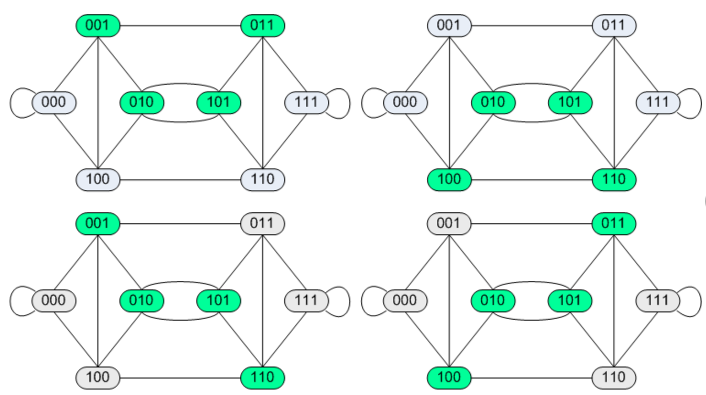
m=2, n=3

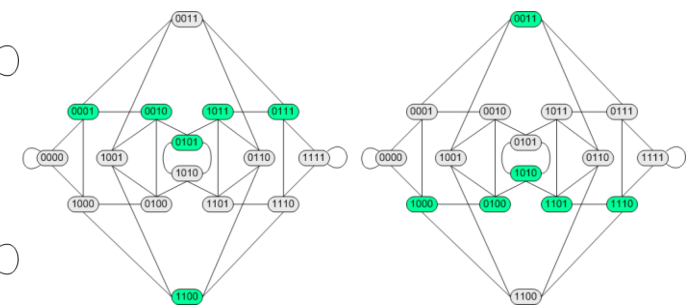


m=2, n=4



(none)





Known Solutions

- Directed case
  - For m=2, n=2 the min code length =3
  - For all other (m,n), V. Horan proved that the min code length = $m^n - m^{n-1}$
- Undirected case
  - Brute force (exhaustive search) results on Condor:

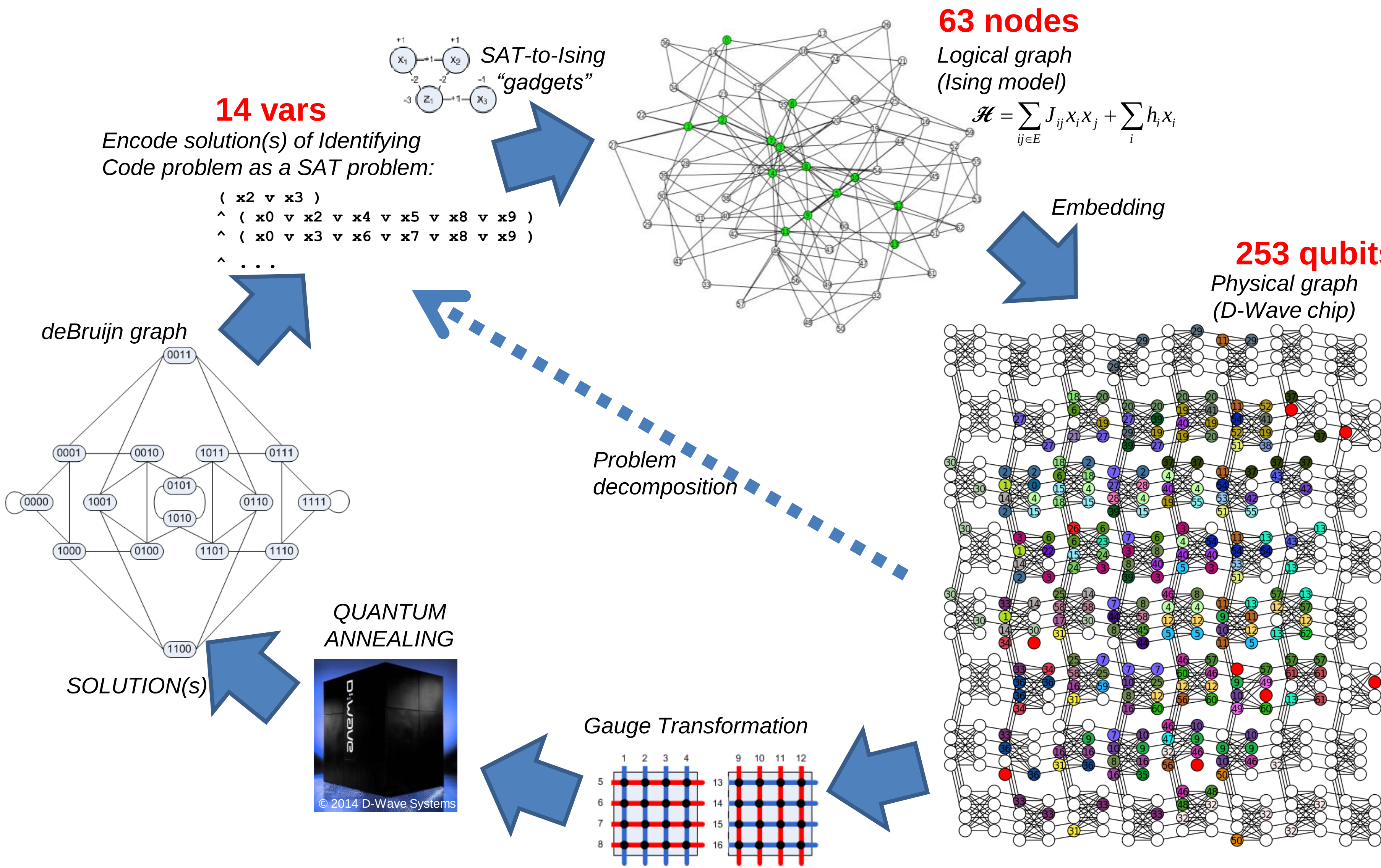
Solved on Vesuvius

Min code length (# solns)		n					
		2	3	4	5	6	7
m	2	NA (0)	4 (4)	6 (2)	12 (58)	≤24	≤110
	3	4 (3)	9 (4366)	?	?	?	?
	4	5 (396)	?	?	?	?	?
	5	6 (240)	?	?	?	?	?
	6	8 (82,890)	?	?	?	?	?
	7	9	?	?	?	?	?
	8	≤10	?	?	?	?	?

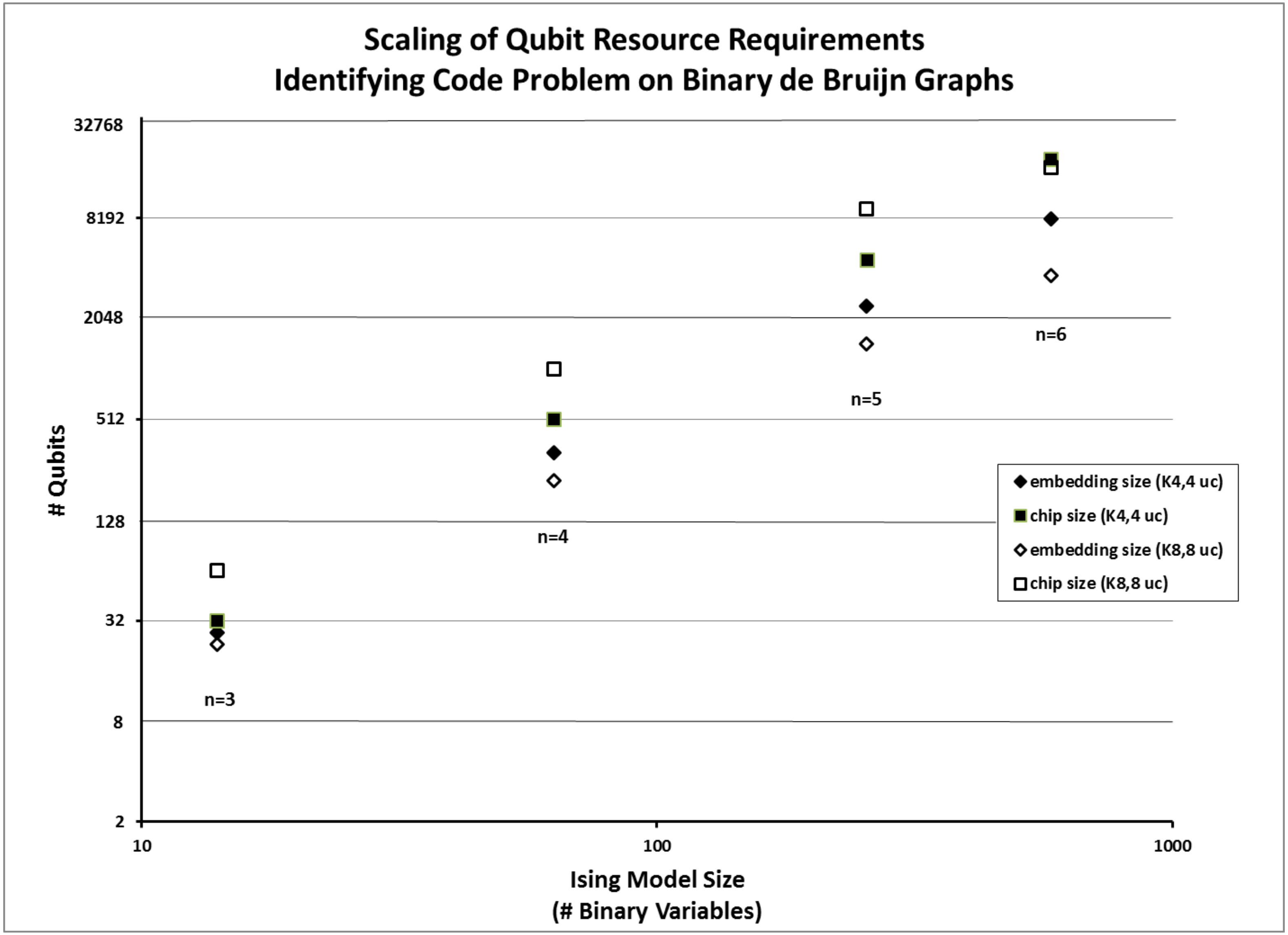
BLUE entries found by S. Bak using Z3 SMT solver
≤ indicates upper bound

- Somewhat reminiscent of the Ramsey number problem

Solution Approach



Estimates of qubits to solve larger instances (m=2 case)



Quantum-Assisted Training of Deep Neural Networks

References

Our Paper

- Adachi, S.H., Henderson, M.P. (2015) Application of Quantum Annealing to Training of Deep Neural Networks. http://arxiv.org/abs/1510.06356

