Solving the MAX-cut Problem using QAOA

Armin Ahmadkhaniha
Computer Science PhD program

November 20, 2024

Outline

1. Prerequisite
2. MAX-cut as QUBO
3. What is the ansatz?
4. Codes
5. References

Quadratic Optimization Problem

- Optimization problems with quadratic objective function and linear and quadratic constraints

minimize $x^T Qx + c^T x$

$$Q \in \mathbb{R}^{n \times n}, \quad c \in \mathbb{R}^n$$

subject to

$Ax \leq b$

$x^T Q_i x + a_i^T x \leq r_i$

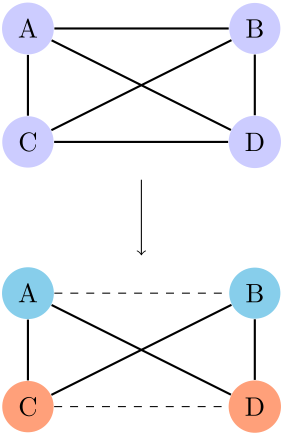
$l_j \leq x \leq u_j$

QUBO

- Special case: Quadratic Unconstrained Binary Optimization (QUBO)
  1. Quadratic objective function
  2. No variable constraints
  3. Binary optimization variables

$$\textbf{minimize } x^T Qx + c^T x$$
$$x \in \{0, 1\}^n,\ Q \in \mathbb{R}^{n \times n}, \quad c \in \mathbb{R}^n$$

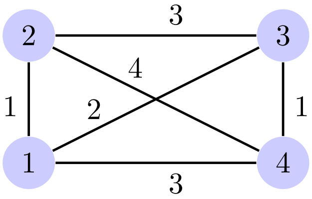
what is MAX-cut?



MaxCut

Weight matrix

$$w = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 4 \\ 2 & 3 & 0 & 1 \\ 3 & 4 & 1 & 0 \end{pmatrix}$$



Cost function

$$C(x) = \sum_{i,j=1}^{n} W_{ij} x_i (1 - x_j)$$

MAX-cut as QUBO

QUBO

QUBO matrix and vector

$$c_i = \sum_{j=1}^{n} W_{ij}, \quad Q_{ij} = -W_{ij}$$

Cost function

$$C(x) = \sum_{i,j=1}^{n} x_i Q_{ij} x_j + \sum_{i=1}^{n} c_i x_i$$

$$= x^T Q x + c^T x$$

$$H_C|x\rangle = C(x)|x\rangle$$

QUBO cost function

$$C(x) = \sum_{i,j=1}^{n} x_i Q_{ij} x_j + \sum_{i=1}^{n} c_i x_i$$

Hamiltonian operator

$$H_C = \sum_{i,j=1}^{n} \frac{1}{4} Q_{ij} Z_i Z_j - \sum_{i=1}^{n} \frac{1}{2} \left(c_i + \sum_{j=1}^{n} Q_{ij} \right) Z_i + \left(\sum_{i,j=1}^{n} \frac{Q_{ij}}{4} + \sum_{i=1}^{n} \frac{c_i}{2} \right)$$

Matrix exponential

Matrix exponential

For a matrix $\mathbf{M}$, the matrix exponential is defined as the power series:

$$e^{\mathbf{M}} = \sum_{k=0}^{\infty} \frac{1}{k!}\mathbf{M}^k$$

$$e^{-i\theta X} = R_X(2\theta) = \begin{pmatrix} \cos(\theta) & -i\sin(\theta) \\ -i\sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$e^{-i\theta Y} = R_Y(2\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$e^{-i\theta Z} = R_Z(2\theta) = \begin{pmatrix} e^{-i\theta} & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

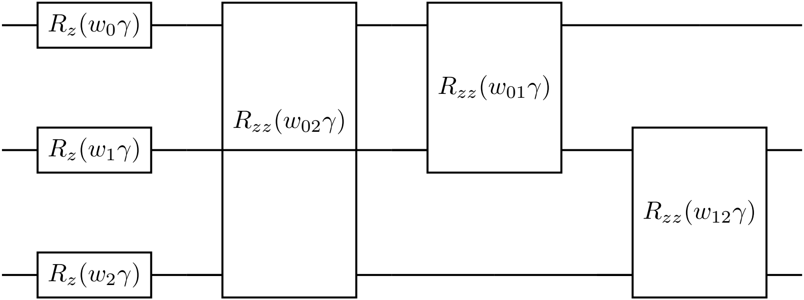
Pauli matrices

Cost Layer

$$H_C = \sum_{i,j=1}^{n} \frac{1}{4} Q_{ij} Z_i Z_j - \sum_{i=1}^{n} \frac{1}{2} \left(c_i + \sum_{j=1}^{n} Q_{ij} \right) Z_i$$

$$e^{-i\gamma H_C} = \prod_{i,j=1}^{n} R_{Z_i Z_j} \left(\frac{1}{2} Q_{ij} \gamma \right) \prod_{i=1}^{n} R_{Z_i} \left(\left(c_i + \sum_{j=1}^{n} Q_{ij} \right) \gamma \right)$$

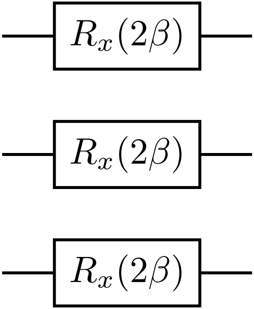
Quantum Circuit



Mixer Layer

$$H_M = \sum_{i=1}^{n} X_i \qquad X = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$e^{-i\beta H_M} = \prod_{i=1}^{n} R_X(2\beta)$$



Show us some coding!!!!

`https://github.com/ArminAhmadkhaniha/QAOA-and-maxcut/blob/main/qaoamax.ipynb`

References

- E. Farhi, J. Goldstone, and S. Gutmann, A Quantum Approximate Optimization Algorithm, ArXiv:1411.4028 [Quant-Ph] (2014)

Thank you for your time and attention!

Questions and Discussion

Contact Information:

- **Email:** ahmadkha@mcmaster.ca