QUBO Bus routes

Febrero 2025

1 Introduction

1.1 Problem description

Find the best routing strategy for several lines of urban transport vehicles, considering the following statements:

- Distances are not symmetric (in general, $d(A, B) \neq d(B, A)$).
- Bus stops are set based on social and demographic factors.
- The number of bus stops per line, as well as the number of lines are hyperparameters of the problem.

We want to set L bus lines in N stops. As the distances between stops are not symmetric, we can define the binary variables as

$$x_{ij}^{l} \in \{0, 1\} \tag{1}$$

Where the value is 1 only if the line l goes at some point from stop i to stop j. The global function that we want to minimize is

$$f(x_{00}^{0}, x_{01}^{0}, ..., x_{0N-1}^{0}, x_{10}^{0}, ..., x_{N-1N-1}^{0}, x_{00}^{1}, ..., x_{N-1N-1}^{L-1}) = \sum_{l}^{L-1} \sum_{i}^{N-1} \sum_{j}^{N-1} D_{ij} x_{ij}^{l} \tag{2}$$

where the matrix of distances D_{ij} is not symmetric and of order $N \times N$.

1.2 QUBO formulation

A Quadratic unconstrained binary optimization (QUBO) is a optimization problem which is defined for a set of binary variables $z = (z_1, z_2, ..., z_n) \in \{0, 1\}^n$ which minimizes the quantity

$$f(z) = z^T Q z \tag{3}$$

for Q a symmetric (or triangular superior) matrix of size $(n \times n)$. We need to reformulate our problem to fit into this definition. Some simplifications may be necessary in order to do so.

2 QUBO Formulation of the bus routes problem

2.1 Previous considerations and further assumptions

At an starting point, notice that, if we wanted to express all the variables of our problem in a QUBO formulation, we would need to 'unravel' the three indexes variables x_{ij}^{l} into a vector y_k with $N \times N \times L$ components. On one hand, this approach, equipped with suitable constraints, would give the best route for all the bus lines at the same time. However, it is unfeasible as the scalability gets extremely compromised when the number of stops or the number of lines taken into consideration start to grow.

To surpass this problem we need to make some assumptions in the definition of the QUBO model. The first one, not relevant for this part of the project, is to divide the whole landscape of bus stops into zones, agrupping them and solving the optimization problem in different levels. Then, the city is divided into "district", which are componend of "neighbourhoods" which have only a dozen of bus stops, making the scalability much stronger.

The second assumption will be to consider bidirectionl routes. This makes again the distances symmetric. However, the fact that they were obtained by computing the time that one spend to go from A to B, which normally is different from the time of goung from B to A makes the mean distance a "time" distance which is more accurate than taking directly half of the distance in meters which separate both A and B in an straight line. This assumptions has an effect on our parameters, which is to define a new distances matrix

$$D'_{ij} = \frac{D_{ij} + D_{ji}}{2} \tag{4}$$

The next assumption in the chase of better scalability is to consider individual lines and, after optimize for each one of them, refine the solutions which fulfill a global condition on all the lines. With this last simplification the QUBO model is equivalent to the Traveling Salesman Problem (TSP) with start and end stops different from each other. We will take the TSP representation of the variables as well as the common constraints imposed in this kind of problems. We will call this approach TS Bus Line Problem.

2.2 Formulating the TS Bus Line Problem and the cost function

With the previous assumptions, some of the variables of our problem are no longer needed. For example, as we only consider one line, the superindex l can be discarded. One could also think that, as the distanes are now symmetric, the variables x_{ij} and x_{ji} give the same information and we can drop half of them, reducing the number of binary variables to $\frac{N\times(N-1)}{2}$. However, the shape of the constraints with this amount of variables is extremely complicated, leading to the use of slack variables and incongruences. Therefore, to stick to the TSP whose constraints has been well explored we will define the following quantities

- $N \in \mathbb{N}$ is the total number of stops considered.
- $p \in \{1, ..., N\}$ is the total number of 'travels' (or steps) that the bus line has to complete.
- $x_{start} \in \{0, ..., N-1\}$ is the index of the initial stop.
- $x_{end} \in \{0, ..., N-1\}$ is the index of the final stop.

In principle, the route can be made circular by setting $x_{start} = x_{end}$. In addition, if $p = N$ the line has to end in the same stop it started after going through all the other stops. We will focus only in linear-bidirectional routes, therefore we do not allow for $x_{start} = x_{end}$ or $p = N$.

