A domain-wall encoding of discrete variables*

AQC 2021

Nicholas Chancellor

June 23, 2021

Joint Quantum Centre

Durham University

Engineering and Physical Sciences Research Council

*Based on results from arχiv:2102.12224 (with co-authors Jie Chen and Tobias Stollenwerk) and background from other sources

Relation to other work

Domain-wall encoding:

- Steve Abel (Thursday): Simulating field theories
- Raouf Dridi (Thursday, poster): Optimisation studies done in collaboration with Quantum Computing Inc.
- Jie Chen (Friday): Applied to real world network problem

Non-domain-wall work I am involved in:

- Adam Callison (Tuesday, but recorded): Energetic perspective on diabatic annealing
- Viv Kendon (Tuesday, poster): Noise in unstructured quantum-walk/AQC hybrid search
- Jemma Bennett (Tuesday, poster): Error suppression

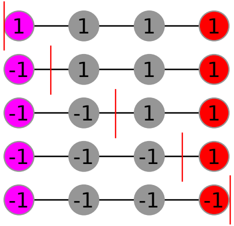
Discrete variables into binary, three ways*

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 method we discuss here

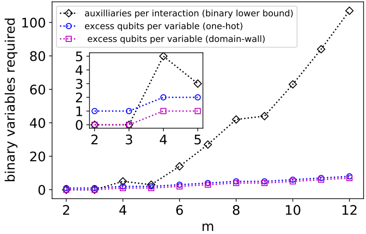
encoded value	qubit configuration
0	1111
1	-1111
2	-1-111
3	-1-1-11
4	-1-1-1-1



*For details see: Chancellor, Quantum Sci. Technol. 4 045004

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 $\rightarrow$ auxilliary variables
- Fair counting needs to include auxilliary variables as well



This is a losing proposition for general interactions*

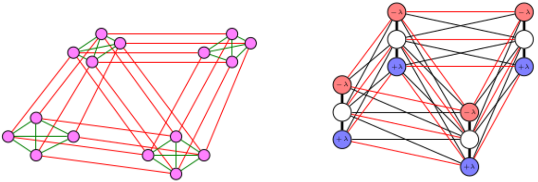
*Extensive discussion of this point recently added to arχiv:2102.12224; binary may still be best for interactions with special structure, example, variable multiplication: Joseph et. al. Phys. Rev. A 103, 032433

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)



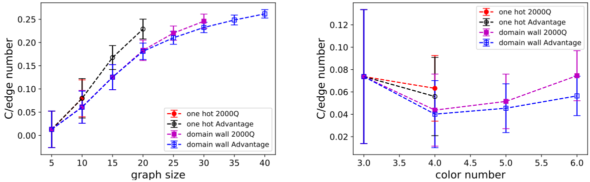
three-colouring $\rightarrow$ randomly generated edges with 50% probability
k-colouring $\rightarrow$ twice as many nodes as colours, random edges with 75% probability

[*]see Chancellor, Quantum Sci. Technol. 4 045004

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

* arχiv:2102.12224

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	2.27×10^{-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 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

*arχiv:2102.12224

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	1.05×10^{-9}		1.42×10^{-9}		2.87×10^{-2}		8.21×10^{-2}		1.31×10^{-12}		2.11×10^{-5}	
5 color (b,w)	91	1	78	1	34	18	23	59	88	1	91	1
5 color p	1.88×10^{-26}		1.32×10^{-22}		1.82×10^{-2}		≈ 1		1.45×10^{-25}		1.88×10^{-26}	
6 color(b,w)	99	0		FAIL	59	15		FAIL		FAIL	99	0
6 color p	1.58×10^{-30}				1.28×10^{-7}						1.58×10^{-30}	
7 color(b,w)	92	0	FAIL	FAIL		FAIL		FAIL		FAIL	FAIL	
7 color p	2.02×10^{-28}											

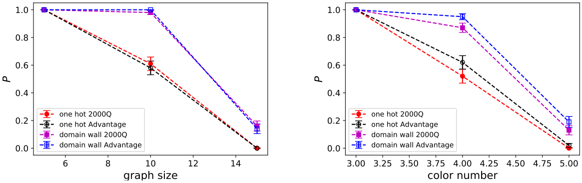
- Domain-wall 2000Q beats one-hot Advantage (in a statistically significant way)
- Trend continue up to size where no longer possible to embed in 2000Q (FAIL)
- One case where 2000Q beats advantage for the same decoding (one-hot)*

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

*arχiv:2102.12224

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

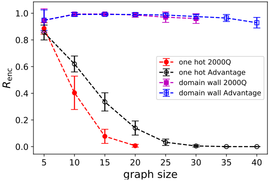
All QPU-encoding combinations found optimal solution at smallest size $\rightarrow$ explains no "winners" in hypothesis testing

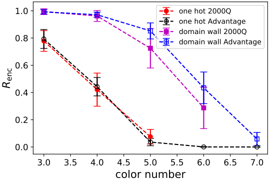
*arχiv:2102.12224

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

---

* arχiv:2102.12224

Results summary

- Binary encoding
  - Losing proposition for generic interaction due to higher order terms*
  - Best strategy in specific cases where higher order terms not needed or included in hardware
- Encoding makes a bigger difference to solution optimality even than choosing a more advanced processor
- Domain wall constraints seem much less "fragile"
- Encoding still helps with chain breaks, but advantage is smaller $\rightarrow$ QPU structure makes a bigger difference

Experiments didn't find any metrics where one-hot does better

No observed downside to using domain-wall encoding, but some major advantages

*see degree-of-freedom counting argument in Chancellor, Quantum Sci. Technol. 4 045004, can't do better than domain-wall with only quadratic