Towards Large scale Quantum optimization

$$m = O(n^k)$$

where m: number of binary variables, n = number of qubits and $K>1$ is the tunable parameter.

The number of parameters and circuit depth display mid linear and sublinear scaling in 'm'.

Key word: <u>Qubit Efficient Encoding</u>

For $m = 7000$ (competitive results as compared to classical)
For $m = 2000$, $n = 17$ Max Cut approximation ratio $\underline{\underline{0.941}}$
(trapped-ion)

$m = O(n^k)$ ~ solving for binary optimization problems of size 'm' polynomially larger than the number of qubits (n) used.

Encoding happens : the m variables into Pauli Correlations across k qubits, ($K \in \Theta$ of our choice)

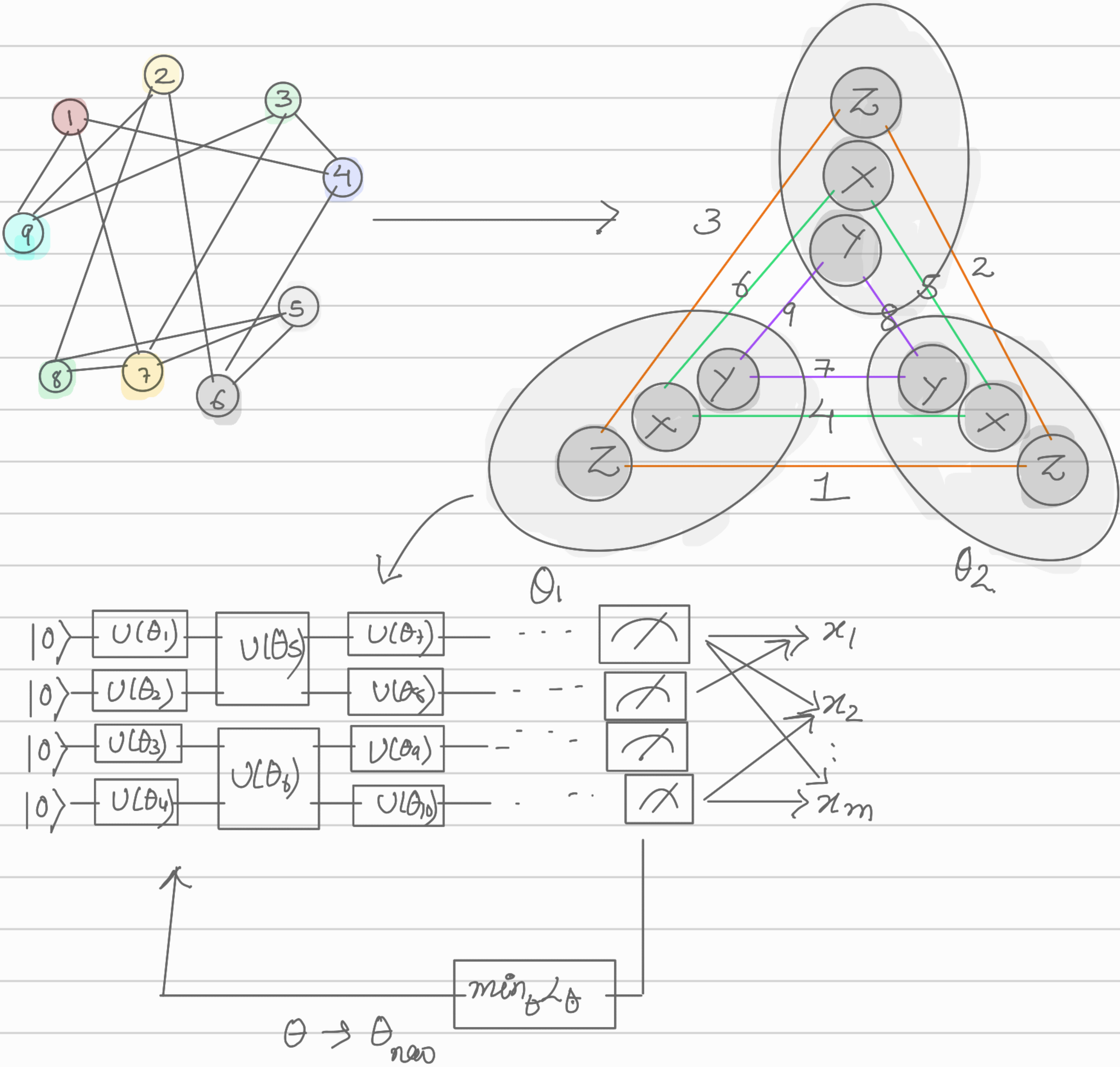
then a parameterized circuit is trained so that its output correlations minimizes a non-linear loss function suitable for gradient descent.

correlation

The co-relation between two operators A and B is often expressed as the expectation value of the product of A and B.

Eq: Pauli matrices $\sigma_1, \sigma_2, \sigma_3$ and state ρ

$$\mathrm{Corr}(\sigma_i, \sigma_j) = \mathrm{Tr}(\rho \cdot \sigma_i \cdot \sigma_j)$$



An exemplary MaxCut problem of $m = 9$ vertices is encoded into 2-body Pauli matrix correlation across $n=3$ qubits. Pauli string encodes the vertices. For example:

the binary variable 'x_1' of vertex 1 is encoded in the expectation values of $Z_1 \otimes Z_2 \otimes I_3$, supported on qubits 1 and 2, while x_9 is encoded in $Y_1 \otimes I_2 \otimes Y_3$. This corresponds to a quadratic space compression of m variables into

$n = O(m^{1/2})$ qubits

since here we have a two body correlation. ($Z_1 \otimes Z_2 \otimes I_3$)
but if we have an 'K' body correlation, we can attain polynomial compressions of order k.

$$n = O(m^{1/k}) \quad \text{or} \quad n = O(\sqrt[k]{m})$$

when two operators commute, it means that the order in which you measure them doesn't affect the outcome.

$$[A, B] = AB - BA = 0$$

or you can simultaneously measure the corresponding physical quantities with high precision. The order of measurement becomes irrelevant.

The Pauli set chosen is composed of three subsets of mutually commuting Pauli strings. This allows one to experimentally estimate all 'm' correlations using only 3 measurement settings throughout.

Pauli set $\in \{X, Y, Z, I\}$

mutually commuting Pauli strings: subsets of Pauli operators that commute with each other. In this context, three subsets of Pauli strings are chosen, and within each subset, all the Pauli string commute with each other.

