QUBO transformation using Eigenvalue Decomposition. Amit Verma and Mark Lewis.

Tomás Ricardo Basile Álvarez

July 3, 2022

Abstract

The paper uses eigenvalue decomposition of the Q matrix of a Qubo problem to transform the problem into a new one in which it may be easier to find optimal solutions.

Table of Contents

QUBO

Definition

QUBO: Quadratic Unconstrained Binary Optimization is a combinatorial optimization problem that given a polynomial $f_Q : \{0,1\}^n \to \mathbb{R}$ of the form:

$$f_Q(\vec{x}) = \sum_{i=1}^{n} \sum_{j=1}^{i} Q_{ij} x_i x_j$$

with $x_i \in \{0,1\}$ and coefficients $q_{ij} \in \mathbb{R}$, consists of finding a binary vector $\vec{x}^* \in \{0,1\}^n$ that maximizes f_Q.

QUBO

Definition

QUBO: Quadratic Unconstrained Binary Optimization is a combinatorial optimization problem that given a polynomial $f_Q : \{0,1\}^n \to \mathbb{R}$ of the form:

$$f_Q(\vec{x}) = \sum_{i=1}^{n} \sum_{j=1}^{i} Q_{ij} x_i x_j$$

with $x_i \in \{0,1\}$ and coefficients $q_{ij} \in \mathbb{R}$, consists of finding a binary vector $\vec{x}^* \in \{0,1\}^n$ that maximizes f_Q.

- Another way of writing the function is $f_Q(\vec{x}) = \vec{x}^T Q \vec{x}$, where Q is a symmetric $n \times n$ matrix.

QUBO

Definition

QUBO: Quadratic Unconstrained Binary Optimization is a combinatorial optimization problem that given a polynomial $f_Q : \{0,1\}^n \to \mathbb{R}$ of the form:

$$f_Q(\vec{x}) = \sum_{i=1}^{n} \sum_{j=1}^{i} Q_{ij} x_i x_j$$

with $x_i \in \{0,1\}$ and coefficients $q_{ij} \in \mathbb{R}$, consists of finding a binary vector $\vec{x}^* \in \{0,1\}^n$ that maximizes f_Q.

- Another way of writing the function is $f_Q(\vec{x}) = \vec{x}^T Q \vec{x}$, where Q is a symmetric $n \times n$ matrix.
- Many problems such as TSP, VRP, etc. can be given in a QUBO formulation.

Spectral Decomposition

Because matrix Q is symmetric, it has a base of orthonormal eigenvectors $\vec{c_i}$ with real eigenvalues λ_i (that is, $Q\vec{c_i} = \lambda_i \vec{c_i}$). Furthermore, Q acting on a vector x can be written as:

$$Q = \sum_{i=1}^{n} \lambda_i \vec{c_i}\vec{c_i}^T \quad , \quad Q\vec{x} = \sum_{i=1}^{n} \lambda_i (\vec{c_i} \cdot \vec{x})\vec{c_i}$$

Spectral Decomposition

Because matrix Q is symmetric, it has a base of orthonormal eigenvectors $\vec{c_i}$ with real eigenvalues λ_i (that is, $Q\vec{c_i} = \lambda_i \vec{c_i}$). Furthermore, Q acting on a vector x can be written as:

$$Q = \sum_{i=1}^{n} \lambda_i \vec{c_i}\vec{c_i}^T \quad , \quad Q\vec{x} = \sum_{i=1}^{n} \lambda_i (\vec{c_i} \cdot \vec{x})\vec{c_i}$$

For example:

$$\begin{bmatrix} -7 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 5 \end{bmatrix} = -7.56 \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix} \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix}^t + 2.44 \begin{bmatrix} -0.03 \\ -0.77 \\ 0.63 \end{bmatrix} \begin{bmatrix} -0.03 \\ -0.77 \\ 0.63 \end{bmatrix}^t + 7.12 \begin{bmatrix} 0.19 \\ 0.61 \\ 0.76 \end{bmatrix} \begin{bmatrix} 0.19 \\ 0.61 \\ 0.76 \end{bmatrix}^t$$

Figure: An example of a matrix decomposition.

