Quantum computing and materials science: A practical guide to applying quantum annealing to the configurational analysis of materials

B. Camino ✉ iD ; J. Buckeridge iD ; P. A. Warburton iD ; V. Kendon iD ; S. M. Woodley iD

Check for updates

View Online   Export Citation   CrossMark

28 June 2023 13:11:23

Quantum computing and materials science: A practical guide to applying quantum annealing to the configurational analysis of materials

View Online   Export Citation   CrossMark

B. Camino,[1,a)] (ID) J. Buckeridge,[2] (ID) P. A. Warburton,[3,4] (ID) V. Kendon,[5] (ID) and S. M. Woodley[1] (ID)

AFFILIATIONS

[1]Department of Chemistry, University College London, London WC1H 0AJ, United Kingdom
[2]School of Engineering, London South Bank University, SE10 AA London, United Kingdom
[3]London Centre for Nanotechnology, University College London, WC1H 0AH London, United Kingdom
[4]Department of Electronic & Electrical Engineering, University College London, WC1E 7JE London, United Kingdom
[5]Department of Physics, University of Strathclyde, G4 0NG Glasgow, United Kingdom

[a)]Author to whom correspondence should be addressed: b.camino@ucl.ac.uk.

ABSTRACT

Using quantum computers for computational chemistry and materials science will enable us to tackle problems that are intractable on classical computers. In this paper, we show how the relative energy of defective graphene structures can be calculated by using a quantum annealer. This simple system is used to guide the reader through the steps needed to translate a chemical structure (a set of atoms) and energy model to a representation that can be implemented on quantum annealers (a set of qubits). We discuss in detail how different energy contributions can be included in the model and what their effect is on the final result. The code used to run the simulation on D-Wave quantum annealers is made available as a Jupyter Notebook. This Tutorial was designed to be a quick-start guide for the computational chemists interested in running their first quantum annealing simulations. The methodology outlined in this paper represents the foundation for simulating more complex systems, such as solid solutions and disordered systems.

I. INTRODUCTION

Quantum chemistry is considered to be one of the first fields that could benefit from the development of quantum hardware and algorithms.[1] The first applications of quantum computing to solve the electronic structure problem date back to 2005[2] and the field has since then received significant attention.[3–13] A detailed overview of the state-of-the-art methods developed in this field can be found in the following two review papers (and references therein): McArdle *et al.*[3] focus on using digital quantum computers to solve the electronic structure problem *ab initio*, while Cao *et al.*[4] is aimed at the expert computational chemist and highlights the advantages quantum computing brings to the field.

At this early technological stage, several strategies to employ quantum effects in computing are being explored.[14] The two most well-developed at present are quantum gate-based[15] and quantum annealing.[16] The former can be thought of as the quantum equivalent of classical computers, where the classical bits are replaced by quantum bits (qubits) and classical gates are replaced by quantum gates, which operate in discrete time. The latter relies on the quantum adiabatic theorem and requires the problem of interest to be mapped to a physical set of qubits, whose states are allowed to evolve by slowly changing the Hamiltonian parameters of the quantum annealer, at all times remaining in the instantaneous ground state. However, current quantum annealers experience coupling to their thermal environment beyond the ~10–100 ns timescale.[17] Quantum annealers, therefore, behave in a way that is intermediate between the idealized quantum adiabatic theorem and inevitable experimental limitations.

From a chemical point of view, quantum gate-based algorithms have been employed mainly to solve the Fermion

28 June 2023 13:11:23

Hamiltonian that describes the motion of electrons in molecules and materials, i.e., the Shrödinger equation. Typically, the terms that account for electron correlation make this equation intractable to solve classically for all but the simplest systems. Thus, post-Hartree-Fock (post-HF) approaches have been developed such as configurational interaction (CI) and ideally applied in the full-CI limit.[18] Several algorithms have proved to be promising in this field. However, results are still limited due to the size and stability of available quantum hardwares.[3–5,19]

