Quadratization in Pseudo-Boolean Optimization and Adiabatic Quantum Computing

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

Nicholas Chancellor[2]
[2]*Department of Physics, Durham University, DH1 3LE, South Road, Durham, UK.*

Nikesh S. Dattani[3,4,5]
[3]*Harvard-Smithsonian Center for Astrophysics,*
[4]*McMaster University, and*
[5]*Hertford College, Oxford University.*

A collaborative, evolving, open review paper on k-local to 2-local transformations (quadratizations) in classical computing, quantum annealing, and universal adiabatic quantum computing.

I. INTRODUCTION

Every computation can be done by minimizing a Hamiltonian of either one of the following forms:

$$H = \sum_i^n (\alpha_i z_i + \beta_i x_i) + \sum_{ij}^n (\alpha_{ij} z_i z_j + \beta_{ij} x_i x_j), \tag{1}$$

$$H = \sum_i^n (\alpha_i z_i + \beta_i x_i) + \sum_{ij}^n (\alpha_{ij} z_i z_j + \beta_{ij} x_i z_j), \tag{2}$$

where the z and x variables denote the Pauli matrices:

$$z \equiv \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \; x \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{3}$$

the α and β coefficients are real numbers; and the Appendix will teach any unfamiliar readers how to interpret this notation.

We know that every computation can be done this way because the minimization can be done by adiabatic quantum computation (AQC), and it has been proven [1, 2] that AQC can simulate any circuit-based quantum computation with overhead that grows at most polynomially with the size of the problem, and we know that circuit-based quantum computation can simulate any classical computation with at most polynomial overhead. If we remove all the terms in Eq. 1 or Eq. 2 that have x operators, we get an Ising Hamiltonian that is quadratic:

$$H = \sum_i^n \alpha_i z_i + \sum_{ij}^n \alpha_{ij} z_i z_j, \tag{4}$$

and we no longer know of any proof that ***any*** computation can be done by minimizing such a Hamiltonian with only polynomial overhead compared to circuit-based quantum

computation or even classical computation, but plenty of very interesting problems can still be formulated as the minimization or approximate minimization of a quadratic Ising Hamiltonian, such as neural network training [3], computerized image denoising [4–7], integer factorization [8], and Ramsey number determination [9, 10], just to name a few. However unlike Eq. 2, plenty of devices that can minimize Ising Hamiltonians already exist, and with further restrictions on the α coefficients, D-Wave has made devices with $n = 2000$, and this number has roughly doubled every 14 months since the first 1-qubit demonstration in 2003, a phenomenon analogous to Moore's law of classical computer growth, which has become known as Rose's Law.

Furthermore, ***no classical computer has ever been able to minimize certain quadratic Ising Hamiltonians faster than the D-Wave machines to date*** [11, 12], so being able to turn computations into the form of Eq. 4 appears to be promising in terms of demonstrating the first useful application of quantum supremacy (which is the term used for quantum devices outperforming classical computation devices). Moreover, for problems like neural networks and computerized image denoising, the best known classical computation algorithms also work by turning the problems into minimizations of Ising Hamiltonians, so even if you are happy with continuing to only use classical computer and have no interest in using quantum annealing devices, turning a problem into a quadratic Ising Hamiltonian minimization problem can be very powerful for some computations.

For neural networks, image denoising, integer factorization, and Ramsey number determination, it is rather easy to turn the problem into the form:

$$H = \sum_{i}^{n} \alpha_i z_i + \sum_{ij}^{n} \alpha_{ij} z_i z_j + \sum_{ijk}^{n} \alpha_{ijk} z_i z_j z_k + \sum_{ijkl}^{n} \alpha_{ijkl} z_i z_j z_k z_l \ldots, \tag{5}$$

but then ***quadratizing*** Eq. 5 into the quadratic Ising form often requires the introduction of auxiliary variables, and since minimizing Eq. 5 often has cost $\mathcal{O}(2^n)$, it is often extremely desirable to do the quadratization with ***as few auxiliary variables as possible***. Apart from quadratizing with minimum number of variables, it is also often desirable to reduce the number of non-zero α coefficients, to keep the non-zero α coefficients bounded or with specific values, to reduce the number of positive (non-submodular) α coefficients, to reduce the spectral ratio of the Hamiltonian ('maximum energy minus minimum energy' divided by 'energy of the second minimum minus the energy of the global minimum'), and to reduce the number of energy levels close to the global minimum. Adjusting the energy landscape in order to accomplish all of these things apart from reducing the number of auxiliary variables needed has been termed Energy Landscape Manipulation (ELM) which is an entirely separate topic, sometimes even more important than quadratizing with as few auxiliary variables as possible, nonetheless this Review simply focuses on methods for the latter aim.

It has been shown in [7] that a general function of the form Eq. 5 with degree k, can be quadratized using $O(n^{k/2})$ auxiliary variables. However, it is often possible to quadratize with far fewer auxiliary variables (for example when the auxiliary variables for quadratizing one term are reused to quadratize other terms that have some of the same variables in them) or to quadratize without any auxiliaries at all (ex. if a Groebner basis can be found with all basis functions being quadratic, or if some symmetries in the Hamiltonian can be found that make it possible to quadratize without auxiliaries).

Part I

Hamiltonians with only z terms (pseudo-Boolean functions)

II. METHODS THAT INTRODUCE ZERO AUXILIARY VARIABLES

A. Deduction Reduction (Deduc-reduc)

Summary

We look for *deductions* (e.g. $b_1 b_2 = 0$) that hold at the global minimum. These can be found by *a priori* knowledge of the given problem, or by enumerating solutions of a small subset of the variables. We can then substitute high-order terms using the low-order terms of the deduction, and add on a penalty term to preserve the ground states [13].

Cost

- 0 auxiliary variables needed.
- The computational cost of the search for deductions is difficult to estimate. The approximate worst-case complexity is $\mathcal{O}(n^{d+1}2^m)$ where m is the number of variables in a 'small' problem, n is the total number of qubits and d is the maximum degree of deductions we are searching for. We suggest $10 \leq m \leq 20$, so that a small problem involves checking roughly $1,000$ to $1,000,000$ states, and $d = 2$. See the appendix for more details.

Pros

- No auxiliary variables needed.

Cons

- When deductions cannot be determined naturally (as in the Ramsey number determination problem, see Example XII), deductions need to be found by 'brute force', which scales exponentially with respect to m. For highly connected systems (systems with a large number of non-zero α_{ij} coefficients), the value of m required to find even one deduction can be prohibitively large.