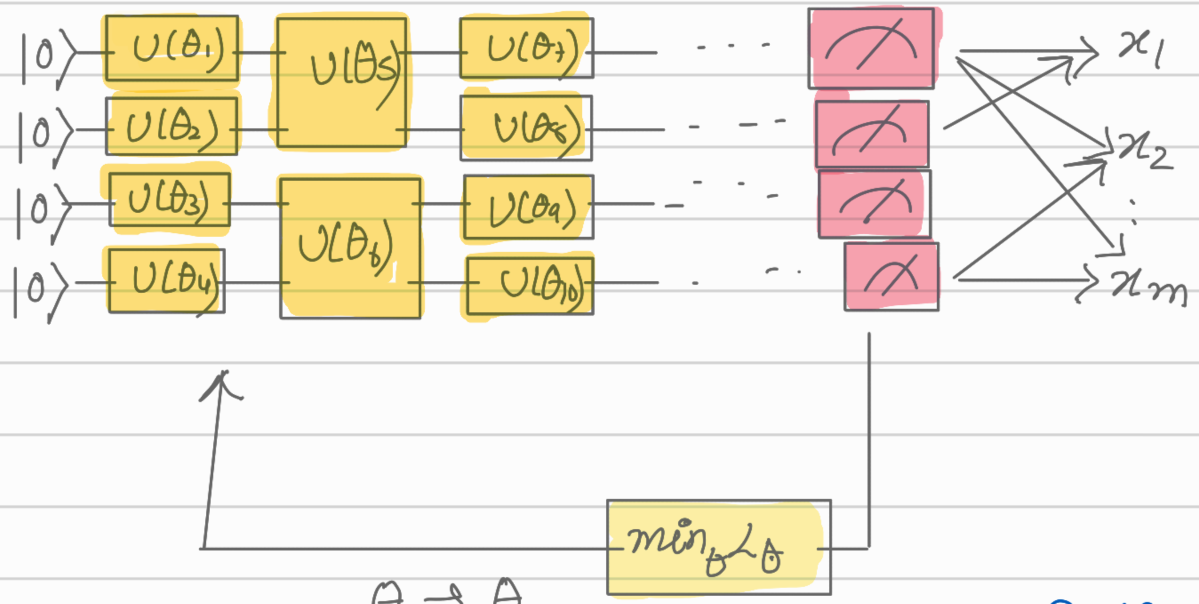
Eg: Sub 1 $\{\sigma_x, \sigma_y\}$ Sub 2: $\{\sigma_z\}$ Sub 3: $\{I\}$
Here each subset commutes within the subset, but may not commute with operators from other subsets.

Sub 1 allows correlation involving observables in X and Y subset 2 allows estimation of correlations in Z and so on.

By measuring these three subsets, one can experimentally estimate all the correlations of interest without the need for a larger number of distinct measurement settings.

It reduces the complexity of the experimental setup and potentially minimizes the errors associated with changing measurement configurations.

Quantum-Classical optimization
trained a parameterized Quantum circuit.



Loss function on the Pauli Expectation values.

solution: once the circuit is trained, we read out its output 'x' from the correlations across single-qubit measurement outcomes. Finally an efficient classical bit-swap search around x to find potential better solutions nearby. The result of that search, x^* is the final output of the solver.

Questions:

☐ How to encode? What Pauli basis to use? How it decides?

- [ ] what are we training the POC against? How to recover the output?
- [ ] what is a classical bit-swap search and how does it helps?

RESULTS!

Quantum solvers with Polynomial space compression

We solve combinatorial optimization problems over m binary variables, n qubits and $k \in \mathbb{Q}$ of our choice.

$$m = O(n^k)$$

Such a compression is achieved by encoding the variables into 'm' - Pauli Matrix correlations across multiple qubits.

$[m] = \{1, 2, \ldots, m\}$ Set of binary variables.

Let $x = \{x_i\}_{i \in [m]}$ $x = \{x_1, x_2, x_3 \ldots, x_m\}$

choose a specific subset $\Pi = \{\Pi_i\}_{i \in [m]}$ of $m \leq 4^n - 1$ ($2^{2n} - 1$) traceless Pauli strings Π_i

i.e of n-fold tensor products of identity ($\mathbb{I}$) or Pauli (X, Y and Z) matrices, excluding the n-qubit identity matrix $\mathbb{I}^{\otimes n}$

Pauli-correlation Encoding

$$x_i = \mathrm{sgn}(\langle \Pi_i \rangle) \quad \forall\, i \in [m]$$

sgn is the sign function and $\langle \Pi_i \rangle = \langle \psi | \Pi_i | \psi \rangle$. We focus on strings with k single qubit traceless Pauli matrices. In particul

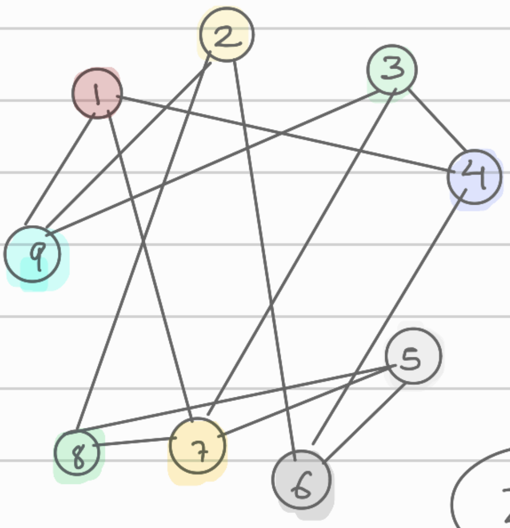
$\Pi^{(k)} = \{\Pi_1^{(k)}, \ldots, \Pi_m^{(k)}\}$ where each $\Pi_i^{(k)}$ is a permutation of either