Quantum annealing in its current hardware implementation solves problems that can be formulated as an Ising model. Unfortunately, the electronic Hamiltonian cannot be mapped to the Ising form without an exponential growth of the number of variables required to solve the problem, as discussed from a classical point of view in Xia *et al.*[20] This introduces an intrinsic limitation in using quantum annealers for solving this type of problem. Nevertheless, several algorithms to solve the electronic structure problem through quantum annealing have been explored for small systems.[8–10,21–23]

Quantum annealing is suited to solving optimization and global minimum search problems.[24] Therefore, it finds applications in chemistry, where several configurations of the same structure are energetically viable. For example, in the analysis of the conformation of transition states,[25] configurations of dense polymers mixtures,[26] protein design,[27,28] screening of chemicals to discover new materials[29] and materials design[30] such as configurational analysis of doped materials.[31]

Here, we will focus on this last application using vacancies in graphene as a test model to explore how to map a chemical system to an Ising Hamiltonian and to the quadratic unconstrained binary optimisation (QUBO) model. These models are the respective natural language for QA and chemistry and they both belong to the binary quadratic model (*BQM*) class of problems. The goal of this Tutorial is to present an easy-to-follow example of how to apply quantum annealing to solve a chemical problem, which can be formulated as follows: "given a periodic lattice of atoms, represented by a unit cell, atom coordinates, and periodic boundary conditions (PBCs), what is the lowest energy configuration (or configurations) when N_{vac} sites are replaced by vacancies." This problem was studied by Carnevali *et al.*[31] using a discrete quadratic model, relying on two qubits per site, to represent the graphene structure. In this paper, a binary quadratic model, relying on a one-qubit-per-site implementation is used. Furthermore, the energy model introduced in by Carnevali *et al.*[31] is expanded in Sec. II G to better capture the chemistry of a more complex system.

In this paper, we are interested in providing a description of the procedure to build the energy model. The model system was purposefully chosen to be a simple test case. The energy model we discuss in Sec. II E describes the effect of the vacancy on the energy in a very simplified way. Nevertheless, by tuning the interaction between species (in this case carbon atoms and vacancies), the model can be extended, for example, to the simulation of solid solutions, spin configurations, and stoichiometric nanoparticles.

Despite the simplicity of the problem, it can be seen as the test model for the combinatorial optimization class of problems, which are often found in chemistry and materials science. The configuration space to explore is proportional to the binomial coefficient $\binom{N_{sites}}{N_{vac}}$ where N_{sites} is the number of sites in the unit cell and N_{vac} is the number of vacancies. For small values of N_{sites} or N_{vac}, the problem can be treated exactly by completing an exhaustive search, i.e., by calculating the energy of each possible outcome. However, as the size of the cell increases, the problem becomes quickly intractable (for example, for $N_{sites} = 50$ and $N_{vac} = 10$, there are more than 1 billion possible configurations). Various heuristic classical algorithms are available to tackle this type of problem, for example, Monte–Carlo methods,[32,33] random walks,[34] tree search optimization,[35] tabu algorithm,[36] and simulated annealing (SA).[37–41] However, these approaches suffer from limitations, such as getting stuck in local minima, sampling the same location more than once, and completely missing the best configurations because of the randomness of the sampling. A method to overcome these limitations is, therefore, required to analyze large configurational spaces encountered in materials science.

Simulated annealing borrows ideas from the thermal annealing of metals. A fictional temperature, which decreases as the annealing evolves, is introduced as a tunable parameter. By reducing the temperature, the size of the accessible configurational space decreases during annealing. The process typically stops when either a minimum threshold on the energy or the maximum number of iterations, defined by the user, has been reached.[42] The progress of the SA can be determined by using sophisticated methods such as holding points and checking to see if the holding point is within a local basin.[43] Quantum annealing (QA) is based on the same principles as SA but uses an auxiliary quantum mechanical system to find the minimum energy of the system. Unlike the SA algorithm that uses classical search techniques, QA exploits two quantum mechanical phenomena: superposition and tunneling.[24] The former means that at the beginning of the anneal, the qubits represent all possible final states, corresponding to all combinations of 1 and 0 simultaneously (see Appendix A). The latter means that the system can move from one configuration to another by tunneling through the energy barrier. Therefore, it is possible for QA to escape deep local minima that trap other optimization techniques.