In each step, the line goes from a stop i to another stop j with a cost of D_{ij} for that travel. As we want to take track of the travels of the line and the ordering of them, we have to take two different vectors for the initial position and the final position. Then, for a single trip were the line start in an unknown stop x_i and end in another stop x_j, the cost function will be computed as

$$\begin{pmatrix} x_0 & x_1 & ... & x_{N-1} \end{pmatrix} \begin{pmatrix} 0 & d_{0,1} & d_{0,2} & ... & d_{0,N-1} \\ d_{1,0} = d_{01} & 0 & d_{1,2} & ... & d_{1,N-1} \\ ... & ... & ... & ... & ... \\ d_{N-1,0} & d_{N-1,1} & d_{N-1,2} & ... & 0 \end{pmatrix} \begin{pmatrix} x_{N+0} \\ x_{N+1} \\ ... \\ x_{N+N-1} \end{pmatrix} \tag{5}$$

If the line has a total of $p+1$ stops, which means is makes, p connections between stops, the total cost function will be

$$f_{cost} = \sum_{k=0}^{p-1} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \left(x_{Nk+i} \cdot x_{N(k+1)+j} \cdot D_{ij} \right) \tag{6}$$

with the total lenght of the binary vector being $R = N(p+1)$. In the Q matrix representation that we are looking for, the cost function will be an $R \times R$ matrix with internal block matrices

$$Q_{cost} = \begin{pmatrix} 0_{N\times N} & D_{ij} & 0_{N\times N} & ... & 0_{N\times N} \\ 0_{N\times N} & 0_{N\times N} & D_{ij} & ... & 0_{N\times N} \\ ... & ... & ... & ... & ... \\ 0_{N\times N} & 0_{N\times N} & 0_{N\times N} & ... & D_{ij} \\ 0_{N\times N} & 0_{N\times N} & 0_{N\times N} & ... & 0_{N\times N} \end{pmatrix} \tag{7}$$

Finally, as for solving the QUBO with Pulser we require a symmetric Q matrix, we can simply do

$$Q^{'}_{cost} = Q_{cost} + Q^{\intercal}_{cost} \tag{8}$$

By doing this we simply penalized double the distances but the global minimum remains the same, allowing for a symmetric Q matrix.

2.3 Constraints

After stating the distances cost matrix we have to impose some constraints to find valid solutions. The restrictions are included in the QUBO formulation as

$$H_{total} = H_{cost} + \lambda H_{constraints} \tag{9}$$

Were the $H_{constraint}$ should be a quadratic function of the binary variables.
Or, similarly

$$Q_{total} = Q_{cost} + \lambda Q_{constraints} \tag{10}$$

The common constraints in the TSP are the folllowing

2.3.1 Constraint 1: In each step, the line is in only one stop

For this constraint the formulation is

$$\sum_{k=0}^{p} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \Big(x_{Nk+i} \cdot x_{Nk+j} \Big) = 0 \quad \{i, j | i < j\} \tag{11}$$

This constraint is already quadratic and, for $\lambda > 0$ penalizes solutions which have more than one node activated at the same time in a step k. To make it symmetric we allow also for $i > j$ in the summation. This only makes the penalization stronge but does not change the shape of the contraint.

2.3.2 Constraint 2: In each step, the bus line should be in one stop (not dissapear)

As we have stated that, at each step, the bus lines can only be at one stop at most, we only have to ensure that the total amount of stops visited in the whole bus line is adequate. Therefore

$$\sum_{k=0}^{R-1} x_k = p + 1 \tag{12}$$

To insert this condition in the QUBO model, we have to make it quadratic as

$$\Big(\sum_{k=0}^{R-1} x_k - (p+1) \Big)^2 = \sum_{k=0}^{R-1} x_k^2 + 2 \sum_{k<l}^{R-1} x_k x_l - 2(p+1) \sum_{k=0}^{R-1} x_k + p^2 \tag{13}$$

Again, to make it symmetric and suitable for the QUBO model we end up with

$$\sum_{k=0}^{R-1} (1 - 2(p+1)) x_k^2 + \sum_{k \neq l}^{R-1} x_k x_l \tag{14}$$

2.3.3 Constraint 3: At most one visit to each stop

This constraint is fulfilled with

$$\sum_{k=0}^{R-1} \sum_{f=k+N;stepN}^{R-1} x_k \cdot x_f = 0 \tag{15}$$

This condition is already quadratic. In the original TSP as the Salesman has to return to the initial stop, the las step is not taken into account in this constraint. Howeverm as we are interseted only in linear-bidirectional routes, we have to check that the last stop is not repeated among the already visited stops.

2.3.4 Constraint 4: Start stop

To incentivate the bus line to start in a preferred start stop, we can add a negative penalty weight $\lambda_4 < 0$ to the node x_i where $i = x_{start}$. As the variable are binary, this condition is already quadratic.

2.3.5 Constraint 5: End stop

Similarly, to incentivate the bus line to end in a certain end stop $x_{end} = j$, we can add a negative penalty weight $\lambda_5 < 0$ to the node x_{Np+j}.

