Compiling AQC problems to run on hardware
Lecture 3

Nike Dattani
nike@hpqc.org

HPQC Labs

2022 11 28

Quantum Computing

Q&A for engineers, scientists, programmers, and computing professionals interested in quantum computing

6.8k	8.4k	84%	16k	1.7k	6.9	4y
questions	answers	answered	users	visits/day	questions/day	site age

	questions	answers	answered	users	visits/day	questions/day	site age
Quantum Computing Q&A for engineers, scientists, programmers, and computing professionals interested in quantum computing	6.8k	8.4k	84%	16k	1.7k	6.9	4y
Matter Modeling Q&A for materials modelers and data scientists	2.6k	3.1k	89%	5.2k	754	3.4	2y7m

Quantum Computing

Q&A for engineers, scientists, programmers, and computing professionals interested in quantum computing

6.8k	8.4k	84%	16k	1.7k	6.9	4y
questions	answers	answered	users	visits/day	questions/day	site age

Matter Modeling

Q&A for materials modelers and data scientists

2.6k	3.1k	89%	5.2k	754	3.4	2y7m
questions	answers	answered	users	visits/day	questions/day	site age

Matter Modeling

Q&A for materials modelers and data scientists

2.6k	3.1k	89%	5.2k	754	3.4	2y7m
questions	answers	answered	users	visits/day	questions/day	site age

VASP

GROMACS
FAST. FLEXIBLE. FREE.

LAMMPS

QUANTUMESPRESSO

siesta ™

ORCA

AutoDock 4

VMD

pyscf

Molpro
Quantum Chemistry Software

GAMESS

CP2K

Q-CHEM ™

ASE

WANNIER90

Molecular Dynamics Simulati
CHARMM

pymatgen
(Python Materials Genomics)

PSI4

MRCC

Abinit

WIEN2k

Factoring 143

	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0	
Multiplier					1	p_2	p_1	1	p
					1	q_2	q_1	1	x q
Binary-multiplication					1	p_2	p_1	1	
				q_1	p_2q_1	p_1q_1	q_1		
			q_2	p_2q_2	p_1q_2	q_2			
		1	p_2	p_1	1				
Carry	z_{67}	z_{56}	z_{45}	z_{34}	z_{23}	z_{12}			
	z_{57}	z_{46}	z_{35}	z_{24}					
Product	1	0	0	0	1	1	1	1	143

$$5 - 3p_1 - p_2 - q_1 + 2p_1q_1 - 3p_2q_1 + 2p_1p_2q_1 - 3q_2 + p_1q_2 + 2p_2q_2 + 2p_2q_1q_2$$

p_1	p_2	q_1	q_2	F
0	0	0	0	5
0	0	0	1	2
0	0	1	0	4
0	0	1	1	1
0	1	0	0	4
0	1	0	1	3
0	**1**	**1**	**0**	**0**
0	1	1	1	1
1	0	0	0	2
1	**0**	**0**	**1**	**0**
1	0	1	0	3
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	3

$$\rightarrow p = 11, q = 13$$

$$\rightarrow p = 13, q = 11$$

4 variables:

