Noname manuscript No.
(will be inserted by the editor)

The set partitioning problem in a quantum context

Rafael Cacao · Lucas R. C. T. Cortez · Jackson Forner · Hamidreza Validi · Ismael R. de Farias Jr. · Illya V. Hicks

Received: date / Accepted: date

Abstract The set partitioning problem and its decision variant (i.e., the exact cover problem) are combinatorial optimization problems that were historically crucial in the quantum optimization community. This problem is also employed in the main problem of the branch-and-price approach in many real-world optimization problems, including, but not limited to, redistricting and scheduling. Motivated by recent claims on the capability of quantum computers in "solving" hard combinatorial optimization problems, we propose a quadratic unconstrained binary optimization (QUBO) formulation for the set partitioning problem with tight penalty coefficients. We also employ five reduction techniques of Garfinkel and Nemhauser (*Operations Research*, 1969) to reduce the size of an existing set of benchmark instances. We finally use variational quantum eigensolver (VQE) as a heuristic to find feasible solutions for the problem. Our computational experiments show the efficacy of employing the tight penalty coefficients and the existing classical reduction techniques in the quantum context. Our codes and data are available on GitHub.

Keywords The set partitioning problem · quadratic unconstrained binary optimization · tight penalty coefficients · variational quantum eigensolver

Mathematics Subject Classification (2020) MSC code1 · MSC code2 · more

Rafael Cacao · Lucas R. C. T. Cortez · Hamidreza Validi · Ismael R. de Farias Jr.
Industrial, Manufacturing and Systems Engineering, Texas Tech University
E-mail: {rafael.cacao,lucas.cortez,hvalidi,ismael.de-farias}@ttu.edu

Jackson Forner
Operations Research and Engineering Management, Southern Methodist University
E-mail: jforner@mail.smu.edu

Illya V. Hicks
Computational Applied Mathematics and Operations Research, Rice University
E-mail: ivhicks@rice.edu

1 Introduction

The recent development of quantum computers and claims on quantum advantage[1] have motivated operations researchers to develop quadratic unconstrained binary optimization (QUBO) formulations for "solving" combinatorial optimization problems. Although there are severe doubts about the existence of quantum advantage, research shows that quantum machines can "solve" combinatorial optimization problems faster than their classical counterparts. While we provide no claim on quantum advantage, we explore the capability of a quantum simulator in finding feasible solutions for the set partitioning problem. We show that our proposed tight QUBO formulation, along with a nice set of polytime reduction techniques [6], can help improve the quality of solutions obtained by a quantum simulator.

Studying the set partitioning and exact cover problems in the quantum context dates back to 1998, when no large quantum computer was available. Farhi and Gutmann [4] reformulate the exact cover as a combinatorial branch-and-bound problem on a decision tree. They prove that if the classical approach can solve the problem in polytime, then the quantum strategy can do so. Farhi et al. [3] propose a quantum adiabatic evolution algorithm for the exact cover problem and claim that "future" quantum computers might be able to outperform the classical ones on hard instances of the NP-complete problems. Young et al. [16] explain how their quantum adiabatic algorithm (QAA), using Monte Carlo simulations, outperforms the classical Davis-Putnam algorithm for solving the exact cover problem with at most 128 variables. In subsequent work, Young et al. [17] employ the same simulation procedure to simulate the QAA for the instances of the exact cover problem that contain at most 256 variables. Furthermore, Vikstal et al. [15] use the quantum approximate optimization algorithm (QAOA) to solve the tail-assignment problem that is reduced to the exact cover problem in a branch-and-price approach. Svensson et al. [13] employ QAOA in a branch-and-price framework to solve the set partitioning problem.

In the optimization context, Garfinkel and Nemhauser [6] introduce the set partitioning problem as the "set covering [problem] with equality constraints." This hard problem is usually employed in the main problem of the branch-and-price approaches for solving large-scale and real-world optimization problems (e.g., political redistricting and crew scheduling). Lewis et al. [9] develop an unconstrained binary quadratic optimization model in which all the equality constraints are penalized in the objective function. Similarly, Alidaee et al. [1] propose a QUBO model for solving the set packing problem. One can easily convert a set packing problem formulation to a set partitioning one by adding slack variables. Glover et al. [7] propose a QUBO formulation for the set partitioning problem with arbitrary large penalty coefficients. Interested readers are referred to Punnen [12] for more information on QUBO models.

[1] The goal of demonstrating that a quantum computer with a tailored quantum algorithm can efficiently perform a task that its classical counterparts cannot.

2 Background

In this section, we provide some background on the set partitioning problem and on the variational quantum eigensolver (VQE) algorithm, a hybrid quantum-classical algorithm.

2.1 The set partitioning formulation

The set partitioning problem is a well-known NP-hard problem [5]. We define this problem as follows.

Problem: The set partitioning problem.
Input: Partition set P, $w_p \geq 0$ for each partition $p \in P$ and item set I.
Output: (if any exist) A partition $p_1, \ldots, p_\ell$ of the item set I with smallest integer ℓ such that each item is assigned to exactly one partition.

For every partition $p \in P$, decision variable x_p is 1 if partition p is selected, and 0 otherwise. For every partition $p \in P$, we also define I_p as the set of items that are included in partition p.

$$\min \sum_{p \in P} w_p x_p \tag{1a}$$

$$\sum_{p \in P : i \in I_p} x_p = 1 \qquad \forall i \in I \tag{1b}$$

$$x_p \in \{0, 1\} \qquad \forall p \in P. \tag{1c}$$

Here, constraints (1b) imply every item must be assigned to exactly one possible partition.