The hardware for the simulation reported in this paper is made commercially available by D-Wave[44] via their quantum cloud service Leap™. The tools required to build the models and interact with Leap™ are distributed via the open-source project Ocean, hosted on the D-Wave GitHub page.[45] The Advantage_system5.2, which implements a Pegasus Quantum Processing Unit (QPU) Architecture,[46] was used for the results discussed in Sec. II.

The structure of this paper is chosen to guide the reader through (a) the mapping of the problem to the D-Wave hardware and (b) the building of the energy model. The methodology discussed here, together with the code made available at Ref. 47, gives the necessary resources to computational chemists and materials scientists interested in incorporating quantum annealing into their workflow. In Sec. II A, the analogies between simulations on classical and quantum computers are drawn. The Ising and QUBO models are introduced in Sec. II B. In Sec. II C, we discuss the workflow to map the vacancies in the graphene problem to the QA hardware and run the annealing. Section II D explores whether quantum annealing is needed for this type of simulation and what

28 June 2023 13:11:23

other methods could be used instead. In Sec. II E, the energy model is built first in terms of objectives and then, in Secs. II F and II G, constraints are added. The concept of qubit and its Bloch sphere representations are discussed in Appendix A. The mapping of the problem to the QPU hardware is presented in Appendix B. The idea of a graph representation of the chemical structure is introduced in Appendix C. The testing and tuning of the model parameters can be found in Appendix D. Considerations on the configuration multiplicity are discussed in Appendix E. Finally, the QA-simulated energy is compared to density functional theory (DFT) energies obtained by using the CRYSTAL[48] code and discussed in Appendix F.

II. THE MODEL AND RESULTS

A. Analogies with the "classical" quantum chemical approach

The maturity of the technology used in classical computers means that the mapping of the materials chemistry problem to the hardware level is, generally, taken care of by the design of the high-level programming language used to write the materials and molecular modeling software. On the other hand, being at the dawn of the application of quantum computing to chemistry, we are required to invest some time thinking about how to map the chemical system to the quantum hardware. In this section, we outline the differences between the classical and quantum computational materials chemistry workflows.

Computational chemists and materials scientists are familiar with translating a chemical system into a system of functions that

FIG. 1. Left: computational materials chemistry workflow for classical computers. Right: computational materials chemistry workflow for quantum computers. The vector **x** represents the lowest energy configuration of the system.

can be implemented on a classical computer. A simplified typical workflow is summarized on the left side of Fig. 1. First, the system is defined, from a chemical point of view, in terms of the number and types of atoms, and their possible positions. Periodic boundary conditions (PBCs) are applied to the unit cell to model extended periodic systems. This step represents the first abstraction. The system then needs to be defined in terms of mathematical functions to be represented on a computer. Although there are, in theory, no constraints on the types of functions that can be used, the properties of some of them are better suited at describing electron distributions, e.g., Slater Type Orbitals (STOs) and Gaussian functions for molecules, and plane waves for crystals. These functions are parameterized in order to describe the position of the atoms and the corresponding electron distribution in the systems under study.

The next step involves selecting the electronic Hamiltonian. These Hamiltonians have been developed and implemented on classical computers for several decades.[48–51] They each tend to perform better for a specific class of structures or properties, but they have, in general, broad applicability. Therefore, typically, the user will select an out-of-the-box method (HF,[52] LDA,[49] B3LYP,[53,54] etc.) implemented in the code they are using and will not devote time to developing the energy model for each simulation. The outcome of the simulation will be the ground state energy of the system, the single-particle energy levels and other properties, such as the optimized atomic geometry, vibrations or optical spectra, etc.