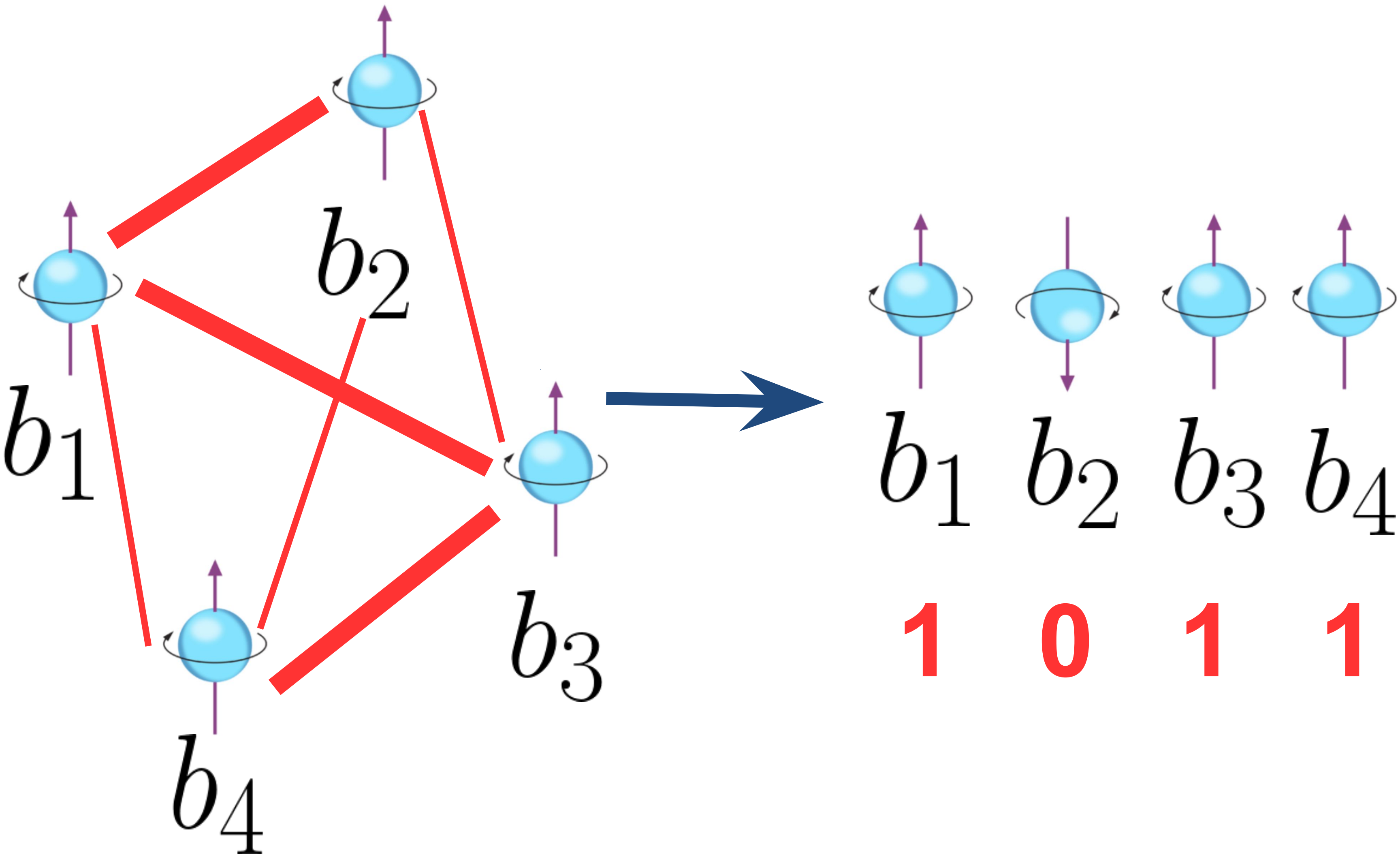
Search through 2^4 possibilities

5000 variables:

Search through 2^{5000} possibilities

$$5 - 3p_1 - p_2 - q_1 + 2p_1q_1 - 3p_2q_1 + 2p_1p_2q_1 - 3q_2 + p_1q_2 + 2p_2q_2 + 2p_2q_1q_2$$

$$5 - 3b_1 - b_2 - b_3 + 2b_1b_3 - 3b_2b_3 + 2b_1b_2b_3 - 3b_3 + b_1b_4 + 2b_2b_4 + 2b_2b_3b_4$$



A problem:

$$5 - 3b_1 - b_2 - b_3 + 2b_1b_3 - 3b_2b_3 + 2b_1b_2b_3 - 3b_3 + b_1b_4 + 2b_2b_4 + \underline{2b_2b_3b_4}$$