2.2 The variational quantum eigensolver

Peruzzo et al. [11] introduced the variational quantum eigensolver algorithm in the context of quantum chemistry. The problem is to minimize the total energy of the system represented by its Hamiltonian. The optimization is done in a hybrid quantum-classical manner:

1. A parameterized quantum circuit is used to prepare a quantum state that is used to estimate the expectation value of the Hamiltonian.
2. Then a classical solver will vary the circuit's parameters to try to minimize the expectation value of the energy associated with the Hamiltonian.
3. Once the parameters are updated, the parameterized circuit generates another quantum state to calculate a new expectation value for the energy, initiating the algorithm loop again.

The ground state of a system is the eigenvector that minimizes the eigenvalue of the system's Hamiltonian H. Let $\mathbb{A}$ be the index set of the eigenvectors of H. Then for every $i \in \mathbb{A}$, we have $H|h_i\rangle = \lambda_i |h_i\rangle$ with $|h_i\rangle$ and λ_i be the corresponding eigenvector and eigenvalue, respectively. Let $\left|h_{gs}\right\rangle$ be an eigenvector with the smallest eigenvalue λ_{gs}[2]. Then for every $i \in \mathbb{A}$, we have

$$\left\langle h_{gs}\middle|H\middle|h_{gs}\right\rangle = \lambda_{gs} \leq \lambda_i = \langle h_i|H|h_i\rangle .$$

So, we can write H as a diagonal matrix using its eigenvectors.

$$H = \sum_{i\in\mathbb{A}} \lambda_i |h_i\rangle \langle h_i| .$$

Let $|\psi\rangle$ be a normalized state with expected value $\langle\psi|H|\psi\rangle$. The following argument shows that $\langle\psi|H|\psi\rangle$ is a valid upper bound for the ground state eigenvalue λ_{gs}.

$$\begin{aligned}
\langle\psi|H|\psi\rangle &= \left\langle \psi \middle| \sum_{i\in\mathbb{A}} \lambda_i |h_i\rangle \langle h_i| \middle| \psi \right\rangle \\
&= \sum_{i\in\mathbb{A}} \lambda_i \langle\psi|h_i\rangle \langle h_i|\psi\rangle \\
&= \sum_{i\in\mathbb{A}} \lambda_i |\langle\psi|h_i\rangle|^2 \\
&\geq \sum_{i\in\mathbb{A}} \lambda_{gs} |\langle\psi|h_i\rangle|^2 \\
&= \lambda_{gs}.
\end{aligned}$$

By iterating over different states, one can tighten the bound around the ground state energy of a system. Starting with an initial guess $|\psi_0\rangle$, a parameterized quantum circuit, represented by the unitary linear transformation $U(\theta)$, acts on $|\psi_0\rangle$ and gives us an estimate for $\left\langle\psi(\theta)\middle|H\middle|\psi(\theta)\right\rangle$. Then, a classical optimizer minimizes the return value of the circuit by varying the values of θ.

3 QUBO Formulations

One can move constraints (1b) with appropriate penalty coefficients to the objective function (1a) and write a QUBO formulation. The QUBO formulation corresponding to the integer programming (IP) model (1) can be written as

$$\min z(x) := \sum_{p\in P} w_p x_p + \sum_{i\in I} c_i \left(\sum_{p\in P : i\in I_p} x_p - 1 \right)^2 . \tag{2}$$

Assuming the feasibility of the set partitioning instances and nonnegativity of w, we provide two types of penalty coefficients: (i) naive, and (ii) tight.

[2] Here, index gs stands for ground state.

Remark 1 (Naive penalty coefficients) Let c be a penalty vector with $c_i > \sum_{p \in P} w_p$ for every item $i \in I$ and $\hat{x}$ be an optimal solution of the QUBO formulation (2). Then $\hat{x}$ is optimal for the IP model (1).

We propose a polytime row-and-column elimination algorithm that returns a tight penalty coefficient vector. Algorithm 1 considers the worst-case penalty coefficients at each iteration and eliminates a row and a column correspondingly.

Algorithm 1 Calculate penalty coefficients

Require: (I, P)
1: let $M \leftarrow I$ and $N \leftarrow P$ be the row set and column set, respectively.
2: $c^* \leftarrow 0$
3: **for** every row $i \in M$ **do**
4: let $w^i_{\max} := \max_{p \in P: i \in I_p}\{w_p\}$
5: **while** $M \neq \varnothing$ and $N \neq \varnothing$ **do**
6: let $i_{\max} \in \operatorname{argmax}_{i \in M}\{w^i_{\max}\}$ (if tie, choose a row whose next largest w is minimum)
7: let $p_{\max}$ be a partition with cost $\max_{i \in M}\{w^i_{\max}\}$ (if tie, choose a partition with minimum $|I_p|$)
8: $c^* \leftarrow c^* + w_{i_{\max}}$
9: $M \leftarrow M - \{i_{\max}\}$
10: $N \leftarrow N - \{p_{\max}\}$
11: **return** c^*

Let $m := |M|$ and $n := |N|$ be the number of rows and columns corresponding to the number of items and partitions, respectively. The following proposition shows that our proposed algorithm runs in polytime.

Proposition 1 *Algorithm 1 runs in* $\min\{m, n\}O(m \log m + mn)$ *time.*

Proof We note that lines (3)-(4) run in $mO(n \log n)$. Regarding the *while* loop in line (5), we have $\min\{m, n\}$ iterations. At each iteration of the *while* loop, we have time complexity (i) $m \log m$ for sorting rows and (ii) mn for breaking ties. Hence, we have time complexity $\min\{m, n\}O(m \log m + mn)$ in total.

The following proposition provides a tight penalty coefficient vector for the QUBO formulation (2).