Example

Consider the Hamiltonian:

$$H_{4\text{–local}} = b_1 b_2 (10 + b_3 + b_3 b_4) + b_1 (b_3 - 3) + b_2 (b_3 - 2b_3 - b_4 - 1) + F(x_3, x_4, x_5, \ldots, x_N) \quad (6)$$

where F is any polynomial in b_i for $i \geq 3$.
Since the 10 coefficient of $b_1 b_2$ is greater than the sum of all of the other coefficients involving b_1 or b_2, it must be the case that $b_1 b_2 = 0$. Specifically, for the 4 assignments of (b_3, b_4), we see that $b_1 b_2 = 0$ at every minimum of $H_{4\text{–local}} - F$.

Using deduc-reduc we have:

$$H_{2\text{–local}} = 12 b_1 b_2 + b_1 (b_3 - 3) + b_2 (b_3 - 2b_3 - b_4 - 1) + F(x_3, x_4, x_5, \ldots, x_N) \quad (7)$$

which has the same global minima as $H_{4\text{–local}}$ but one fewer quartic and one fewer cubic term.

Bibliography

- Original paper, with application to integer factorization: [13].

B. ELC Reduction

Summary

An Excludable Local Configuration (ELC) is a partial assignment of variables that make it impossible to achieve the minimum. We can therefore add a term that corresponds to the energy of this ELC without changing the solution to the minimization problem. In practice we can eliminate all monomials with a variable in which a variable is set to 0, and reduce any variable set to 1. Given a general Hamiltonian we can try to find ELCs by enumerating solutions of a small subset of variables in the problem [6].

Cost

- 0 auxiliary variables needed.
- For n qubits there is $\binom{n}{m}$ ways to choose m of them and 2^m assignments of these m variables, therefore $\mathcal{O}\left(2^m\binom{n}{m}\right)$ operations required to enumerate all possible cases for all possible subsets of size $m \leq n$ variables.
- Approximate methods exist which have been shown to be much faster and give good approximations to the global minimum [6].

Pros

- No auxiliary variables needed.

Cons

- No known way to find ELCs except by 'brute force', which scales exponentially with respect to m.
- ELCs do not always exist.

Example

Consider the Hamiltonian:

$$H_{3\text{-local}} = b_1b_2 + b_2b_3 + b_3b_4 - 4b_1b_2b_3. \tag{8}$$

If $b_1b_2b_3 = 0$, no assignment of our variables will we be able to reach a lower energy than if $b_1b_2b_3 = 1$. Hence this gives us 12 ELCs, and one example is $(b_1, b_2, b_3) = (1, 0, 0)$ which we can use to form the polynomial:

$$H_{2\text{-local}} = H_{3\text{-local}} + 4b_1(1 - b_2)(1 - b_3) \tag{9}$$

$$= b_1b_2 + b_2b_3 + b_3b_4 + 4b_1 - 4b_1b_2 - 4b_1b_3. \tag{10}$$

In both cases Eqs. (8) and (10), the only global minima occur when $b_1b_2b_3 = 1$.

Bibliography

- Original paper and application to computerized image denoising: [6].

C. Groebner Bases

Summary

Given a set of polynomials, a Groebner basis is another set of polynomials that have exactly the same zeros. The advantage of a Groebner basis is it has nicer algebraic properties than the original equations, in particular they tend to have smaller degree polynomials. The algorithms for calculating Groebner bases are generalizations of Euclid's algorithm for the polynomial greatest common divisor.

Work has been done in the field of 'Boolean Groebner bases', but while the variables are Boolean the coefficients of the functions are in $\mathbb{F}_2$ rather than $\mathbb{Q}$.

Cost

- 0 auxiliary variables needed.
- $\mathcal{O}\left(2^{2^n}\right)$ in general, $\mathcal{O}(d^{n^2})$ if the zeros of the equations form a set of discrete points, where d is the degree of the polynomial and n is the number of variables [14].

Pros

- No auxiliary variables needed.
- General method, which can be used for other rings, fields or types of variables.

Cons

- Best algorithms for finding Groebner bases scale double exponentially in n.
- Only works for Hamiltonians whose minimization corresponds to solving systems of discrete equations (RICHARD, why is this the ONLY case?).

Example

Consider the following pair of equations:

$$b_1b_2b_3b_4 + b_1b_3 + b_2b_4 - b_3 = b_1 + b_1b_2 + b_3 - 2 = 0. \tag{11}$$

Feeding these to Mathematica's `GroebnerBasis` function, along with the binarizing $b_1(b_1 - 1) = \ldots = b_4(b_4 - 1) = 0$ constraints, gives a Groebner basis:

$$\{b_4b_3 - b_4, b_2 + b_3 - 1, b_1 - 1\}. \tag{12}$$

From this we can immediately read off the solutions $b_1 = 1$, $b_2 = 1 - b_3$ and reduce the problem to $b_3b_4 - b_4 = 0$. Solving this gives a final solution set of: $(b_1, b_2, b_3, b_4) = (1,0,1,0), (1,0,1,1), (1,1,0,0)$.

Bibliography

- Reduction and embedding of factorizations of all bi-primes up to $200,000$: [15].

D. Split Reduction

Summary

It has been shown in [16] that, if multiple runs of a minimization algorithm is permitted, it is possible to reduce a lot of the problem by conditioning on the most connected variables. We call each of these operations a *split*.

Cost

Exponential in the number of splits, as the number of problems to solve doubles with every split.

Pros

- This method can be applied to any problem and can be very effective on problems with a few very connected variables.

Cons

- Exponential cost in the worst case.

Example

Consider the simple objective function

$$H = 1 + b_1b_2b_5 + b_1b_6b_7b_8 + b_3b_4b_8 - b_1b_3b_4. \tag{13}$$

In order to quadratize H, we first have to choose a variable to split over. In this case b_1 is the obvious choice since it is present in the most terms and contributes to the quartic term.

We then obtain two different problems:

$$H_0 = 1 + b_3b_4b_8 \tag{14}$$
$$H_1 = 1 + b_2b_5 + b_6b_7b_8 + b_3b_4b_8 - b_3b_4. \tag{15}$$

At this point, we could split H_0 again and solve it entirely, or use a qubit we saved in the previous split to quadratize our only problem.

