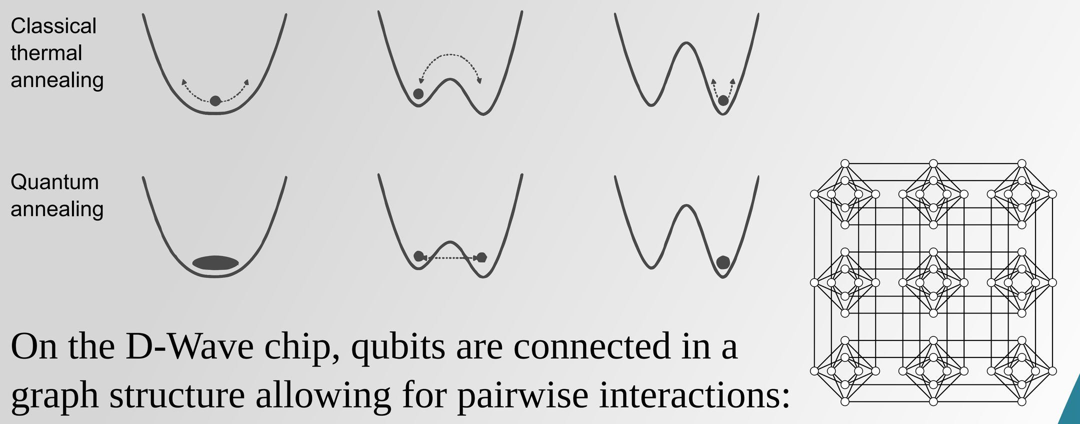
Finding Maximum Cliques on a Quantum Annealer

Background on D-Wave

The D-Wave quantum annealer is a hardware realization of classical (thermal) simulated annealing, a wide-spread optimization technique which minimizes a function by proposing random moves to escape local minima:



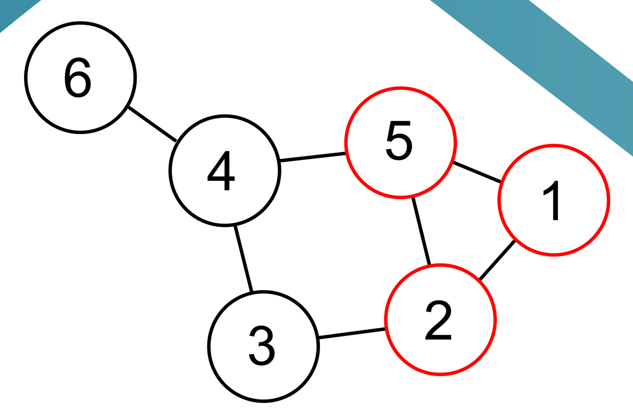
On the D-Wave chip, qubits are connected in a graph structure allowing for pairwise interactions:

D-Wave minimizes a sum of linear and quadratic contributions weighted by given constants a_i, $a_{ij} \in \mathbb{R}$, called **Hamiltonian**:

$$f(q) = \sum_{i \in V} a_i q_i + \sum_{(i,j) \in E} a_{ij} q_i q_j$$

Ising: $q_i \in \{-1,+1\}$ Qubo: $q_i \in \{0,+1\}$

The Maximum Clique Problem



We consider *maximum clique (*MC), a classical NP-hard graph problem.

Applications: network analysis, bioinformatics, computational chemistry.

Let $G = (V,E)$ be an undirected graph. A clique is a subset $S \subseteq V$ forming a complete subgraph (any two vertices of S are connected by an edge in G). A maximal clique is a clique of maximal size.

Qubo for Maximum Clique

We use the equivalence of MC to maximum independent set problem:
For a graph H, S is independent set if any two vertices $v,w \in S$ are not connected in H.