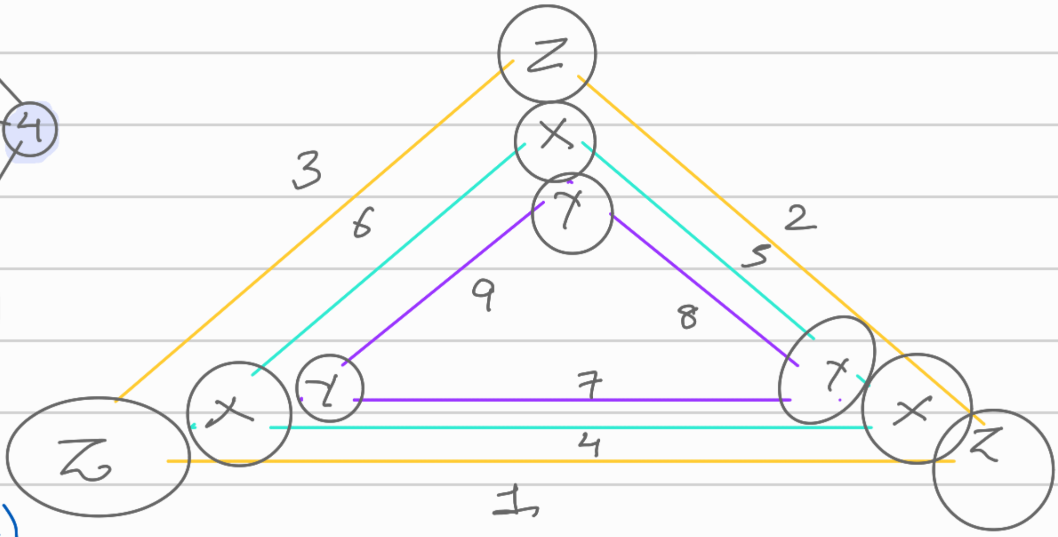
$$X^{\otimes k} \otimes I^{n-k}, \; Y^{\otimes k} \otimes I^{n-k} \text{ or } Z^{\otimes k} \otimes I^{n-k}$$

For eg.: for two body Pauli Matrix correlation

$k = 2$ and $m = 9$

$$\Pi^{(2)} = \{\Pi_1^{(2)}, \ldots, \Pi_m^{(2)}\}$$





$m = 9$ (binary variable)

$$\Pi^{(2)} = \{\Pi_1^{(2)}, \Pi_2^{(2)}, \Pi_3^{(2)}, \ldots \Pi_9^{(2)}\}$$

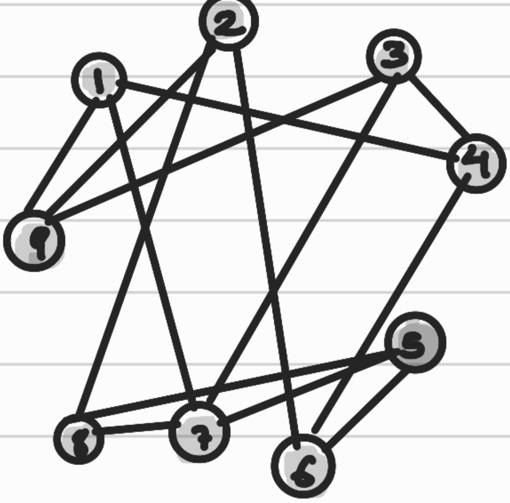
$$\Pi_1^{(2)} = Z^{\otimes 2} \otimes I^{3-2} \qquad Z_1 \otimes Z_2 \otimes I_3$$

$$\Pi_2^{(2)} = I^{3-2} \otimes Z^{\otimes 2} \qquad Z_2 \otimes Z_3 \otimes I_1$$

and so on.

How encoding is done?

Encode the 'm' variables into Pauli correlations across k qubits, for 'k' an integer of our choice. ($m = O(n^k)$)



$m = 9$ here

$n = 3$ qubits

$k = 2$, body cor.

2-body Pauli correlations

$1 = Z_1 \otimes Z_2 \otimes I_3$ $\quad q_1$ and q_2

and so on

$3 = Z_1 \otimes I_2 \otimes Z_3$ $\quad q_1$ and q_3

$8 = I_1 \otimes Y_2 \otimes Y_3$ $\quad q_2$ and q_3

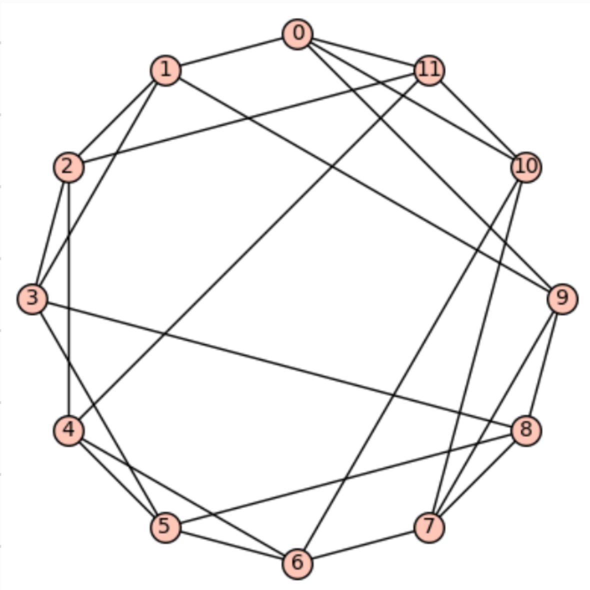
This corresponds to a quadratic space compression of 'm' variables into $n = O(m^{1/2})$ qubits

$m = O(n^k)$

and if we have k body correlation

$n = O(m^{1/k})$

For eg: 12 vertices



what does a 3-body correlation looks like?