Related Work

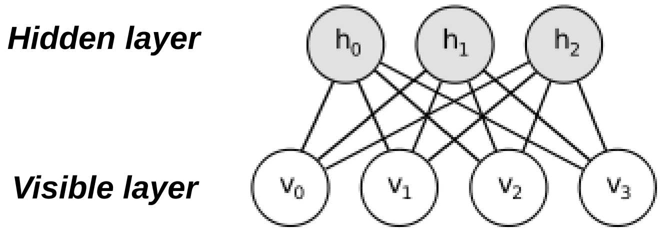
- Denil, M., de Freitas, N. (2011). Toward the implementation of a quantum RBM. NIPS*2011 Workshop on Deep Learning and Unsupervised Feature Learning.
- Dumoulin, V., Goodfellow, I.J., Courville, A., Bengio, Y. (2014) On the Challenges of Physical Implementations of RBMs. AAAI 2014: 1199-1205.
- Rose, G. (2014) First ever DBM trained using a quantum computer https://dwave.wordpress.com/2014/01/06/first-ever-dbm-trained-using-a-quantum-computer/
- Benedetti, M., Realpe-Gómez, J., Biswas, R., Perdomo-Ortiz, A. (2015) Estimation of effective temperatures in a quantum annealer & its impact in sampling applications: A case study towards deep learning applications. http://arxiv.org/abs/1510.07611
- Amin, M.H., Andriyash, E., Rolfe, J., Kulchytskyy, B., Melko, R. (2016) Quantum Boltzmann Machine. https://arxiv.org/abs/1601.02036

Beyond Quantum Annealing / D-Wave

- Wiebe, N., Kapoor, A., Svore, K.M. (2014) Quantum Deep Learning. http://arxiv.org/abs/1412.3489

Idea: How quantum sampling is applied to training of RBMs

- Restricted Boltzmann Machine model:



Energy functional *W = weights; b,c = biases*

$$E(v,h) = -\sum_i b_i v_i - \sum_j c_j h_j - \sum_{ij} W_{ij} v_i h_j$$

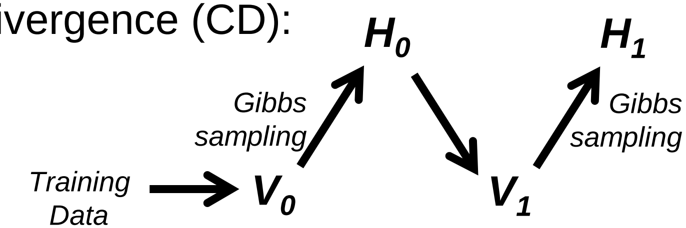
Joint probability distribution

$$P(v,h) = \frac{e^{-E}}{Z} \quad \text{where} \quad Z = \sum_{v,h} e^{-E}$$

- Weight updates are determined by the formula

$$\Delta w_{ij} \propto \frac{\partial \log P}{\partial w_{ij}} = < v_i h_j >_{data} - < v_i h_j >_{model}$$

- Second term is intractable; this has motivated approximate schemes such as Contrastive Divergence (CD):



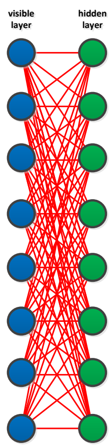
"Contrastive Divergence" (CD-1)

$$\Delta w_{ij} \propto \langle H_1 V_1 \rangle - \langle H_0 V_0 \rangle$$

- However, CD can take many iterations to converge (related to slow mixing of Gibbs sampling)
- We attempt to use *quantum sampling* to estimate the "intractable" term directly
  - Quantum sampling has the potential to mix faster (e.g. due to *tunneling)*

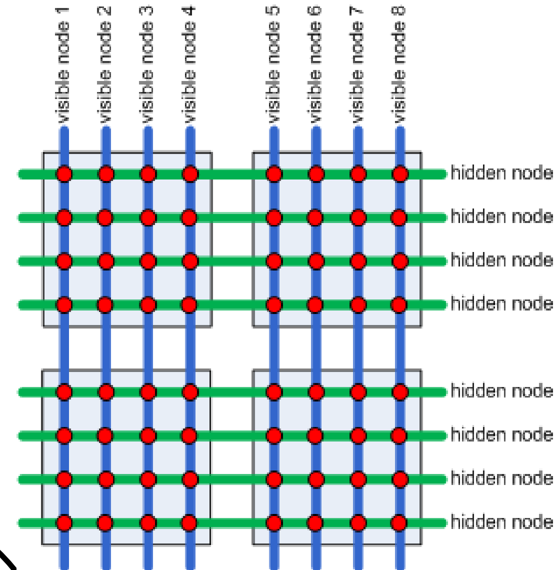
Quantum-assisted training

RBM



$$E(v,h) = -bv - ch - vWh$$

$$P(v,h) = \frac{e^{-E}}{Z}$$



Embedding

$$P(v,h) = \frac{e^{-\beta_{eff}\mathcal{H}_f}}{Z'}$$ ***(ansatz)***

Update weights

$$w_{ij}^{(t+1)} = \alpha w_{ij}^{(t)} + \epsilon[\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}]$$

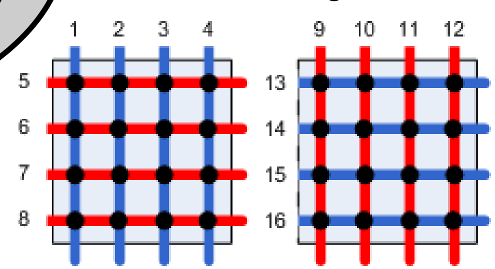
CLASSICAL COMPUTER

Training Outer Loop

Estimate model expectations using quantum sampling

$$< v_i h_j >_{model} \sim \overline{v_i\, h_j} = \frac{1}{N}\sum_{n=1}^{N} v_i^{(n)}\, h_j^{(n)}$$

Gauge transf.



