

# Where We Are and Where We Are Going With Quantum Annealing and Related Algorithms

Jülich Spring School 2023

Nicholas Chancellor

June, 24 2022



## Potentially relevant UK projects for this audience

Collaborative computational project on quantum computing (CCP-QC)

- ▶ Work with other CCPs (academic projects) to find uses for quantum computing within scientific research
- ▶ Idea is to use quantum computing to solve hard problems which come up in academic research rather than industry
- ▶ <https://ccp-qc.ac.uk/>

Quantum Enhanced and Verified Exascale Computing (QEVEC)

- ▶ Work on how quantum coprocessors can (eventually) support exascale computing
- ▶ Multiple projects looking at a variety of applications
- ▶ <https://excalibur.ac.uk/projects/qevec/>

Contact Viv Kendon at [viv.kendon@strath.ac.uk](mailto:viv.kendon@strath.ac.uk) if you are interested in potential collaborations

# Please Interrupt With Questions!

- ▶ I am generally very “anti-slides” when it comes to teaching...
  - ▶ but in this case there are too many plots to show to do it any other way
- ▶ The main reason: I think it makes it more intimidating to ask questions and go “off script”
- ▶ If you have a question your classmates are probably wondering the same thing
- ▶ Fine if we don't get through all the slides or have to skip details



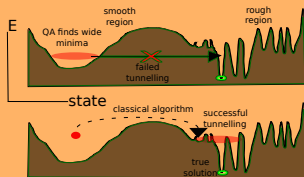
# Structure of This Lecture

- ▶ Setting the scene
  - ▶ Big picture: quantum computing and hybrid algorithms
  - ▶ (Why) should I care about quantum annealing?
- ▶ Adiabatic quantum computing and beyond
  - ▶ Strengths of this concept
  - ▶ How and why we need to go beyond the adiabatic theorem
- ▶ Hybrid (quantum/classical) computing in quantum annealing
  - ▶ Different variants of reverse annealing
  - ▶ What this gives us algorithmically
- ▶ Encoding and interaction graphs
  - ▶ Example of importance: domain-wall encoding

# The importance of hybrid algorithms\*

In the near term:

- ▶ Quantum computers may be very powerful in some ways but...
- ▶ Will remain very limited in others



For them to be genuinely useful, we must take advantage of the computational power, while circumventing the limitations

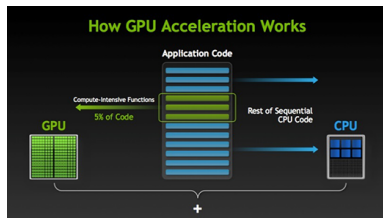
- ▶ This naturally indicates a coprocessor arrangement
- ▶ Fundamentally hybrid → computational model involves both classical and quantum steps
- ▶ This is different from just being supported by classical computation, see paper for full details

\*see: [Callison and Chancellor Phys. Rev. A 106, 010101 \(2022\)](#)

# This isn't a new idea in computing\*

Classical computing already makes heavy use of coprocessors:

- ▶ Graphics cards → good for highly parallel processing
- ▶ Application specific integrated circuits
- ▶ Neuromorphic devices → structures similar to natural neural networks



Wikimedia commons, created by user Mmanss, CC attribution share-alike

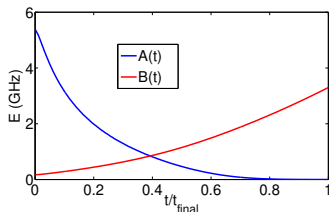
- ▶ No reason to think the same logic won't apply for quantum
- ▶ Needs fast interconnects and collocation with classical (HPC) resources (Jülich is a leader on this front)

\*see: [Callison and Chancellor Phys. Rev. A 106, 010101 \(2022\)](#)

# Adiabatic quantum computing

Traditional picture:

- ▶ Map an NP-hard optimization problem to a Hamiltonian, unknown ground state is solution
- ▶ *Slowly* change from a (driver) Hamiltonian with an easily prepared ground state to problem Hamiltonian
- ▶ Adiabatic theorem of quantum mechanics  $\rightarrow$  success probability arbitrarily close to 100 % by running long enough



$$H(t) = A(t)H_{\text{driver}} + B(t)H_{\text{problem}}$$

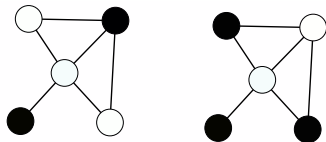
## Example of Ising problem mapping \*

Have:

- ▶ Binary variables  $Z_i \in \{-1, 1\}$
- ▶ Minimisation over Hamiltonian made of single and pairwise terms  $H_{\text{Ising}} = \sum_i h_i Z_i + \sum_{j>i} J_{i,j} Z_i Z_j$

Want:

- ▶ Maximum independent set: how many vertexes on a graph can we colour so none touch?  $\rightarrow$  NP hard



Method:

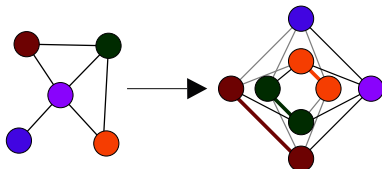
1. For an edge between vertex  $i$  and  $j$  add  $Z_i + Z_j + Z_i Z_j \rightarrow$  penalizes colouring ( $Z = 1$ ) adjacent vertexes
2. Add  $-\lambda Z_i$  to reward coloured vertexes ( $0 < \lambda < 1$ )

\*Taken from the notes of a physics level 3 computing project I wrote, full notes at: [http://nicholas-chancellor.me/QOpt\\_project\\_final.pdf](http://nicholas-chancellor.me/QOpt_project_final.pdf)



## Minor embedding

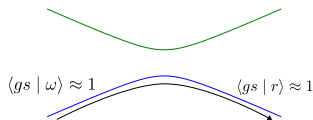
- ▶ Strong 'ferromagnetic' ( $-Z_i Z_j$ ) coupling energetically penalizes variables disagreeing
- ▶ If strong enough than entire 'chain' acts as a single variable
- ▶ Mathematically corresponds to mapping one graph to graph minors of another



Can embed arbitrary graphs into quasi-planar hardware graph with polynomial ( $n^2$  for fully connected) overhead  $\rightarrow$  Ising model **restricted to hardware graph** is also NP-hard

In practice this leads to a large overhead  $\rightarrow$  important consideration for solving real problems

# Advantages and disadvantages of the adiabatic picture



## Theoretically satisfying

- Algorithm is effectively deterministic  $\rightarrow$  “always” succeeds
- Intuitive picture involving only ground and first excited state

## Let's assume $P \neq NP$ \*

- Algorithm succeeds roughly 100% of the time
- Total runtime needs to be exponential in size of problem  $\rightarrow$  system needs to remain coherent for exponentially long time\*

\*For those unfamiliar with complexity theory, this is basically saying “let's assume that hard optimization problems exist”, most experts believe  $P \neq NP$