$$b_a\left(-b_1+b_2+b_3\right)-b_1b_2-b_1b_3+b_1$$

Cubic

$$b_a\left(-b_1+b_2+b_3\right)-b_1b_2-b_1b_3+b_1$$

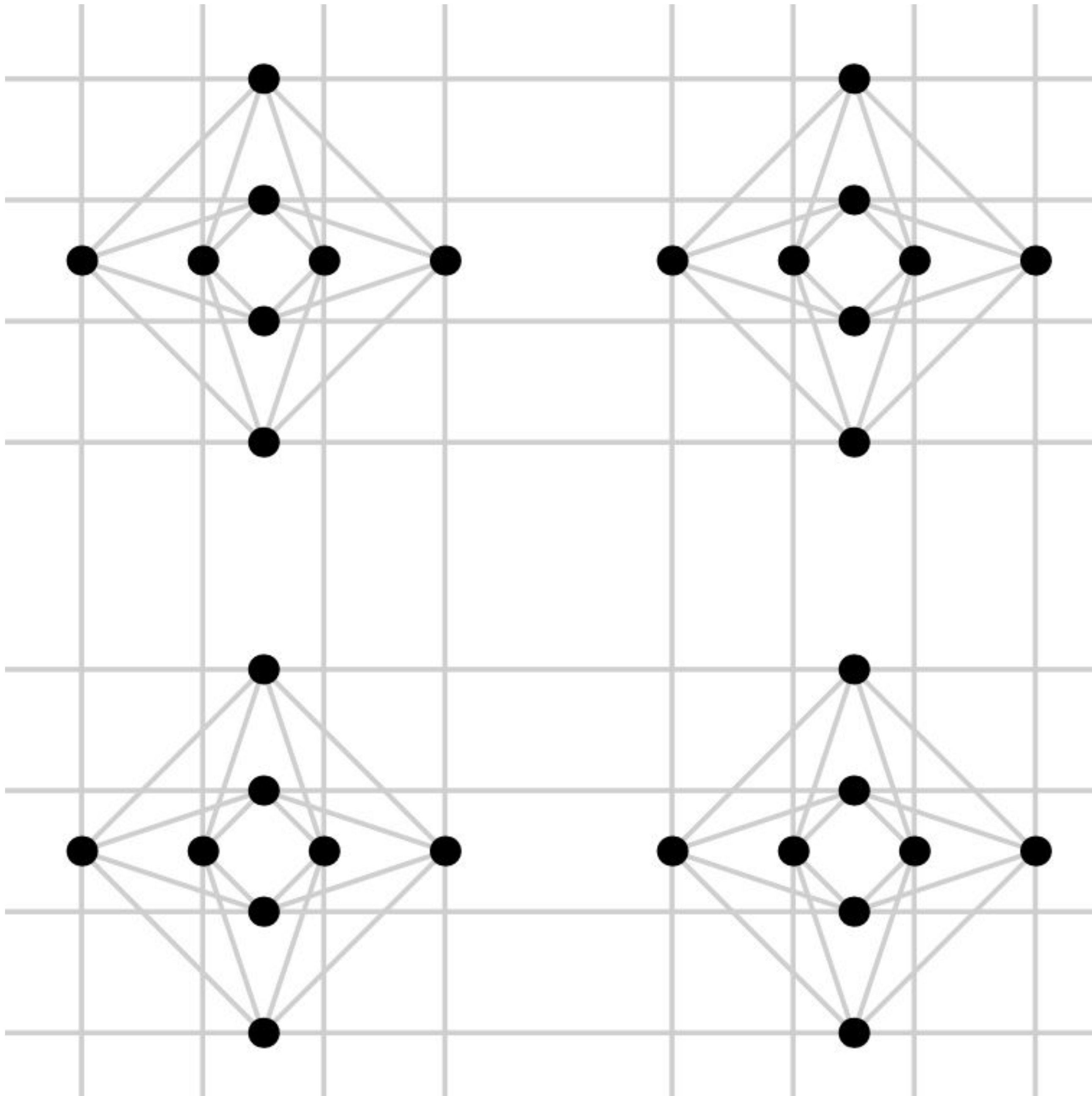
Quadratic

Compile a problem to run on AQC hardware:

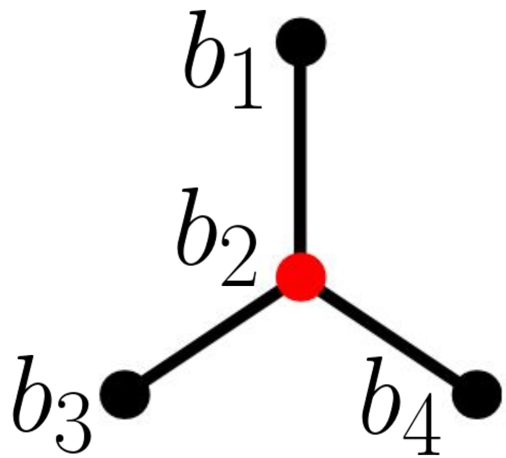
- Turn it into a minimization problem
- Remove unnecessary qubits ($a + b = 1 + 2c \Rightarrow c = 0$)
- **Quadratize** cubic terms into quadratics!
- One final thing…

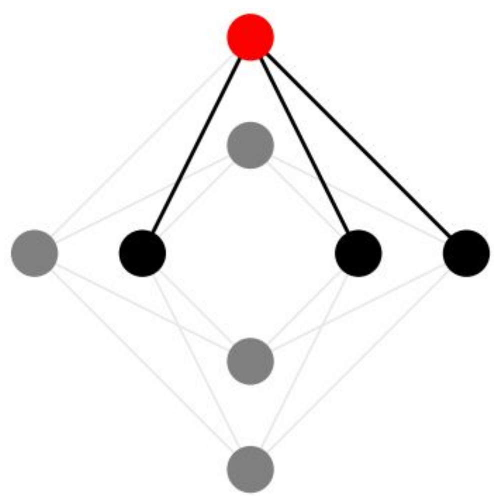
One way to **compile** the factoring problem to run on quantum hardware:

- Turn it into a minimization problem
- Remove unnecessary qubits ($a + b = 1 + 2c \Rightarrow c = 0$)
- **Quadratize** cubic terms into quadratics!
- One final thing… **Graph embedding…**

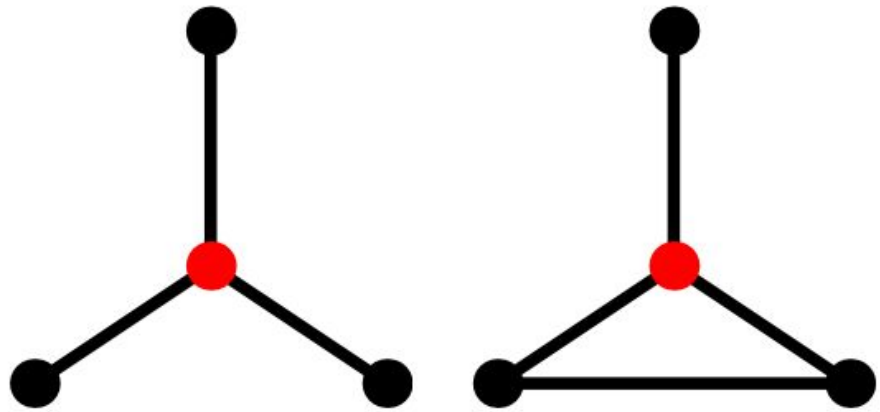


$$5 - 3b_1 - b_2 - b_3 + 2b_1b_3 - \underline{3b_2b_3} + 2b_1b_2b_3 - 3b_3 + \underline{b_1b_2} + \underline{2b_2b_4} + 2b_2b_3b_4$$

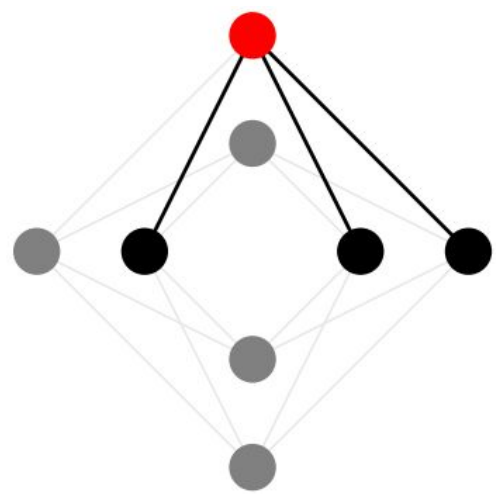




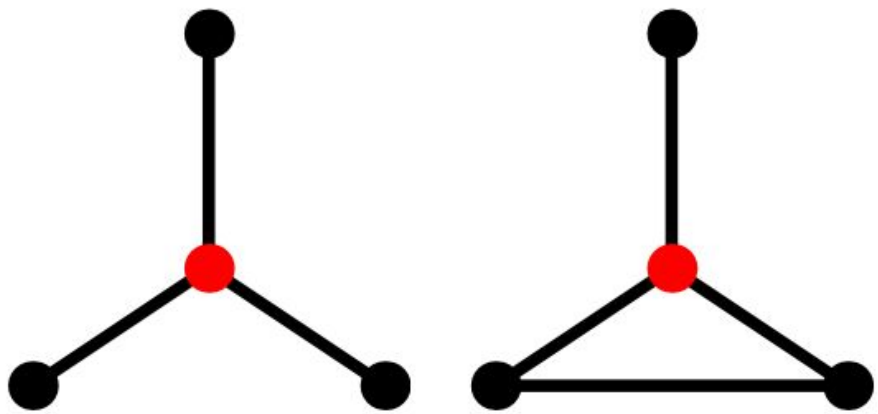
$$5 - 3b_1 - b_2 - b_3 + \underline{2b_1b_3} - \underline{3b_2b_3} + 2b_1b_2b_3 - 3b_3 + \underline{b_1b_2} + \underline{2b_2b_4} + 2b_2b_3b_4$$