An independent set of $H = (V, \bar{E})$ is a clique in $G = (V, E)$. Constrained minimization:

$$\text{maximize}_{x_i \in \{0,1\}} \sum_{i=1}^{N} x_i \qquad \text{subject to} \sum_{(i,j) \in \bar{E}} x_i x_j = 0$$

The equivalent formulation as unconstrained minimization (Qubo):

$$H = -A \sum_{i=1}^{N} x_i + B \sum_{(i,j) \in \bar{E}} x_i x_j$$

with A = 1 and B = 2 (Lucas, 2014).
Disadvantage: $O(N^2)$ quad. terms, limited D-Wave solubility.

D-Wave Solvers

D-Wave Inc. provide tools to submit Qubo/Ising problems to the annealer, to perform the annealing and to post-process the output:

- ► *Sapi*: "Solver API", highest level control, set annealing cycles or post-processing, load pre-computed embeddings for complete 45 vertex graphs
- ► *QBsolv*: heuristic for instances ≥ 1000 qubits, identifies signif. rows/columns of Hamiltonian, solves subproblem on D-Wave
- ► *QSage*: black-box solver for bitstrings of arbitrary size, tabu search enhanced with DW-generated samples

Experiments: Small graphs with no special structure

Graph	Max. clique size	Runtime [s]						
		Sapi	PPHa	QBsolv	fmc	pmc	SA	Gurobi
p= 0.3	5	0.15	0.15	0.05	$8 \cdot 10^{-6}$	$3 \cdot 10^{-5}$	0.15	102
p= 0.5	8	0.15	0.15	0.06	$3 \cdot 10^{-4}$	$5 \cdot 10^{-5}$	0.37	38
p= 0.7	13	0.15	0.15	0.04	0.002	8.10^{-5}	0.19	33
p= 0.9	20	0.15	0.15	0.04	0.135	8.10^{-5}	0.28	2

Set-up: 45 vertex graphs, random edges with probability p

Results: Every software solver returns correct solution on small random graphs fitting D-Wave's architecture.

Gurobi solves the dual problem (maximum independent set) thus leading to reversed graph densities and timings.

Main observations: *pmc* is an order of magnitude faster than all other methods, D-Wave yields constant time solutions but no quantum speed-up detectable.

Classical Solvers

We benchmark against the following classical approaches:

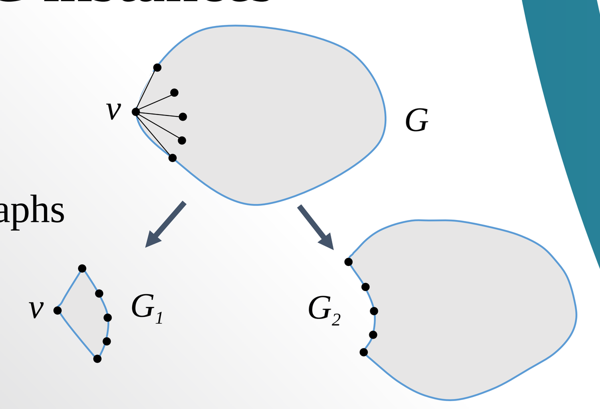
- ► *SA-Ising*: All-purpose simulated annealing.
- ► *SA-clique*: Simulated annealing specifically for cliques of size m (Geng et al., 2007).
- ► *Fast Max-Clique Finder (fmc, pmc)*: Exact and heuristic efficient search algorithms for max. cliques in sparse graphs.
- ► *Post-processing heuristics alone (PPHa)*: D-Wave's server-side post-processing step applied to random initial solution.
- ► *Gurobi*: Mathematical programming solver for linear, mixed-integer and quadratic programs (Gurobi Optimization Inc., 2015). Applied to the dual of maximum clique (maximum independent set) on the complement graph.