2.4 QUBO model

With all those constraints the QUBO model is complete and we can obtain, given a set of penalty weights $\vec{\lambda} = \big(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5\big)$, the total Q matrix as

$$Q = Q_{cost} + \sum_{i=1}^{3} \lambda_i Q_i - \sum_{j=4}^{5} \lambda_j Q_j \tag{16}$$

The problem now is to obtain a set of suitable weights which makes the global minimum of the total function a valid solution (which fulfill all the constraints) without forgetting to minimize the distances cost function.

3 Weights optimization

The process of optimization of the weights is as follows

1. Select the number of stops N, the number of travels p, the start stop x_{start} and the end stop x_{end}.
2. Compute the upper bound of costs as $C_{UB} = (1, 1, ..., 1)Q_{cost}(1, 1, ..., 1)^\intercal$.
3. Create an initial array of weights $\lambda_i = factor \cdot C_{UB}$. We will select $factor = 0.01$.
4. Compute the initial Q matrix and solve the QUBO problem with any method (we have chosen the DWave's Simulated Annealer).
5. If the solution is valid, the lambdas are taken as optimal. End the iteration. If not, continue the iteration.
6. (Brute force) Compute all the combinations of binary vectors and order them based on the total value of $x^\intercal Qx$.
7. For the n best combinations count which constraints they fail to fulfill.
8. Raise the value of λ_i by adding to it a quantity based on the amount of combinations which has failed to fulfill the constraint i. In our case we sum a quantity equal to a scaling factor multiplied by the percentage of failed combinations.
9. With the new array of weights, go to step 4.

For instance, for $N = 5$, $p = 3$, $x_{start} = 1$, $x_{end} = 3$, after iterate 6 times with scaling factor 0.1, the solver with 1000 reads and taking into account the best 30 combinations, one obtain

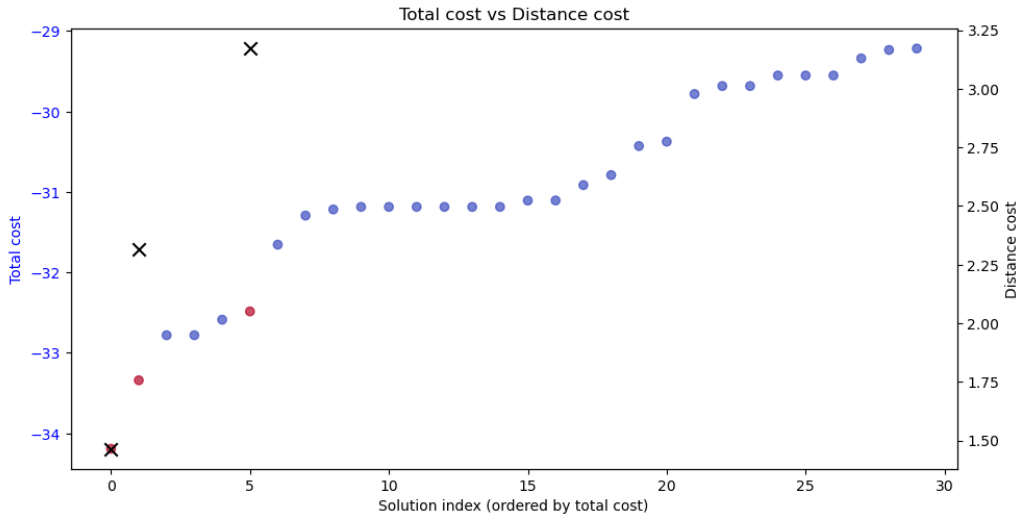

Figure 1: Brute force found best combinations. Red dots are valid solutions. Crosses are the bidirectional distance (cost function)

By increasing the number of iterations, the valid solutions will get closer to the minimum, but the cost function of distances will gradually get less importance to the point that the global minimum could not be the best valid solution.

3.1 Analysis of weights

Computing solutions by brute force is highly inefficient. Therefore, the objective of this optimization of weights is to find a rule of selecting the best ones which are suitable for different situations. To start with something, we can fix the parameters $N = 5$ and $p = 2$ and see how the optimal weights change if we select different start and end nodes.

For all possible combinations, the mean values and variance of the weights are

λ	μ	σ^2
λ_1	4.334	0.0113
λ_2	2.300	0.0014
λ_3	1.234	0.0041
λ_4	2.632	0.0072
λ_5	4.142	0.0015

