Tensor Network Hackathon - Team 8
QUBO reformulation of the knapsack problem

Marco Tesoro

May 21, 2024

Introduction

The $0-1$ knapsack problem (KP), also known as the *Integer Weight Knapsack Problem (IWKP)*, is one of the best-known and most important combinatorial optimization problems. It occurs as a special case in a wide variety of optimization problems and has been generalized in various ways [KPP13]. Like many intriguing combinatorial optimization problems, KP is also NP-hard.

In this guide, we show how to reformulate a generic instance of the KP into the corresponding **Quadratic Unconstrained Binary Optimization (QUBO)** problem. This reformulation is analogous to a spinglass Hamiltonian with all-to-all interacting spin-$1/2$ variables, allowing the application of quantum algorithms to search for optimal solutions to the original problem. This reformulation constitutes the starting point of this hackathon project.

We now recall the mathematical formulation of the problem. We are given a backpack characterized by a maximum capacity $\mathcal{C}$ and a set of n objects, each with a positive integer profit $p_j \in \mathbb{N}$ and a positive integer weight $w_j \in \mathbb{N}$. The objective is to select a subset of items to include in the backpack such that the total profit $\mathcal{V} = \sum_{j=1}^{n} p_j x_j$ is maximized, while the total weight $\mathcal{W} = \sum_{j=1}^{n} w_j x_j$ does not exceed the capacity of the backpack. Introducing the binary optimization variables

$$x_j = \begin{cases} 1 & \text{if item } j \text{ is included in the knapsack} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

we can express this problem as the following *Binary Integer Linear Program (BILP)*:

$$\begin{aligned} &\arg\max_{\vec{x}\in\{0,1\}^n} \mathcal{V} \\ &\text{s.t. } \sum_{j=1}^{n} w_j x_j \leqslant \mathcal{C}. \end{aligned} \tag{2}$$

Additionally, the parameters defining a specific instance of the KP must satisfy the following non-triviality conditions:

$$\begin{aligned} &w_j \leqslant \mathcal{C} \\ &\sum_{j=1}^{n} w_j > \mathcal{C}. \end{aligned} \tag{3}$$

QUBO reformulation

Since we aim to leverage quantum algorithms, specifically the Quantum Approximate Optimization Algorithm (QAOA), to find the optimal solution to the KP, one possible approach is to convert the BILP into an equivalent QUBO problem. QUBOs are widely used for solving optimization problems on quantum computers [Luc14], and involve transforming a constrained optimization problem into an unconstrained one, where the variables are binaries and the constraints are incorporated into the objective function as penalty terms.

To handle the capacity constraint in Eq. 2, we introduce $\mathcal{C}$ auxiliary binaries, $y_\omega \in \{0,1\}$ for $1 \leqslant \omega \leqslant \mathcal{C}$, known as *slack variables*, defined as

$$y_\omega = \begin{cases} 1 & \text{if the final weight of the knapsack is } \omega \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

and we transform the capacity constraint into the following quadratic penalty term

$$P(\vec{x},\vec{y}) = \left(1 - \sum_{\omega=1}^{\mathcal{C}} y_\omega\right)^2 + \left(\sum_{\omega=1}^{\mathcal{C}} \omega y_\omega - \sum_{j=1}^{n} w_j x_j\right)^2. \tag{5}$$

This formulation ensures that the final backpack weight can take on only one value (first term in Eq. 5) and that the total weight given by the objects in the knapsack matches the claimed value (second term in Eq. 5). We add this penalty term to the cost function to obtain the QUBO objective function that must be minimized:

$$f_{kp}(\vec{x},\vec{y}) = -\sum_{j=1}^{n} p_j x_j + \lambda\, P(\vec{x},\vec{y}) \tag{6}$$

where $\lambda > 0$ is a penalty constant that must be tuned in order to enforce the constraint within the optimization. This approach introduces additional binary variables equal to the value of the maximum capacity $\mathcal{C}$, making the total number of optimization variables in the equivalent QUBO problem $n_{QUBO} = n + \mathcal{C}$.

However, we can encode the $\mathcal{C}$ possible values of the final total weight more efficiently. Let $M = \lfloor \log_2 \mathcal{C} \rfloor$ be the number of bits required to represent $\mathcal{C}$. We can use only $M+1$ auxiliary binaries $b_0, b_1, \ldots, b_M$ instead of the $\mathcal{C}$ variables $y_1, y_2, \ldots, y_{\mathcal{C}}$ to perform the following replacement

$$\sum_{\omega=1}^{\mathcal{C}} \omega y_\omega \longrightarrow \sum_{\alpha=0}^{M-1} 2^\alpha b_\alpha \;+\; (\mathcal{C} + 1 - 2^M) b_M \tag{7}$$

This substitution automatically satisfies the first term in Eq. 5, as the bitstring $b_0 b_1 \cdots b_M$ corresponds to one and only one value of the final weight in the binary representation given by Eq. 7. Thus, the penalty term is now given by

$$P(\vec{x},\vec{b}) = \left(\sum_{\alpha=0}^{M-1} 2^\alpha b_\alpha + (\mathcal{C}+1-2^M) b_M - \sum_{j=1}^{n} w_j x_j\right)^2 \tag{8}$$

and requires $n_{QUBO} = n + \lfloor \log_2 \mathcal{C} \rfloor + 1$ optimization variables.