Georg Hahn, Hristo Djidjev (PI), Guillaume Chapuis, Guillaume Rizk

Computer, Computational, and Statistical Sciences Division
Los Alamos National Laboratory

ISTI NSEC: Efficient combinatorial optimization using quantum computing

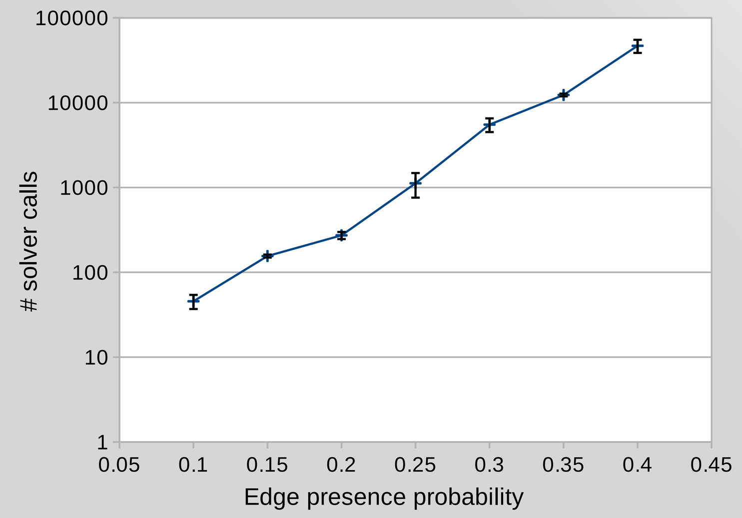
Solving large MC instances

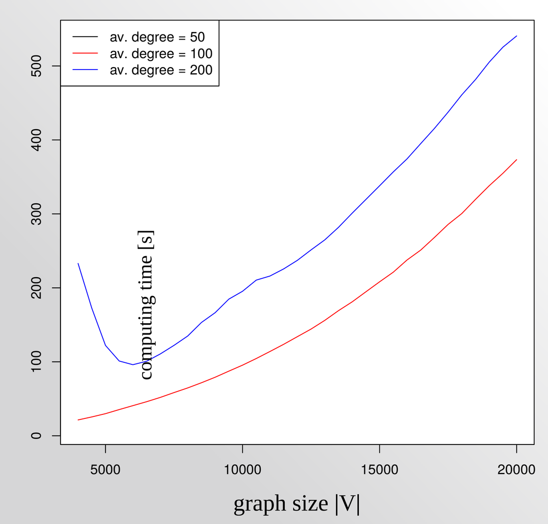


Idea: Remove edges not belonging to a maximum clique and split into subgraphs of at most 45 vertices.

Algorithm: Start with list $L=\{G\}$ and iterate until subgraphs fit DW:

- ► *Extract k-core*: maximal subgraph whose vertices have degree ≥ k; any clique C of size $k+1$ of G is also a clique of the k-core
- ► *Graph partitioning*: divide G into cores C_i and distance one neighbors H_i (halo); maximal clique is equal to the max clique in one of the partitions
- ► *Vertex splitting*: like partitioning but with $C_1=\{v\}$ for $v \in V$ and $C_2 = V \setminus \{v\}$





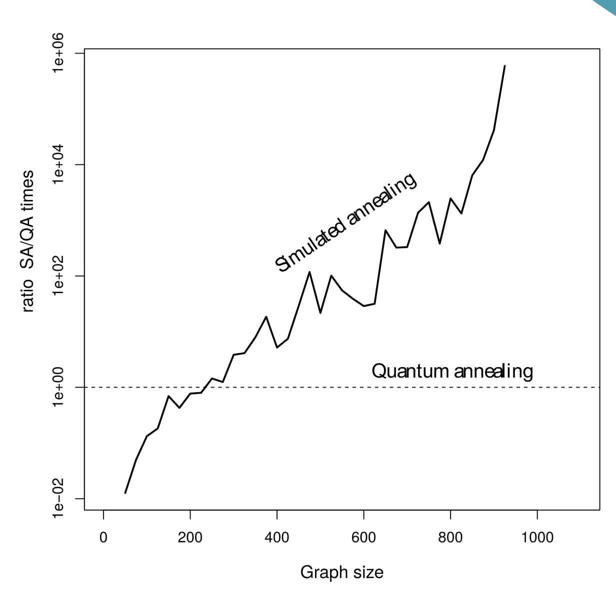
Set-up (left): 500 vertices, edge probability p from 0.1 to 0.4 in steps of 0.05