Proposition 2 *Let c be a penalty vector with $c_i > c^*$ for every item $i \in I$, where c^* is the penalty coefficient returned by Algorithm 1. Suppose $\hat{x}$ is an optimal solution of the* QUBO *model* (2)*. Then $\hat{x}$ is optimal for IP model* (1)*.*

Proof We show that for every item $i \in I$, we have $\sum_{p \in P: i \in I_p} \hat{x}_p = 1$. Suppose not. Then there is an item $j \in I$ for which $\sum_{p \in P: j \in I_p} \hat{x}_p \neq 1$. Because $w_p \geq 0$ for any partition $p \in P$ and by the feasibility of the instances, there is a solution x^* for which (i) $\sum_{p \in P: j \in I_p} x^*_p = 1$, and (ii) its objective value is less than that of $\hat{x}$ as

$$z(x^*) \leq c^* < c_j \leq z(\hat{x}).$$

Here, the first inequality holds by the feasibility of x^*. The second inequality holds by the assumption. The last inequality holds because $\sum_{p \in P : j \in I_p} \hat{x}_p \neq 1$. This contradicts optimality of $\hat{x}$ for the QUBO formulation (2).

4 The Set Partitioning Hamiltonian

We recall $m := |I|$ and $n := |P|$ as the sizes of sets P and I, respectively. Without loss of generality, we define $I := \{1, \dots, m\}$ and $P := \{1, \dots, n\}$. For every item $i \in I$ and every partition $p \in P$, we define $a_{ip} = 1$ if $i \in I_p$ and $a_{ip} = 0$ otherwise. Then, we have

$$\sum_{p \in P} w_p x_p + \sum_{i \in I} c_i \left(\left(\sum_{p \in P : i \in I_p} x_p \right) - 1 \right)^2 = \sum_{p \in P} w_p x_p + \sum_{i \in I} c_i \left(\left(\sum_{p \in P} a_{ip} x_p \right) - 1 \right)^2 .$$

To convert the objective function into a Hamiltonian, we need to substitute constants c for $c \cdot I^{\otimes n}$ and x_p for $\frac{1}{2} I^{\otimes n} - Z_p$ with

$$Z_p = \left(\bigotimes_{j \in P : j < p} I \right) \otimes Z \otimes \left(\bigotimes_{j \in P : j > p} I \right)$$

as described by Glover et al. [7]. Thus, we have

$$\begin{aligned} H &= \frac{1}{2} \sum_{p \in P} w_p \left(I - Z_p \right) + \sum_{i \in I} c_i \left[\frac{1}{2} \left(\sum_{p \in P} a_{ip} \left(I - Z_p \right) \right) - I \right]^2 \\ &= \frac{1}{2} \sum_{p \in P} w_p \left(I - Z_p \right) + \sum_{i \in I} c_i \left[I - \left(\sum_{p \in P} a_{ip} \left(I - Z_p \right) \right) \right. \\ &\left. + \frac{1}{4} \left(\sum_{p \in P} \sum_{q \in P} a_{ip} a_{iq} \left(I - Z_p - Z_q + Z_p Z_q \right) \right) \right] . \end{aligned}$$

One can rewrite H without using parameters a as follows.

$$H = I - \left(\sum_{p \in P : i \in I_p} (I - Z_p) \right) + \frac{1}{2} \sum_{p \in P} w_p (I - Z_p) + \sum_{i \in I} c_i \left[\frac{1}{4} \left(\sum_{p \in P : i \in I_p} \sum_{q \in P : i \in I_q} (I - Z_p - Z_q + Z_p Z_q) \right) \right].$$

5 Computational Enhancement

Matrix reduction (i.e., deleting some rows and columns of a matrix) has a long history in mixed integer programming [2,8]. Specifically, Garfinkel and Nemhauser [6] proposed five reduction techniques for the set partitioning problem that are explained below. In Section 6, we observe how these reductions help us to solve instances of the set partitioning problem on a quantum simulator.

Reduction 1: If there is an empty row in A (i.e., a row with all 0 entries), then the set-partitioning problem has no solution.

Reduction 2: If there is a row with exactly one unit entry, say in position t, then we must have $x_t = 1$ in every solution of the problem. For every $j \in \{1, 2, \ldots, m\}$, we remove row R_j with $a_{jt} = 1$ because the j^{th} equality of $Ax = b$ already holds with $x_t = 1$. Furthermore, we can set $x_q = 0$ for every $q \in \{1, 2, \ldots, n\}$ with $a_{jq} = 1$ and $q \neq t$.

Reduction 3: For every pair of rows $\{i, j\} \in \binom{[m]}{2}$ with $R_j \geq R_i$, we delete row R_j. Moreover, we delete any column A_t such that $a_{jt} = 1$ and $a_{it} = 0$ and set the corresponding variable $x_t = 0$.

Reduction 4: For every column index $t \in \{1, \ldots, n\}$, we delete column A_t and set the corresponding variable $x_t = 0$ if $A_t = \sum_{j \in \Omega} A_j$ and the cost coefficients satisfy $c_t \geq \sum_{j \in \Omega} c_j$ for some subset $\Omega \subseteq \{1, \ldots, n\} \setminus \{t\}$. We propose an IP model to find columns that this reduction technique can remove. For every $t \in [n]$, we have

$$\min \sum_{j \in [n] \setminus \{t\}} c_j y_j \tag{3a}$$

$$A[t] = \sum_{j \in [n] \setminus \{t\}} A_j y_j \tag{3b}$$

$$c_t \geq \sum_{j \in [n] \setminus \{t\}} c_j y_j \tag{3c}$$

$$y_j \in \{0, 1\} \qquad \forall j \in [n] \setminus \{t\}. \tag{3d}$$

Reduction 5: Suppose that neither $R_j \geq R_k$ nor $R_k \geq R_j$ for rows R_j and R_k. We define $J := \{t \in \{1,\ldots,n\} \mid a_{jt} > a_{kt}\}$ and $K := \{t \in \{1,\ldots,n\} \mid a_{kt} > a_{jt}\}$. If there exists a row R_ℓ with $a_{\ell t} = 1$ for all $t \in K$, and there is some $t \in J$, say $t = q$, with $a_{\ell q} = 1$, then we have $x_q = 0$.