Test Case: “Coarse Grained” MNIST

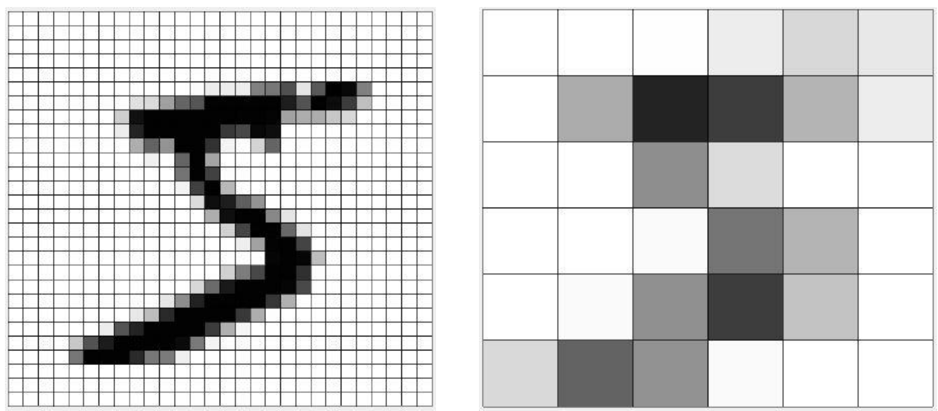
MNIST data set (http://yann.lecun.com/exdb/mnist)

- **Handwritten digits 0-9**
- **60,000 training and 10,000 test set images with truth labels**
- **Each image consists of 784 greyscale pixels (28x28)**

To fit the problem on Vesuvius, we “coarse-grained” the images:

- **We discarded 2 pixels on each edge, leaving a 24x24 image**
- **We computed the average pixel value over each 4x4 block, resulting in a coarse-grained 6x6 image**

original image (28x28)



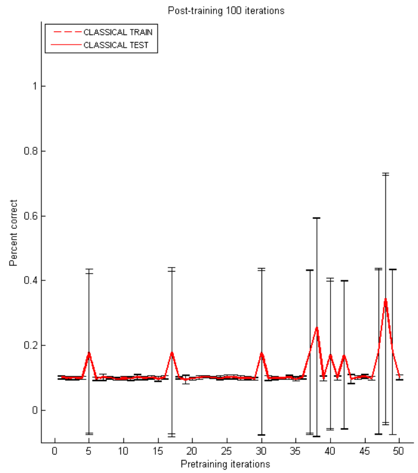
coarse-grained image (6x6)

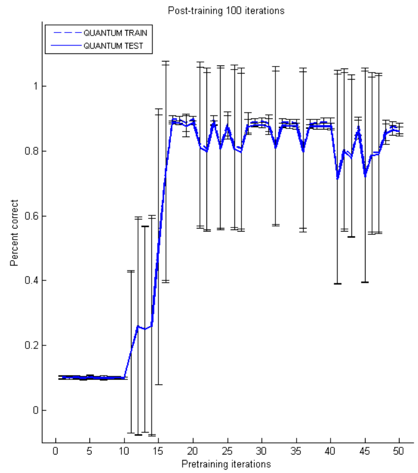
Original and coarse-grained versions of image from MNIST data set (handwritten digit 5)

- **We discarded the 4 corners, resulting in 32 super-pixels**
- **A more challenging recognition problem than the real MNIST!**