$(3)^2 = 9$ variables

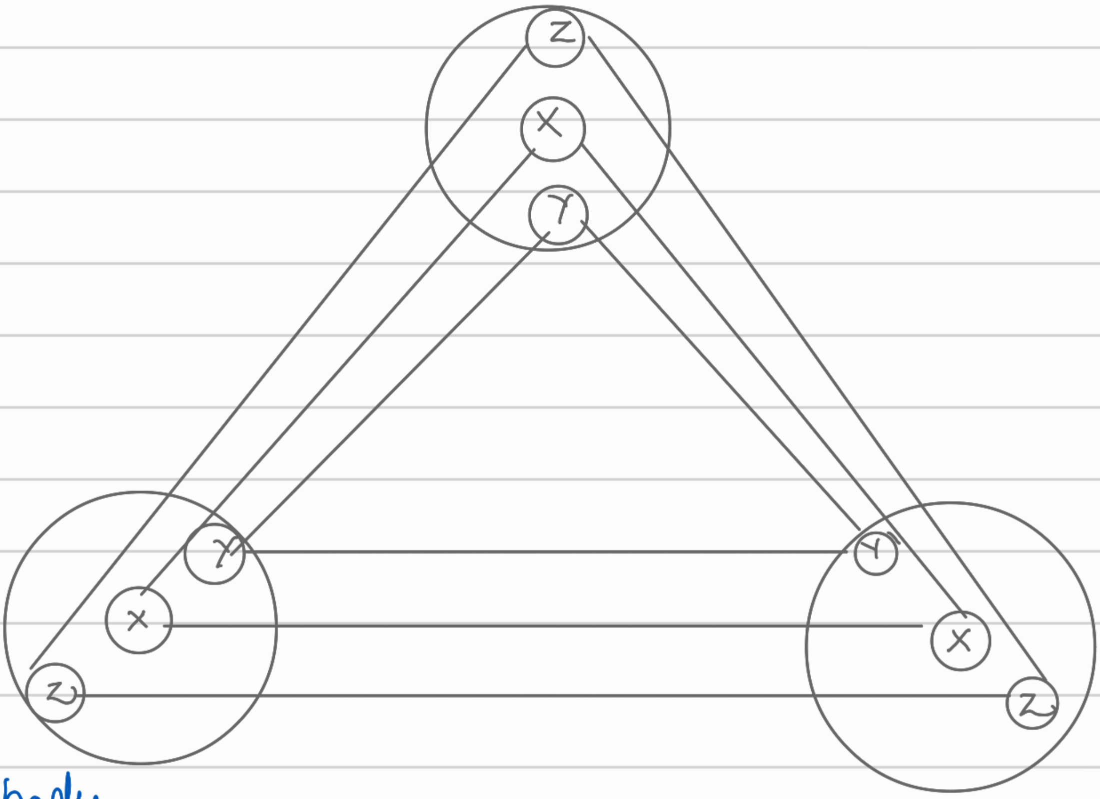
$(3)^3 = 27$ variables

but

how to achive 3 body corr?

Quantum solvers with Polynomial space Compression

Compression is achieved by encoding the variables into m Pauli-matrix correlations across multiple qubits



3-body

$$\left.\begin{matrix} Z_1 \otimes Z_2 \otimes Z_3 \\ Z_2 \otimes Z_3 \otimes Z_1 \\ Z_3 \otimes Z_1 \otimes Z_2 \end{matrix}\right]3 \qquad \left.\begin{matrix} X_1 \otimes X_2 \otimes X_3 \\ X_2 \otimes X_3 \otimes X_1 \\ X_3 \otimes X_1 \otimes X_2 \end{matrix}\right]3 \quad \text{||ly } \left. Y \right]3$$

$$\underbrace{\qquad\qquad\qquad\qquad}_{9}$$

$[m] := \{1, 2, \ldots, m\}$

$\downarrow$

Classical binary variables.

Let $x := \{x_i\}_{i \in [m]}$ denote the string of optimization variables

$$\boxed{m = 9}$$

$$x = \{x_1, x_2, \ldots, x_9\}$$

and then choose a specific subset $\Pi = \{\Pi_i\}_{i \in [m]}$ of $m \leq 4^n - 1$ traceless Pauli string Π_i

$m \leq 4^3 - 1 \qquad m \leq 63$ traceless Pauli string

$x_i := \mathrm{sgn}(\langle \Pi_i \rangle) \quad \forall i \in [m]$

$X^{\otimes k} \otimes \mathbb{I}^{\otimes n-k} \qquad Y^{\otimes k} \otimes \mathbb{I}^{\otimes n-k} \qquad Z^{\otimes k} \otimes \mathbb{I}^{\otimes n-k}.$

This is experimentally convenient, since only three measurement settings are required throughout.

Usually all possible permutation for the encoding yields

$$m = 3\binom{n}{k}$$

For

$n = 3$ qubit $\quad k = 2$

$$m \Rightarrow 3\binom{3}{2} \Rightarrow 3\left(\frac{3!}{2!\,1!}\right)$$

$$m = \frac{3}{2} n(n-1) \qquad \boxed{m = 9}$$

$n = 4$ qubit $\quad k = 2$

$$m = 3\binom{4}{2} \Rightarrow 3\left(\frac{4!}{2!\,2!}\right) = \boxed{18}$$

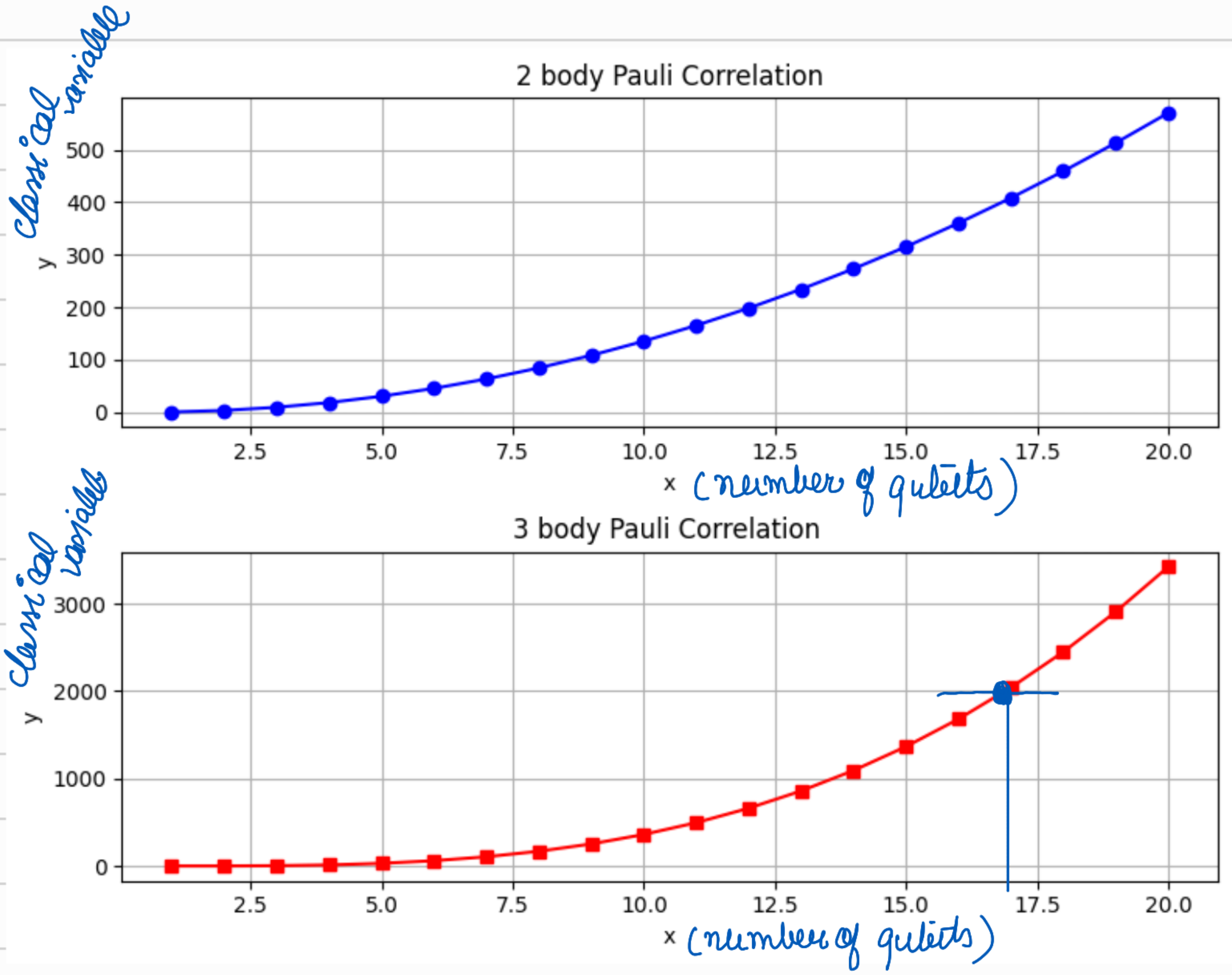
$$m = \frac{1}{2} n(n-1)(n-2) \qquad \boxed{m = 18}$$