Table 1 illustrates the reduction results on an existing set of instances provided by Svensson et al. [13]. In this set of instances, the number of items is fixed to 24, and the number of partitions varies between 6 to 20. The table shows that the classical reduction techniques of Garfinkel and Nemhauser [6] reduce the number of items and partitions by up to 83 and 90 percent, respectively. We also note that the number of items is reduced by at least 4 units for every benchmark instance, i.e., 17% of item reduction for every instance.

Table 1: Reduction results on the benchmark set partitioning instances of Svensson et al. [13] with 24 items.

	Original num. of		New num. of		Reduction (%)	
ID	items	parts.	items	parts.	items	parts.
6.1	24	6	**6**	**2**	75.00	66.67
6.2	24	6	**4**	**4**	83.33	33.33
6.3	24	6	**8**	6	66.67	00.00
8.1	24	8	**6**	**2**	75.00	75.00
8.2	24	8	**9**	**4**	62.50	50.00
8.3	24	8	**8**	**6**	66.67	25.00
8.4	24	8	**14**	8	41.67	00.00
10.1	24	10	**10**	**2**	58.33	80.00
10.2	24	10	**8**	**4**	66.67	60.00
10.3	24	10	**12**	**6**	50.00	40.00
10.4	24	10	**16**	**8**	33.33	20.00
10.5	24	10	**10**	10	58.33	00.00
12.1	24	12	**7**	**2**	70.83	83.33
12.2	24	12	**12**	**4**	50.00	66.67
12.3	24	12	**13**	**6**	45.83	50.00
12.4	24	12	**15**	**8**	37.50	33.33
14.1	24	14	**12**	**2**	50.00	85.71
14.2	24	14	**20**	**4**	16.67	71.43
14.3	24	14	**18**	**6**	25.00	57.14
14.4	24	14	**18**	**8**	25.00	42.86
14.5	24	14	**20**	**10**	16.67	28.57
14.6	24	14	**14**	**12**	41.67	14.29
20.1	24	20	**20**	**2**	16.67	90.00
20.2	24	20	**20**	**4**	16.67	80.00
20.3	24	20	**18**	**6**	25.00	70.00
20.4	24	20	**18**	**10**	25.00	50.00
20.5	24	20	**18**	**14**	25.00	30.00
20.6	24	20	**20**	**14**	16.67	30.00
20.7	24	20	**20**	**16**	16.67	20.00
20.8	24	20	**18**	**18**	25.00	10.00
20.9	24	20	**20**	20	16.67	00.00

6 Computational Experiments

The computational experiments are divided into two parts. In the first part, we fix the circuit configuration setup to run our experiments on the benchmark instances of Svensson et al. [13] in the second part. So, we generate *feasible* instances with partitions of size between 4 and 16 and an item set of size 18. For each partition size, we generate 200 random instances that result in 2,600 *feasible* random instances in total. The second set of experiments aims to test the computational power of tight penalty coefficients and the classic reductions on a set of 31 benchmark instances provided by Svensson et al. [13]. These instances have partition sizes of 6 to 20 with 24 items. We generate the weights associated with all partitions randomly within a range of natural numbers from 1 to 10. Computational experiments are conducted on the Nocona partition at Texas Tech University's High-Performance Computing Center (HPCC). The machines ran on CentOS 8.1, with 240 PowerEdge C6252 nodes and 2 AMD EPYC™ 7702 processors per node. We use the open-source library Qiskit [14] in our implementations. The VQE algorithm employs COBYLA as the classical optimizer. We check the feasibility of solutions by the Gurobi solver. Our codes and data are available at our GitHub repository.

6.1 Circuit configurations and setup

In this section, we test different circuit configurations to find an appropriate setup for running our final experiments on the benchmark instances of Svensson et al. [13]. We run our configuration experiments based on (i) three single-qubit gate combinations, (ii) two control gate combinations, and (iii) two entanglement strategies. This results in twelve different configurations that are tested on 200 randomly generated feasible instances for partition sizes between 4 and 16 and the item set size of 18. Thus, we have a total of 2,600 instances and 31,200 experiments.

In our experiments, the single qubit gates are (i) rotation-Y gate (ry), (ii) rotation-Y gate followed by rotation-Z gate (ryrz) and (iii) rotation-Z gate followed by rotation-Y gate (rzry). Furthermore, the control gates are (i) control-Y gate (cy) and (ii) control-Y gate followed by control-Z gate on the same pair of qubits (cycz). We also employ two entanglement strategies: (i) linear entanglement, and (ii) structured entanglement. Linear entanglement refers to control gates between neighbor qubits. Structured entanglement refers to control gates between qubits representing partitions containing a similar item. We give this name as it depends on the structure of the set partitioning problem.