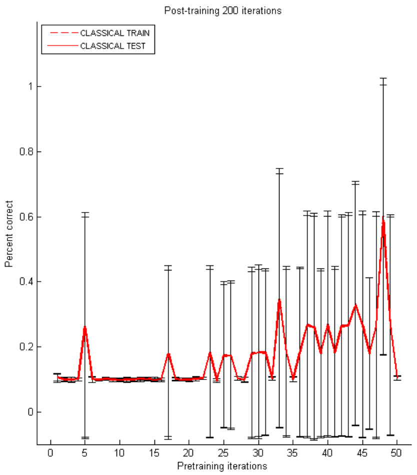
Results for CG-MNIST Data Set

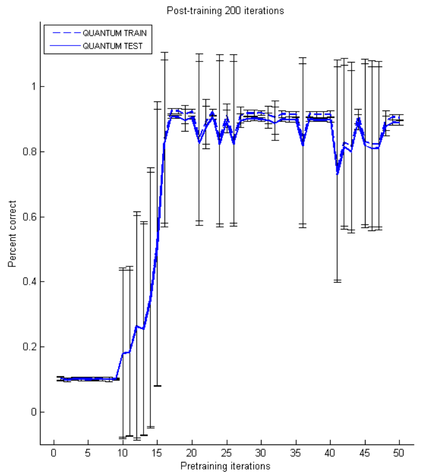
100 post-training iterations



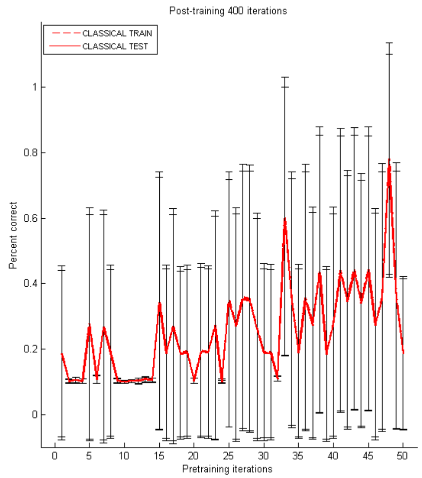


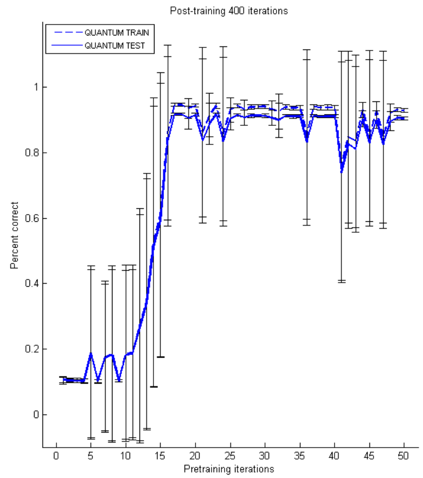
200 post-training iterations



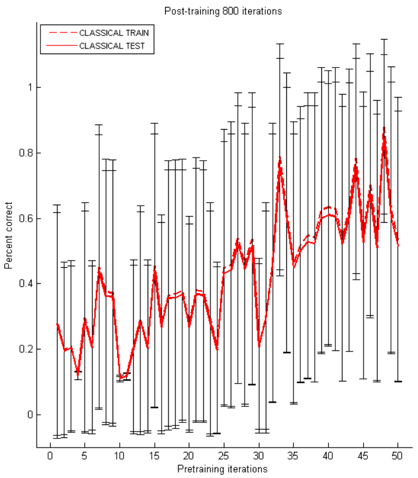


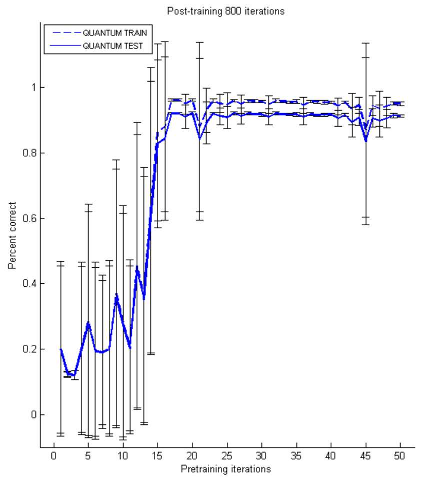
400 post-training iterations





800 post-training iterations





Scaling up to larger RBMs

Example: Paths to 1024x1024 RBM

Grid size (NxN unit cells)	Unit Cell Size: 8	16	32	64	128	256	512	1024	2048	
1							512	1024	2048	*# qubits*
							256	512	1024	*RBM size (NxN)*
2					512	1024	2048	4096		
					128	256	512	1024		
4			512	1024	2048	4096	8192			
			64	128	256	512	1024			
8	512	1024	2048	4096	8192	16384				
	32	64	128	256	512	1024				
16	2048	4096	8192	16384	32768					
	64	128	256	512	1024					
32	8192	16384	32768	65536						
	128	256	512	1024						
64	32768	65536	131072							
	256	512	1024							
128	131072	262144								
	512	1024								
256	524288									
	1024									

Bigger unit cells
Need design improvements

High qubit overhead
Very long chains
Need ICE improvements

NOTE: This is assuming full bipartite RBM graphs. Sparser RBMs may be acceptable and would have better scaling.

Summary

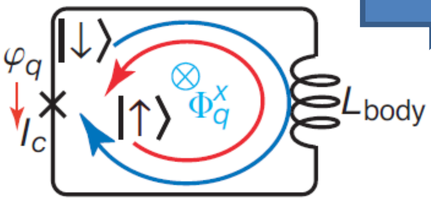
- **Progress in understanding problem types that are good candidates for near-term applications of Quantum Annealing**
- **Not yet able to demonstrate “quantum supremacy” for these applications, but …**
- **We are optimistic that this could occur in the foreseeable future, for specific problem types**
- **Further progress needed in Quantum Annealing hardware**
  - Increased qubit connectivity and control precision, and reduced processor noise, would be especially helpful for the applications described above

BACKUP SLIDES

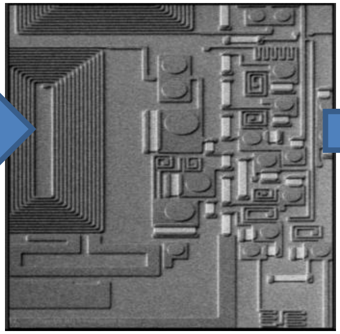
D-Wave hardware overview

Qubit implementation

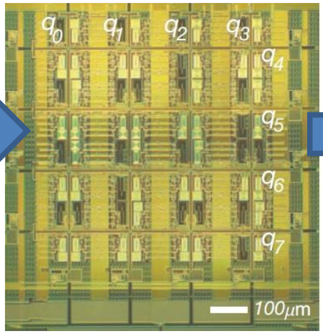
- rf SQUID Flux Qubit
- Compound-Compound Josephson Junction