$n = 4$ qubit $\quad k = 3$

$$m = 3\binom{4}{3} \Rightarrow 3\left(\frac{4!}{3!\,1!}\right) = \boxed{12}$$

Mostly deal with quadratic and cubic compression
$k = 2$ $\qquad$ $k = 3.$

$m = \frac{3}{2} n(n-1) \qquad m = \frac{1}{2}(n(n-1)(n-2))$



Using 2-body and 3-body Pauli Correlations can help us run 500 - 3000 binary variables long problems with using 20 qubits at max.

The goal of the parameter optimization is to minimize the non linear loss function

$$\mathcal{L} = \sum_{(i,j)} W_{ij} \tanh(\alpha \langle \Pi_i \rangle) \tanh(\alpha \langle \Pi_j \rangle) + \mathcal{L}^{reg.}$$

$$x_i = \mathrm{sgn}(\langle \Pi_i \rangle)$$

where $\langle \Pi_i \rangle = \langle \Psi | \Pi_i | \Psi \rangle$

---

we solve combinatorial optimizations over $m = O(n^k)$ binary variables using only n qubits, for k a suitable integer of our choice. most probably 2 or 3.

Such a compression is achieved by encoding the variables into m- Pauli matrix correlations.

X_1X_2 X_2X_3 X_3X_1 ~ 9 Pauli Correlations
Z_1Z_2 Z_2Z_3 Z_3Z_1 ~ 9 binary variables
Y_1Y_2 Y_2Y_3 Y_3Y_1 ~ 3 qubits

kind of like $(3,1,p)$ but better.

set of integers from 1 to m

$[m] := \{1, 2, \ldots m\}$ like here $[m] := \{1, 2, \ldots 9\}$

Let

$x := \{x_i\}_{i \in [m]}$ $x := \{x_1, x_2, \ldots x_9\}$
string of optimization variables.

$\Pi := \{\Pi_i\}_{i \in [m]}$ of $m \leq 4^n - 1$ traceless Pauli strin
$m \leq 4^3 - 1$ $m \leq 63$

$\{\Pi_1, \Pi_2, \ldots \Pi_9\}$.

Quantum Solvers with Polynomial Space Compression

$m = O(n^k)$
n - number of qubits
m - binary classical variables (Pauli matrix correlation)
k - number of correlations

$[m] := \{1, 2, 3, \ldots, m\}$

$x := \{x_i\}_{i \in [m]}$ denote the string of optimization var.

$\Pi := \{\Pi_i\}_{i \in [m]}$ of $m \leq 4^n - 1$ traceless Pauli string

Pauli Correlation Encoding

$$x_i := \text{sgn}\left(\langle \Pi_i \rangle\right) \quad \forall \ i \in [m] \quad (1)$$

$\langle \Pi_i \rangle = \langle \psi | \Pi_i | \psi \rangle$ Expectation value of that Pauli string over quantum state $|\psi\rangle$

Sufficient conditions for the Encoding

Deriving sufficient conditions on the magnitudes of Pauli string correlators for encoding arbitrary bit strings into valid quantum states as per (1)

For an arbitrary bit string x, we define

$$\rho_i = \frac{\mathbb{1} + x_i \Pi_i}{2^n} \qquad \text{Tr}[\rho_i \Pi_j] = x_j \delta_{ij}$$

x_i is the i^{th} bit of x.

ρ_i is diagonal in the eigenbasis of Π_i, with eigenvalues

$$\frac{(\mathbb{1} \pm x_i)}{2^n} \geq 0$$

$$\rho = \frac{1}{m} \sum_{i=1}^{m} \rho_i = \frac{\mathbb{1}}{2^n} + \frac{1}{2^n} \sum_{i=1}^{m} \frac{x_i}{m} \Pi_i$$