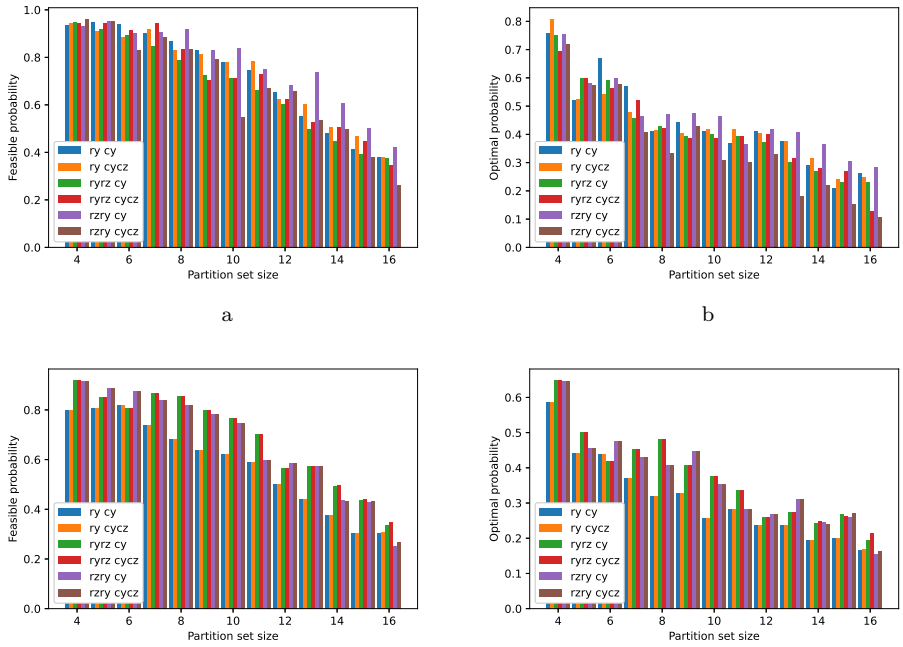

Fig. 1: Average feasible and optimal probability for different number of partitions and circuit configurations. Figures (a) and (b) show the average feasible and optimal probabilities using linear entanglement. Figures (c) and (d) show the average feasible and optimal probabilities using structured entanglement.

We also employ tight penalty coefficients obtained by Algorithm 1. Furthermore, the ansatz is initialized as a random point by the VQE algorithm and the circuit depth is set to one. We measure the quality of the results in terms of the average probability of obtaining a feasible solution and the average probability of obtaining an optimal solution for each instance size. Figure 1 illustrates the average probability results in terms of the number of qubits (size of the partition set). When the number of qubits is 16 (the largest number of qubits in the experiment), Figure 1 shows that we obtain average feasible and optimal probabilities up to 42% and 28%, respectively. We note that these percentages are obtained under configuration rzry-cy with linear entanglement. This configuration performs best for larger values of qubits (i.e., between 13 and 16) in terms of average feasible and optimal probabilities. We also observe that introducing a problem-derived entanglement strategy (i.e., structured strategy) does not provide a clear computational advantage or disadvantage, as observed by Nannicini [10] too. Therefore, linear entanglement is preferred as it requires less preprocessing. For benchmark instances, we con-

duct our experiments under configuration rzry-cy with linear entanglement in Sections 6.2 and 6.3.

6.2 Tight vs. naive penalty coefficients

In this section, we discuss the effect of employing tight penalty coefficients, which are returned by Algorithm 1 in Section 3, against naive ones. To assess the effect of tight penalty coefficients, we compare the probabilities of obtaining feasible and optimal solutions under naive and tight penalty coefficients. Table 2 illustrates the effect of designing tight penalty coefficients on a set of benchmark instances provided by Svensson et al. [13]. Our experiments show the superiority of the QUBO formulation with tight penalty coefficients over the one with naive penalties. In better words, Table 2 shows that 29% of instances have a greater probability of obtaining feasible or optimal solutions under the tight policy. On the other hand, only 13% of instances have a greater probability of obtaining feasible or optimal solutions under the naive policy. We conjecture a better performance of the tight model for larger instances; however, we cannot test larger instances due to the limitations on the number of qubits we can test in our simulations. Furthermore, Table 2 shows that our quantum simulation cannot solve instances with 20 partitions. In Section 6.3, we will see how reduction techniques discussed in Section 5 can improve obtaining both feasible and optimal solutions for the benchmark instances with 20 partitions.

6.3 Final experiments

We conduct the final set of experiments on the benchmark instances of Svensson et al. [13] with the following considerations.

1. We employ the tight penalty coefficients returned by Algorithm 1 in Section 3;
2. We apply the reduction techniques of Garfinkel and Nemhauser [6] discussed in Section 5; and
3. We adopt the best circuit configuration discussed in Section 6.1, i.e., the single-qubit gates rotation-Z followed by rotation-Y and the linear entanglement of the control-Y.

Table 3 shows the performance improvement (in terms of the probability of obtaining feasible or optimal solutions) of the VQE for 68% of the benchmark instances. After applying the reductions and tight penalty coefficients, we obtain

- feasible solutions for 15 out of 31 instances that reach no feasible solution beforehand.
- optimal solutions for 16 out of 31 instances that reach no optimal solution beforehand.

Table 2: The VQE results with naive and tight penalty coefficients on the benchmark set partitioning instances of Svensson et al. [13] with 24 items.

	No. of	Prob. w naive		Prob. w tight	
ID	parts.	Feasible	Optimal	Feasible	Optimal
6.1	6	1.00	1.00	1.00	1.00
6.2	6	0.00	0.00	**1.00**	0.00
6.3	6	1.00	0.00	1.00	0.00
8.1	8	0.00	0.00	0.00	0.00
8.2	8	0.00	0.00	0.00	0.00
8.3	8	0.00	0.00	0.00	0.00
8.4	8	1.00	0.00	1.00	0.00
10.1	10	0.00	0.00	0.00	0.00
10.2	10	0.00	0.00	0.00	0.00
10.3	10	**0.99**	0.00	0.00	0.00
10.4	10	0.00	0.00	**0.99**	0.00
10.5	10	0.98	0.00	**1.00**	0.00
12.1	12	0.00	0.00	0.00	0.00
12.2	12	0.00	0.00	**0.94**	0.00
12.3	12	1.00	1.00	1.00	1.00
12.4	12	0.00	0.00	**0.02**	**0.02**
14.1	14	0.00	0.00	0.00	0.00
14.2	14	**0.01**	0.00	0.00	0.00
14.3	14	0.00	0.00	0.00	0.00
14.4	14	0.00	0.00	0.00	0.00
14.5	14	0.00	0.00	**0.01**	0.00
14.6	14	0.00	0.00	**0.01**	0.00
20.1	20	0.00	0.00	0.00	0.00
20.2	20	0.00	0.00	**0.02**	0.00
20.3	20	0.00	0.00	0.00	0.00
20.4	20	0.00	0.00	0.00	0.00
20.5	20	**0.01**	0.00	0.00	0.00
20.6	20	0.00	0.00	0.00	0.00
20.7	20	0.00	0.00	**0.18**	0.00
20.8	20	**0.05**	0.00	0.00	0.00
20.9	20	0.00	0.00	0.00	0.00

After applying the reductions and tight penalty coefficients, we also note that we obtain at least one feasible solution for 5 out of 9 instances with 20 partitions that reach no feasible solution beforehand.