Now we examine the quantum computational materials chemistry workflow. In the first step, analogous to the classical workflow, the chemical system is defined in terms of atomic numbers, atomic coordinates, and PBCs. Then, the structure needs to be mapped to the quantum hardware. Some recent methods are able to map a classical basis set (e.g., STO-3G) directly to hardware. As discussed in Sec. I, this approach requires a large number of qubits.[55] In this work, we will use a different approach: each site of the structure is mapped to a single logical qubit. The problem can, therefore, be encoded on a binary vector (whose length depends on both the system being investigated and the energy model, as discussed below). A simple mapping function (and reverse mapping function) is developed in Sec. II E to perform this task for our chosen example system. For a problem to be solved on a quantum annealer, it needs to be expressed in a binary quadratic model (*BQM*) form. This model is discussed in Sec. II B.

B. The ising and QUBO models

As discussed in Sec. II A, we need to map our problem to a binary quadratic problem to be able to solve it on a quantum annealer. Among the *BQM* class of problems, there are the Ising Hamiltonians

$$E(s) = \sum_i h_i s_i + \sum_{j>i} J_{i,j} s_i s_j \quad s_i \in \{+1, -1\} \tag{1}$$

and the quadratic unconstrained binary optimisation (QUBO) method

$$E(x) = x^T Q x = \sum_i Q_{i,i} x_i + \sum_i \sum_{j>i} Q_{i,j} x_i x_j \quad x_i \in \{0, 1\}. \tag{2}$$
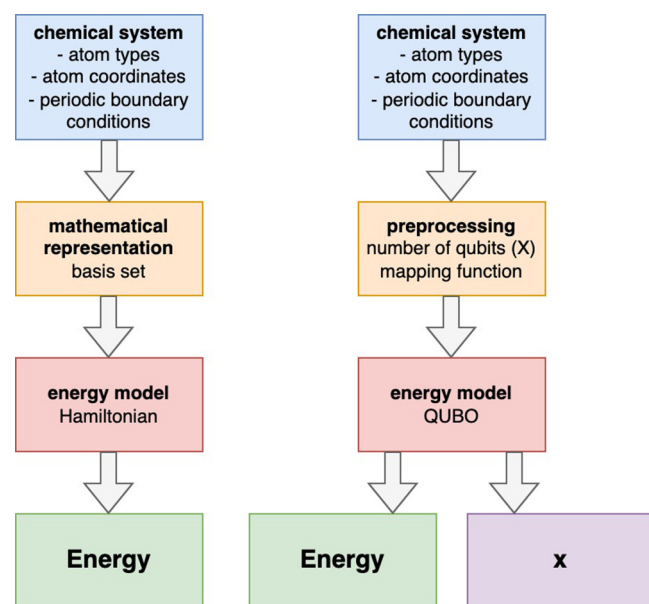
28 June 2023 13:11:23

The linear term in the right-hand side of Eq. (2) originates from the fact that the x_i variables are binary and, since $0^2 = 0$ and $1^2 = 1$, x_i^2 can be replaced by x_i.

During the quantum annealing, the state of the qubit is represented by a vector that can have any value on a sphere, called the Bloch sphere (see Appendix A for details). At the end of the annealing, the projection of the vector on the z axis of the sphere is measured for all qubits used to map the problem and returned as the result of the calculation. In the Ising model, the variables s_i can be either -1 or $+1$. The term h_i is the bias[56] applied to the qubit i and $J_{i,j}$ the coupling between qubits i and j. In the QUBO model, the variables can be either 0 or 1. A problem formulated as an Ising model can easily be converted into a QUBO problem and vice versa by using $s_i = (2x_i - 1)$. In this work, the QUBO formulation of the problem was chosen because it allows for a matrix representation of the Hamiltonian that is more familiar to the computational chemist.