Because matrix Q is symmetric, it has a base of orthonormal eigenvectors $\vec{c_i}$ with real eigenvalues λ_i (that is, $Q\vec{c_i} = \lambda_i \vec{c_i}$). Furthermore, Q acting on a vector x can be written as:

$$Q = \sum_{i=1}^{n} \lambda_i \vec{c_i}\vec{c_i}^T \quad , \quad Q\vec{x} = \sum_{i=1}^{n} \lambda_i (\vec{c_i} \cdot \vec{x})\vec{c_i}$$

For example:

$$\begin{bmatrix} -7 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 5 \end{bmatrix} = -7.56 \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix} \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix}^t + 2.44 \begin{bmatrix} -0.03 \\ -0.77 \\ 0.63 \end{bmatrix} \begin{bmatrix} -0.03 \\ -0.77 \\ 0.63 \end{bmatrix}^t + 7.12 \begin{bmatrix} 0.19 \\ 0.61 \\ 0.76 \end{bmatrix} \begin{bmatrix} 0.19 \\ 0.61 \\ 0.76 \end{bmatrix}^t$$

Figure: An example of a matrix decomposition.

Therefore, the QUBO function we want to maximize is:

$$f_Q(\vec{x}) = \vec{x}^T Q\vec{x} = \sum_{i=1}^{n} \lambda_i (\vec{c_i} \cdot \vec{x})(\vec{x} \cdot \vec{c_i}) = \sum_{i=1}^{n} \lambda_i (\vec{x} \cdot \vec{c_i})^2$$

Table of Contents

The paper proposes a way of finding better vectors $\vec{x}$ that maximize the function $f_Q(\vec{x}) = \sum_{i=1}^{n} \lambda_i (\vec{x} \cdot \vec{c_i})^2$.

The paper proposes a way of finding better vectors $\vec{x}$ that maximize the function $f_Q(\vec{x}) = \sum_{i=1}^{n} \lambda_i (\vec{x} \cdot \vec{c_i})^2$.

To do it, they propose transforming the function into another one that takes into account the eigenvalues λ_i with biggest absolute value. Then they optimize this new function.

1. Find the first k biggest eigenvalues of Q and their eigenvectors and sort them in decreasing order.

1. Find the first k biggest eigenvalues of Q and their eigenvectors and sort them in decreasing order.
2. Augment the original matrix Q with a penalty or reward term dependent on a chosen "penalty" parameter called M. That is, we transform the function $f_Q(\vec{x}) = \vec{x}^T Q\vec{x}$ we need to maximize into:

$$\vec{x}^T Q\vec{x} + \sum_{i=1}^{k} M \operatorname{sign}(\lambda_i)(\vec{x}^T \cdot \vec{c}_i)^2.$$

1. Find the first k biggest eigenvalues of Q and their eigenvectors and sort them in decreasing order.
2. Augment the original matrix Q with a penalty or reward term dependent on a chosen "penalty" parameter called M. That is, we transform the function $f_Q(\vec{x}) = \vec{x}^T Q \vec{x}$ we need to maximize into:

$$\vec{x}^T Q \vec{x} + \sum_{i=1}^{k} M \,\mathrm{sign}(\lambda_i)(\vec{x}^T \cdot \vec{c}_i)^2.$$

3. Solve this new QUBO problem, which whose solutions will possibly be better for the original problem.

1. Find the first k biggest eigenvalues of Q and their eigenvectors and sort them in decreasing order.
2. Augment the original matrix Q with a penalty or reward term dependent on a chosen "penalty" parameter called M. That is, we transform the function $f_Q(\vec{x}) = \vec{x}^T Q\vec{x}$ we need to maximize into:

$$\vec{x}^T Q\vec{x} + \sum_{i=1}^{k} M \operatorname{sign}(\lambda_i)(\vec{x}^T \cdot \vec{c}_i)^2.$$

3. Solve this new QUBO problem, which whose solutions will possibly be better for the original problem.

Example (with $M = 1$ and $k = 1$):

$$Q' = \begin{bmatrix} -7 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 5 \end{bmatrix} - 1 \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix} \begin{bmatrix} -0.98 \\ 0.14 \\ 0.13 \end{bmatrix}^t = \begin{bmatrix} -7 & 2 & 2 \\ 2 & 4 & 2 \\ 2 & 2 & 5 \end{bmatrix} - 1 \begin{bmatrix} 0.96 & -0.14 & -0.13 \\ -0.14 & 0.02 & 0.02 \\ -0.13 & 0.02 & 0.02 \end{bmatrix}$$

$$= \begin{bmatrix} -7.96 & 2.14 & 2.13 \\ 2.14 & 3.98 & 1.98 \\ 2.13 & 1.98 & 4.98 \end{bmatrix}$$

$$f_{Q'}(\vec{x}) = \vec{x}^T Q\vec{x} + \sum_{i=1}^{k} M\text{sign}(\lambda_i)(\vec{x}^T \cdot \vec{c_i})^2$$