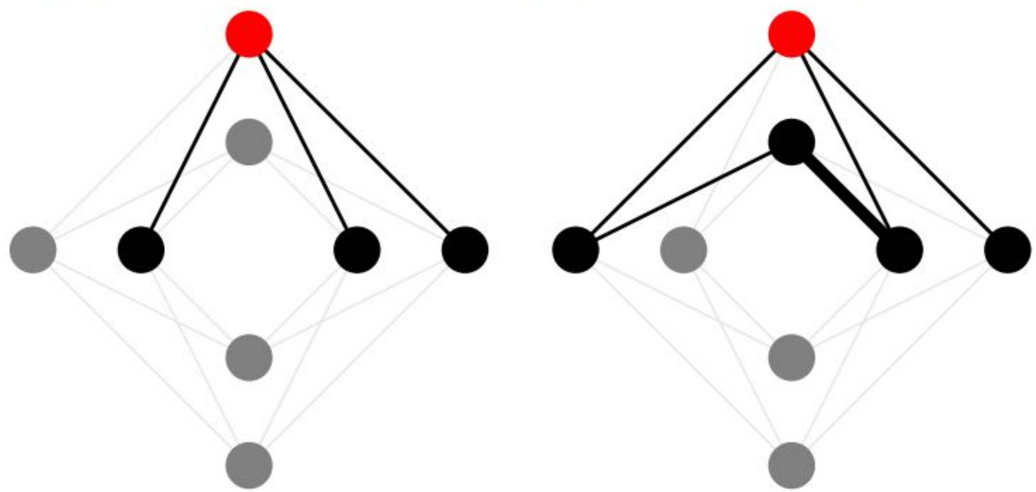
(a) *Propeller* **(b)** *Coat Hanger*



$$5 - 3b_1 - b_2 - b_3 + \underline{2b_1b_3} - \underline{3b_2b_3} + 2b_1b_2b_3 - 3b_3 + \underline{b_1b_2} + \underline{2b_2b_4} + 2b_2b_3b_4$$



(a) *Propeller* **(b)** *Coat Hanger*

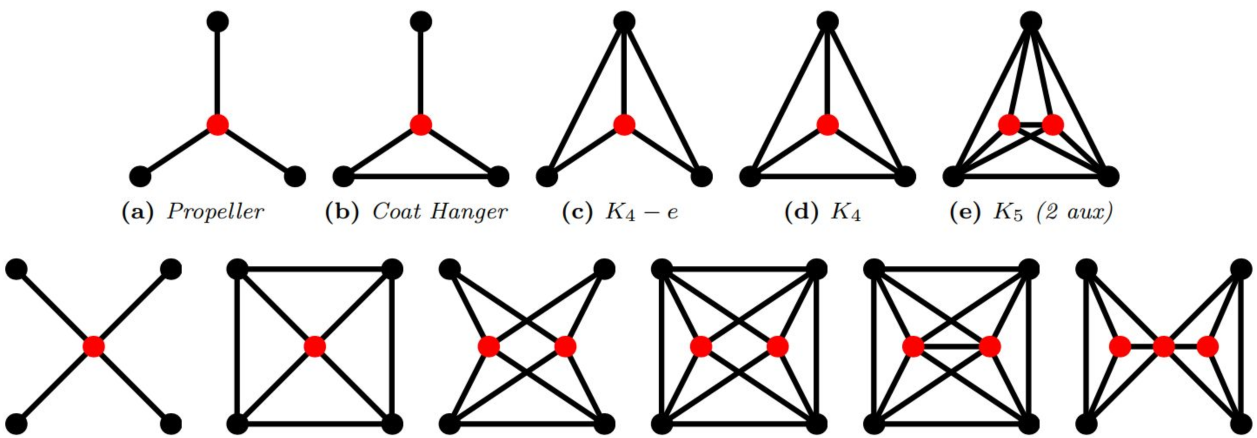


Embedding quadratization gadgets on Chimera and Pegasus graphs

Nike Dattani*
Harvard-Smithsonian Center for Astrophysics

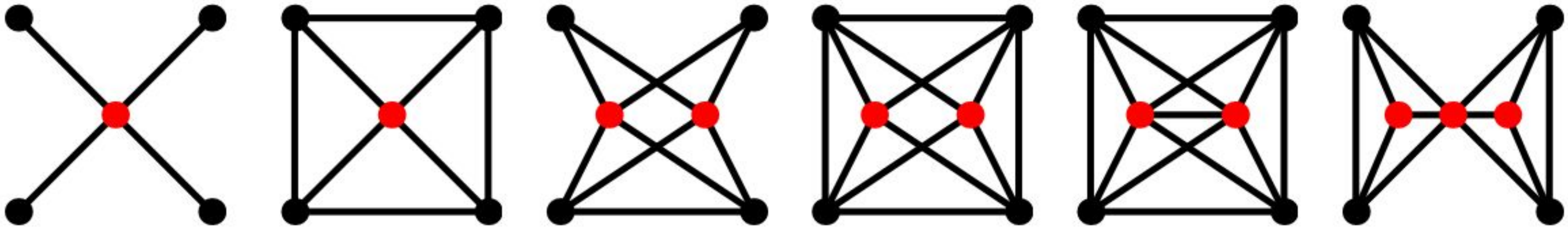
Nicholas Chancellor†
Durham University, Joint Quantum Centre

