Solving assignment problems via Quantum Computing: a case-study in train seating arrangement

Ilaria Gioda*, Davide Caputo†, Edoardo Fadda*, Daniele Manerba‡,
Blanca Silva Fernández†, and Roberto Tadei*

*Department of Control and Computer Engineering, Politecnico di Torino, 10129 Torino, Italy
Email: ilaria.gioda@studenti.polito.it, {edoardo.fadda, roberto.tadei}@polito.it

†Data Reply s.r.l., 10126 Torino, Italy
Email: {da.caputo, b.silvafernandez}@reply.it

‡Department of Information Engineering, Università degli Studi di Brescia, 25123 Brescia, Italy
Email: daniele.manerba@unibs.it

***Abstract*—In recent years, researchers have oriented their studies towards new technologies based on quantum physics that should resolve complex problems currently considered to be intractable. This new research area is called Quantum Computing. What makes Quantum Computing so attractive is the particular way with which quantum technology operates and the great potential it can offer to solve real-world problems. This work focuses on solving assignment-like combinatorial optimization problems by exploiting this novel computational approach. A case-study, denoted as the Seating Arrangement Optimization problem, is considered. It is modeled through the Quadratic Unconstrained Binary Optimization paradigm and solved through two tools made available by the *D-Wave Systems* company, QBSolv, and a quantum-classical hybrid system. The obtained experimental results are compared in terms of solution quality and computational efficiency.**

I. Introduction

Combinatorial Optimization (CO) is one of the most studied research fields in the area of optimization. The application of this research area extends to many sectors, and more and more researchers are active in model and solve effectively and efficiently the problems belonging to this category. Among others, one of the most recent and innovative modeling approaches to formulate a CO problem is the so-called Quadratic Unconstrained Binary Optimization (QUBO) paradigm. Among the various approaches for solving combinatorial optimization problems in the QUBO form, in recent years, researchers have begun to be oriented towards a new computational frontier, as the *Quantum Computing*. This paper focuses on analyzing this new computational approach, specifically for the resolution of assignment problems. We analyze the Seating Arrangement Optimization problem (SAOP) as a case-study, which was first formulated as a QUBO problem and then solved through the use of some tools made available by *D-Wave Systems*, a Canadian company specializing in quantum computing. In particular, quantum systems suitable for solving optimization problems are called quantum annealers; they exploit the physical concept that everything in nature tends to evolve towards equilibrium (see [1]). It is worth noting that there exist alternative ways to implement quantum-like algorithms as the one explored in [4].

The remaining part of this paper is organized as follows. Section II describes the Quadratic Unconstrained Binary Optimization (QUBO) paradigm and reports on the existing solvers dedicated to problems in this particular form, including the quantum technologies offered by *D-Wave Systems*. Section III presents the case-study considered, the Seating Arrangement Optimization problem. The problem is first described and modeled as a quadratic problem. Then an equivalent QUBO formulation is derived. Section IV describes and compares the computational results of the experimental analysis. Section V provides conclusions and a brief discussion on possible future works.

II. Quantum Computing solvers

The leader company that works with quantum annealers is *D-Wave Systems Inc.*. In particular, this organization deals with building and studying quantum technologies and, for some years, has allowed external people to use their quantum annealers to solve specific commercial problems, especially combinatorial optimization ones. Quantum annealers are designed to solve complex combinatorial optimization problems in a particular formulation, the Quadratic Unconstrained Binary Optimization (QUBO) one. The goal of a QUBO model is to find an optimal solution by minimizing an objective function in the form

$$\min_{\mathbf{x}\in\{0,1\}^{|N|}} \mathbf{x}^T Q\mathbf{x} \tag{1}$$

where $\mathbf{x}$ is a column vector of binary variables of size $|N|$ and Q an upper-triangular $|N| \times |N|$ matrix, called QUBO matrix. Not all optimization problems come in this form. However, many of them can be rewritten as a QUBO model. The constraints identified for the problem must be readjusted and converted into penalties to form the actual objective function (1) that has to be minimized. Specifically, as in classical

Lagrangean relaxations, the purpose of these penalties is to prevent the optimizer from choosing solutions that violate the constraints. They involve the addition of a positive quantity, therefore not favorable to the minimization objective in case of infeasible solutions [3]. Some standard ways of creating this translation from classical constraints can be found in [3]. By using this formulation, two solution methods are available: *QBSolv* and *D-Wave Systems*.