\*for other approaches see: [Crosson and Lidar, Nature Reviews Physics volume 3, pages 466-489 \(2021\)](#)

# What can be done?

Restore coherence somehow

- Error correction, difficult to do in continuous time, but progress being made
- Low temperature dissipation can restore coherence → would have to be very low temperature

- Have to mitigate errors for a very long time
- Not the subject of this presentation



Succeed with low probability

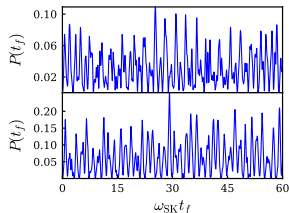
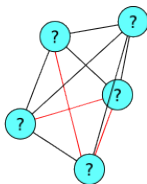
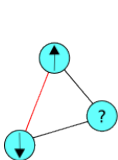
- *Total* runtime is still exponential in problem size
- Each run is short → exponentially many needed to hit right answer

- Exponentially low success each run is *conceptually* unsatisfying...
- ... but much less demanding for coherence
- back to this later, simpler setting first

Lottery

# Start simple: continuous time quantum walk on spin glass

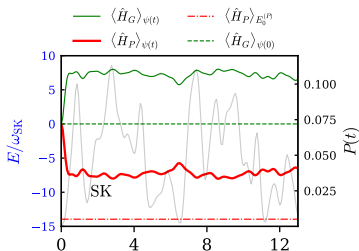
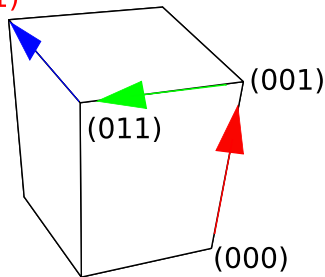
- ▶ Start with an equal positive superposition of all solutions,  $|\omega\rangle = \frac{1}{\sqrt{N}} \sum_i |i\rangle$
- ▶ Evolve with a fixed Hamiltonian  $H_{\text{walk}} = \gamma H_{\text{hop}} + H_{\text{problem}}$
- ▶  $H_{\text{hop}} = -\sum_i \sigma_i^x \rightarrow$  superposition is ground state
- ▶  $H_{\text{problem}} = \sum_{i,j} J_{i,j} \sigma_i^z \sigma_j^z + \sum_i h_i \sigma_i^z$  where  $h_i$  and  $J_{i,j}$  drawn from the same Gaussian distribution
- ▶ Measure after random short period of time, repeat many times



See [Adam Callison et al 2019 New J. Phys. 21 123022](#) for details, work with Adam Callison, Viv Kendon, and Florian Mintert

# How is this a 'walk'? How does it find solutions?

(111)



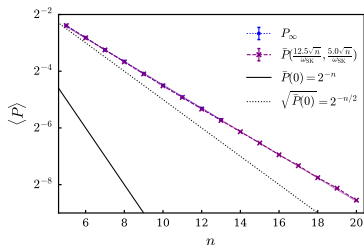
- ▶  $H_{\text{hop}}$  effectively forms a hypercube with a bitstring at each vertex, probability amplitude 'walks' between different states
- ▶  $H_{\text{problem}}$  contributes phases which guide the walk

Energy is conserved  $\langle H_{\text{walk}} \rangle_{t=0} = \langle H_{\text{walk}} \rangle_{t>0}$  since the system starts in the ground state of  $H_{\text{hop}}$ :

$$\langle H_{\text{problem}} \rangle_{t>0} - \langle H_{\text{problem}} \rangle_{t=0} = \langle H_{\text{hop}} \rangle_{t=0} - \langle H_{\text{hop}} \rangle_{t>0} \leq 0$$

Walk seeks out 'good' solutions!

## How well this works, numerically extracting scaling



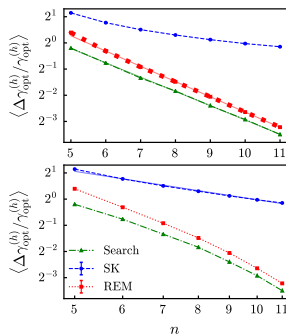
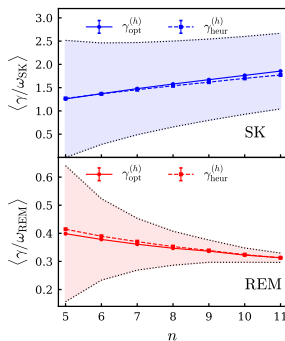
- ▶  $N = 2^n$  possible bitstrings, one correct solution, runtime scales as inverse probability
- ▶ Scaling of  $\frac{1}{N^{0.417}}$  better than both classical guessing ( $\frac{1}{N}$ ) and  $\frac{1}{\sqrt{N}}$  unstructured (Grover like) quantum search

The structure of the problem (correlations in bitstring energies) is playing a role in the computational mechanism, otherwise could not beat  $\frac{1}{\sqrt{N}}$  scaling

# Compare to problems without correlations

Random Energy Model (REM): each bitstring is assigned a random independent energy

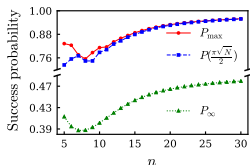
- ▶ Dynamics become dominated by a single close avoided crossing, require fine tuned  $\gamma$ , technically difficult, may not be possible to find correct value
- ▶ Requires single long run for high success probability  $\rightarrow$  need long coherence time



## Compare to unstructured search

- ▶ Continuous time analog to Grover search
- ▶ Problem Hamiltonian is a single marked bitstring  $|m\rangle\langle m|$ :

$$H(s) = -(1 - s) \sum_i \sigma_i^x + s(1 - |m\rangle\langle m|)$$



- ▶ Same optimal speedup as gate model finds solution in time  $\sqrt{N}$  rather than  $N$  from classical guessing/exhaustive search
- ▶ Exponentially sensitive to parameter setting (values of  $s$ )
- ▶ Succeeds with  $O(1)$  probability after  $\sqrt{N}$  runtime

For a detailed study of AQC and QW, see [Morley et. al. Phys. Rev. A 99, 022339 \(2019\)](#), for practical implementation, see [Dodds et. al. Phys. Rev. A 100, 032320 \(2019\)](#)\*

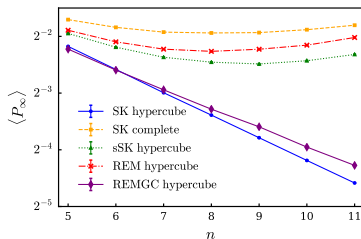


## (Why) is the effect of correlations interesting?

If the bitstring energies are uncorrelated no *classical* algorithm could do better than random guessing, why?

