Challenges of making quantum computing a reality

AMBER group meeting

Nick Chancellor*

February 9, 2024

*for more about me see http://nicholas-chancellor.me and contained links

My Background

Bachelor in Engineering Physics from Colorado School of Mines

PhD from University of Southern California in Physics

Post-doc at UCL, annealing experiments

Post-doc at Durham Physics hybrid quantum/classical
$\rightarrow$ UKRI Innovation fellowship at Durham
$\rightarrow$ teaching fellow at Durham (+ 50% consulting)

Lectureship in Newcastle School of computing (50% +consulting)

Quantum computing

Big idea: harness the fundamental physics of discrete systems (quantum mechanics) to solve important problems

- We know it works in theory: quantum search of unstructured database with N entries in a time proportional to $\sqrt{N}$
- This is not possible without using quantum mechanics (only option without QM is random guess or exhaustive search)

(011) (010)

(111)

(000)

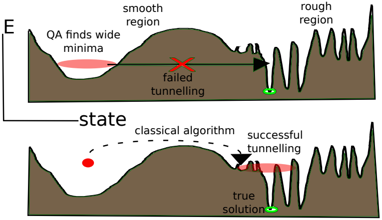
(001) (110)

...but how do we use real, imperfect, quantum machines to solve problems people care about?

Applying Quantum computing

How do we use real, imperfect, quantum machines to solve problems people care about?

1. Only use them for what they are good at do the rest classically hybrid quantum/classical algorithms



2. Find the right problems → need to be the right shape and size for near term the machines... and still be problems people care about

But first... some background on continuous time QC

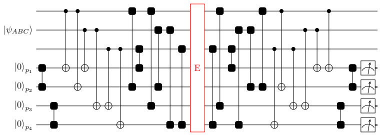
Two different approaches to quantum computing

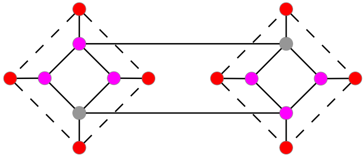
'Gate' based quantum computing

- Discrete quantum operations on qubits
- Construct 'circuits' out of these gates
- Detect and correct errors to reduce effect of noise

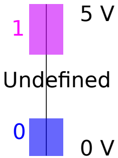
Continuous time

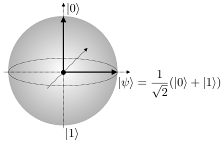
- Map problems directly to physical system
- Allow quantum physics to help search solution space
- Low temperature environment could help solve problems





Why I focus* on continuous time





Classical bits: fundamentally discrete $\rightarrow$ 0 or 1, nothing in between

Lends itself to a discrete *digital* description: bit flips either happen or they don't

Quantum bits: continuous rotations are possible

Breaking operations up into discrete chunks is not natural $\rightarrow$ an (exact) bit flip is just as hard as any other rotation

Bonus feature: applied gate based algorithms similar to continuous time operations $\rightarrow$ cont. time algorithms have implications for gate-model

---

*I am still interested in gate-model quantum computing and have done multiple gate-model projects in the past

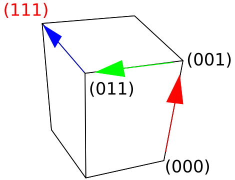
Getting physics to solve hard problems $\rightarrow$ transverse field Ising model

Physics Language, Hamiltonian:

$$H = -A(t) \sum_{i}^{n} X_i + B(t) \left(\sum_{i}^{n} h_i Z_i + \sum_{i,j}^{n} J_{ij} Z_i Z_j \right)$$

What this means in non-physics language:

$\sum_i^n X_i \rightarrow$ Bit flips, hops state through n dimensional hypercube



$\sum_i^n h_i Z_i + \sum_{i,j}^n J_{ij} Z_i Z_j \rightarrow$ Ising spin glass, defines interesting problem to be solved (as bitstring energies) more on next slides