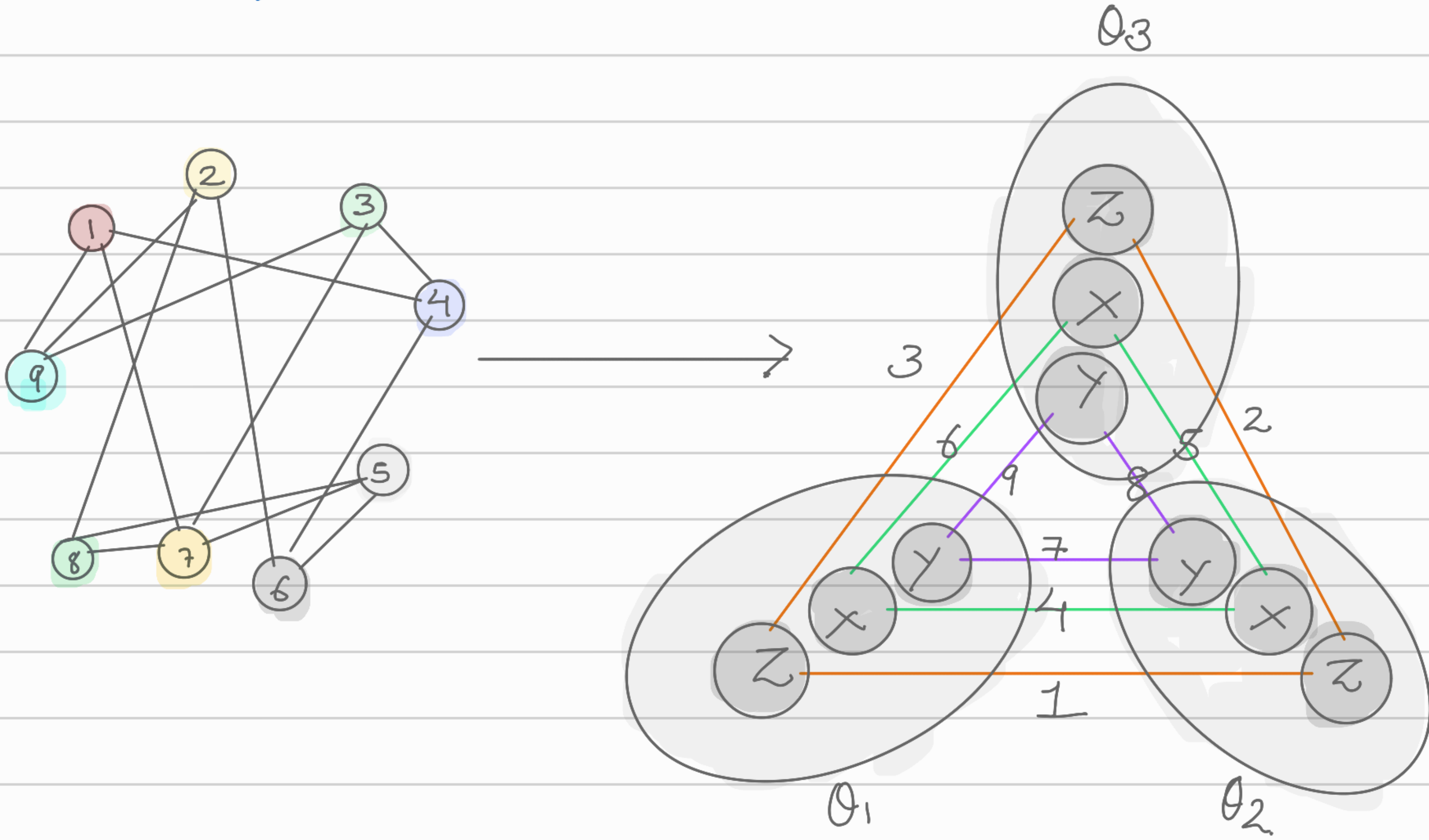
Convex combination of positive semi-definite matrices, it is also positive semi-definite. Moreover it satisfies

$$\text{Tr}[\rho \Pi_i] = \frac{x_i}{m} \qquad \Theta\left(\frac{1}{m}\right)$$

We focus on strings with k single-qubit traceless Pauli we focus on strings with k single qubit Pauli matrices. In particular we consider

$$\Pi^{(k)} := \{ \Pi_1^{(k)}, \dots, \Pi_m^{(k)} \}$$ where each $\Pi_i^{(k)}$ is a

permutation of either $X^{\otimes k} \otimes \mathbb{1}^{\otimes n-k}$, $Y^{\otimes k} \otimes \mathbb{1}^{\otimes n-k}$ or $Z^{\otimes k} \mathbb{1}^{\otimes n-k}$



Like in this figure we have

$Z_1 \otimes Z_2 \otimes \mathbb{1}_3$ $\qquad$ $\mathbb{1}_1 \otimes Z_2 \otimes Z_3$ and similarly for X and Y.

($n=3$ $k=2$; in this case)

$\Pi^{(k)}$ is the union of 3 sets of mutually-commuting strings.

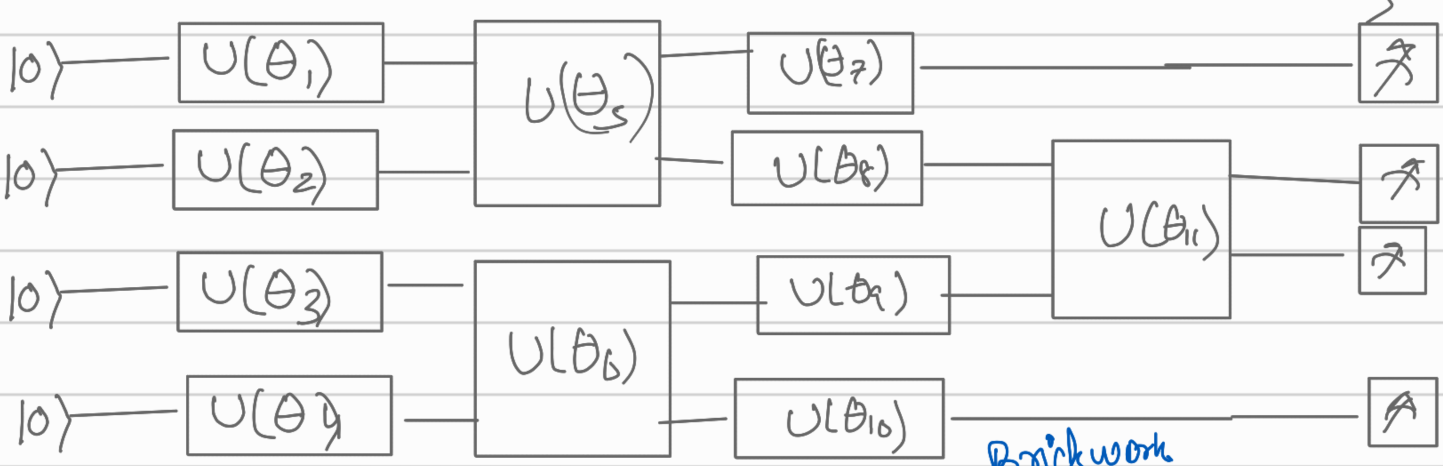
Using all possible permutations of the encoding yields,

$$m = 3\binom{n}{k}$$

We'll focus on $k=2$ and $k=3$

$m = \frac{3}{2} n(n-1)$ $\qquad$ $m = \frac{1}{2} n(n-1)(n-2)$

$x_i := \mathrm{sgn}(\langle \psi | \Pi_i | \psi \rangle) \quad \forall\, i \in [m]$



Brickwork

We parameterize the state in Eq (1) as the output of a quantum circuit with parameters θ.

$$\mathcal{L} = \sum W_{ij} \tanh(\alpha \langle \Pi_i \rangle) \tanh(\alpha \langle \Pi_j \rangle) + \mathcal{L}^{(reg)}$$

THE ALGORITHM

$m = O(n^k)$

$m \to$ number of binary variables

$n \to$ number of qubits

$k \to$ 2 or 3

$[m] := \{1, 2, \ldots, m\}$

number of binary variables

number of Pauli-Matrix correlations.