7 Conclusion and future work

In this paper, we explore an important combinatorial optimization problem in the quantum context: the set partitioning problem. We propose tight penalty coefficients for the QUBO formulation of this problem. We also employ a classical set of reduction techniques to decrease the size of the instances on a quantum simulator. Our experiments show the efficacy of employing tight penalty coefficients and classical reductions in "solving" an existing set of experiments on a quantum simulator. For future works, one can

Table 3: The VQE results with and without reduction and tight penalty coefficients on the benchmark set partitioning instances of Svensson et al. [13] with 24 items.

		Prob. w/o reduction & tight pen. coeff.		Prob. w reduction & tight pen. coeff.	
ID	No. of parts.	Feasible	Optimal	Feasible	Optimal
6.1	6	1.00	1.00	1.00	1.00
6.2	6	1.00	0.00	1.00	**1.00**
6.3	6	1.00	0.00	1.00	**1.00**
8.1	8	0.00	0.00	**1.00**	**1.00**
8.2	8	0.00	0.00	**1.00**	**0.53**
8.3	8	0.00	0.00	**1.00**	0.00
8.4	8	1.00	0.00	1.00	0.00
10.1	10	0.00	0.00	**1.00**	**1.00**
10.2	10	0.00	0.00	**1.00**	**0.04**
10.3	10	0.00	0.00	0.00	0.00
10.4	10	0.99	0.00	**1.00**	**1.00**
10.5	10	1.00	0.00	1.00	0.00
12.1	12	0.00	0.00	**1.00**	**1.00**
12.2	12	0.94	0.00	**1.00**	**0.90**
12.3	12	1.00	1.00	1.00	1.00
12.4	12	0.02	0.02	**1.00**	**1.00**
14.1	14	0.00	0.00	**1.00**	**1.00**
14.2	14	0.00	0.00	**1.00**	**1.00**
14.3	14	0.00	0.00	**1.00**	**1.00**
14.4	14	0.00	0.00	**1.00**	0.00
14.5	14	**0.01**	0.00	0.00	0.00
14.6	14	**0.01**	0.00	0.00	0.00
20.1	20	0.00	0.00	**1.00**	**1.00**
20.2	20	0.02	0.00	**1.00**	**0.85**
20.3	20	0.00	0.00	**1.00**	**1.00**
20.4	20	0.00	0.00	**1.00**	**1.00**
20.5	20	0.00	0.00	**1.00**	0.00
20.6	20	0.00	0.00	0.00	0.00
20.7	20	**0.18**	0.00	0.00	0.00
20.8	20	0.00	0.00	0.00	0.00
20.9	20	0.00	0.00	**0.01**	0.00

- test the effect of tight penalty coefficients and reduction techniques for larger sets of instances as we believe that the effect of tight penalty coefficients should be more noticeable on larger instances of the set partitioning problem; and
- develop tight penalty coefficients and fixing procedures for other crucial optimization problems in the quantum context.

Acknowledgement

The authors acknowledge the High Performance Computing Center (HPCC) at Texas Tech University for providing computational resources that have contributed to the research results reported within this paper.

Dedication

This paper is dedicated to the memory of our coauthor, Ismael R. de Farias Jr., who sadly passed away on Wednesday, August 3rd, 2022.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Alidaee, B., Kochenberger, G., Lewis, K., Lewis, M., Wang, H.: A new approach for modeling and solving set packing problems. European Journal of Operational Research **186**(2), 504–512 (2008)
2. Balinski, M.L., Quandt, R.E.: On an integer program for a delivery problem. Operations research **12**(2), 300–304 (1964)
3. Farhi, E., Goldstone, J., Gutmann, S., Lapan, J., Lundgren, A., Preda, D.: A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. Science **292**(5516), 472–475 (2001)
4. Farhi, E., Gutmann, S.: Quantum computation and decision trees. Physical Review A **58**(2), 915 (1998)
5. Garey, M.R., Johnson, D.S.: Computers and intractability, vol. 174. Freeman San Francisco (1979)
6. Garfinkel, R.S., Nemhauser, G.L.: The set-partitioning problem: set covering with equality constraints. Operations Research **17**(5), 848–856 (1969)
7. Glover, F., Kochenberger, G., Hennig, R., Du, Y.: Quantum bridge analytics I: a tutorial on formulating and using qubo models. Annals of Operations Research pp. 1–43 (2022)
8. Kolner, T.: Some highlights of a scheduling matrix generator system. In: United Airlines, presented at the Sixth AGIFORS Symposium (1966)
9. Lewis, M., Kochenberger, G., Alidaee, B.: A new modeling and solution approach for the set-partitioning problem. Computers & Operations Research **35**(3), 807–813 (2008)
10. Nannicini, G.: Performance of hybrid quantum-classical variational heuristics for combinatorial optimization. Phys. Rev. E **99**, 013304 (2019). DOI 10.1103/PhysRevE.99.013304. URL `https://link.aps.org/doi/10.1103/PhysRevE.99.013304`
11. Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.H., Zhou, X.Q., Love, P.J., Aspuru-Guzik, A., O'Brien, J.L.: A variational eigenvalue solver on a photonic quantum processor. Nature Communications **5**, 4213 (2014)
12. Punnen, A.P.: The quadratic unconstrained binary optimization problem. Tech. rep., Springer (2022)
13. Svensson, M., Andersson, M., Grönkvist, M., Vikstål, P., Dubhashi, D., Ferrini, G., Johansson, G.: A hybrid quantum-classical heuristic to solve large-scale integer linear programs. arXiv preprint arXiv:2103.15433 (2021)
14. tA v, A., ANIS, M.S., Abby-Mitchell, Abraham, H., AduOffei, Agarwal, R., Agliardi, G., Aharoni, M., Ajith, V., Akhalwaya, I.Y., Aleksandrowicz, G., Alexander, T., Amy, M., Anagolum, S., Anthony-Gandon, Araujo, I.F., Arbel, E., Asfaw, A., Ashimine, I.E., Athalye, A., Avkhadiev, A., Azaustre, C., BHOLE, P., Bajpe, V., Banerjee, A., Banerjee, S., Bang, W., Bansal, A., Barkoutsos, P., Barnawal, A., Barron, G., Barron, G.S., Bello, L., Ben-Haim, Y., Bennett, M.C., Bevenius, D., Bhatnagar, D., Bhatnagar, P., Bhobe, A., Bianchini, P., Bishop, L.S., Blank, C., Bolos, S., Bopardikar, S., Bosch, S., Brandhofer, S., Brandon, Bravyi, S., Bryce-Fuller, Bucher, D., Burgholzer, L., Burov, A., Cabrera, F., Calpin, P., Capelluto, L., Carballo, J., Carrascal, G., Carriker, A., Carvalho, I., Chakrabarti, R., Chen, A., Chen, C.F., Chen, E., Chen, J.C., Chen, R., Chevallier, F., Chinda, K., Cholarajan, R., Chow, J.M., Churchill, S., CisterMoke, Claus, C., Clauss, C., Clothier, C., Cocking, R., Cocuzzo, R., Connor, J.,