To solve H_1, we can split again on b_8, resulting in problems:

$$H_{1,0} = 1 + b_2b_5 + b_6b_7 \tag{16}$$
$$H_{1,1} = 1 + b_2b_5 + b_3b_4. \tag{17}$$

Now both of these problems are quadratic. Hence we have reduced our original, hard problem into 3 easy problems, requiring only 2 extra runs of our minimization algorithm, and without needing any auxiliary variables.

Bibliography

- Original paper and application to Ramsey number calculation: [16].

III. METHODS THAT INTRODUCE AUXILIARY VARIABLES TO QUADRATIZE A SINGLE TERM

A. Negative Term Reduction

Summary

For a negative term $-b_1 b_2 ... b_k$, introduce a single auxiliary variable b_a and make the substitution:

$$-b_1 b_2 \dots b_k = \min_{b_a} \left((k - 1 - \sum_i b_i) b_a \right). \tag{18}$$

Cost

- 1 auxiliary variable for each k-local term.

Pros

- All resulting quadratic terms are submodular (have negative coefficients).
- Can reduce arbitrary order terms with only 1 auxiliary.

Cons

- Only works for negative terms.

Example

$$H_{6-\text{local}} = -2 b_1 b_2 b_3 b_4 b_5 b_6 + b_5 b_6, \tag{19}$$

has a unique minimum energy of -1 when all $b_i = 1$.

$$H_{2-\text{local}} = 2 \left(5 b_a - b_1 b_a - b_2 b_a - b_3 b_a - b_4 b_a - b_5 b_a - b_6 b_a \right) + b_5 b_6 \tag{20}$$

has the same unique minimum energy, and it occurs at the same place (all $b_i = 1$), with $b_a = 1$.

Bibliography

- Original paper: [17].
- Discussion: [5], [7].

B. Positive Term Reduction

Summary

By considering the negated literals $\bar{b}_i = 1 - b_i$, we recursively apply the previous method to $b_1 b_2 \dots b_k = -\bar{b}_1 b_2 \dots b_k + b_2 b_3 \dots b_k$. The final identity is:

$$b_1 b_2 \dots b_k = \min_{b_a} \left(\sum_{i=1}^{k-2} b_{a_i} \left(k - i - 1 + b_i - \sum_{j=i+1}^{k} b_j \right) \right) + b_{k-1} b_k \tag{21}$$

Cost

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

Pros

- Works for positive monomials.

Cons

- $k - 1$ non-submodular quadratic terms.

Example

$$b_1 b_2 b_3 b_4 = \min_{b_a} b_{a_1} (2 + b_1 - b_2 - b_3 - b_4) + b_{a_2} (1 + b_2 - b_3 - b_4) + b_3 b_4 \tag{22}$$

Bibliography

- Summary: [18].

C. Ishikawa's Symmetric Reduction (Positive Term Reduction)

Summary

This method rewrites a positive monomial using symmetric polynomials, so all possible quadratic terms are produced and they are all non-submodular:

$$b_1 \dots b_k = \min_{b_{a_1},\dots,b_{a_{n_k}}} \left(\sum_{i=1}^{n_k} b_{a_i} \left(c_{i,d} \left(-\sum_{j=1}^{k} b_j + 2i \right) - 1 \right) + \sum_{i<j} b_i b_j \right) \tag{23}$$

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
- $\mathcal{O}(kt)$ for a k-local Hamiltonian with t terms.

Pros

- Works for positive monomials. About half as many auxiliary variables for each k-order term as the previous method.

Cons

- $\mathcal{O}(k^2)$ quadratic terms are created, which may make chimerization more costly.
- $\frac{k(k-1)}{2}$ non-submodular terms.
- Worse than the previous method for quartics, with respect to submodularity.

Example

$$b_1 b_2 b_3 b_4 = \min_{b_a} (3 - 2b_1 - 2b_2 - 2b_3 - 2b_4) b_a + b_1 b_2 + b_1 b_3 + b_1 b_4 + b_2 b_3 + b_2 b_4 + b_3 b_4 \tag{24}$$

Bibliography

- Original paper and application to image denoising: [5].

D. Asymmetric Reduction

Summary

Similar to other methods of reducing one term, this method can reduce a positive cubic monomial into quadratic terms using only one auxiliary variable, while introducing fewer non-submodular terms than the symmetric version.

RICHARD: this is asymmetric w.r.t. b1,b1,b3, so isn't there more options for how to transform the b1b2b3 in this case?

The identity is given by:

$$b_1 b_2 b_3 = \min_{b_a} \left(b_a - b_2 b_a - b_3 b_a + b_1 b_a + b_2 b_3 \right) \tag{25}$$

Cost

1 auxiliary qubit per positive cubic term.

Pros

- Works on positive monomials.
- Fewer non-submodular terms than Ishikawa Reduction.

Cons

- Only been shown to work for cubics.

Bibliography

- Original paper and application to computer vision: [19].

E. Bit flipping

Summary

For any variable b, we can consider the negation $\bar{b} = 1 - b$. The process of exchanging b for $\bar{b}$ is called *flipping*. Using bit-flipping, an arbitrary function in n variables can be represented using at most $2^{(n-2)}(n-3)+1$ variables, though this is a gross overestimate.

Can be used in many different ways:

1. Flipping positive terms and using III A, recursively;

2. For $\alpha < 0$, we can reduce $\alpha\bar{b}_1\bar{b}_2...\bar{b}_k$ very efficiently to submodular form using III A. A generalized version exists for arbitrary combinations of flips in the monomial which makes reduction entirely submodular [5];

3. When we have quadratized we can minimize the number of non-submodular terms by flipping.

4. We can make use of both b_i and $\bar{b}_i$ in the same Hamiltonian by adding on a sufficiently large penalty term: $\lambda(b_i + \bar{b}_i - 1)^2 = \lambda(1 + 2b_i\bar{b}_i - b_i - \bar{b}_i)$. This is similar to the ideas in reduction by substitution or deduc-reduc. In this way, given a quadratic in n variables we can make sure it only has at most n nonsubmodular terms if we are willing to use the extra n negation variables as well (so we have $2n$ variables in total).

Cost

- None, as replacing b_i with it's negation $\bar{b}_i$ costs nothing except a trivial symbolic expansion.

Pros

- Cheap and effective way of improving submodularity.

- Can be used to combine terms in clever ways, making other methods more efficient.

Cons

- Unless the form of the Hamiltonian is known, spotting these 'factorizations' using negations is difficult.

- We need an auxiliary variable for each b_i for which we also want to use $\bar{b}_i$ in the same Hamiltonian.