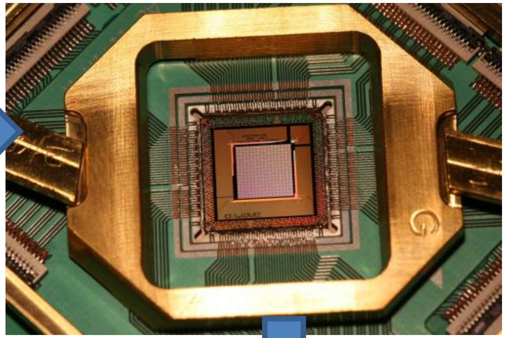
Niobium on silicon



8-qubit unit cell

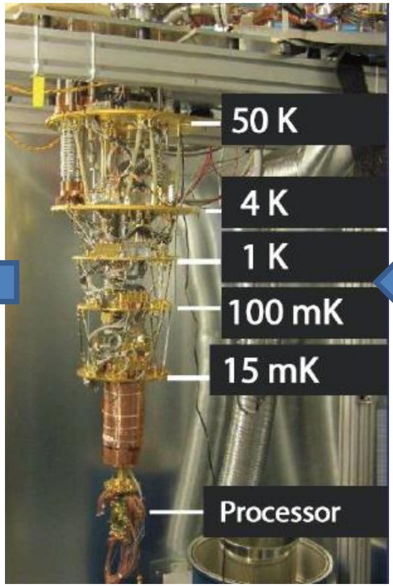


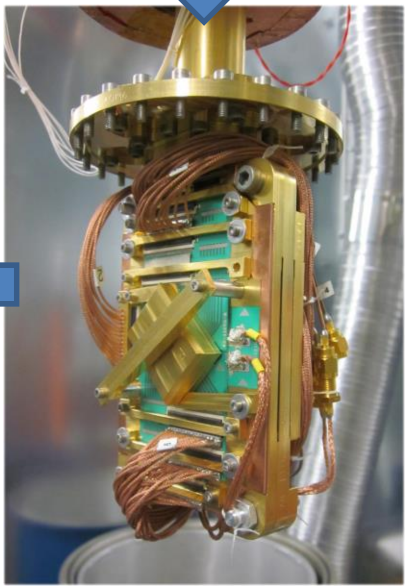
1152-qubit "Washington" chip



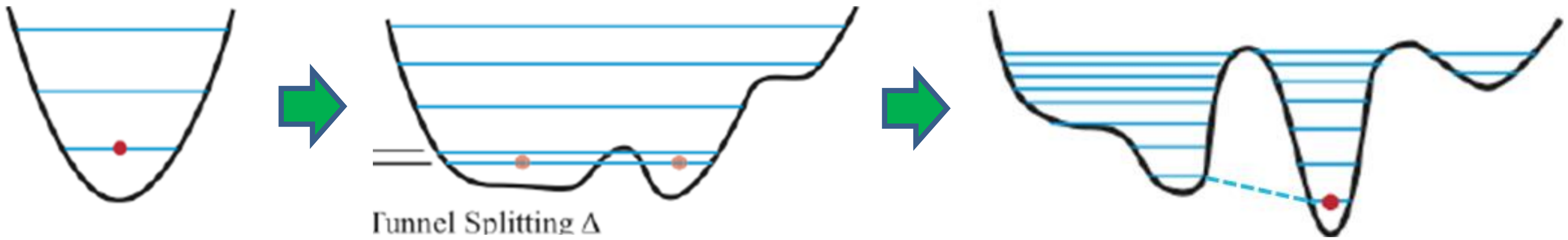
Magnetically shielded enclosure (10^{-9} Tesla)

Pulse tube dilution refrigerator





Quantum Adiabatic Algorithm & Quantum Annealing



- Suppose we want to find the ground state of Hamiltonian $\mathcal{H}_f$
- Start with a Hamiltonian $\mathcal{H}_i$ with a known & easily prepared ground state
- "Slowly" evolve from $\mathcal{H}_i$ to $\mathcal{H}_f$

$$\mathcal{H}(t) = \left(1 - \frac{t}{T}\right)\mathcal{H}_i + \frac{t}{T}\mathcal{H}_f \qquad 0 \leq t \leq T$$

- Adiabatic theorem: we will end up in the ground state of $\mathcal{H}_f$ if:

$$T > \frac{C\,\|\dot{\mathcal{H}}\|}{\Delta^2} \qquad \Delta = \min_t\left(E_1(t) - E_0(t)\right)$$

- Caveats:
    - Ideal adiabatic conditions (closed system)
        - Non-ideal (open system) → "***Quantum Annealing***" (heuristic)
    - In general we don't know the minimum spectral gap Δ

Is it really “Quantum”? Is there evidence for entanglement?