$x := \{x_i\}_{i \in [m]}$ denote the string of optimization variable.

$x := \{x_0, x_1, x_2, \ldots, x_m\}$

choose a specific subset

$\Pi := \{\Pi_i\}_{i \in [m]}$ of $m \le 4^n - 1$ traceless Pauli String Π_i

$\Pi = \{\Pi_1, \Pi_2, \dots, \Pi_m\}$

for 3 qubits
$m \le 4^3 - 1 \qquad m \le 63$

for 2 qubit
$m \le 4^2 - 1 \qquad m \le 15$

Hybrid quantum-classical solver for binary optimization problems of size 'm' polynomially larger than the number of qubits 'n' used.

A parameterized quantum circuit is trained so that its output correlations minimize a non-linear loss function suitable for gradient descent. Then a simple classical post-processing is used to obtain the solution bit string. (bit-swap)

$m = O(n^k)$

We define Pauli-correlation encoding (PCE) relative to Π as

$$x_i = \mathrm{sgn}(\langle \Pi_i \rangle) \quad \forall\, i \in [m]$$

m = number of variables (classical) and/or number of Pauli correlation

For each qubit there are 4 possibilities I, X, Y and Z. Since we have 'n'-qubits, the total number of Pauli correlations is 4^n but now since we are excluding the n-qubit identity matrix $\mathbb{1}^{\otimes n}$. We are left with $m \le 4^n - 1$ traceless Pauli strings

Eg: $\qquad n = 2$ qubits

	$\mathbb{I}$	X_2	Y_2	Z_2
$\mathbb{I}$	NOT ALLOWED	$X_2 \otimes \mathbb{I}$	$Y_2 \otimes \mathbb{I}$	$Z_2 \otimes \mathbb{I}$
X_1	$X_1 \otimes \mathbb{I}$	$X_1 \otimes X_2$	$X_1 \otimes Y_2$	$X_1 \otimes Z_2$
Y_1	$Y_1 \otimes \mathbb{I}$	$Y_1 \otimes Y_2$	$Y_1 \otimes Y_2$	$Y_1 \otimes Z_2$
Z_1	$Z_1 \otimes \mathbb{I}$	$Z_1 \otimes X_2$	$Z_1 \otimes Y_2$	$Z_1 \otimes Z_2$

15 elements

Π_i

similarly for $n = 3$ qubits, there will be 63 possible Pauli correlation $\boxed{m \leq 4^{n-1}}$

Pauli Correlation Encoding (PCE)

$$x_i := \text{sgn}(\langle \Pi_i \rangle) \text{ for all } i \in [m]$$

Eg:

$$x_1 = \text{sgn}(\langle \Pi_i \rangle) \Rightarrow \text{sgn}(\langle \psi | \Pi_i | \psi \rangle)$$

$$\Rightarrow \text{sgn}(\langle \psi | X_1 \otimes \mathbb{I} | \psi \rangle)$$

and

$$\boxed{\begin{aligned} \text{sgn}(-ve) &= -1 \\ \text{sgn}(+ve) &= +1 \end{aligned}}$$

How to code?

Given binary variables

$$x := \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$$

We the encode them in 3 qubits

$x_1 = Z_1 \otimes Z_2 \otimes \mathbb{I}$	$x_4 = X_1 \otimes X_2 \otimes \mathbb{I}$	$x_7 = Y_1 \otimes Y_2 \otimes \mathbb{I}$
$x_2 = \mathbb{I} \otimes Z_2 \otimes Z_3$	$x_5 = \mathbb{I} \otimes X_2 \otimes X_3$	$x_8 = \mathbb{I} \otimes Y_2 \otimes Y_3$
$x_3 = Z_1 \otimes \mathbb{I} \otimes Z_3$	$x_6 = X_1 \otimes \mathbb{I} \otimes X_3$	$x_9 = Y_1 \otimes \mathbb{I} \times Y_3$

circuit creation will be the same.

function should depend on
i) number of binary variables
ii) number of qubits

for the time being we'll only focus on $\boxed{k = 2,3}$ they are good enough.

Global MS gate

for 2-qubit case it is reduce to Rxx gate.

or $x = \{x_0, x_1, x_2, x_3\}$

in 2 qubits

$x_0 = Z_1 \otimes Z_2 \qquad x_2 = Y_1 \otimes Y_2$

$x_1 = X_1 \otimes X_2 \qquad x_3 = Z_1 \otimes \mathbb{I}$

$x_i = \text{sgn}(\langle \Pi_i \rangle) \quad \forall i \in [m]$

$$\langle \Pi_i \rangle = \langle \underbrace{\psi_i}_{\text{Quantum circuit}} | \underbrace{\Pi_i}_{\text{Operator}} | \psi_i \rangle$$

Problem of interest is a Max-Cut

$$\max \sum_{i,j} w_{ij} (1 - x_i x_j) \qquad \text{weighted max cut}$$

$$\mathcal{L} = \sum w_{ij} \tanh(\alpha \langle \Pi_i \rangle) \tanh(\alpha \langle \Pi_j \rangle) + \mathcal{L}^{(reg)}$$

$$L^{(reg)} = \beta \nu \left[\frac{1}{m} \sum_{i \in V} \tanh(\alpha \langle \Pi_i \rangle)^2 \right]^2$$

CODE

a) How to generate the Hamiltonian?

i) How to decide on number of qubits?