Example

By bit-flipping b_2 and b_4, i.e. substituting $b_2 = 1 - \bar{b}_2$ and $b_4 = 1 - \bar{b}_4$, we see that:

$$H = 3b_1b_2 + b_2b_3 + 2b_1b_4 - 4b_2b_4 \tag{26}$$
$$= -3b_1\bar{b}_2 - \bar{b}_2b_3 - 2b_1\bar{b}_4 - \bar{b}_2\bar{b}_4 + 5b_1 + b_3 + 4\bar{b}_2 + 4\bar{b}_4 - 4. \tag{27}$$

The first expression is highly non-submodular while the second is entirely submodular.

Bibliography

- Original paper: [5].

IV. METHODS THAT INTRODUCE AUXILIARY VARIABLES TO QUADRATIZE MULTIPLE TERMS WITH THE SAME AUXILIARIES

A. Reduction by Substitution

Summary

Pick a variable pair (b_i, b_j) and substitute $b_i b_j$ with a new auxiliary variable $b_{a_{i,j}}$. Enforce equality in the ground states by adding some scalar multiple of $b_i b_j - 2b_i b_{a_{i,j}} - 2b_i b_{a_{i,j}} + 3b_{a_{i,j}}$ or similar.

Cost

- 1 auxiliary variable per reduction.

Pros

- Variable can be used across the entire Hamiltonian, reducing many terms at once.
- Simple.

Cons

- Inefficient for single terms as it introduces many auxiliary variables compared to Ishikawa reduction, for example.
- Introduces quadratic terms with large positive coefficients, making them highly non-submodular.
- Determining optimal substitutions is expensive.

Example

We pick the pair (b_1, b_2) and combine.

$$b_1 b_2 b_3 + b_1 b_2 b_4 \mapsto b_3 b_a + b_4 b_a + b_1 b_2 - 2b_1 b_a - 2b_1 b_a + 3b_a \tag{28}$$

Bibliography

- Original paper: [20]

B. FGBZ Reduction (Fix-Gruber-Boros-Zabih)

Summary

Here we consider a set C of variables which occur in multiple monomials throughout the Hamiltonian. Each application 'rips out' this common component from each term [21][18].

Let $\mathcal{H}$ be a set of monomials, where $C \subseteq H$ for each $H \in \mathcal{H}$ and each monomial H has a weight α_H. The algorithm comes in 2 parts: when all $\alpha_H > 0$ and when all $\alpha_H < 0$. Combining the 2 gives the final method:

1. $\alpha_H > 0$

$$\sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in H} b_j = \min_{b_a} \left(\sum_{H\in\mathcal{H}} \alpha_H \right) b_a \prod_{j\in C} b_j + \sum_{H\in\mathcal{H}} \alpha_H (1-b_a) \prod_{j\in H\setminus C} b_j \tag{29}$$

2. $\alpha_H < 0$

$$\sum_{H\in\mathcal{H}} \alpha_H \prod_{j\in H} b_j = \min_{b_a} \sum_{H\in\mathcal{H}} \alpha_H \left(1 - \prod_{j\in C} b_j - \prod_{j\in H\setminus C} b_j \right) b_a \tag{30}$$

Cost

- One auxiliary variable per application.
- In combination with III A, it can be used to make an algorithm which can reduce t positive monomials of degree d in n variables using $n + t(d-1)$ auxiliary variables in the worst case.

Pros

- Can reduce the connectivity of a Hamiltonian, as it breaks interactions between variables.

Cons

- $\alpha_H > 0$ method converts positive terms into negative ones of same order rather than reducing them, though these can then be reduced more easily.
- $\alpha_H < 0$ method only works for $|C| > 1$, and cannot quadratize cubic terms.

Example

First let $C = b_1$ and use the positive weight version:

$$\begin{aligned} b_1b_2b_3 + b_1b_2b_4 &\mapsto 2b_{a_1}b_1 + (1-b_{a_1})b_2b_3 + (1-b_{a_1})b_2b_4 && (31)\\ &= 2b_{a_1}b_1 + b_2b_3 + b_2b_4 - b_{a_1}b_2b_3 - b_{a_1}b_2b_4 && (32)\end{aligned}$$

now we can use III A:

$$\begin{aligned} -b_{a_1}b_2b_3 - b_{a_1}b_2b_4 &\mapsto 2b_{a_2} - b_{a_1}b_{a_2} - b_{a_2}b_2 - b_{a_2}b_3 + 2b_{a_2} - b_{a_1}b_{a_2} - b_{a_2}b_2 - b_{a_2}b_4 && (33)\\ &= 4b_{a_2} - 2b_{a_1}b_{a_2} - 2b_{a_2}b_2 - b_{a_2}b_3 - b_{a_2}b_4. && (34)\end{aligned}$$

Bibliography

- Original paper and application to image denoising: [21].

V. METHODS THAT REPRODUCE THE FULL SPECTRUM

These techniques will transform k-local functions to 2-local functions that not only have the same ground state as the k-local function, but also the entire input/output spectrum of the k-local functions will be preserved in the low-lying energy space of the corresponding 2-local function (which also has higher energy states due to the auxiliary variables added).

A. Recursive Order Reduction

Summary

For each k-local term an auxilliary variable and a quadratic penalty function to constrain the auxilliary variable, is used to reduce the term's order to $(k-1)$-local. This is repeated recursively until the term is 2-local.

Cost

- $k-2$ auxiliary variables to reduce each k-local terms.
- At most $k\,t$ for a k-local Hamiltonian of t terms.

Pros

- Simple itertive method conducive to automated reduction.
- Chain like structure means that long range connectivity not required.

Cons

- Not very symmetric with respect to variables.
- Usually formulated in terms of $b_i b_j$ interactions, therefore not conducive to specialized hardware design since hardware usually has native $z_i z_j$ interactions.

Example

Let us assume we want to reproduce the spectrum of $H_4 = b_1 b_2 b_3 b_4$, first note that

$$P(b_{a_k}, b_i, b_j) = 3b_{a_k} + b_i b_j - 2b_i b_{a_k} - 2b_j b_{a_k} \tag{35}$$

will yield an energy of $E = 0$ iff $b_{a_k} = b_i b_j$, and $E > 0$ otherwise. We first reduce the order of the polynomial by 1 by applying this penalty term with a large positive prefactor λ.

$$H_{3-\text{local}} = b_1 b_2 b_{a_1} + \lambda(3b_{a_1} - 2b_{a_1}(b_3 + b_4) + b_3 b_4) \tag{36}$$

Performing the same trick again yields a 2-local Hamiltonian:

$$H_{2-\text{local}} = b_1 b_{a_2} + \lambda\,(3\,b_{a_2} - 2b_{a_2}(b_2 + b_{a_1}) + b_2 b_{a,1}) + \lambda\,(3\,b_{a_1} - 2b_{a_1}(b_3 + b_4) + b_3 b_4) \tag{37}$$

Bibliography

- Possibly the original paper [ask Boros and Gruber for advice on where it came first]: [24].
- Used in: [10, 23].