Energy of one bitstring tells nothing about energy of neighbours

Complementary to the search-like mechanism usually attributed to quantum algorithms



Not just a 'one off' difference between spin glasses and REM, but more general between correlated and uncorrelated energies

## Rapid quenches?

Energy conservation argument extended to any monotonic (closed system) quench

$$H(t) = A(t) H_{\text{drive}} + B(t) H_{\text{problem}} \quad \frac{A(t)}{B(t)} \geq \frac{A(t + \delta t)}{B(t + \delta t)} \forall t$$

Sketch of proof:

1. Trotterize time evolution:  $A(t) \rightarrow A(t + \delta t)$  and  $B(t) \rightarrow B(t + \delta t)$  and apply  $|\psi(t + \delta t)\rangle = \exp(-iH(t)\delta t)|\psi(t)\rangle$  in separate steps
2. Rescale time so that Hamiltonian always resembles (energy conserving) quantum walk  $H_{\text{eff}}(\Gamma(t)) = \Gamma(t) H_{\text{drive}} + H_{\text{problem}}$
3. In rescaled version  $\Gamma(t) \geq \Gamma(t + \delta t)$  (lowest  $\langle H_{\text{drive}} \rangle$  is  $-n$ )  $\therefore \langle H_{\text{eff}}(\Gamma(t)) \rangle_{\psi(t)} - \Gamma(t) n \geq \langle H_{\text{eff}}(\Gamma(t + \delta t)) \rangle_{\psi(t)} - \Gamma(t + \delta t) n$
4. Because  $\langle H_{\text{eff}}(\Gamma(t)) \rangle_{\psi(t)} \geq -\Gamma(t) n \forall t$ ,  $\langle H_{\text{problem}} \rangle_{\psi(t)} \leq 0 \forall t$

Details can be found in [Callison et. al. PRX Quantum 2, 010338](#)

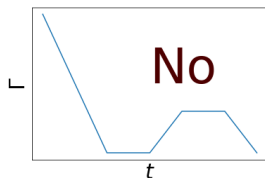
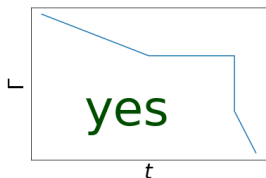
# A very general result!

For result to hold (to be better than random guessing on average):

1. Monotonic  $\Gamma(t) \geq \Gamma(t + \delta t)$  where  $\Gamma(t) = \frac{A(t)}{B(t)}$
2. Start in ground state of  $H_{\text{drive}}$
3. Driver not gapless  $\rightarrow$  not a concern for real problems

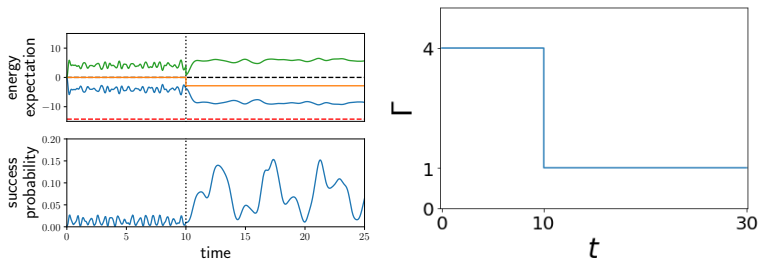
What is allowed:

1. No limit on how fast algorithm runs
2. Discontinuities in  $\Gamma(t)$  are ok
3.  $H_{\text{drive}}$  does not need to be diagonal in an orthogonal basis to  $H_{\text{problem}}$   $\rightarrow$  starting state can be biased



# Intuitive example: two stage quantum walk

Perform a quantum walk at  $\gamma_1$ , and then use result as an input state for a second walk at  $\gamma_2 < \gamma_1$



- ▶ Energy expectations: Green =  $\gamma_{1,2} \langle H_{\text{drive}} \rangle$ ; Blue =  $\langle H_{\text{problem}} \rangle$ ; Gold =  $\gamma_{1,2} \langle H_d \rangle + \langle H_{\text{problem}} \rangle$
- ▶ Total energy conserved except for at dashed line where  $\gamma$  decreases
- ▶ Non-instantaneous quench effectively infinite stage quantum walk

# Why is the rapid quench result important?

General, but rather weak:

Any monotonic quench at least as good as measuring the initial state

1. Design protocols to maximize dynamics → don't need to worry about dynamics being counter-productive
2. A **biased** search can already start from a very good guess  
[more discussion on this later](#)
3. Mechanism to understand dynamics very far from adiabatic limit

## Connection to gate model algorithms: Quantum Alternating Operator Ansatz (QAOA)

- ▶ Apply\*  $H_{\text{driver}}$  and  $H_{\text{problem}}$  sequentially rather than simultaneously
  - ▶ Can simulate quantum annealing in the limit of many repetitions
  - ▶ Machine learning usually used to optimise controls
- 
- ▶ Recent work by others\* shows optimal QAOA looks very similar (but not identical) to simulated quantum annealing
  - ▶ Is the (approximate) energy conservation mechanism the reason for this behaviour?

---

\*QAOA literature calls these mixer and phase separator, but I will use the quantum annealing terminology to avoid confusion

\*Brady et. al. [arXiv:2107.01218](https://arxiv.org/abs/2107.01218), *Phys. Rev. Lett.* **126**, 070505 (2021)

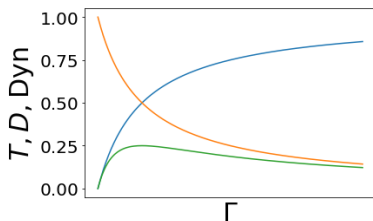
## Quantifying dynamics in a two state subspace

Transfer coefficient, transfer between computational basis states:  
 $T^{(jk)} = \frac{2\Gamma(t)|\langle k|H_{\text{drive}}|j\rangle|}{2\Gamma(t)|\langle k|H_{\text{drive}}|j\rangle|+|\Delta_{jk}|}$  (where  $\Delta_{jk}$  is the total difference in diagonal matrix elements) **Blue** in figure

Disequilibrium coefficient, amount which  $H_{\text{problem}}$  breaks the initial equilibrium:

$D^{(jk)}$ , defined the same as  $T^{(jk)}$ , but in the diagonal basis of  $H_{\text{drive}}$  rather than the computational basis **Gold** in figure

$\chi^{(jk)} = T^{(jk)} D^{(jk)}$  quantifies total dynamics **Green** in figure

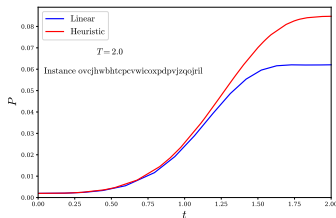
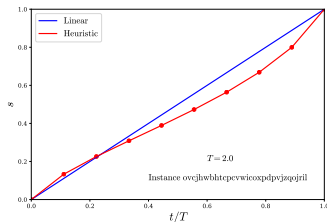


Unlike spectral gap, these quantities can be efficiently calculated for large problems!