Finally, the KP QUBO cost function is:

$$f_{KP}(\vec{x},\vec{b}) = -\sum_{j=1}^{n} p_j x_j \;+\; \lambda \left(\sum_{\alpha=0}^{M-1} 2^\alpha b_\alpha + (\mathcal{C}+1-2^M) b_M - \sum_{j=1}^{n} w_j x_j\right)^2 \tag{9}$$

with $\lambda > 0$, $x_j \in \{0,1\}$ for $j = 1, \ldots, n$, $b_\alpha \in \{0,1\}$ for $\alpha = 0, \ldots, M$ and $M = \lfloor \log_2 \mathcal{C} \rfloor$. By explicitly computing the square of the second term in Eq. 9 we obtain

$$\begin{aligned} f_{KP}(\vec{x},\vec{b}) = {} & \sum_{j=1}^{n} (\lambda w_j^2 - p_j) x_j \;+\; \lambda \sum_{\alpha=0}^{M-1} 2^{2\alpha} b_\alpha \;+\; \lambda (\mathcal{C}+1-2^M)^2 b_M + \\ & + \lambda \Bigg[2 \sum_{j'>j}^{n} w_j w_{j'} x_j x_{j'} \;+\; \sum_{\alpha'>\alpha}^{M-1} 2^{\alpha+\alpha'+1} b_\alpha b_{\alpha'} \;+ \\ & \quad - \sum_{j=1}^{n} \sum_{\alpha=0}^{M-1} 2^{\alpha+1} w_j x_j b_\alpha - 2(\mathcal{C}+1-2^M) \sum_{j=1}^{n} w_j x_j b_M \;+ \\ & \quad + 2(\mathcal{C}+1-2^M) \sum_{\alpha=0}^{M-1} 2^\alpha b_\alpha b_M \Bigg] \end{aligned} \tag{10}$$

which is of the form

$$f(\vec{v}) = C_0 + \sum_{k=1}^{n_{QUBO}} \mathcal{Q}_{kk} v_k + \sum_{k'>k}^{n_{QUBO}} \mathcal{Q}_{kk'} v_k v_{k'} \tag{11}$$

provided the optimization variables are arranged in a larger vector

$$\vec{v} = (x_1, x_2, \ldots, x_j, \ldots, x_n, b_0, b_1, \ldots, b_\alpha, \ldots, b_M)^T \tag{12}$$

from which we can derive the so-called QUBO matrix $\mathcal{Q}$. This symmetric matrix has diagonal elements $\mathcal{Q}_{kk}$ representing the coefficients of the terms involving a single binary, while the off-diagonal elements $\mathcal{Q}_{kk'}$ constitute the interaction strengths between different variables. The matrix $\mathcal{Q}$ will serve as the starting point for applying the QAOA variational quantum algorithm to solve the KP.

Mapping the problem to a classical Hamiltonian

We reformulate the QUBO problem as a ground-state search of a quantum many-body Hamiltonian, by mapping the cost function in Eq. 10 into a spinglass Hamiltonian [Luc14]. This involves converting the binary decision variables $\nu_k \in \{0, 1\}$ into spin-$\frac{1}{2}$ variables $\sigma_k \in \{-1, 1\}$ using the following linear transformation:

$$\sigma_k = 2\nu_k - 1. \tag{13}$$

Through this transformation, we obtain the following classical many-body Ising-like Hamiltonian:

$$\mathcal{H} = \mathrm{Cte} \cdot \mathbb{1} + \sum_{k=1}^{n_{\mathrm{QUBO}}} h_k \sigma_k + \sum_{k'>k}^{n_{\mathrm{QUBO}}} J_{kk'} \sigma_k \sigma_{k'} \tag{14}$$

where the couplings are derived from the QUBO matrix elements:

$$\begin{aligned}
\mathrm{Cte} &= C_0 + \frac{1}{2} \sum_{k=1}^{n_{\mathrm{QUBO}}} \mathcal{Q}_{kk} + \frac{1}{4} \sum_{k'>k}^{n_{\mathrm{QUBO}}} \mathcal{Q}_{kk'} \\
h_k &= \frac{1}{2} \mathcal{Q}_{kk} + \frac{1}{4} \sum\nolimits_{k'=k+1}^{n_{\mathrm{QUBO}}} \mathcal{Q}_{kk'} \\
J_{kk'} &= \frac{1}{4} \mathcal{Q}_{kk'}.
\end{aligned} \tag{15}$$

Finally, we promote the classical spin variables to quantum operators using the Pauli spin operator along the z-direction to obtain a many-qubit quantum Hamiltonian characterized by the couplings in Eq. 15.

References

[KPP13] Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer Science & Business Media, March 2013.

[Luc14] Andrew Lucas. Ising formulations of many NP problems. *Front. Phys.*, 2, February 2014. Publisher: Frontiers.