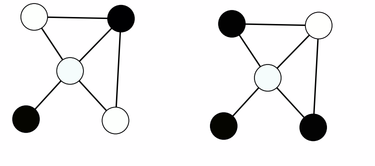
Example of Ising problem mapping *

Have:

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

Want:

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



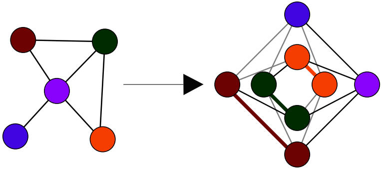
Method:

1. For an edge between vertex i and j add $Z_i + Z_j + Z_i Z_j \rightarrow$ penalizes colouring ($Z = 1$) adacent 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

Minor embedding

- Strong 'ferromagnetic' ($-Z_iZ_j$) 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

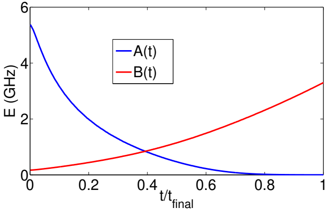


Can embed arbitrary graphs into quasi-planar hardware graph with polynomial (n^2 for fully connected) overhead → Ising model restricted to hardware graph is also NP-hard

In practice this leads to a large overhead → important consideration for solving real problems → can be mitigated by better encoding as discussed later

Actually solving problems

Quantum Hamiltonians generalize classical Monte Carlo algorithms eg. simulated annealing



$$H = -A(t) \sum_{i}^{n} X_i + B(t) \left(\sum_{i}^{n} h_i Z_i + \sum_{i,j}^{n} J_{ij} Z_i Z_j \right)$$

- Parameter sweeps can be used to solve problems
- Low temperature dissipation can help too

Understanding details not necessary for big picture

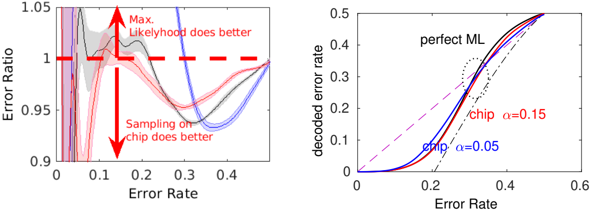
Continuous time quantum computing

Physical system maps interesting computer science problem

Physics of system can be leveraged algorithmically to solve problems*: powerful marriage of physics and CS

Example: Maximum entropy inference on a *physical* quantum annealer NC et. al. Scientific Reports vol. 6, 22318 (2016)

- Thermal states maximize entropy $\rightarrow$ can be used to decode communications
- Superconducting quantum device produces (approximately) these distributions, can beat less powerful classical techniques



*of course there are many details here I don't have time to discuss

The importance of hybrid algorithms*

In the near term:

- Quantum computers may become very computationally 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

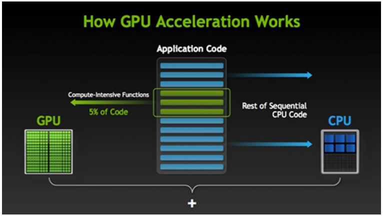
- This naturally indicates a coprocessor arrangement
- Fundamentally hybrid $\rightarrow$ computational model involves both classical and quantum steps
- This is different from just being supported by classical computation, for example through error correction or embedding calculations, see paper for 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 $\rightarrow$ good for highly parallel processing
- Application specific integrated circuits
- Neuromorphic devices $\rightarrow$ structures similar to natural neural networks



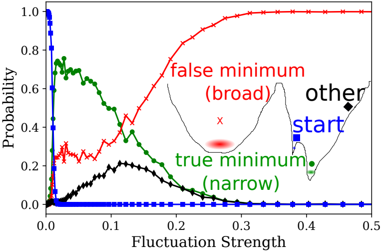
- No reason to think the same logic won't apply for quantum
- Needs fast interconnects and collocation with classical (HPC) resources

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

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
- My early work on subject New J. Phys. 19 023024 (2017)

⋆figure: Chancellor and Kendon Phys. Rev. A 104, 012604 (2021)

†I used to have a slide listing reverse annealing papers, now too many to fit on a slide