B. Chained Three Body Parity Operators

Summary

Goal: Guarantee that $b_{a,k} = b_i b_j$ can be chained together to make large product terms consisting of b. The penalty term $P(b_{a_k}, b_i, z_j) = \mp b_{a_k} b_i b_j$ guarantees that $b_{a_k} = \pm b_i b_j$. The three local term can be made from gadgets.

This method was originally only used to reproduce the ground state of high locality terms, but states of the "wrong" parity ($q_k = \mp b_i b_j$) will all have the same energy as well, so it reproduces the full spectrum.

Cost

- The best known gadget for a three local Ising term uses one auxilliary qubit. Based on this gadget an $n \geq 4$ body Ising term can be made using $3 + 2(n-4)$ auxilliary qubits.

Pros

- Natural transmon implementation [26].
- Chain like structure means that long range connectivity not required.

Cons

- Does not preserve degeneracy, ground state will retain orginal degeneracy, but excited states will have degeneracy multipled by $n-3$
- Not very symmetric.

Example

Let us use this method to reproduce the spectrum of the following five local Hamiltonian $b_1 b_2 b_3 b_4 b_5$. The following three local gadget reproduces the spectrum of the three local term $\pm b_1 b_2 b_3$

$$P_{\pm}(b_1, b_2, b_3; \lambda) = \lambda \left(b_1 b_2 + b_2 b_3 + b_3 b_1 + 2b_a(b_1 + b_2 + b_3)\right) \mp (b_1 + b_2 + b_3 + 2b_a). \tag{38}$$

Using thee copies of this three local gadget as a building block, and using two additional auxilliary variables, the spectrum of the five local term $b_1 b_2 b_3 b_4 b_5$ can be reproduced by the following Hamiltonian

$$H_{2-\text{local}} = P_+(b_1, b_2, b_{a_1}; \lambda) + P_+(b_{a_1}, b_3, b_{a_2}; \lambda) + P_+(b_{a_2}, b_4, b_5; \lambda). \tag{39}$$

Bibliography

- Original proposal with transmon implementation: [26].

C. Symmetry Based Mappings

Summary

Auxilliary qubits can be made to "count" the number of logical qubits in the 1 configuration. By applying single qubit terms to the auxilliary qubits, the spectrum of *any* permutation symmetric Hamiltonian can be reproduced.

Cost

- For a k local coupler requires k auxilliary qubits.

Pros

- Natural flux qubit implementation [32].
- Single gadget can reproduce any permutation symmetric spectrum.
- High degree of symmetry means this method is natural for some kinds of quantum simulations [33].

Cons

- Requires coupling between all logical qubits and from all logical qubits to all auxilliary qubits.
- Requires single body terms of increasing strength as k is increased.

Example

A 4 qubit gadget guarantees that the number of auxillary bits in the 0 state is equal to the number of logical bits in the 1 state

$$H_{4-\text{count}} = \sum_{i=2}^{4}\sum_{j=1}^{i-1} z_i z_j - \frac{1}{2}\sum_{i=1}^{4} z_i + \sum_{i=1}^{4}\sum_{j=1}^{4} z_i z_{a_j} + \frac{1}{2}\left(5z_{a_1} + z_{a_2} - 3\,z_{a_3} - 7z_{a_4}\right). \tag{40}$$

To replicate the spectrum of $z_1 z_2 z_3 z_4$, we add

$$H_{4-\text{local}} = z_{a_1} - z_{a_2} + z_{a_3} - z_{a_4} + \lambda H_{4-\text{count}}, \tag{41}$$

where λ is a large number.

For the spectrum of $b_1 b_2 b_3 b_4$, we implement,

$$H_{4-\text{local}} = -\frac{1}{2} z_{a_4} + \lambda H_{4-\text{count}}. \tag{42}$$

Bibliography

- Paper on flux qubit implementation: [32]
- Paper on max-k-sat mapping: [34]
- Talk including use in quantum simulation: [33]

Part II

Hamiltonians quadratic in z and linear in x (Transverse Field Ising Hamiltonians)

The Ising Hamiltonian with a transverse field in the x direction is possible to implement in hardware:

$$H = \sum_i \left(\alpha_i^{(z)} z_i + \alpha_i^{(x)} x_i\right) + \sum_{ij} \left(\alpha_{ij}^{(zz)} z_i z_j\right). \tag{43}$$

D. ZZZ-TI-CBBK: Transvese Ising from ZZZ, by Cao, Babbush, Biamonte, and Kais (2015)

There is only one reduction in the literature for reducing a Hamiltonian term to the transverse Ising Hamiltonian, and it works on 3-local zzz terms, by introducing an auxiliary qubit with label a:

$$\alpha z_i z_j z_k \to \alpha^I + \alpha_i^z z_i + \alpha_j^z z_j + \alpha_k^z z_k + \alpha_a^z z_a + \alpha_a^x x_a + \alpha_{ia}^{zz} z_i z_a + \alpha_{ja}^{zz} z_j z_a + \alpha_{ka}^{zz} z_k z_a \tag{44}$$

$$\begin{aligned}
\alpha^I &= \tfrac{1}{2}\left(\Delta + \left(\tfrac{\alpha}{6}\right)^{2/5} \Delta^{3/5}\right) \\
\alpha_i^z &= -\tfrac{1}{2}\left(\left(\tfrac{7\alpha}{6} + \left(\tfrac{\alpha}{6}\right)^{3/5} \Delta^{2/5}\right) - \left(\tfrac{\alpha\Delta^4}{6}\right)^{1/5}\right) \\
\alpha_j^z &= \alpha_i^{(z)} \\
\alpha_k^z &= \alpha_i^{(z)} \\
\alpha_a^z &= \tfrac{1}{2}\left(\Delta - \left(\tfrac{\alpha}{6}\right)^{2/5} \Delta^{3/5}\right) \\
\alpha_a^x &= \left(\tfrac{\alpha\Delta^4}{6}\right)^{1/5} \\
\alpha_{ia}^{zz} &= -\tfrac{1}{2}\left(\left(\tfrac{7\alpha}{6} + \left(\tfrac{\alpha}{6}\right)^{3/5} \Delta^{2/5}\right) + \left(\tfrac{\alpha\Delta^4}{6}\right)^{1/5}\right) \\
\alpha_{ja}^{zz} &= \alpha_{ia}^{(zz)} \\
\alpha_{ka}^{zz} &= \alpha_{ja}^{(zz)}
\end{aligned}$$