Note that a big value of M biases the solution heavily towards the direction of the biggest eigenvector. While a small value of M doesn't change the original function that much and are inconsequential.

$$f_{Q'}(\vec{x}) = \vec{x}^T Q\vec{x} + \sum_{i=1}^{k} M\text{sign}(\lambda_i)(\vec{x}^T \cdot \vec{c_i})^2$$

Note that a big value of M biases the solution heavily towards the direction of the biggest eigenvector. While a small value of M doesn't change the original function that much and are inconsequential.
Similarly, it is important to decide the number k of eigenvectors we want to take into account.

Computational Experiments and Results

- They solve QUBO problems from different datasets and compare the solutions when they used their method to transform Q and when they didn't to see if there was an advantage.

- They solve QUBO problems from different datasets and compare the solutions when they used their method to transform Q and when they didn't to see if there was an advantage.
- The QUBO problems were solved using an optimizer called CPLEX or one they call PRlocal which is based on path relinking and tabu search.

ORlib database

- First they solved QUBO problems from the ORlib database.
- For small problems, the solutions obtained by CPLEX optimizer with the original and transformed problem were the same, no significant difference was obtained.

- First they solved QUBO problems from the ORlib database.
- For small problems, the solutions obtained by CPLEX optimizer with the original and transformed problem were the same, no significant difference was obtained.
- For a bigger problem, the distribution of eigenvalues was the one shown on the left and the improvement of the transformed problem over the original is shown on the right.

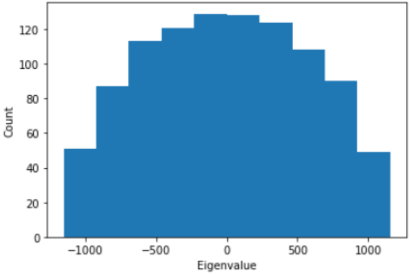
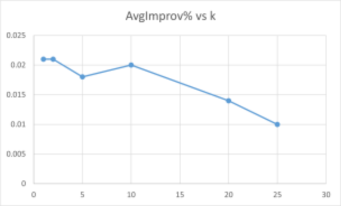


Fig. 1: Frequency distribution of eigenvalues for *bqp_1000_1*

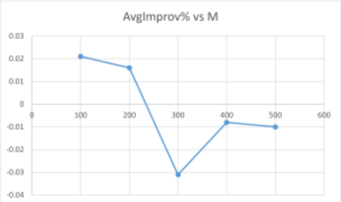
	Average Improvement % Over Base Problem				
k	M=100	M=200	M=300	M=400	M=500
1	0.069	0.031	-0.015	-0.169	-0.362
2	0.076	0.055	-0.031	-0.236	-0.955
5	0.053	-0.076	-0.26	-0.958	-2.431
10	0.064	-0.079	-0.761	-2.452	-3.889
20	0.056	-0.426	-1.431	-3.017	-4.343
25	-0.022	-0.555	-1.753	-3.115	-4.658

Palubeckis database

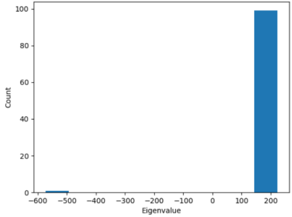
- They did the problem for bigger datasets (with similar normal eigenvalue distribution) from the Palubeckis database and solved them using the PRlocal heuristic for the base and transformed problem. The sensitivity analysis for k and M are shown in the following graphs.



(a) Sensitivity analysis of k



- Finally, they solved problems from the MDG database. This database has big problems with a more concentrated eigenvalue decomposition. For example, the eigenvalue decomposition for a problem of this database is:



- As you can see, a few eigenvalues lie around -500 while the rest are around 200. The principal components dominate other components and the method works better. The improvement of solving the original problem and the transformed one using the PRlocal heuristic is shown below.

Instance Type	Nodes	Avg Improvement % Over Base Problem
MDG-a	500,2000	1.0
MDG-b	500,2000	6.0
MDG-c	3000	9.7

Table of Contents

Conclusions

A technique that modifies the Q matrix based on the principal eigenvalues of Q is presented. This techniques gives improved solutions, specially when there are few dominant eigenvalues.
It is important to adjust the value of M and k to get the greatest benefit, but it is not obvious how to choose them.