Correa, F., Crockett, Z., Cross, A.J., Cross, A.W., Cross, S., Cruz-Benito, J., Culver, C., Córcoles-Gonzales, A.D., D, N., Dague, S., Dandachi, T.E., Dangwal, A.N., Daniel, J., DanielAja, Daniels, M., Dartiailh, M., Davila, A.R., Debouni, F., Dekusar, A., Deshmukh, A., Deshpande, M., Ding, D., Doi, J., Dow, E.M., Downing, P., Drechsler, E., Drudis, M.S., Dumitrescu, E., Dumon, K., Duran, I., EL-Safty, K., Eastman, E., Eberle, G., Ebrahimi, A., Eendebak, P., Egger, D., EgrettaThula, ElePT, Elsayed, I., Emilio, Espiricueta, A., Everitt, M., Facoetti, D., Farida, Fernández, P.M., Ferracin, S., Ferrari, D., Ferrera, A.H., Fouilland, R., Frisch, A., Fuhrer, A., Fuller, B., GEORGE, M., Gacon, J., Gago, B.G., Gambella, C., Gambetta, J.M., Gammanpila, A., Garcia, L., Garg, T., Garion, S., Garrison, J.R., Garrison, J., Gates, T., Gavrielov, N., Gentinetta, G., Georgiev, H., Gil, L., Gilliam, A., Giridharan, A., Glen, Gomez-Mosquera, J., Gonzalo, de la Puente González, S., Gorzinski, J., Gould, I., Greenberg, D., Grinko, D., Guan, W., Guijo, D., Guillermo-Mijares-Vilarino, Gunnels, J.A., Gupta, H., Gupta, N., Günther, J.M., Haglund, M., Haide, I., Hamamura, I., Hamido, O.C., Harkins, F., Hartman, K., Hasan, A., Havlicek, V., Hellmers, J., Herok, Ł., Hill, R., Hillmich, S., Hincks, I., Hong, C., Horii, H., Howington, C., Hu, S., Hu, W., Huang, C.H., Huang, J., Huisman, R., Imai, H., Imamichi, T., Ishizaki, K., Ishwor, Iten, R., Itoko, T., Ivrii, A., Javadi, A., Javadi-Abhari, A., Javed, W., Jianhua, Q., Jivrajani, M., Johns, K., Johnstun, S., Jonathan-Shoemaker, JosDenmark, JoshDumo, Judge, J., Kachmann, T., Kale, A., Kanazawa, N., Kane, J., Kang-Bae, Kapila, A., Karazeev, A., Kassebaum, P., Kato, T., Kehrer, T., Kelso, J., Kelso, S., van Kemenade, H., Khanderao, V., King, S., Kobayashi, Y., Kovi11Day, Kovyrshin, A., Krishna, J., Krishnakumar, R., Krishnamurthy, P., Krishnan, V., Krsulich, K., Kumkar, P., Kus, G., LNoorl, LaRose, R., Lacal, E., Lambert, R., Landa, H., Lapeyre, J., Lasecki, D., Latone, J., Lawrence, S., Lee, C., Li, G., Liang, T.J., Lishman, J., Liu, D., Liu, P., Lolcroc, M, A.K., Madden, L., Maeng, Y., Maheshkar, S., Majmudar, K., Malyshev, A., Mandouh, M.E., Manela, J., Manjula, Marecek, J., Marques, M., Marwaha, K., Maslov, D., Maszota, P., Mathews, D., Matsuo, A., Mazhandu, F., McClure, D., McElaney, M., McElroy, J., McGarry, C., McKay, D., McPherson, D., Meesala, S., Meirom, D., Mendell, C., Metcalfe, T., Mevissen, M., Meyer, A., Mezzacapo, A., Midha, R., Millar, D., Miller, D., Miller, H., Minev, Z., Mitchell, A., Mohammad, A., Moll, N., Montanez, A., Monteiro, G., Mooring, M.D., Morales, R., Moran, N., Morcuende, D., Mostafa, S., Motta, M., Moyard, R., Murali, P., Murata, D., Müggenburg, J., NEMOZ, T., Nadlinger, D., Nakanishi, K., Nannicini, G., Nation, P., Navarro, E., Naveh, Y., Neagle, S.W., Neuweiler, P., Ngoueya, A., Nguyen, T., Nicander, J., Nick-Singstock, Niroula, P., Norlen, H., NuoWenLei, O'Riordan, L.J., Ogunbayo, O., Ollitrault, P., Onodera, T., Otaolea, R., Oud, S., Padilha, D., Paik, H., Pal, S., Pang, Y., Panigrahi, A., Pascuzzi, V.R., Perriello, S., Peterson, E., Phan, A., Pilch, K., Piro, F., Pistoia, M., Piveteau, C., Plewa, J., Pocreau, P., Possel, C., Pozas-Kerstjens, A., Pracht, R., Prokop, M., Prutyanov, V., Puri, S., Puzzuoli, D., Pythonix, Pérez, J., Quant02, Quintiii, Rahman, R.I., Raja, A., Rajeev, R., Rajput, I., Ramagiri, N., Rao, A., Raymond, R., Reardon-Smith, O., Redondo, R.M.C., Reuter, M., Rice, J., Riedemann, M., Rietesh, Risinger, D., Rivero, P., Rocca, M.L., Rodríguez, D.M., RohithKarur, Rosand, B., Rossmannek, M., Ryu, M., SAPV, T., Sa, N.R.C., Saha, A., Ash-Saki, A., Salman, A., Sanand, S., Sandberg, M., Sandesara, H., Sapra, R., Sargsyan, H., Sarkar, A., Sathaye, N., Savola, N., Schmitt, B., Schnabel, C., Schoenfeld, Z., Scholten, T.L., Schoute, E., Schuhmacher, J., Schulterbrandt, M., Schwarm, J., Schweigert, P., Seaward, J., Sergi, Serrano, D.E., Sertage, I.F., Setia, K., Shah, F., Shammah, N., Shanks, W., Sharma, R., Shaw, P., Shi, Y., Shoemaker, J., Silva, A., Simonetto, A., Singh, D., Singh, D., Singh, P., Singkanipa, P., Siraichi, Y., Siri, Sistos, J., Sistos, J., Sitdikov, I., Sivarajah, S., Slavikmew, Sletfjerding, M.B., Smolin, J.A., Soeken, M., Sokolov, I.O., Sokolov, I., Soloviev, V.P., SooluThomas, Starfish, Steenken, D., Stypulkoski, M., Suau, A., Sun, S., Sung, K.J., Suwama, M., Słowik, O., Taeja, R., Takahashi, H., Takawale, T., Tavernelli, I., Taylor, C., Taylour, P., Thomas, S., Tian, K., Tillet, M., Tod, M., Tomasik, M., Tornow, C., de la Torre, E., Toural, J.L.S., Trabing, K., Treinish, M., Trenev, D., TrishaPe, Truger, F., TsafrirA, Tsilimigkounakis, G., Tsuoka, K., Tulsi, D., Tuna, D., Turner, W., Vaknin, Y., Valcarce, C.R., Varchon, F., Vartak, A., Vazquez, A.C., Vijaywargiya, P., Villar, V., Vishnu, B., Vogt-Lee, D., Vuillot, C., WQ, Weaver, J., Weidenfeller, J., Wieczorek, R., Wildstrom, J.A., Wilson,