Figure 2: *Gadget graphs.* Graphs showing the connectivity between qubits in quadratization gadgets for cubic to quadratic gadgets (top row), and quartic to quadratic gadgets (bottom row). Red vertices represent auxiliary qubits and black vertices represent logical qubits. Black edges denote the existence of a quadratic term in the gadget, involving the two corresponding qubits represented by vertices connected by the edge. *Linear and constant terms in the gadgets are completely ignored here.*

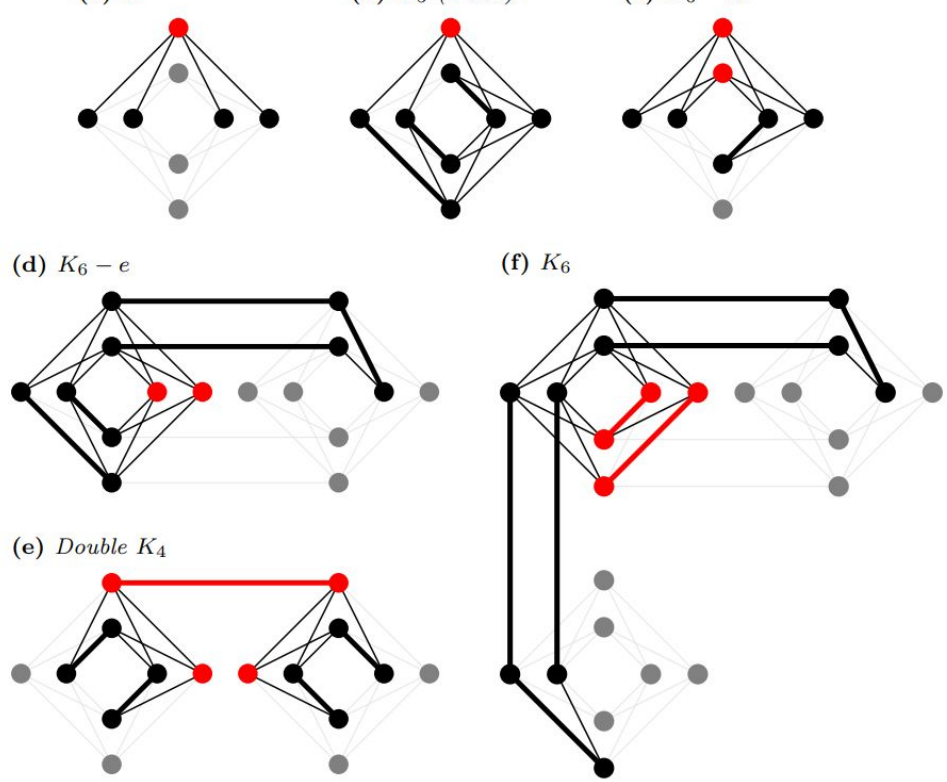


(a) *Propeller* **(b)** *Coat Hanger* **(c)** $K_4 - e$ **(d)** K_4 **(e)** K_5 *(2 aux)*

(f) X **(g)** K_5 *(1 aux)* **(h)** $K_6 - 4e$ **(i)** $K_6 - e$ **(j)** K_6 **(k)** *Double* K_4



(f) X **(g)** K_5 *(1 aux)* **(h)** $K_6 - 4e$ **(i)** $K_6 - e$ **(j)** K_6 **(k)** *Double* K_4



How many qubits to factor RSA-232?

5,893 qubits (in a degree-4 optimization problem)

How many qubits to factor RSA-232?

5,893 qubits (in a degree-4 optimization problem)

18,766 qubits (after quadratization into degree-2)

How many qubits to factor RSA-232?

5,893 qubits (in a degree-4 optimization problem)

18,766 qubits (after quadratization into degree-2)

5.5 billion qubits (on D-Wave, after graph embedding!)

How many qubits to factor RSA-232?

5,893 qubits (in a degree-4 optimization problem)

18,766 qubits (after quadratization into degree-2)

5.5 billion qubits (on D-Wave, after graph embedding!)

H_2 molecule

$$
\begin{aligned}
H &= -0.812611 + 0.171201 Z_0 + 0.171201 Z_1 - 0.2227965 Z_2 - 0.2227965 Z_3 \\
&= +0.16862325 Z_1 Z_0 + 0.12054625 Z_2 Z_0 + 0.165868 Z_2 Z_1 + 0.165868 Z_3 Z_0 \\
&= +0.12054625 Z_3 Z_1 + 0.17434925 Z_3 Z_2 - 0.04532175 X_3 X_2 Y_1 Y_0 \\
&= +0.04532175 X_3 Y_2 Y_1 X_0 + 0.04532175 Y_3 X_2 X_1 Y_0 - 0.04532175 Y_3 Y_2 X_1 X_0
\end{aligned}
$$

How many qubits to factor RSA-232?