# Finding optimal annealing schedules

$$H(t) = A(t) H_{\text{drive}} + B(t) H_{\text{problem}}$$

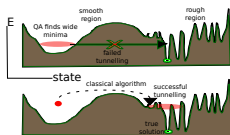
- ▶ Define  $A(t) = (1 - s(t))$  and  $B(t) = s(t)$
- ▶ Set  $\frac{\partial s}{\partial t} \propto \frac{1}{\chi}$
- ▶ Compare to linear schedule  $s \propto t$  for single SK instance



Heuristic performs better than linear schedule!



# Hybrid protocols using this mechanism?



Known techniques:

Dissipative reverse annealing [NC 2017 New J. Phys. 19 023024](#) as implemented on D-Wave devices

Relies on dissipation, not suitable for coherent algorithms

Coherent reverse annealing [Perdomo-Ortiz et. al. Quantum Inf Process \(2011\) 10: 33. doi:10.1007/s11128-010-0168-z](#)

Involves three separate Hamiltonians, not compatible with rapid sweep proof in [Callison et. al. PRX Quantum 2, 010338](#)

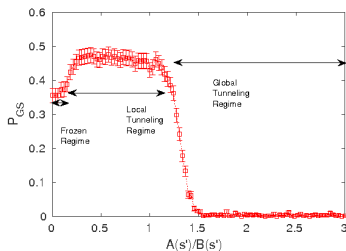
Biased driver Hamiltonian [Chinese Physics Letters, 30 1 010302](#) and [Tobias Graß Phys. Rev. Lett. 123, 120501 \(2019\)](#)

Compatible with proof in [Callison et. al. PRX Quantum 2, 010338](#), can apply the mechanisms discussed here

# A subroutine for hybrid quantum/classical optimization

Basic requirement: needs to be able to incorporate outside information to solve problem

- ▶ One way to do this → search preferentially around candidate solution



How to do this experimentally: (dissipative) Reverse annealing

- ▶ Seed in guess solution on D-Wave quantum annealer
- ▶ Quantum fluctuations plus dissipation search locally
- ▶ See [New J. Phys. 19 023024 \(2017\)](#)

# Obligatory slide: D-Wave controversy

Two separate controversies:

1) Are the dynamics actually quantum? **Yes!**

- ▶ Lots of evidence, most striking is simulation of extremely quantum KT phase transition **Nature 560 456–460 (2018)**
- ▶ Classical models reproduce **some** behaviours, expected → mean field approximation

2) Can it **beat improve** classical computing? **Open question**

- ▶ No conclusive speedup\* demonstrated yet
- ▶ Not what this talk is about

- ▶ Currently largest scale device to study algorithmic application of quantum mechanics
- ▶ **Good science can be done regardless of answer to question 2!**

\*For optimisation, there is some evidence of a speedup for simulation, **King et. al. Nature Communications 12, 1113 (2021)**

## Reverse annealing in algorithms (incomplete list)\*

1. Start from one solution to find other solution ([D-Wave whitepaper 14-1018A-A](#))
  - ▶ [Finding other solution 150x more likely than forward](#)
2. Search locally around classical solution ([arXiv:1810.08584](#))
  - ▶ Start from greedy search solution
  - ▶ [Speedup of 100x over forward annealing](#)
3. Iterative search ([arXiv:1808.08721](#))
  - ▶ Iteratively increase search range until new solution found
  - ▶ [Forward annealing could not solve any, reverse solved most](#)
  - ▶ See also: [arXiv:2007.05565](#)
4. Quantum simulation([Nature 560 456–460 \(2018\)](#))
  - ▶ Seed next call with result from previous
  - ▶ [Seeding with previous state makes simulation possible](#)
5. Monte Carlo and Genetic like algorithms
  - ▶ Quantum assisted genetic algorithm QAGA ([arXiv:1907.00707](#))
  - ▶ [Finds global optima quickly where other methods struggle](#)
  - ▶ Theoretical discussion (my work) ([NJP 19, 2, 023024 \(2017\)](#) and [arXiv:1609.05875](#))

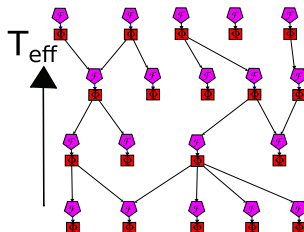
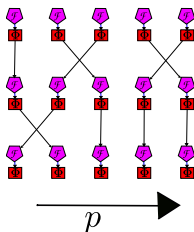
---

\*forward annealing= traditional non-hybrid method 

# Hybrid quantum/classical, what's next?

## 1. More sophisticated algorithms

- ▶ Except for QAGA, all experiments have been very simple algorithms
- ▶ Move to more complex ones based on current state of art (particularly the state of the art for specific problems)
- ▶ Develop theoretical framework: inference primitive → **NC Nat Comput (2022)**. <https://doi.org/10.1007/s11047-022-09905-2>



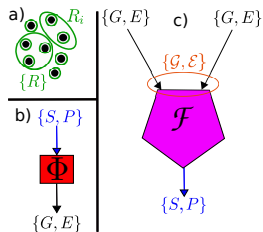
## 2. Understand and improve protocols

- ▶ Understand how these protocols actually work under realistic conditions

## More on hybrid...

Fundamental question: When/how to use a call to a physical device which is very powerful but also very constrained

Discussion so far has been under the context of quantum, but actually much more general → heterotic computing

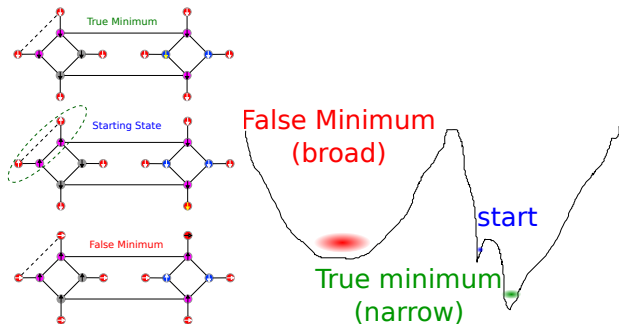


Many of the ideas from my work would equally apply to other powerful optimisation subroutines

Interesting future work in taking these ideas outside of quantum computing

## Solving problems in a dissipative setting

Ising model gives us rich control to design energy landscapes, compare behaviour of D-Wave quantum annealers with different noise levels

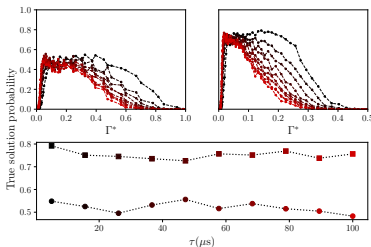


Core idea: starting state near a true minima, and further away from a false minima which would 'trick' forward annealing  
dashed coupling  $J_t$  controls barrier between start and true minimum

Details in [Chancellor and Kendon PRA 104, 012604 \(2021\)](#)