Including all coefficients and factorizing, we get:

$$\alpha z_i z_j z_k \to \left(\Delta + \frac{\alpha\Delta^4}{6}^{1/5} (z_i + z_j + z_k)\right)\left(\frac{1 - z_a}{2}\right) + \frac{\alpha\Delta^4}{6}^{1/5} x_a \tag{45}$$

$$+ \left(\left(\frac{\alpha}{6}\right)^{2/5} \Delta^{3/5} - \left(\frac{7\alpha}{6} + \left(\frac{\alpha}{6}\right)^{3/5} \Delta^{2/5}\right)(z_i + z_j + z_k)\right)\left(\frac{1 + z_a}{2}\right) \tag{46}$$

The low-lying spectrum (eigenvalues ***and*** eigenvectors) of the right side of Eq. (44) will match those of the left side to within a spectral error of ϵ as long as $\Delta = \mathcal{O}\left(\epsilon^{-5}\right)$.

Cost

- 1 auxiliary qubit
- 8 auxiliary terms not proportional to $\mathbb{1}$.

Example

E. 1B1-TI: 1-by-1 gadget for $ZZ\ldots Z \rightarrow$ Transverse Ising (Present Work)

Cost

- $k-2$ auxiliary qubits
- auxiliary terms not proportional to $\mathbb{1}$.

Example

F. SD-TI: Sub-division gadget for $ZZ\ldots Z \rightarrow$ Transverse Ising (Present Work)

Cost

- $k/2$

Example

Part III

General Quantum Hamiltonians

The most general time-independent Hamiltonian for n qubits acted on only by Pauli operators is:

$$H = \alpha I + \sum_{i}^{n} \left(\alpha_i^{(z)} z_i + \alpha_i^{(x)} x_i + \alpha_i^{(y)} y_i \right) + \tag{47}$$

$$\sum_{ij}^{n} \left(a_{ij}^{(zz)} z_i z_j + a_{ij}^{(zx)} z_i x_j + a_{ij}^{(zy)} z_i y_j + a_{ij}^{(xz)} x_i z_j + a_{ij}^{(xx)} x_i x_j + a_{ij}^{(xy)} x_i y_j + a_{ij}^{(yz)} y_i z_j + a_{ij}^{(yx)} y_i x_j + a_{ij}^{(yy)} y_i y_j \right) + \tag{48}$$

$$\sum_{ijk}^{n} \left(a_{ijk}^{(zzz)} z_i z_j z_k + a_{ijk}^{(zzx)} z_i z_j x_k + a_{ijk}^{(zzy)} z_i z_j y_k + a_{ijk}^{(zxz)} z_i x_j z_k + \cdots + a_{ijk}^{(yyy)} y_i y_j y_k \right) + \tag{49}$$

$$\sum_{ijkl\cdots n}^{n} \left(a_{ijkl\cdots n}^{(zzzz\cdots z)} z_i z_j z_k z_l \cdots z_n + a_{ijkl\cdots n}^{(zzz\cdots x)} z_i z_j z_k \cdots x_n + a_{ijkl\cdots n}^{(zzz\cdots y)} z_i z_j z_k \cdots y_n + a_{ijkl\cdots n}^{(zz\cdots xz)} z_i z_j \cdots x_{n-1} z_n + \cdots + a_{ijkl\cdots n}^{(yyy\cdots y)} y_i y_j y_k \cdots y_n \right) \tag{50}$$

We wish to find a 2-local Hamiltonian of the form of Eq. (2) or Eq. (1) that is equivalent to Eq. (47) in terms of reproducing certain desired properties (e.g. same ground state(s), same ground eigenvalue(s), same full eigenspectrum, etc.) to within the desired precision.

VI. 1-BY-1 GADGETS

A 1B1 gadget allows k-local terms to be quadratized one step at a time, where at each step the term's order is reduced by at most one. In each step, a k-local term is reduced to $(k-1)$-local, contrary to SD (sub-division) gadgets which can reduce k-local terms to $(1/2)$-local in one step.

A. 1B1-KKR (Kempe, Kitaev, Regev, 2006)

Summary

Group all k-local terms together and express their sum as a sum of products of commuting matrices s_{ij}:

$$H_{k-\text{local}} = \sum_i \prod_j^k s_{ij} + H_{(k-1)-\text{local}}. \tag{51}$$

Define auxiliary qubits labeled by a_{ij} and make the transformation:

$$\sum_i \prod_{ij}^k s_{ij} \mapsto \frac{\Delta}{4} \left(9 - \sum_i \left(\sum_j z_{a_{ij}} \right)^2 - \sum_j z_{a_{ij}}^2 + \frac{1}{\sqrt[3]{6\Delta}} x_{a_{ij}} \right) + \Lambda. \tag{52}$$

The result will be a $(k-1)$-local Hamiltonian with the same low-lying spectrum as $H_{k-\text{local}}$ to within ϵ as long as $\Delta = \Theta(\epsilon^{-3})$.

Cost

- The number of auxiliary qubits needed is the number of s_{ij} matrices in Eq. (53).
- $\Delta = \Theta(\epsilon^{-3})$

B. 1B1-OT (Oliviera & Terhal, 2008)

Summary

We wish to reduce the k-local term:

$$H_{k-\text{local}} = \alpha \prod_j^k s_j. \tag{53}$$

Define one auxiliary qubit labeled by a and make the transformation:

$$H_{k-\text{local}} \to -\left(\frac{\alpha}{2}\right)^{1/3} \Delta^{2(1-r)} s_k \left(\frac{1-z_a}{2}\right) + \left(\frac{\alpha}{2}\right)^{1/3} \frac{\Delta^r}{\sqrt{2}} \left(s_{k-1} - s_{k-2}\right) x_a \tag{54}$$

$$+ \frac{1}{2}\left(\frac{\alpha}{2}\right)^{2/3} \left(\Delta^{r-1} s_{k-1} + \text{sgn}(\alpha)\sqrt{2}\Delta^{-1/4} \prod_j^{k-2} s_j\right)^2 + \frac{\alpha}{4}\left(1 + 2\text{sgn}^2\alpha\Delta^{3/2-2r}\right) s_k. \tag{55}$$

The result will be a $(k-1)$-local Hamiltonian with the same low-lying spectrum as $H_{k-\text{local}}$ to within ϵ as long as $\Delta = \Theta\left(\epsilon^{-3}\right)$.