QBSolv is an open-source solver released in January 2017, which runs on the CPU like traditional solvers. Its goal is to solve significant QUBO problems with high connectivity. The solver strategy consists of partitioning significantly large QUBO problems into smaller components and applying a specified sampling method (the classical Tabu Search algorithm, by default) independently to each of these pieces to find the minimum value required for the optimization. Further technical details on QBSolv can be found in [2].

D-Wave Systems allows to submit and solve a problem modeled as QUBO on a remote quantum computer. To do this, in 2018, the computing company made available to users a cloud service, the *D-Wave's Leap*, and a set of Python APIs, the Solver API (SAPI), that allow any developer to access and submit any problem to the *D-Wave Quantum System*.

III. A case-study: The Seating Arrangement Optimization Problem

The considered case-study focuses on passenger transport on high-speed trains considering the Italian Government's new regulations on social distancing due to the COVID-19 pandemic.

The railway companies have currently adapted their passenger positioning strategies by embracing a seating arrangement as a "checkerboard" pattern, i.e., with the allocation of passengers to alternate seats, to counter the spread of the COVID-19 virus. Nevertheless, with the adoption of this strategy, the filling capacity of the wagons has dropped to 50% of the total capacity, leading to a drastic reduction in the high-speed rail operators' earnings. This is due to the mismatch between the costs necessary for activating the railway transport lines and the revenues obtained from ticket sales. From now on, we will denote the examined case-study as the Seating Arrangement Optimization problem.

The objective of the Seating Arrangement Optimization problem (SAOP) is to fill the train wagon as much as possible within the restrictions on social distancing due to the COVID-19 health emergency. Still, it aims to maximize the number of passengers belonging to the same family or living group in adjacent seats. Although the focus of the problem can be extended to the entire train, the study refers to only one wagon. Then, for a multi-wagon train, the procedure will be run for each wagon separately. Furthermore, we assumed a static situation: just one train segment, i.e., a trip between two adjacent stations, is considered so that the number of passengers and their social relationships are known beforehand without any changes during the travel.

Some fundamental elements characterize the SAOP. A set of passengers that has to be transported on a high-speed train is given. During the ticket reservation procedure, each passenger is associated with a unique identifier, the booking ID, which can be shared or not with other passengers. The important assumption of the problem is that people with the same booking ID belong to the same family or living group. This condition, therefore, assumes they can be excluded from the social distancing impositions prescribed by the regulations against the spread of the COVID-19 virus. A high-speed train's wagon is then considered. The wagon has a certain number of seats. Each seat is represented by a pair of coordinates, a row and a column number, which collocate it into a grid. Finally, it is necessary to consider the following requirements:

- allocation of one and only one seat to each one of the considered passengers (avoid that a passenger has more than one seat assigned to him);
- allocation of one passenger at most to each seat (avoid different passengers being assigned the same seat);
- allocation of not adjacent (in front/behind/left/right) seats to people belonging to different families (identified by different booking IDs).

Let us consider the following sets and parameters:

- $R = \{1, 2, \dots, r_{max}\}$: set of seats row numbers;
- $C = \{1, 2, \dots, c_{max}\}$: set of seats column numbers;
- K: set of booking IDs;
- n_k: nb. of passengers with the same booking ID $k \in K$.

Moreover, let us define the variable

$$x_{(r,c),k} := \begin{cases} 1 & \text{if a passenger with booking ID } k \text{ is} \\ & \text{assigned to seat with row and column } (r,c) \\ 0 & \text{otherwise} \end{cases}$$

for each row $r \in R$, column $c \in C$, and booking ID $k \in K$. Then, a natural quadratic programming model for the SAOP can be stated as:

$$\max \sum_k \sum_{(r,c)} x_{(r,c),k} \cdot x_{(r+1,c),k} + \\ + \sum_k \sum_{(r,c)} x_{(r,c),k} \cdot x_{(r,c+1),k} \quad (2)$$

subject to

$$\sum_{(r,c)} x_{(r,c),k} = n_k, \quad k \in K \quad (3)$$

$$\sum_k x_{(r,c),k} \leq 1, \quad r \in R,\ c \in C \quad (4)$$

$$x_{(r,c),k} \cdot x_{(r+1,c),k'} = 0, \\ r \in R \setminus \{r_{max}\},\ c \in C,\ k, k' \in K, k \neq k' \quad (5)$$

$$x_{(r,c),k} \cdot x_{(r,c+1),k'} = 0, \\ r \in R,\ c \in C \setminus \{c_{max}\},\ k, k' \in K, k \neq k' \quad (6)$$

$$x_{(r,c),k} \in \{0, 1\}, \quad r \in R,\ c \in C,\ k \in K. \quad (7)$$

The objective function (2) maximizes the number of passengers with the same booking ID assigned to adjacent seats. Constraints (3) state that each passenger with a given booking ID is assigned to one seat, while constraints (4) state that each seat is assigned to at most one passenger with a given booking ID. Constraints (5) ensure that two seats, one next to the other (in the same column), are not assigned to passengers with different booking IDs, while constraints (6) ensure that two seats, one in front of the other (in the same row), are not assigned to passengers with different booking IDs. Finally, binary conditions on the variables are stated in (7).