# Need more fluctuations on less noisy QPU

- ▶ Dissipation mediates reverse annealing local search, less coupling to bath  $\rightarrow$  energy dissipates more slowly
- ▶ For the same runtime more fluctuations (higher  $\Gamma$ ) is needed
- ▶ Suggests that spin bath polarisation is not the dominant effect here, otherwise noisier version would need higher  $\Gamma$



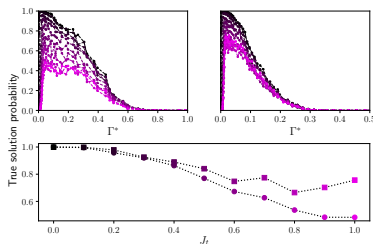
Increasing hold time  $\tau$  allows more tunnelling out of the true minimum at higher  $\Gamma$  (left: lower noise, right: higher) doesn't affect the height of the peak (important for modelling)



# More exciting difference: higher peak with noisier QPU

## Why is this exciting?

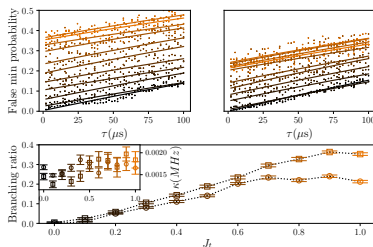
False minimum is further away, more tunnelling from start state means longer range search with lower noise



- ▶ Top left: lower noise true min probability, Right: higher, bottom, peak values, bottom, peak value versus  $J_t$
- ▶ As it gets harder to tunnel to the true minimum the difference between the higher and lower noise QPUs grows

# A simple model: initial branching then tunnelling

$$P_{\text{false}}(\tau) = 1 - [1 - R_{\text{false}}] \exp(-\kappa\tau)$$



- ▶ Branching ratio  $R_{\text{false}}$  indeed higher on lower noise QPU (left fit, squares on bottom)
- ▶ Don't have time for full details, see [Chancellor and Kendon PRA 104, 012604 \(2021\)](#)
- ▶ Confirms model, lowering noise causes more branching to further local minima  $\rightarrow$  longer range  $\tau$  search

## Effect of problem structure and encoding\*

Consider higher-than-binary discrete problems; appear often in real world optimisation, for example:

- ▶ A truck can go down any of three roads...
  - ▶ A tasks can be scheduled at any of five times...
  - ▶ A component can be placed any of seven places on a chip...
- ▶ Define two index objects:

$$x_{i,\alpha} = \begin{cases} 1 & \text{variable } i \text{ takes value } \alpha \\ 0 & \text{otherwise} \end{cases}$$

- ▶ Discrete Quadratic models, (DQM), made from pairwise interactions of  $x$  terms:

$$H_{\text{DQM}} = \sum_{i,j} \sum_{\alpha,\beta} D_{(i,j,\alpha,\beta)} x_{i,\alpha} x_{j,\beta}$$

---

\*Details in [arXiv:2108.12004](https://arxiv.org/abs/2108.12004)

# Discrete variables into binary, three ways

Variable size= $m$

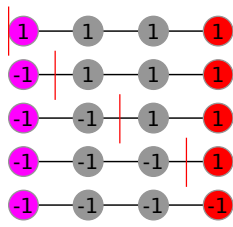
performance metric	binary	one-hot	domain wall*
# binary variables	$\lceil \log_2(m) \rceil$	$m$	$m - 1$
# couplers for encoding	0 if $m = 2^n, n \in \mathbb{Z}$ complicated otherwise	$m(m - 1)$	$m - 2$
intra-variable connectivity	N/A or complicated	complete	linear
maximum order needed for two variable interactions	$2 \lceil \log_2(m) \rceil$	2	2

Binary= assign bitstrings to configurations

One hot= constrain variables so exactly one can be 1

Domain wall= new encoding w/ better performance\*

encoded value	qubit configuration
0	1111
1	-1111
2	-1-111
3	-1-1-11
4	-1-1-1-1

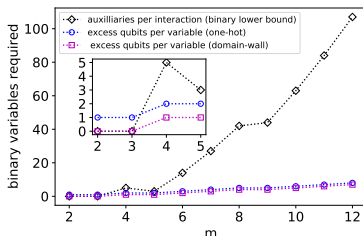


\*For details see: [Chancellor, Quantum Sci. Technol. 4 045004](#)

\*[Chen et. al. IEEE Transactions on Quantum Engineering 3102714 \(2021\)](#)

# Binary encoding

- ▶ A variable of size  $m$  can be encoded in  $\lceil \log_2(m) \rceil$  qubits
- ▶ Arbitrary interactions require high order terms in Hamiltonian
- ▶ Only quadratic interactions  $\rightarrow$  gadgets  $\rightarrow$  auxiliary variables
- ▶ Fair counting needs to include auxiliary variables as well



This is a losing proposition for general interactions\*

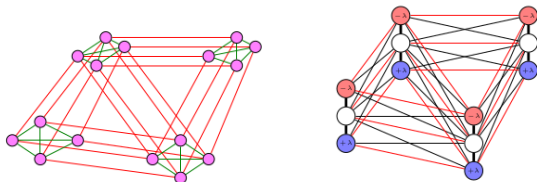
\*Binary may still be best for interactions with special structure, example, variable multiplication: (for example [Joseph et. al. Phys. Rev. A 103, 032433](#))

# Comparing one-hot and domain-wall: colouring problems\*

Simple test problem with structure: penalty between nodes if and only if they are the same colour

Use natural structure of problem to 'spread out' embedding

Four colouring example, 'layered' structure in Domain wall (right), no structure in one hot, (left)

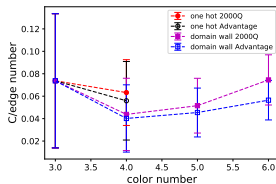
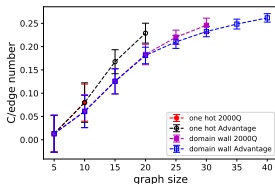


three-colouring  $\rightarrow$  randomly generated edges with 50% probability  
k-colouring  $\rightarrow$  twice as many nodes as colours, random edges with 75% probability

# The results\*

For both  $k$  and three colouring problems the domain-wall encoding performs better on both Advantage and 2000Q D-Wave QPUs

three colouring (left),  $k$ -colouring (right)



$C$ =number of places same colour touches

Even looks like domain-wall on 2000Q out-performs one-hot on Advantage!