Cost

- Only 1 auxiliary qubit.
- $\Delta = \Theta\left(\epsilon^{-3}\right)$

Example

Bibliography

C. 1B1-CBBK (Cao, Babbush, Biamonte, Kais, 2015)

Define one auxiliary qubit labeled by a and make the transformation:

$$H_{k-\text{local}} \to \left(\Delta + \left(\frac{\alpha}{2}\right)^{3/2} \Delta^{1/2} s_k\right)\left(\frac{1-z_a}{2}\right) \tag{56}$$

$$- \frac{\alpha^{2/3}}{2}\left(1+\text{sgn}^2\alpha\right)\left((2\alpha)^{2/3}\,\text{sgn}^2\alpha + \alpha^{1/3} s_k - \sqrt[3]{2}\Delta^{1/2}\right)\left(\frac{1+z_a}{2}\right) \tag{57}$$

$$+ \left(\frac{\alpha}{2}\right)^{1/3} \Delta^{3/4}\left(\prod_j^{k-2} s_j - \text{sgn}(\alpha) s_{k-1}\right) x_a + \text{sgn}(\alpha)\sqrt[3]{2}\alpha^{2/3}\left(\Delta^{1/2}+\Delta^{3/2}\right)\prod_j^{k-1} s_j. \tag{58}$$

The result is $(k-1)$-local and its low-lying spectrum is the same as that of $H_{k-\text{local}}$ when Δ is large enough.

VII. SUBDIVISION GADGETS

Instead of recursively reducing k-local to $(k-1)$-local one reduction at a time, we can reduce k-local terms to $(k/2)$-local terms directly for even k, or to $(k+1)/2$-local terms directly for odd k. Since when k is odd we can add an identity operator to the k-local term to make it even, we will assume in the following that k is even, in order to avoid having to write floor and ceiling functions.

A. SD-OT (Oliviera & Terhal, 2008)

Summary

For any k-local term, we can subdivide it into a product of two $(k/2)$ –local terms, with an auxiliary qubit labeled a:

$$H_{k-\text{local}} \to \Delta\frac{1-z_a}{2} + \sqrt{\frac{\Delta}{2}}\left(s_k - \prod_j^{k-1} s_j\right)x_a \tag{59}$$

Cost

- $\Delta = \frac{\|H_{(k-1)-\text{local}}+\|\epsilon^2}{2}$.
- $\Delta = \Theta\left(\epsilon^{-2}\right)$.

Pros

- only one qubit to reduce k to $\lceil k/2\rceil + 1$

Cons

- Only beneficial for $k \geq 5$.

Example

Bibliography

B. SD-CBBK (Cao, Babbush, Biamonte, Kais 2015)

Summary

For any k-local term, we can subdivide it into a product of two $(k/2)$ –local terms:

$$H_{k-\text{local}} = \alpha H_{1,(k/2)-\text{local}} H_{2,(k/2)-\text{local}} + H_{(k-1)-\text{local}}. \tag{60}$$

Define one qubit a and make the following Hamiltonian is $(k/2)$-local:

$$\Delta \frac{1-z_a}{2} + |\alpha| \frac{1+z_a}{2} + \sqrt{|\alpha|\Delta/2}\left(\text{sgn}(\alpha) H_{1,(k/2-\text{local})} - H_{2,(k/2-\text{local})}\right) x_a \tag{61}$$

The result is a $(k/2)$-local Hamiltonian with the same low-lying spectrum as $H_{k-\text{local}}$ for large enough Δ. The disadvantage is that Δ has to be larger. ***Cost***

- $\Delta \geq \left(\frac{2|\alpha|}{\epsilon} + 1\right)\left(|\alpha| + \epsilon + 22\|H_{(k-1)-\text{local}}\right)$

Pros

- only one qubit to reduce k to $\lceil k/2 \rceil + 1$

Cons

- Only beneficial for $k \geq 5$.

Example

Bibliography

C. Bravyi et al (2008)

Summary

Cost

Example

Bibliography

D. Duan et al (2011)

Summary

Cost

Example

Bibliography

E. BOA (Babbush, O'Gorman, Aspuru-Guzik, 2013)

Summary

Cost

Example

Bibliography

VIII. DIRECT REDUCTION

Here we do not reduce k by one order at a time (1B1 reduction) or by $k/2$ at a time (SD reduction), but we directly reduce k-local terms to 2-local terms.

A. DR-JF (Jordan & Fahri)

Summary

Cost

Example

Bibliography

IX. NON-PERTURBATIVE EMBEDDINGS

A. NP-SJ (Subasi & Jarzynski, 2016)

Summary

Cost

Example

Bibliography

Part IV

Appendix

X. NOTATION

The left Kronecker product by a 2×2 identity matrix is implied by Eqs. (2) and (1) in the following way for qubit operators $q \in \{z, x\}$, assuming $i < j$: Subasi & Jarzynski (2016)

$$q_i = \mathbf{1}^{\otimes i-1} q \mathbf{1}^{\otimes (N-i+1)} \tag{62}$$

$$q_i \bar{q}_j = \mathbf{1}^{\otimes i-1} q \mathbf{1}^{\otimes (j-1+i)} \bar{q} \mathbf{1}^{\otimes (N-j+1)}. \tag{63}$$

Thinking in this way, with this notation might take some time to get used to, but tens of thousands of people are comfortable thinking this way (including any undergraduate student after a 1-semester quantum computing course). You can now see that H is a $2^N \times 2^N$ matrix with elements that are complex numbers. "Minimizing the Hamiltonian" just means finding the eigenvector of this matrix with lowest eigenvalue. On a classical computer, the cost of finding this eigenvector is the cost of diagonalizing the matrix: $\mathcal{O}(2^{3N})$, but undergraduate level quantum mechanics tells us that any physical system's state (wavefunction, ψ_n) is an eigenvector of a Hamiltonian and the eigenvalue E_n is just the energy associated with being in the n^{th} energy level:

$$H\psi_n = E_n \psi_n. \tag{64}$$

Eq. 64 is just the time-independent version of the Schroedinger equation, which you have at least heard of. The diagonal elements are the energies of the n levels and the off-diagonals are associated with the propensity for tunnelling from level n to level m. Any 2^N level system can be represented by N spin-1/2 particles (**qubtis**), and an example of a spin-1/2 particle is simply an electron. Some of the N electrons will have spin up and some will have spin down, hence 2^N possibilities. So instead of $\mathcal{O}(2^{3N})$ operations on a classical computer for finding the eigenvector with lowest eigenvalue (and hence doing the completely arbitrary computation), we can just (for example) put N electrons together with the appropriate H describing their energy, and then cool the system down to its ground state. This ground state is one out of 2^N possible states , and the configuration of spin up and spin down electrons encodes the desired computation.