- Evidence of a quantum signature for 108 qubits:
  S. Boixo, T.F. Rønnow, S.V. Isakov, Z. Wang, D. Wecker, D.A. Lidar, J.M. Martinis, M. Troyer. (2013) Quantum annealing with more than one hundred qubits, http://arXiv.org/abs/1304.4595

- Entanglement “witnesses” & qubit tunneling spectroscopy:
  T. Lanting,et al. (2014) Entanglement in a Quantum Annealing Processor
  https://journals.aps.org/prx/abstract/10.1103/PhysRevX.4.021041

- Evidence for coherent quantum tunneling:
  D. Venturelli, S. Mandrà, S. Knysh, B. O'Gorman, R. Biswas, V. Smelyanskiy. (2014) Quantum Optimization of Fully-Connected Spin Glasses. http://arxiv.org/abs/1406.7553

How can the D-Wave machine work when $T_2 \ll T$?

(single-qubit coherence time) *(annealing time)*

- Experimental measurements of ground state populations
  Dickson, N. G. et al. (2013) Thermally assisted quantum annealing of a 16-qubit problem. Nat. Commun. 4:1903 doi: 10.1038/ncomms2920
  http://www.nature.com/ncomms/journal/v4/n5/full/ncomms2920.html

- Quantum Monte Carlo simulations
  T. Albash, D. Lidar. (2015) How detrimental is decoherence in adiabatic quantum computation?
  http://arxiv.org/abs/1503.08767

Is It Faster than a Classical Computer?

- Jury is still out …

T.F. Rønnow, Z. Wang, J. Job, S. Boixo, S.V. Isakov, D. Wecker, J.M. Martinis, D.A. Lidar, M. Troyer (2014) Defining and detecting quantum speedup. http://arxiv.org/abs/1401.2910

> Most benchmarks to date have been on random Ising problems
> Best classical algorithms: Optimized simulated annealing (Troyer, ETH Zurich), "Selby" code

O. Parekh, J. Wendt, L. Shulenburger, A. Landahl, J. Moussa, J. Aidun (2015) Benchmarking Adiabatic Quantum Optimization for Complex Network Analysis, Sandia Report SAND2015-3025.

> Speedup observed w D-Wave on affinity independent set and "planted" solution benchmarks, but not on other cases.
> Choice of solution criteria (time to optimality vs. near-optimality) can affect benchmark results.

J. King, S. Yarkoni, M.M. Nevisi, J.P. Hilton, C.C. McGeoch (2015) Benchmarking a quantum annealing processor with the time-to-target metric. http://arxiv.org/abs/1508.05087

> Studied "time-to-target" metric for D-Wave 2X vs. HFS, SA for random Ising & frustrated loop benchmarks.

V.S. Denchev, S. Boixo, S.V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis, H. Neven. (2015) What is the Computational Value of Finite Range Tunneling? http://arXiv.org/abs/1512.02206

> For a specific benchmark, found ~10^8 speedup in time to 99% success prob. vs SA on single core.

Graph Minor Embedding

Simple Example - "Subset Sum" Problem

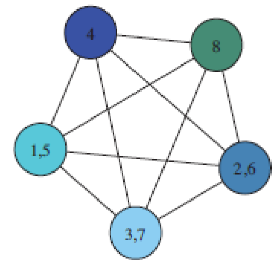
Problem: Find a subset of the numbers $\{2, 3, 5, 7, 11\}$ whose sum is 8.

QUBO Problem
(5 variables)

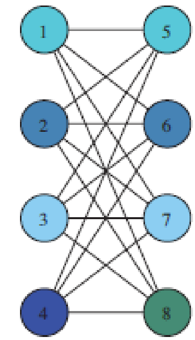
Minimize

$$\boldsymbol{Q}(\boldsymbol{x}) = \left[2x_1 + 3x_2 + 5x_3 + 7x_4 + 11x_5 - 8\right]^2$$

QUBO Graph
(5 logical qubits)

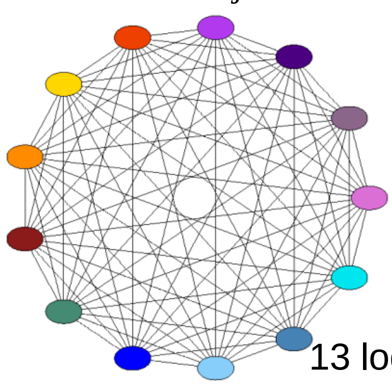


Embedding into D-Wave Unit Cell
(8 physical qubits)