Use hypothesis testing to verify that this is a statistically significant result, test 100 instances on each and see how much each processor/encoding combination wins for all 6 combinations

# Hypothesis testing, three colour\*

Green=statistically significant result (95% confidence)

	Adv. dw/oh		2000Q dw/oh		dw Adv./2000Q		oh Adv./2000Q		(dw, Adv.)/(oh, 2000Q)		(dw, 2000Q)/(oh, Adv.)	
5 node (b,w)	0	0	0	0	0	0	0	0	0	0	0	0
5 node p												
10 node (b,w)	42	0	37	0	2	0	19	21	39	0	40	0
10 node p	$2.27 \times 10^{-13}$		$7.28 \times 10^{-12}$		$2.50 \times 10^{-1}$		$6.82 \times 10^{-1}$		$1.82 \times 10^{-12}$		$9.09 \times 10^{-13}$	
15 node (b,w)	85	2	95	3	32	34	70	22	94	1	91	2
15 node p	$2.47 \times 10^{-23}$		$4.95 \times 10^{-25}$		$6.44 \times 10^{-1}$		$2.67 \times 10^{-7}$		$2.42 \times 10^{-27}$		$4.41 \times 10^{-25}$	
20 node (b,w)	99	0	100	0	43	41	94	3	100	0	93	2
20 node p	$1.58 \times 10^{-30}$		$7.89 \times 10^{-31}$		$4.57 \times 10^{-1}$		$9.60 \times 10^{-25}$		$7.89 \times 10^{-31}$		$1.15 \times 10^{-25}$	
25 node (b,w)	100	0		FAIL	66	20		FAIL		FAIL	98	2
25 node p	$7.89 \times 10^{-31}$				$3.33 \times 10^{-7}$						$3.98 \times 10^{-27}$	
30 node (b,w)	100	0		FAIL	72	20		FAIL		FAIL	97	2
30 node p	$7.89 \times 10^{-31}$				$2.30 \times 10^{-8}$						$7.81 \times 10^{-27}$	
35 node (b,w)	100	0	FAIL	FAIL		FAIL		FAIL		FAIL	FAIL	
35 node p	$7.89 \times 10^{-31}$											
40 node(b,w)	100	0	FAIL	FAIL		FAIL		FAIL		FAIL	FAIL	
40 node p	$7.89 \times 10^{-31}$											

- ▶ Domain-wall 2000Q beats one hot-Advantage (in a statistically significant way)
- ▶ Trend continue up to size where no longer possible to embed in 2000Q (FAIL)
- ▶ Otherwise results are expected → 2000Q worse than Advantage, one hot worse than domain wall



# Hypothesis testing, k colour\*

Green/red=statistically significant result (95% confidence)

	Adv. dw/oh		2000Q dw/oh		dw Adv./2000Q		oh Adv./2000Q		(dw, Adv.)/(oh, 2000Q)		(dw, 2000Q)/(oh, Adv.)	
3 color (b,w)	0	0	0	0	0	0	0	0	0	0	0	0
3 color p												
4 color (b,w)	34	1	37	2	11	3	26	16	44	1	33	7
4 color p	$1.05 \times 10^{-9}$		$1.42 \times 10^{-9}$		$2.87 \times 10^{-2}$		$8.21 \times 10^{-2}$		$1.31 \times 10^{-12}$		$2.11 \times 10^{-5}$	
5 color (b,w)	91	1	78	1	34	18	23	59	88	1	91	1
5 color p	$1.88 \times 10^{-26}$		$1.32 \times 10^{-22}$		$1.82 \times 10^{-2}$		$\approx 1$		$1.45 \times 10^{-25}$		$1.88 \times 10^{-26}$	
6 color (b,w)	99	0		FAIL	59	15		FAIL		FAIL	99	0
6 color p	$1.58 \times 10^{-30}$				$1.28 \times 10^{-7}$						$1.58 \times 10^{-30}$	
7 color (b,w)	92	0	FAIL	FAIL		FAIL		FAIL		FAIL	FAIL	
7 color p	$2.02 \times 10^{-28}$											

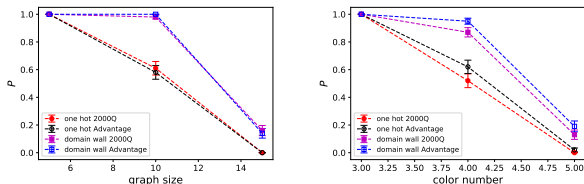
- ▶ Domain-wall 2000Q beats one-hot Advantage (in a statistically significant way)
- ▶ Trend continue up to size where no longer possible to embed in 2000Q (FAIL)
- ▶ One case where 2000Q beats advantage for the same decoding (one-hot)\*

\*This goes away when the decoding strategy for broken chains is changed so probably an artefact of majority vote decoding

\*Chen et. al. IEEE Transactions on Quantum Engineering 3102714 (2021)

# Same pattern holds for probability to find optimal\*

three colouring (left), k-colouring (right)



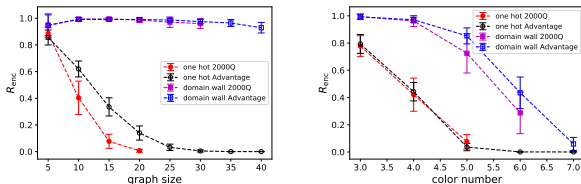
Note that each run was only performed with 100 reads, better results could be attained with more reads

All QPU-encoding combinations found optimal solution at smallest size → explains no “winners” in hypothesis testing

# Digging deeper into performance: encoding failures\*

What fraction of solutions have all one-hot/domain-wall constraints satisfied

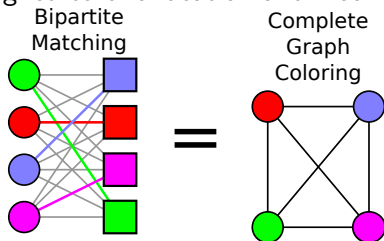
three colouring (left), k-colouring (right)



Domain-wall constraints are much less “fragile” especially with only three colours, makes a much bigger difference than processor structure

# Quadratic Assignment Problem (QAP)

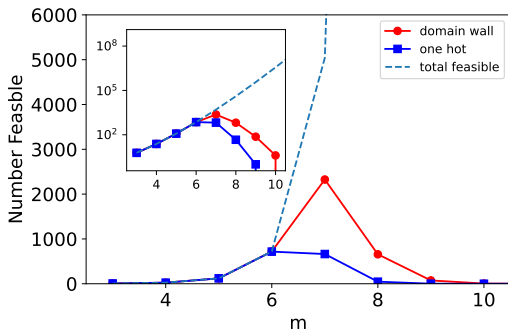
Assign  $m$  facilities to  $m$  locations such that a single facility is only assigned to one location and vice-versa



- ▶ General (hard) version  $\rightarrow$  pairs of assignments are weighted, we use unweighted  $\rightarrow$  **not** hard, but symmetry and large degeneracy useful for analysis
- ▶ Can be thought of as a colouring problem on an  $m$ -node fully connected graph
- ▶  $m!$ -fold degenerate ground state