Table 1: Mean values and variances of optimal weights for $N = 5$ and $p = 2$

The results can be visualized in

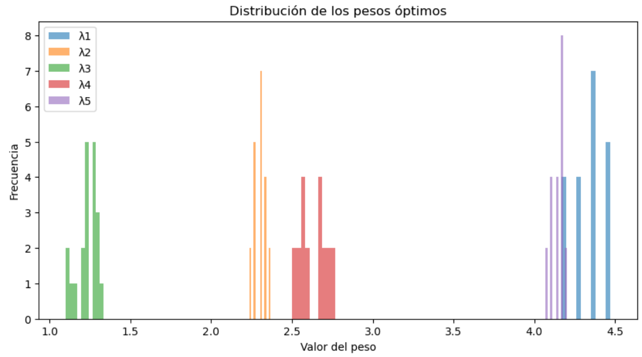

Figure 2: Distribution of weights

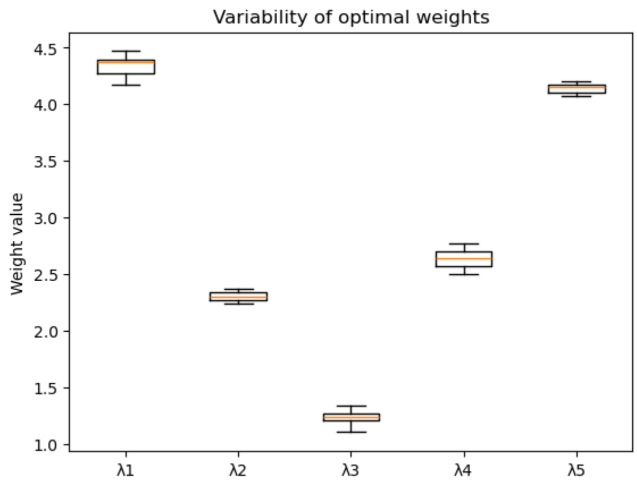

Figure 3: Distribution of weights by value

We can analyze how good are the estimators of the mean value for obtaining a valid solution given a random choice of start and end nodes. For all the possible combinations of start and end nodes we use the mean values of the lambdas and try to find a solutions. The results are that only 40% of the combinations are well solved for this choice of weights.

To improve the solvability we can try to fix the value of the mean weights with the following transformation

$$\lambda_i = (6 - i) \cdot \lambda_{i;mean} \quad i \in \{1, .., 5\} \tag{17}$$

With this simple correction, which gives more importance to a constraint in inverse order, we obtain that all possible combinations (100%) are well solved (the minimum is a valid solution) for this new choice of weights. The question now is to see if, for a different choice of N and p this rule is still valid.

If we try with $N = 4$ and $p = 2$ the results are similar, obtaining a 33% of valid solutions with the mean weights and a 100% with the corrected version.

If compute a joint mean value of the weights for $N = 5$ and $N = 4$ the valid solutions for the mean weights are worse, but, in both cases, the corrected version gives again a 100% of valid solutions.

By now we can keep this corrected version as the global weights to use them in the Quantum solver.

4 Joining more than one bus lines

By the moment we have solved the problem for just one line. In order to include more lines in the same N stos we can do the following

1. Select the number of stops N, the number of travel per line p and the number of lines L. (Note that we will need $p \cdot L > N$.
2. Give a set of start in a way that, if a stop is the initial stop of some line l it cannot be the initial stop of any other line $k \neq l$ or the end stop of any line at all. Do the same for the end stops. These constraints could be relaxed.
3. Select the initial weights as the optimal weights obtained in the previous section.
4. Solve the QUBO model for each of the lines.
5. Check that, joining all the lines, each stop is visited at least one.

Depending on the problem the election of the start and end stops can affect the total distance and give bad solutions which, despite of probably bein vaild, are not the optimal ones. If the start and end stops are not chosen previously, we can scan all the possible combinations and choose the one that gives the lower total distance.

For example, for $N = 5$ and $p = 2$, the total amount of different start-end stops combinations is 60. The best solution using the corrected weights for $N = 5$, $p = 2$ found is

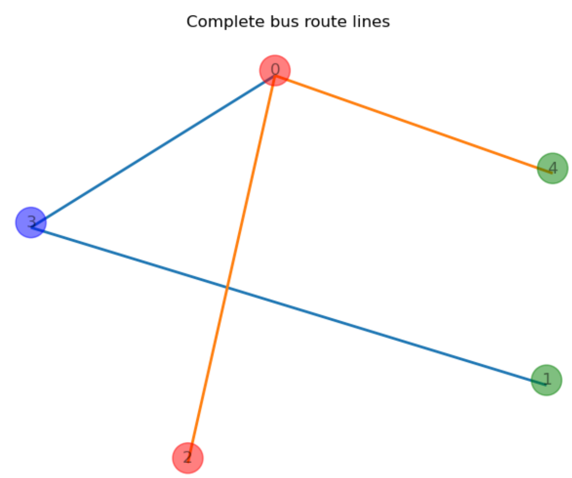

Figure 4: Complete bus routes for $N = 5$ and $p = 2$. Red nodes are starting stops. Green nodes are ending stops.

The total distance taking into account that the lines are bidirectional is 2.2326, where $d = 1$ is the biggest distance in the stops graphs.