A. QUBO formulation

Since the QUBO paradigm asks for an unconstrained model, as the one in (1), the constraints (3)-(6) and the cost function (2) are relaxed and aggregated into a single objective function through non-negative parameters λ's, to be calibrated (see later). In particular, we chose to set these parametric coefficients as numerical and to associate each of them with a specific group of constraints presented in model (2)–(7). We decided to adopt this modeling choice to minimize the number of λ parameters needed, as they represent a non-negligible obstacle during the model calibration.

To do this relaxation, we built a penalty term for each of the identified constraints by following the approach from [3]. Hence, a QUBO formulation for the SAOP problem becomes:

$$\min \lambda_A H_A + \lambda_B H_B + \lambda_C H_C + \lambda_D H_D - H_E \tag{8}$$

where

- the penalty term associated with constraints (3) is
$$H_A = \sum_k (n_k - \sum_{(r,c)} x_{(r,c),k})^2$$
- the penalty term associated with constraints (4) is
$$H_B = \sum_{(r,c)} \sum_{k,k'} x_{(r,c),k} \cdot x_{(r,c),k'}$$
- the penalty term associated with constraints (5) is
$$H_C = \sum_{(r,c)} \sum_{k,k'} x_{(r,c),k} \cdot x_{(r+1,c),k'}$$
- the penalty term associated with constraints (6) is
$$H_D = \sum_{(r,c)} \sum_{k,k'} x_{(r,c),k} \cdot x_{(r,c+1),k'}$$
- the penalty term associated with objective function (2) is
$$H_E = \sum_k \sum_{(r,c)} x_{(r,c),k} \cdot x_{(r+1,c),k} + \sum_k \sum_{(r,c)} x_{(r,c),k} \cdot x_{(r,c+1),k}.$$

Note that, unlike the other penalties, a squaring has been introduced in H_A as it is necessary to be able to grasp the relationship between the values assumed by different variables within the solution.

Starting from (8), the Q matrix of model (1) has been derived. To do that, we need to identify the relationship between the problem's variables. First of all, the single QUBO terms of the function (8) need to be expanded. Then, after the coefficients have been found, they are multiplied by the parametric coefficients λ_A, λ_B, λ_C and λ_D, whose purpose is to give more or less weight to each QUBO penalty such that the constraints are imposed when searching for the solution.

IV. Computational results

This section reports the results obtained by executing several instances of the SAOP modeled as a QUBO using the two tools offered by *D-Wave Systems*, namely, the QBSolv and D-Wave Leap's cloud-based quantum-classical hybrid solver (from now on referred to as D-Wave Hybrid Solver). Initially, the problem size in terms of the number of variables is reported. Then, the two solvers are compared in terms of optimal solutions and computational times.

An ad-hoc data set containing simulated test instances about seats, passengers, and bookings were created for the performed experiments. The input that we provided to our QUBO model has been created based on an indicative estimate of realistic data of a high-speed train. In particular, it was decided to use a wagon consisting of 80 seats, placed in a 4×20 grid, made up of 4 horizontal (the rows) and 20 vertical (the columns) rows. Taking as a reference a reasonable number of passengers for a high-speed train, 1000 passengers have been created. Still, only a small subset of them was used for our restricted experimental analysis. In particular, for the experiments reported in the following, the maximum number of people that have been tested is 52. Since it was necessary to associate a specific booking ID to each passenger, we decided to use 300 distinct booking IDs to make a reasonably homogeneous assignment. The final range of assigned booking IDs is 290 booking IDs, and the minimum number of passengers with the same booking ID is 1 while the maximum is 8. All the experiments have been carried out on a desktop computer with a 1.8 GHz Intel Core i7-8550U processor.

A. Quality of solutions

The quality of the *D-Wave Systems* solvers is now analyzed. After having calibrated the λ parameters in model (8), by using the Python APIs of the Ocean SDK, the two solvers were used to solve different instances of the analyzed problem. The numerical results for the SAOP various instances can be seen in Table I. For each problem instance (identified by "Seats", "Passengers", "Distinct booking IDs" columns), we report in the "Total minimum energy" column the value of the best solution found by each solver (i.e., the minimum value of the expression (8)). Moreover, the number of passengers allocated to seats inside the train wagon (fifth column) and the number of people with the same booking ID correctly assigned to adjacent seats (sixth column) are reported for each solution.