$m = (n)^k$

$9 = (3)^2$

number of binary variables = $\underline{\text{num\_vars}}$

ask for the compression

$$\text{num\_vars} = (n)^{k}$$

ii) Number of Pauli correlation

or $x = \{x_0, x_1, x_2, x_3\}$ in 2 qubits

for 4-vertice max cut this is

$x_0 = Z_1 \otimes Z_2$ $\quad x_2 = Y_1 \otimes Y_2$

$x_1 = X_1 \otimes X_2$ $\quad x_3 = Z_1 \otimes \mathbb{I}$

sparse Pauli op.

$$-2x_0x_1 - 2x_0x_2 - 2x_0x_3 - 2x_1x_2 - 2x_2x_3 + 3x_0 + 2x_1 + 3x_2 + 2x_3$$

$-2[Z_1 \otimes Z_2] - 2[- - - \cdot\cdot \quad]$ $\boxed{\text{end}}$

compute Expectation

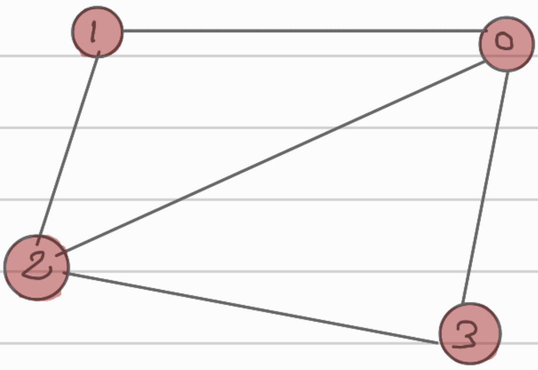
minimize on the loss function defined.

$\langle\psi|H|\psi\rangle \Rightarrow \langle\psi| Z_1\otimes Z_2 + X_1\otimes X_2 + Y_1\otimes Y_2 + Z_1\otimes I|\psi\rangle$

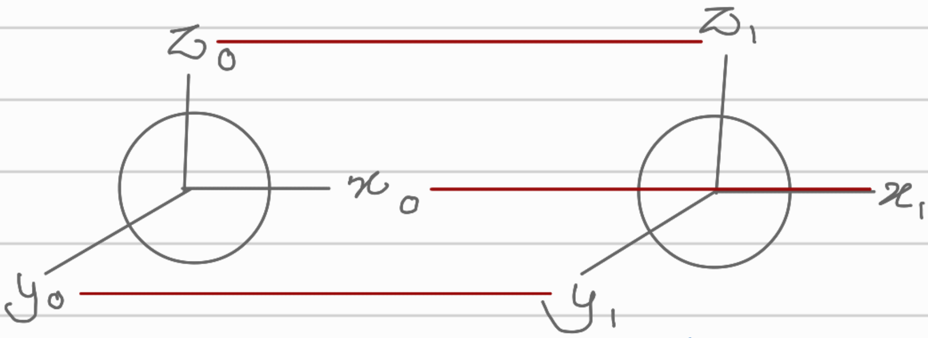
$$\boxed{\langle\psi|Z_1\otimes Z_2|\psi\rangle + \cdots \langle\psi|Z_1\otimes I|\psi\rangle}$$

But then what next?

$-2x_0x_1 - 2x_0x_2 - 2x_0x_3 - 2x_1x_2$
$- 2x_2x_3 + 3x_0 + 2x_1 + 3x_2$
$+ 2x_3$



$x_0 \quad x_1 \quad x_2 \quad x_3$



because we specified $k=2$

$x_0 : Z_0 Z_1$ $\qquad$ $x_2 : Y_0 Y_1$

$x_1 : X_0 X_1$ $\qquad$ $x_3 : Z_0 I_1$

2 body Pauli correlation

$m = n^k$

$m = 2^2$

$m = 4$

Eg: $n = 2$ qubits

	I	X_2	Y_2	Z_2
I	NOT ALLOWED	$X_2\otimes I$	$Y_2\otimes I$	$Z_2\otimes I$
X_1	$X_1\otimes I$	$X_1\otimes X_2$	$X_1\otimes Y_2$	$X_1\otimes Z_2$
Y_1	$Y_1\otimes I$	$Y_1\otimes Y_2$	$Y_1\otimes Y_2$	$Y_1\otimes Z_2$
Z_1	$Z_1\otimes I$	$Z_1\otimes X_2$	$Z_1\otimes Y_2$	$Z_1\otimes Z_2$

15 elements

T_i: $m \le 4^n - 1$

$m \le 4^2 - 1$

$m \le 15$

limit

look for an efficient way to construct the Hamiltonian,
depends on ~ number of qubits

number of binary variables

LOGIC BEHIND THE HAMILTONIAN GENERATION

$[x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8] = [XXI, XIX, IXX, YYI, YIY, IYY, ZZI, ZIZ, IZZ]$

LP

$-2x_0x_3 - 2x_0x_6 - 2x_0x_8 - 2x_1x_5 - 2x_1x_7 - 2x_1x_8 - 2x_2x_3 - \cdots$

$x_0 \cdot x_3 = XXI \cdot YYI \Rightarrow iZiZI \Rightarrow i^2 ZZI \Rightarrow -ZZI$

$$\boxed{\begin{array}{ll} XY = iZ & XZ = -iY \\ YX = -iZ & YZ = iX \\ ZX = iY & ZY = -iX \end{array}}$$

CODING THE LOSS FUNCTION

$$L^{reg} = \beta\nu\left[\frac{1}{m}\sum_{i \in V} \tanh(\alpha\langle \Pi_i \rangle)^2\right]^2$$

$\langle \Pi_i \rangle = \langle \psi | \Pi_i | \psi \rangle$ individual

what should be the weights?

$\tanh(\alpha \cdot \langle \psi | \Pi_1 | \psi \rangle) + \tanh(\alpha \cdot \langle \psi | \Pi_2 | \psi \rangle) + \cdots$

$\boxed{\text{weights } X}$

$$L = \sum_{(i,j \in E)} w_{ij} \tanh(\alpha\langle \Pi_i \rangle)\tanh(\alpha\langle \Pi_j \rangle) + L^{reg}$$

$$\sum_{i,j} w_{ij} \tanh(\alpha\langle\Pi_i\rangle)\tanh(\alpha\langle\Pi_j\rangle) + \boxed{\beta\nu\left[\frac{1}{m}\sum_{i\in V}\tanh(\alpha\langle\Pi_i\rangle)^2\right]}$$

which weights should be used?

single term!

FULL WORKFLOW

OPTIMIZE FUNCTION!
(params)

LOCAL BIT SWAP SEARCH TO LOOK FOR BETTER SOLUTIONS!

CHOICE OF LOSS FUNCTION

$\mathcal{L}$ leverages two main features, the non-linearities from the hyperbolic tangents and regularization term $\mathcal{L}^{(reg)}$ which forces all correlators to have small values.

$$\mathcal{L}^{qua} = \sum w_{ij} \langle\Pi_i\rangle\langle\Pi_j\rangle$$

$$\mathcal{L}^{qua} + \mathcal{L}^{(reg)}$$

EVERY QUBO TO A MAXCUT!