## Experimental tests (unweighed assignment)

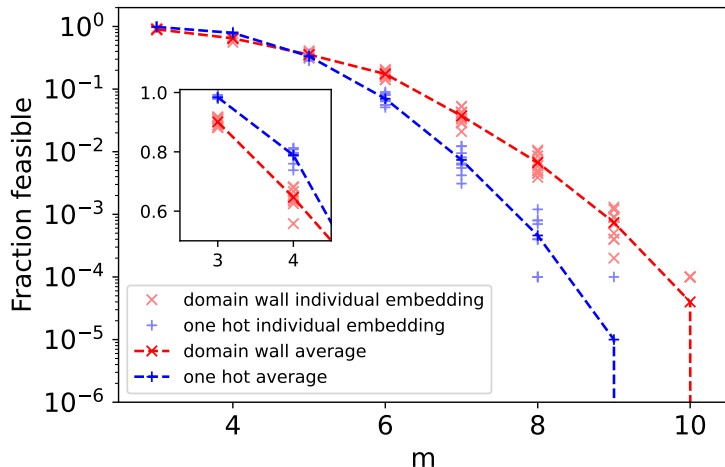
Run on D-Wave *Advantage* annealer 10 embeddings at each size with 10,000 reads for total of 100,000 reads at each size (default settings otherwise)



Able to find all feasible up until about size 6, then both struggle, but domain-wall encoding performs much better.

# Rate of feasible solutions

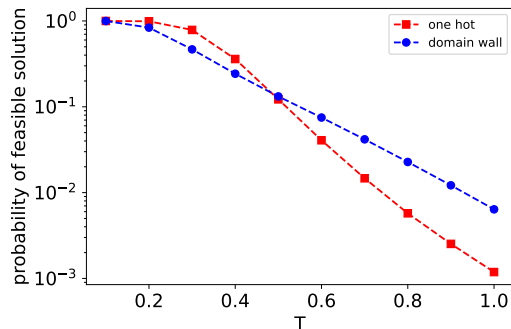
Stars represent fractions of returned solutions which are feasible



At largest size ( $n = 10$ ) domain-wall encoding finds solutions while one-hot finds none.

# One explanation: thermal excitations

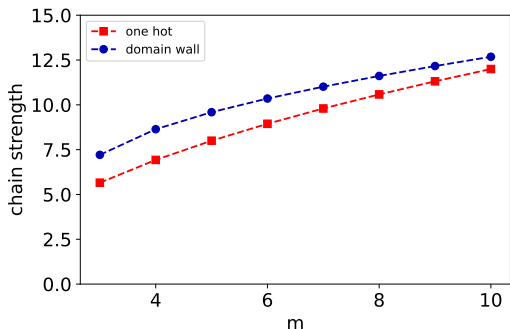
Symmetry of problem means Metropolis algorithm converges quickly, efficient thermal sampling



Probability of feasible solution is better at higher temperature with domain-wall encoding, makes sense one fewer qubit  $\rightarrow$  smaller solution space

# Dynamic range squeezing

Minor embedding chains need to be stronger for larger problems  $\rightarrow$  less range left for problem, effectively higher temperature \*

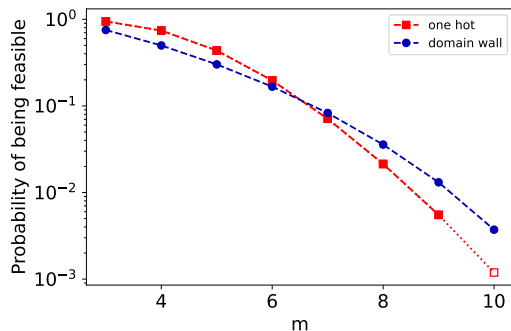


\*we use default “uniform torque compensation” method 



# Thermal equilibrium model

Assuming an energy scale of  $\approx 5$  GHz at the freezing point we find feasible probability for a purely thermal model

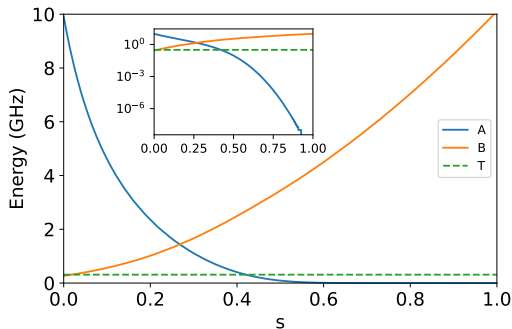


- ▶ Shows same crossover as real data
- ▶ Not in the same location, but...

Estimate of energy scale is rough

Not all sizes will freeze at the same time each will have different scales

## Estimate energy scale



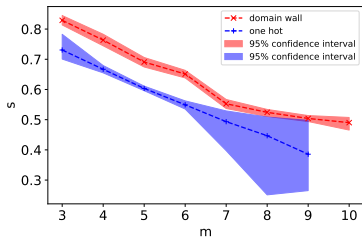
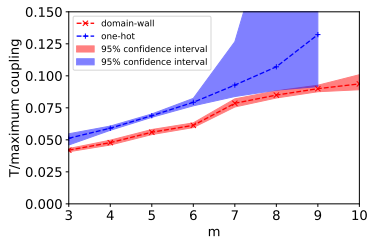
Assume “frozen in” thermal distribution  $\rightarrow$  Kibble-Zurek style approximation

Known physical temperature and experimental success probabilities

Back calculate energy scale ( $B$ ) and therefore freeze point ( $s$ )

Verify that quantum fluctuations ( $A$ ) can be safely ignored at freeze point

# Effective temperature and freeze point

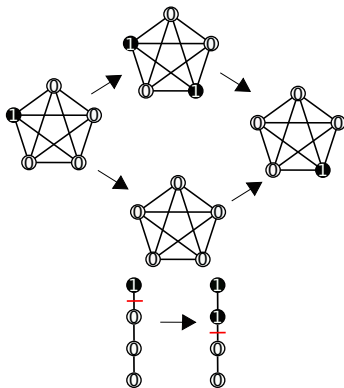


- ▶ Already taken into account embedding strength
- ▶ Domain-wall version effectively sampled at lower temperature  
↔ later freezing

**Encoding has a strong effect on the dynamics of how the problem is solved**

# Why might this be true?

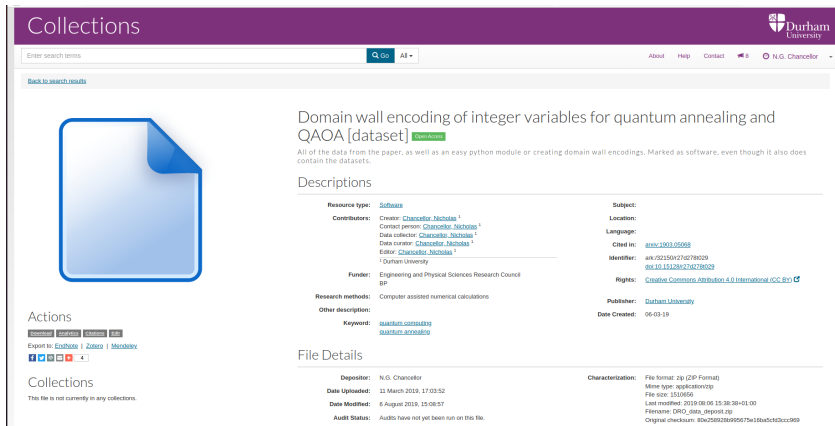
- ▶ One hot value cannot be changed by flipping a single binary variable
- ▶ Domain wall can therefore easier for transverse field to update