5,893 qubits (in a degree-4 optimization problem)
18,766 qubits (after quadratization into degree-2)
5.5 billion qubits (on D-Wave, after graph embedding!)

H_2 molecule

$$
\begin{aligned}
H &= -0.812261\mathbf{1} + 0.171201 Z_0 + 0.171201 Z_1 - 0.2227965 Z_2 - 0.2227965 Z_3 \\
&= +0.16862325 Z_1 Z_0 + 0.12054625 Z_2 Z_0 + 0.165868 Z_2 Z_1 + 0.165868 Z_3 Z_0 \\
&= +0.12054625 Z_3 Z_1 + 0.17434925 Z_3 Z_2 - 0.04532175 X_3 X_2 Y_1 Y_0 \\
&= +0.04532175 X_3 Y_2 Y_1 X_0 + 0.04532175 Y_3 X_2 X_1 Y_0 - 0.04532175 Y_3 Y_2 X_1 X_0
\end{aligned}
$$

Quadratization is a problem!

$$b_1 b_2 \dots b_k \to \left(\sum_{i=1}^{k-2} b_{a_i} (k - i - 1 + b_i - \sum_{j=i+1}^{k} b_j) \right) + b_{k-1} b_k \tag{43}$$

Cost

- $k - 2$ auxiliary variables for each k-local term.

$$b_1 b_2 \dots b_k \to \left(\sum_{i=1}^{k-2} b_{a_i} (k - i - 1 + b_i - \sum_{j=i+1}^{k} b_j) \right) + b_{k-1} b_k \tag{43}$$

Cost

- $k - 2$ auxiliary variables for each k-local term.

$$b_1 \dots b_k \to \left(\sum_{i=1}^{n_k} b_{a_i} \left(c_{i,d} \left(- \sum_{j=1}^{k} b_j + 2i \right) - 1 \right) + \sum_{i<j} b_i b_j \right) \tag{45}$$

where $n_k = \left\lfloor \frac{k-1}{2} \right\rfloor$ and $c_{i,k} = \begin{cases} 1, & i = n_d \text{ and } k \text{ is odd,} \\ 2, & \text{else.} \end{cases}$

Cost

- $\left\lfloor \frac{k-1}{2} \right\rfloor$ auxiliary variables for each k-order term

$$b_1 b_2 \ldots b_k \to \left(\sum_{i=1}^{k-2} b_{a_i} (k - i - 1 + b_i - \sum_{j=i+1}^{k} b_j) \right) + b_{k-1} b_k \tag{43}$$

Cost

- $k - 2$ auxiliary variables for each k-local term.

$$b_1 \ldots b_k \to \left(\sum_{i=1}^{n_k} b_{a_i} \left(c_{i,d} \left(-\sum_{j=1}^{k} b_j + 2i \right) - 1 \right) + \sum_{i<j} b_i b_j \right) \tag{45}$$

where $n_k = \left\lfloor \frac{k-1}{2} \right\rfloor$ and $c_{i,k} = \begin{cases} 1, & i = n_d \text{ and } k \text{ is odd,} \\ 2, & \text{else.} \end{cases}$

Cost

- $\left\lfloor \frac{k-1}{2} \right\rfloor$ auxiliary variables for each k-order term

$$\begin{aligned} b_1 b_2 \cdots b_k \to \alpha^{b} \sum_i b_i + \alpha^{b_{a,1}} \sum_i b_{a_i} + \alpha^{b_{a,2}} b_{a_m} + \alpha^{bb} \sum_{ij} b_i b_j + \alpha^{bb_{a,1}} \sum_i \sum_j^{m-1} b_i b_{a_j} + \\ \alpha^{bb_{a,2}} \sum_i b_i b_{a_m} + \alpha^{b_{a,1} b_{a,1}} \sum_{ij}^{m-1} b_{a_i} b_{a_j} + \alpha^{b_{a,1} b_{a,2}} \sum_i^{m-1} b_{a_i} b_{a_m}, \end{aligned} \tag{51}$$

Cost

$\left\lceil \frac{k}{4} \right\rceil$ auxiliary qubits per positive monomial.

Cost

$\lceil \log k \rceil$ auxiliary qubits per positive monomial.

Cost

$\lceil \log k \rceil$ auxiliary qubits per positive monomial.

$$b_1 \ldots b_k \to \frac{1}{2}\left(2^{m+1} - k + \sum_i b_i - \sum_i^m 2^i b_{a_i}\right)\left(2^{m+1} - k + \sum_i b_i - \sum_i^m 2^i b_{a_i} - 1\right)$$

$\lceil \log {}^{k}\!/_{2} \rceil$ auxiliary qubits per positive monomial.