XI. FURTHER EXAMPLES

Example Here we show how deductions can arise naturally from the Ramsey number problem. Consider $\mathcal{R}(4,3)$ with $N = 4$ nodes. Consider a Hamiltonian:

$$H = (1 - z_{12})(1 - z_{13})(1 - z_{23}) + \ldots + (1 - z_{23})(1 - z_{24})(1 - z_{34}) + z_{12}z_{13}z_{14}z_{23}z_{24}z_{34}. \quad (65)$$

See [16] for full details of how we arrive at this Hamiltonian.

Since we are assuming we have no 3-independent sets, we know that $(1 - z_{12})(1 - z_{13})(1 - z_{23}) = 0$, so $z_{12}z_{13}z_{23} = z_{12}z_{13} + z_{12}z_{23} + z_{13}z_{23} - z_{12} - z_{13} - z_{23} + 1$. This will be our deduction.

Using deduc-reduc we can substitute this into our 6-local term to get:

$$H = 2(1 - z_{12})(1 - z_{13})(1 - z_{23}) + \ldots + (1 - z_{23})(1 - z_{24})(1 - z_{34}) + \quad (66)$$

$$z_{14}z_{24}z_{34}(z_{12}z_{13} + z_{12}z_{23} + z_{13}z_{23} - z_{12} - z_{13} - z_{23} + 1). \quad (67)$$

We could repeat this process to remove all 5- and 4-local terms without adding any auxiliary qubits. Note in this case the error terms added by deduc-reduc already appear in our Hamiltonian.

ACKNOWLEDGMENTS

[1] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, SIAM Review **50**, 755 (2008).

[2] J. D. Biamonte and P. J. Love, Physical Review A **78**, 012352 (2008).

[3] M. V. Altaisky, N. N. Zolnikova, N. E. Kaputkina, V. A. Krylov, Y. E. Lozovik, and N. S. Dattani, Applied Physics Letters **108**, 103108 (2016).

[4] H. Ishikawa, in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2009) pp. 2993–3000.

[5] H. Ishikawa, IEEE Transactions on Pattern Analysis and Machine Intelligence **33**, 1234 (2011).

[6] H. Ishikawa, in *2014 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2014) pp. 1362–1369.

[7] M. Anthony, E. Boros, Y. Crama, and A. Gruber, (2015).

[8] N. S. Dattani and N. Bryans, (2014), arXiv:1411.6758.

[9] F. Gaitan and L. Clark, (2012), arXiv:arXiv:1103.1345v3.

[10] Z. Bian, F. Chudak, W. G. Macready, L. Clark, and F. Gaitan, Physical Review Letters **111**, 130505 (2013).

[11] V. S. Denchev, S. Boixo, S. V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis, and H. Neven, Phys. Rev. X **6**, 31015 (2016).

[12] S. Mandrà, Z. Zhu, W. Wang, A. Perdomo-Ortiz, and H. G. Katzgraber, Phys. Rev. A **94**, 22337 (2016).

[13] R. Tanburn, E. Okada, and N. Dattani, (2015), arXiv:1508.04816.

[14] M. Bardet, Algorithms Seminar , 85 (2002).

[15] R. Dridi and H. Alghassi, (2016), arXiv:1604.05796.

[16] E. Okada, R. Tanburn, and N. S. Dattani, , 5 (2015), arXiv:1508.07190.

[17] D. Freedman and P. Drineas, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2 (IEEE, 2005) pp. 939–946.

[18] E. Boros and A. Gruber, (2014), arXiv:1404.6538.

[19] A. C. Gallagher, D. Batra, and D. Parikh, in *CVPR 2011* (IEEE, 2011) pp. 1857–1864.

[20] I. G. Rosenberg, Cahiers du Centre d'Etudes de Recherche Operationnelle **17**, 71 (1975).

[21] A. Fix, A. Gruber, E. Boros, and R. Zabih, in *2011 International Conference on Computer Vision* (IEEE, 2011) pp. 1020–1027.

[22] E. Boros and P. L. Hammer, Discrete Applied Mathematics **123**, 155 (2002).

[23] A. Perdomo, C. Truncik, I. Tubert-Brohman, G. Rose, and A. Aspuru-Guzik, Phys. Rev. A **78**, 12320 (2008).

[24] J. D. Biamonte, Phys. Rev. A **77**, 52331 (2008).

[25] G. De las Cuevas and T. S. Cubitt, Science **351**, 1180 (2016).

[26] M. Leib, P. Zoller, and W. Lechner, Quantum Science and Technology **1**, 15008 (2016).

[27] W. Lechner, P. Hauke, and P. Zoller, Science Advances **1** (2015), 10.1126/sciadv.1500838.

[28] F. Pastawski and J. Preskill, Phys. Rev. A **93**, 52325 (2016).

[29] T. Albash, W. Vinci, and D. A. Lidar, Phys. Rev. A **94**, 22327 (2016).

[30] A. W. Glaetzle, R. M. W. van Bijnen, P. Zoller, and W. Lechner, Nature Communications **8**, 15813 (2017).

[31] A. Rocchetto, S. C. Benjamin, and Y. Li, Science Advances **2** (2016), 10.1126/sciadv.1601246.

[32] N. Chancellor, S. Zohren, and P. A. Warburton, npj Quantum Information **3**, 21 (2017).

[33] N. Chancellor, "Max-k-SAT, Multi-Body Frustration, & Multi-Body Sampling on a Two Local Ising System," AQC 2016 https://www.youtube.com/watch?v=aC-6hg_h3EA (2016).

[34] N. Chancellor, S. Zohren, P. Warburton, S. Benjamin, and S. Roberts, Scientific Reports **6** (2016), 10.1038/srep37107.