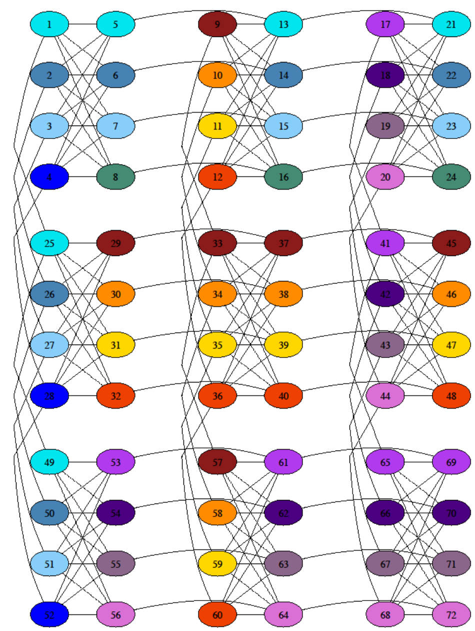
Worst Case - Complete Graph

$$x^* = \arg\min_{x} \sum_{ij \in E} Q_{ij} x_i x_j$$



13 logical qubits

embedding



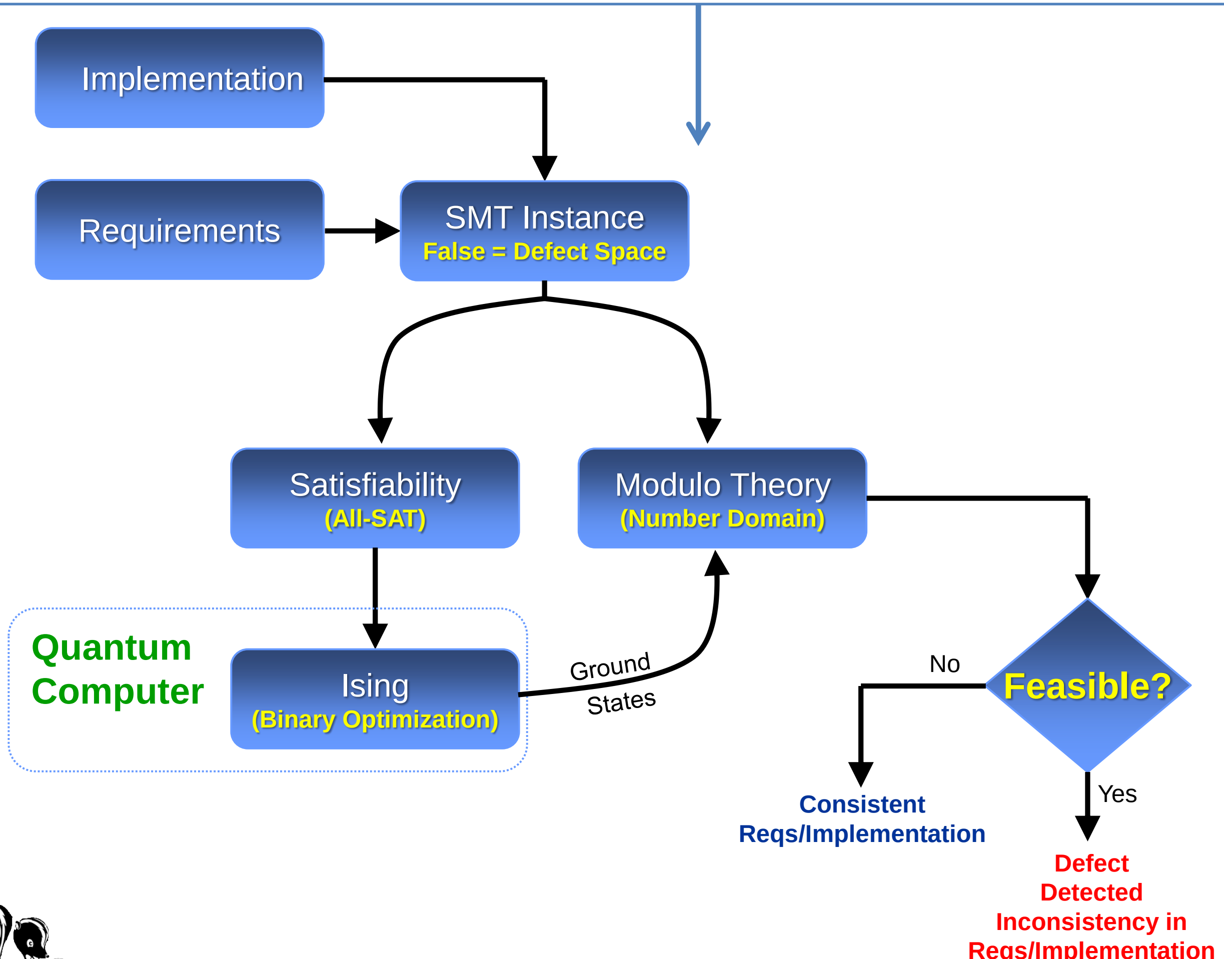
72 physical qubits

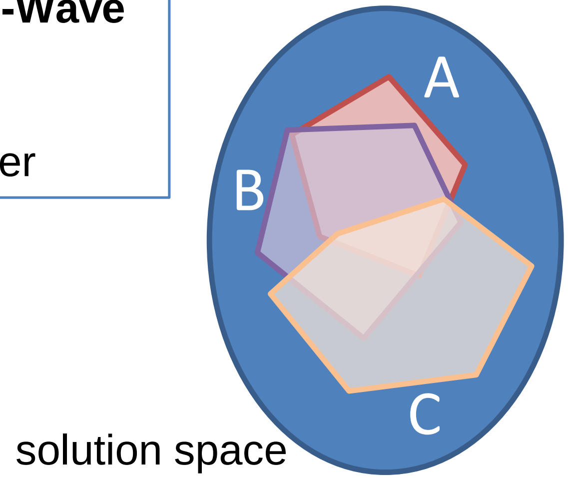
Ref: C. Klymko, B.D. Sullivan, T.S. Humble, Adiabatic Quantum Programming: Minor Embedding With Hard Faults, http://arXiv.org/abs/1210.8395.

Software Verification & Validation (V&V)

Software verification and validation can be posed with the D-Wave chip in the loop with a formal methods approach

- Canadian start-up QRA is designing the front end
- Lockheed Martin engineers integrate with the quantum annealer





V&V can be attacked with quantum optimization and machine learning