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

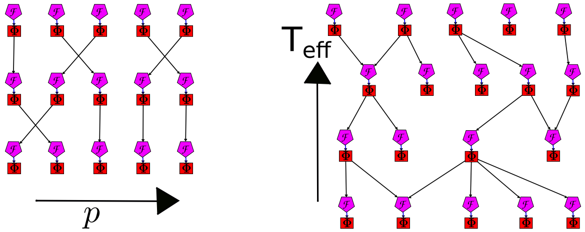
2) Can it ~~beat~~ improve classical computing? Open but positive

- Evidence of speedup for artificial problems
- 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!**

Hybrid quantum/classical, what's next?

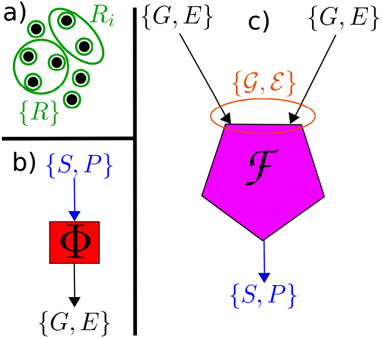
1. More sophisticated algorithms
   - Most work so far has been very simple algorithms
   - Move to more complex ones based on current state of art (particularly the state of the art for specific problems)
   - Develop theoretical framework: inference primitive $\rightarrow$ Chancellor, Natural Computing 22, 737-752, (2023)



2. Understand and improve protocols
   - Understand how these protocols actually work under realistic conditions

Abstract representations of (locally) biased search

In a high level sense biased search behaves similarly regardless of the underlying physical operations



- Won't go through details here, but important to develop a "higher level" (i.e. graphical) picture
- https://doi.org/10.1007/s11047-022-09905-2; Natural Computing (2022) takes some first steps, but more to be done

The effect of encoding: domain-wall 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 model, made from pairwise interactions of x terms:

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

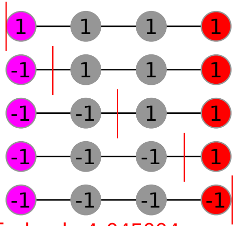
Discrete variables into binary, three ways

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)

Improved performance on maximum three colouring*

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

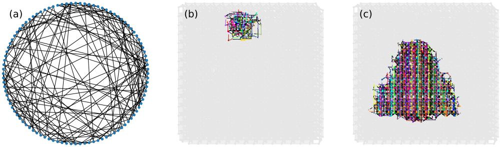
- Domain-wall on 2000Q beats one-hot on Advantage (100 total each size b=number better, w=number worse, p=statistical significance)
- Trend continue up to size where no longer possible to embed in 2000Q (FAIL), similar results for k-colouring (not shown)
- Worth trying if you have discrete problems to encode

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

Other work on encoding

- Consumer data project with dunnhumby (the company behind tesco club card)
- Sponsored student project by Puya Mirkirimi (and the idea behind the project was his)
- Example “promotion-cannabalisation” problem (details of problem not important at this level of discussion)

Replace some quadratic constraints with variationally chosen linear constraints, doesn't always work, but when it does it changes the embedding from (c) to (b)



Puya has a very good talk on this work and it would be good to invite him to give it

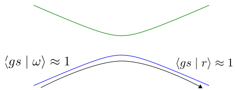
Understanding how quantum annealing computes

Real machines are noisy, hard to keep coherent

Unless P=NP (when quantum machines included), all quantum algorithms for NP-hard problems must do one or both:

1. Protect from all noise for exponential time
2. Succeed with exponentially low probability

Most theory is in the adiabatic limit, succeeds with probability $O(1)$ $\rightarrow$ remain coherent exponentially long, not practical



Need theory/numerics to understand experimentally achievable protocols $\rightarrow$ run many times with low probabilities

Understanding based on energy rather than traditional adiabatic picture Callison et. al. PRX Quantum 2, 010338 (2021) (physics-heavy so I won't go through it here)

What makes a good early use case?