**Need to consider underlying physics with encoding**

# Want to try it yourself?

Python code to create domain wall encodings available at <https://collections.durham.ac.uk/>: “Domain wall encoding of integer variables for quantum annealing and QAOA [dataset]”\*



The screenshot shows the Durham University Collections website. The header is purple with the university logo and navigation links. A search bar is present. The main content area features a large blue document icon on the left and a detailed record on the right. The record title is 'Domain wall encoding of integer variables for quantum annealing and QAOA [dataset]' with a green 'Open Access' badge. Below the title is a short description: 'All of the data from the paper, as well as an easy python module or creating domain wall encodings. Marked as software, even though it also does contain the datasets.' The 'Descriptions' section is divided into two columns. The left column lists 'Resource type: Software', 'Contributors' (Creator, Contact person, Data collector, Data curator, Editor), 'Funder' (Engineering and Physical Sciences Research Council), 'Research methods' (Computer assisted numerical calculations), and 'Other description' (Keyword: quantum computing, quantum annealing). The right column lists 'Subject', 'Location', 'Language', 'Cited in' (arXiv:1903.05068), 'Identifier' (arXiv:1903.05068, doi:10.15128/r27d278t029), 'Rights' (Creative Commons Attribution 4.0 International), 'Publisher' (Durham University), and 'Date Created' (06-03-19). Below this is the 'File Details' section with a table of metadata: Depositor (N.G. Chancellor), Date Uploaded (11 March 2019), Date Modified (6 August 2019), Audit Status (Audits have not yet been run on this file), Characterization (File format: zip), File size (1510656), Last modified (2019-08-06), Filename (DRO\_data\_deposit.zip), and Original checksum (80x258928d995675e18ba5c830cc959).

Collections

Enter search terms   All

[Back to search results](#)

## Domain wall encoding of integer variables for quantum annealing and QAOA [dataset] Open Access

All of the data from the paper, as well as an easy python module or creating domain wall encodings. Marked as software, even though it also does contain the datasets.

### Descriptions

<b>Resource type:</b> <a href="#">Software</a>	<b>Subject:</b>
<b>Contributors:</b> Creator: <a href="#">Chancellor, Nicholas</a> <sup>1</sup> Contact person: <a href="#">Chancellor, Nicholas</a> <sup>1</sup> Data collector: <a href="#">Chancellor, Nicholas</a> <sup>1</sup> Data curator: <a href="#">Chancellor, Nicholas</a> <sup>1</sup> Editor: <a href="#">Chancellor, Nicholas</a> <sup>1</sup> <sup>1</sup> Durham University	<b>Location:</b>
<b>Funder:</b> Engineering and Physical Sciences Research Council BP	<b>Language:</b>
<b>Research methods:</b> Computer assisted numerical calculations	<b>Cited in:</b> <a href="#">arxiv:1903.05068</a>
<b>Other description:</b> <b>Keyword:</b> <a href="#">quantum computing</a> <a href="#">quantum annealing</a>	<b>Identifier:</b> <a href="#">arXiv:1903.05068</a> <a href="#">doi:10.15128/r27d278t029</a>
	<b>Rights:</b> <a href="#">Creative Commons Attribution 4.0 International (CC BY)</a>
	<b>Publisher:</b> <a href="#">Durham University</a>
	<b>Date Created:</b> 06-03-19

### File Details

<b>Depositor:</b> N.G. Chancellor	<b>Characterization:</b> File format: zip (ZIP Format)
<b>Date Uploaded:</b> 11 March 2019, 17:03:52	File type: application/zip
<b>Date Modified:</b> 6 August 2019, 15:08:57	File size: 1510656
<b>Audit Status:</b> Audits have not yet been run on this file.	Last modified: 2019-08-06 15:38:38+01:00
	Filename: DRO_data_deposit.zip
	Original checksum: 80x258928d995675e18ba5c830cc959

**Actions**

[Download](#) [Upload](#) [Delete](#) [Edit](#)

Export to: [EndNote](#) | [Zotero](#) | [Mendeley](#)

[Facebook](#) [Twitter](#) [LinkedIn](#) [Reddit](#) [StumbleUpon](#)

**Collections**

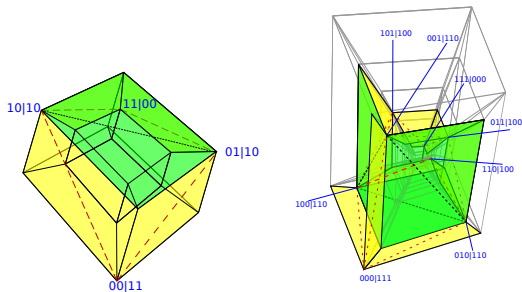
This file is not currently in any collections.

\*<https://doi.org/10.15128/r27d278t029>

## bonus slides

## Aside: Implementing unstructured search dynamics \*

- ▶ Naively requires implementation of all  $2^n$  Pauli strings  $\rightarrow$   
 $-|0\rangle\langle 0| = \prod_i \frac{1}{2}(1 - \sigma_i^z)$  not practical
- ▶ However auxiliary qubits can be used to 'count', spins ones in a configuration  $\rightarrow$  Apply penalty to counting spins and implement at second order of perturbation theory



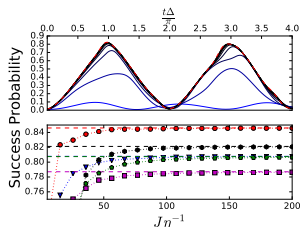
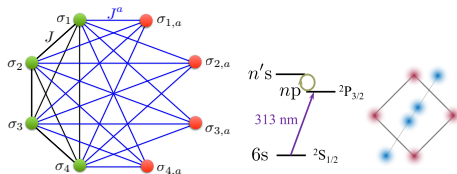
Embeds a hypercube in a hypercube of twice the dimension **Dodds et. al. Phys. Rev. A 100, 032320 (2019)**

\*marked state of  $|0\rangle$  w.l.o.g.

# What is required? (last slide on aside) \*

Need:

- ▶ Nearly all to all interactions, but only two body  $\sigma_i^z \sigma_j^z$ , don't need any higher order
- ▶ Single body  $\sigma_i^z$  and  $\sigma_i^x$  terms
- ▶ ... and that is it, nothing else required, we propose Rydbergs, but maybe one of you has a different idea...



Perturbatively recover simple search dynamics