Cost

$\lceil \log k \rceil$ auxiliary qubits per positive monomial.

$$b_1 \ldots b_k \rightarrow \frac{1}{2}\left(2^{m+1} - k + \sum_i b_i - \sum_i^m 2^i b_{a_i}\right)\left(2^{m+1} - k + \sum_i b_i - \sum_i^m 2^i b_{a_i} - 1\right)$$

$\lceil \log k/2 \rceil$ auxiliary qubits per positive monomial.

Quadratization in Discrete Optimization and Quantum Mechanics

Nike Dattani[1,*]

[1]*Harvard Smithsonian Center for Astrophysics, Cambridge, Massachusetts*

Quadratization without auxiliary qubits

Reducing multi-qubit interactions in adiabatic quantum computation. Part 1: The "deduc-reduc" method and its application to quantum factorization of numbers

Richard Tanburn[1, *]
[1] *Mathematical Institute, Oxford University, OX2 6GG, Oxford, UK.*

Emile Okada[2, †]
[2] *Department of Mathematics, Cambridge University, CB2 3AP, Cambridge, UK.*

Nikesh S. Dattani[3,4, ‡]
[3] *School of Materials Science and Engineering, Nanyang Technological University, 639798, Singapore, and*
[4] *Fukui Institute for Fundamental Chemistry, 606-8103, Kyoto, Japan*

Reducing multi-qubit interactions in adiabatic quantum computation. Part 2: The "split-reduc" method and its application to quantum determination of Ramsey numbers

Emile Okada[1, *]
[1] *Department of Mathematics, Cambridge University, CB2 3AP, Cambridge, UK.*

Richard Tanburn[2, †]
[2] *Mathematical Institute, Oxford University, OX2 6GG, Oxford, UK.*

Nikesh S. Dattani[3,4, ‡]
[3] *School of Materials Science and Engineering, Nanyang Technological University, 639798, Singapore, and*
[4] *Fukui Institute for Fundamental Chemistry, 606-8103, Kyoto, Japan*

Quadratization without auxiliary qubits

Reducing multi-qubit interactions in adiabatic quantum computation. Part 1: The "deduc-reduc" method and its application to quantum factorization of numbers

Richard Tanburn[1,*]
[1]*Mathematical Institute, Oxford University, OX2 6GG, Oxford, UK.*

Emile Okada[2,†]
[2]*Department of Mathematics, Cambridge University, CB2 3AP, Cambridge, UK.*

Nikesh S. Dattani[3,4,‡]
[3]*School of Materials Science and Engineering, Nanyang Technological University, 639798, Singapore, and*
[4]*Fukui Institute for Fundamental Chemistry, 606-8103, Kyoto, Japan*

Reducing multi-qubit interactions in adiabatic quantum computation. Part 2: The "split-reduc" method and its application to quantum determination of Ramsey numbers

Emile Okada[1,*]
[1]*Department of Mathematics, Cambridge University, CB2 3AP, Cambridge, UK.*

Richard Tanburn[2,†]
[2]*Mathematical Institute, Oxford University, OX2 6GG, Oxford, UK.*

Nikesh S. Dattani[3,4,‡]
[3]*School of Materials Science and Engineering, Nanyang Technological University, 639798, Singapore, and*
[4]*Fukui Institute for Fundamental Chemistry, 606-8103, Kyoto, Japan*

Bill Macready <wgm@dwavesys.com> 13/10/2015

to me, Zhengbing, Fabian, fgaitan

Hi Nike

We started implementing the split-reduc method you described, and have a few questions.

Compiling your problem to run on quantum hardware:

- **Turn it into a minimization problem**
- **Remove unnecessary qubits** (can use my open source code)
- **Quadratization** (see my book)
- **Graph embedding** (see my papers with N. Chancellor)

Compiling your problem to run on quantum hardware:

- **Turn it into a minimization problem**
- **Remove unnecessary qubits** (can use my open source code)
- **Quadratization** (see my book)
- **Graph embedding** (see my papers with N. Chancellor)

I can compile your code to run on quantum hardware

We have:
- 100 QPU hours/month on D-Wave machines
- **400 QPU hours on the IBM Q20x**
- **$10000 of QPU time on IonQ (2nd cohort!)**

nike@hpqc.org , info@hpqc.org

If you still prefer classical computing, we can share:

- 2048 TB of disk space
- 808 CPU years / year
- 4 GPU years / year

Thank you!

Now we know how to compile problems to run on AQC hardware!

In Lecture 4, we'll look at some of the actual hardware

nike@hpqc.org