Early quantum computers may be powerful but relatively...

expensive

Needs to be a high value problem

Needs to be hard classically, otherwise why bother

small

Low processor throughput, quantum processor runs on ‘small’ (sub)problem (overall problem could still be high throughput)

NP-hard optimization problems and simulations of electrons are two examples which fit these criteria, there are others as well

Image: public domain taken from wikimedia commons

What makes ~~good~~ **the best** early use cases? *

Everything on the previous slide and...

Problem mapping overheads need to be low

Right size and shape of problem to map to existing machines or special purpose which could be built

Needs hardware *and* problem mapping expertise

structure of interesting instances needs to be understood

Needs application domain experts

Hybrid quantum/classical to get the best out of the machine

Classical algorithms where quantum subroutines can be incorporated

Needs domain *and* quantum expertise

Fundamentally multidisciplinary

*see: arχiv: 2006.05846

Relevant UK projects on finding problems

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/**

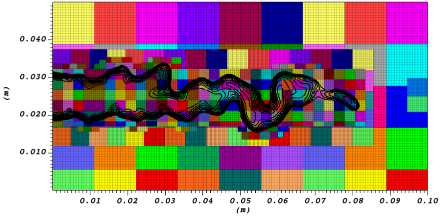
Quantum Enhanced and Verified Exascale Computing (QEVEC)

- 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 me or Viv Kendon at **viv.kendon@strath.ac.uk** if you are interested in potential collaborations

QEVEC work at Durham

- Hired fluid simulation expert postdoc at Durham (Omer Rathore, background in simulating flames)
- Near term application of quantum annealing to optimise load balancing for simulations
- Longer term options for incorporating quantum linear algebra subroutines (HHL) into protocols
- Crucial to optimise input and output to quantum subprocessor



Omer could come and give a full talk on this work but would probably wait until more finished

Representing quantum operations visually

Project by my student (Laur Nita), to produce video games to teach quantum computing,he has also started a company https://www.quarksinteractive.com/

Why visual representations?

- Matrix algebra can be intimidating
- Stereotypes about who should and shouldn't be "good at math"
- Some people may think better visually than in equations
- Add an element of fun/gamification to learning

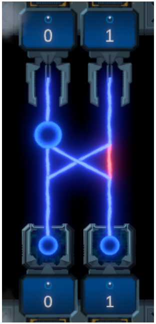
What we want

- Represent arbitrary quantum operations on small systems (but more than one qubit)
- Should be exact rather than approximate (at least up to machine precision)
- Full representation, don't "hide" degrees of freedom

Constructing our representation

State vector $\rightarrow$ coloured balls representing real and imaginary parts
Matrix operator $\rightarrow$ coloured edges (colour represents phase, thickness magnitude)
Rules for adding complex numbers $\rightarrow$ rules for what balls do when they collide

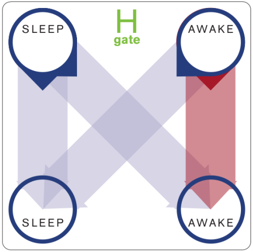
$$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}|0\rangle \rightarrow$$



Worked with colleagues in Durham Education to test learning potential, see: Nita et. al. Research in Science & Technological Education (2021) and arχiv:2106.07077

Saving Schödinger's cat

Card game adaptation for APS physics quest* Single qubit (two level system), want to get the cat into the "awake" state rather than "asleep"



- Ball graphics → cutout cat tokens
- Gates → cards with paths drawn on them
- Computer program → set of rules on how tokens interfere

*https://www.aps.org/programs/outreach/physicsquest/pq21.cfm

Key Message

The key takeaway from this talk should be:

To succeed quantum computing needs a lot of contributions from people who aren't "quantum people"

It will never work if it is just quantum physicists trying to do everything

School of Computing seems like an ideal place to do something:

- Seems to be a growing interest in quantum
- Lots of people doing non-quantum things here as well, probably interesting use cases and other connections
- Plenty of projects already going and future opportunities to expand (see previous)