The two solvers seem to perform well, most of the time reaching the goal of allocating people with the same booking

TABLE I
OPTIMAL SOLUTIONS OBTAINED BY RUNNING THE QUBO MODEL INSTANCES WITH THE *D-Wave Systems* SOLVERS

Seats	**Passengers**	**Distinct booking IDs**	**Solver**	**Nb. of passengers with an assigned seat**	**Nb. of passengers with same booking ID assigned to adjacent seats**	**Total minimum energy**
80	11	3	QBSolv	11	10	-464.300
			D-Wave Hybrid	10	11	-464.300
80	16	4	QBSolv	16	15	-682.800
			D-Wave Hybrid	16	15	-682.800
80	19	5	QBSolv	19	18	-762.000
			D-Wave Hybrid	19	18	-762.000
80	23	7	QBSolv	23	21	-849.400
			D-Wave Hybrid	23	21	-849.400
80	28	8	QBSolv	28	26	-1067.900
			D-Wave Hybrid	28	26	-1067.900
80	34	9	QBSolv	34	32	-1382.000
			D-Wave Hybrid	34	32	-1382.000
80	39	11	QBSolv	39	37	-1496.700
			D-Wave Hybrid	39	37	-1496.700
80	44	13	QBSolv	44	41	-1646.900
			D-Wave Hybrid	44	41	-1646.900
80	50	14	QBSolv	50	45	-1947.500
			D-Wave Hybrid	50	47	-1958.300
80	51	15	QBSolv	51	46	-1953.000
			D-Wave Hybrid	51	47	-1961.100

ID to adjacent seats. Furthermore, an improvement compared to the passenger transport's current situation has been achieved. Both solvers manage to find at least an acceptable seating arrangement up to 15 booking IDs for a total of 51 passengers, bringing therefore to have a filling percentage of the seats up to 63,75% (instead of the classical 50%).

For most instances, the D-Wave Hybrid Solver finds solutions with the same energy as those found by QBSolv. This means that the solver running on the CPU performs well in solution quality, even without quantum hardware usage. However, there are two cases, i.e., the ones corresponding to the instances with 14 and 15 distinct booking IDs (respectively 50 and 51 passengers), where D-Wave Hybrid Solver finds two lower energy and better solutions than those found by QBSolv. The computational time of QBSolv ranges from 1.2s (for the instance with 11 passengers and 3 distinct booking IDs) up to 18.8s (for the instance with 51 passengers and 15 different booking IDs). Instead, if the D-Wave Hybrid Solver is used for solving the same problems, the computational time ranges from 6.5s to 22.1s. The difference between them lies in how they work: QBSolv works locally on the CPU while D-Wave Hybrid Solver requires remote access via the Internet to a physically remote system shared between multiple users.

V. CONCLUSIONS

This paper has analyzed how assignment-like combinatorial optimization problems can be effectively solved through quantum technology tools. Specifically, we aimed to investigate this innovative computation technique, quantum computing, and explore the advantages and disadvantages that derive from it.

We considered a specific case-study concerning the allocation of passengers to seats on high-speed trains with the recent hygiene and health regulations on social distancing due to the COVID-19 pandemic. The experiments show that the quantum approach is a feasible way to solve the problem effectively.

ACKNOWLEDGMENTS

This study has been proposed by and developed with the contribution of the Quantum Computing Team of Data Reply S.r.l., Torino (Italy).

REFERENCES

[1] Marchenkova A., *"What's the difference between quantum annealing and universal gate quantum computers?"*, https://medium.com/quantum-bits/what-s-the-difference-between-quantum-annealing-and-universal-gate-quantum-computers-c5e5099175a1

[2] Booth, M. & Reinhardt, S.P. *"Partitioning Optimization Problems for Hybrid Classical / Quantum Execution"*, Technical Report (2017)

[3] Glover, F., Kochenberger, G. & Du, Y. *"Quantum Bridge Analytics I: a tutorial on formulating and using QUBO models."* 4OR-Q J Oper Res 17, 335–371 (2019). https://doi.org/10.1007/s10288-019-00424-y

[4] S.B. Hengeveld, N. Rubiano da Silva, D.S. Gonçalves, P.H. Souto Ribeiro, A. Mucherino, Solving the One-dimensional Distance Geometry Problem by Optical Computing, arXiv e-print, arXiv:2105.12118, version 1, May 2021.