# Quantum Algorithms

## Quadratic Unconstrained Binary Optimisation

Input - $x \in \{0,1\}^{n}$ Matrix $Q$ is Upper triangular. ($Q \in \mathbf{R}^{n\times n}$)

Optimisation problem:
$\min_x y = x^T Q x$
for $\max y' = -y$

Constrained to Unconstrained :
Introduce Quadratic penalties into the objective function
The penalties are formulated such that they are 0 for feasible solutions and equal to some positive penalty for infeasible solutions
If penatlies are driven to zero, we get same problem

But why quadratic?

| Classical Constraint | Equivalent Penalty |
|---|---|
| $x + y \leq 1$ | $P(xy)$ |
| $x + y \geq 1$ | $P(1 - x - y + xy)$ |
| $x + y = 1$ | $P(1 - x - y + 2xy)$ |
| $x \leq y$ | $P(x - xy)$ |
| $x_1 + x_2 + x_3 \leq 1$ | $P(x_1 x_2 + x_1 x_3 + x_2 x_3)$ |
| $x = y$ | $P(x + y - 2xy)$ |

Table of a few Known constraint/penalty pairs

Here $x, y$ are binary values and P is some constant.

Given general 0/1 optimisation problem:
$\min y = x^T C x$
$Ax = b, x \in \{0, 1\}$

1. We assume $A, b$ both have only integer components
2. For inequality constraints, we add slack variables
3. Convert $Ax = b$ into quadratic penalties
   1. $(P(Ax = b)^T(Ax = b))$
   2. $x_1 + x_2 \leq 1 \equiv P(x_1 x_2)$
4. Choose $P$ suitably.
5. Add them to the objective function

A program is needed to implement this.
Same can be done for linear programs

QUBO as a Query Model of Computation:
Input : $Q \in \mathbf{R}^{n \times n}, \text{Q is upper triangular}$
Output : $x^* \in \{0, 1\}^n \text{ such that } (x^*)^T Q x^* \text{ is min}$

Advantage:
Lot of Quantum Algorithms are expressed in Query model of computation

**Identifying Cliques, independent sets, etc in the flight graph**

## Ising Problems

NP-Hard Given Graph $G = (V,E)$ Each node assigned $0/1$ A Configuration $\sigma(G)$ is assignment of 0/1 to each node $\sigma_{i}$ is the binary value of $i$ node Each edge $e \in E$ has a weight $w(e)$ Each vertex $v \in V$ has a weight $h(v)$

Objective Function:
$\min_{\sigma} H(\sigma) = \sum_{i,j} w(e_{ij}) \sigma_i \sigma_j + \mu \sum_j h_j \sigma_j$

Same as QUBO (A Quantum technique to solve it, at least physically)

## Adiabatic Quantum Optimisation

NP-Hard To find the minimum soln for a Hamiltonian $H_p$ If we know a Hamiltonian $H_0$ whose minimal solution is easy to find, then we can use it to find the same for $H_p$ Adiabatic quantum computing has been shown to be polynomially equivalent to conventional quantum computing in the circuit model. Stuck at local minima Adiabatic is a slow process

$H(t) = (1 - \frac{t}{T})H_0 + \frac{t}{T}H_p$

Energy of Hamiltonian ?= Total score of our objective function

Measure H(t) at time T to get minimum of $H_p$
T is large

Relation to Binary Integer Linear Programming

Quantum Annealer