J., Winston, E., WinterSoldier, Woehr, J.J., Woerner, S., Woo, R., Wood, C.J., Wood, R., Wood, S., Wootton, J., Wright, M., Xing, L., YU, J., Yaiza, Yang, B., Yang, U., Yao, J., Yeralin, D., Yonekura, R., Yonge-Mallo, D., Yoshida, R., Young, R., Yu, J., Yu, L., Yuma-Nakamura, Zachow, C., Zdanski, L., Zhang, H., Zheltonozhskii, E., Zidaru, I., Zimmermann, B., Zindorf, B., Zoufal, C., a matsuo, aeddins ibm, alexzhang13, b63, bartek bartlomiej, bcamorrison, brandhsn, nick bronn, chetmurthy, choerst ibm, comet, dalin27, deeplokhande, dekel.meirom, derwind, dime10, ehchen, ewinston, fanizzamarco, fs1132429, gadial, galeinston, georgezhou20, georgios ts, gruu, hhorii, hhyap, hykavitha, itoko, jeppevinkel, jessica angel7, jezerjojo14, jliu45, johannesgreiner, jscott2, kUmezawa, klinvill, krutik2966, ma5x, michelle4654, msuwama, nico lgrs, nrhawkins, ntgiwsvp, ordmoj, sagar pahwa, pritamsinha2304, rithikaadiga, ryancocuzzo, saktar unr, saswati qiskit, sebastian mair, septembrr, sethmerkel, sg495, shaashwat, smturro2, sternparky, strickroman, tigerjack, tsura crisaldo, upsideon, vadebayo49, welien, willhbang, wmurphy collabstar, yang.luh, yuri@FreeBSD, Čepulkovskis, M.: Qiskit: An open-source framework for quantum computing (2021). DOI 10.5281/zenodo.2573505
15. Vikstål, P., Grönkvist, M., Svensson, M., Andersson, M., Johansson, G., Ferrini, G.: Applying the quantum approximate optimization algorithm to the tail-assignment problem. Physical Review Applied **14**(3), 034009 (2020)
16. Young, A., Knysh, S., Smelyanskiy, V.: Size dependence of the minimum excitation gap in the quantum adiabatic algorithm. Physical Review Letters **101**(17), 170503 (2008)
17. Young, A., Knysh, S., Smelyanskiy, V.: First-order phase transition in the quantum adiabatic algorithm. Physical Review Letters **104**(2), 020502 (2010)