Set-up (right): graph sizes 3000 to 20000, fixed degree d, edge probability $p = d/(|V|-1)$

Experiments: D-Wave vs SA-clique

Why SA-clique? SA-clique is considered the classical analogue of quantum annealing and thus the closest competitor to D-Wave.

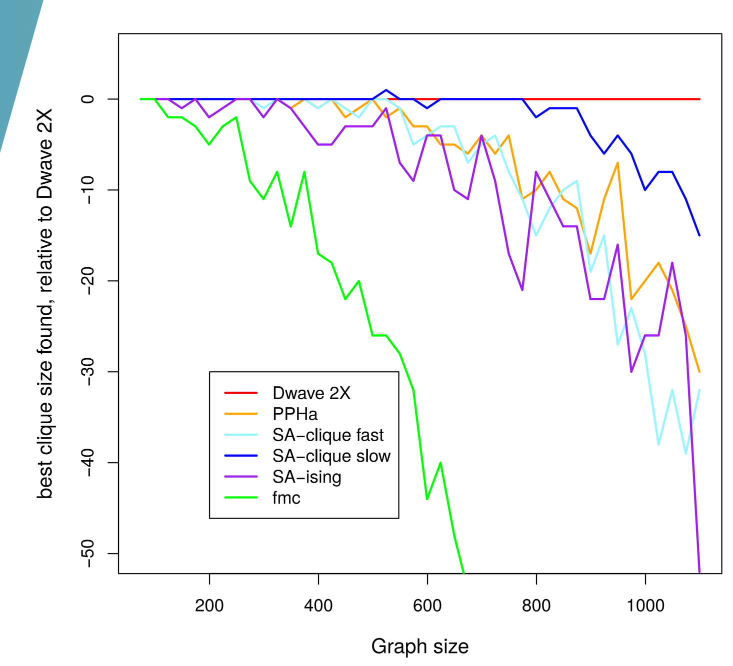
Set-up: 500 anneals on D-Wave on (contracted) chimera graphs, record best solution, then lower cooling schedule of SA-clique until same solution is found.



Conclusions:

- ► Random graphs: too small, optimized classical solvers faster, DW solutions of comparable quality
- ► **No quantum advantage for general instances embeddable on DW**
- ► Special instances designed to fit DW can be magnitudes faster (closer to DW chimera topology=faster)

Experiments: Chimera-like graphs



Motivation: So far no quantum advantage on small graphs. Need comparison on graphs larger than 45 vertices.

Set-up: Use chimera subgraphs generated by contracting edges which always fit the D-Wave topology.

Results: Up to size 400, PPHa finds same result as D-Wave. For size larger than 800, D-Wave is best.

Acknowledgments

The authors acknowledge and appreciate the support provided for this work by the Los Alamos National Laboratory Directed Research and Development Program (LDRD). They thank Dr Denny Dahl for his help while working on D-Wave 2X.

[1] H. Djidjev et.al. (2016). *SIAM CSC* 2016, 1(1): 1-17.
[2] X. Geng et.al. (2007). *Inf Sciences*, 177(22): 5064-5071.
[3] Gurobi Optimization, Inc. (2015). Gurobi optimizer refernce manual.
[4] M. Johnson et.al. (2011). *Nature*, 473: 194-198.
[5] J. King et.al. (2015). *arXiv:1508.05087*, pages 1-29.
[6] A. Lucas (2014). *Frontiers in Physics*, 2(5):1-27.

The poster design was adapted from the design of Felix Breuer: http://blog.felixbreuer.net/2010/10/24/poster.html