The challenge is to formulate the problem we are solving as a minimization function,

$$\underset{\mathbf{x}}{\mathrm{argmin}} \left(\sum_i Q_{i,i} x_i + \sum_i \sum_{j>i} Q_{i,j} x_i x_j \right), \tag{3}$$

where the Q matrix, also called the objective function, is designed in such a manner that the minimum of Eq. (3) corresponds to the lowest energy of the system of interest. Often, problems of practical interest are subject to constraints. However, as the QUBO acronym suggests, in the formulation of the model we are bound to unconstrained optimizations. Therefore, the constraint will be included in the model as a penalty function. This function is designed to increase the energy of the solutions that do not respect the constraint. The development of both objective functions and constraints is the main topic of Secs. II E and II F.

C. The vacancies in graphene workflow

For the calculation of the relative energy of vacancies in graphene, the quantum workflow can be summarized in the following steps:

- step 1: build a graphene supercell large enough to contain the concentration of vacancies that we are interested in simulating. In this paper, we consider all possible structures using a 3×3 supercell of the two-atom primitive cell of graphene with up to three of the 18 carbon atoms removed to form the vacancies. This structure is shown in Fig. 2, where the labeling of the sites is defined, and can be found in the Jupyter Notebook available at Ref. 47. The pymatgen[57] library was used in this step because it allows for a python representation of the system that can be used with the Ocean tools;[45]
- step 2: each site is mapped to a single qubit. This mapping is done via a vector $\mathbf{x}$, with length N_{sites} (in this case 18). The element x_i at position i in the $\mathbf{x}$ vector will be 1 if the site i in the structure displayed in Fig. 2 is occupied by a C atom and 0 otherwise (vacancy);
- step 3: build the BQM representation of the energy model. The details of the model used in this article are discussed in Sec. II B.
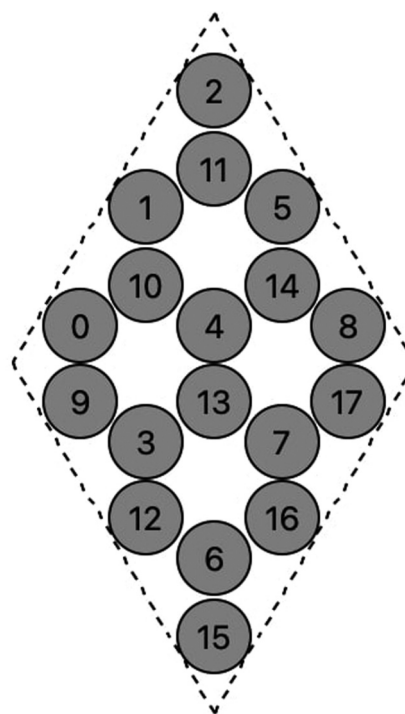

FIG. 2. Graphene 3×3 supercell of the two-atom primitive cell. Gray-filled circles represent the 18 sites for the carbon atoms. The numbering is the one used for the results reported below. The nearest neighbor atoms are the ones where the circles touch. This includes the ones across the periodic boundaries (for example, atoms labeled 0 and 15).

If a specific number of vacancies is chosen, then some additional constraints can be included in the model (this is discussed in detail in Sec. II F);

- step 4: run the quantum annealer. For the problem to be solved on the quantum annealer, each variable needs to be mapped to the hardware. Because of the limited connectivity of currently available hardware, more than one physical qubit might be needed to encode one variable (for a detailed description of this process, see Appendix B). This mapping is called minor embedding and for the results reported in Secs. II E– II G, the default mapping performed by the D-Wave API was selected. The user is also required to define how many times the quantum annealing will be performed. In this paper, the default value of one thousand times was used. The output of this step will be one $\mathbf{x}$ vector for each time the quantum annealing was run. Typically, due to quantum fluctuations and external sources of noise, a distribution of energy and configurations will be returned. If the problem is properly formulated, the lowest energy configuration should correspond to the global minimum of the function. Some of the configurations might be degenerate in energy and, in the specific case discussed in this paper, some configurations might be symmetry equivalent. This will require some post-processing in order to group symmetry-equivalent structures. Finally, by

reversing the mapping function, we can go from the configuration of the **x** vector to a chemical structure containing C atoms and vacancies. Some of the quantum annealing parameters, such as annealing time and chain strength, need to be optimized to improve the quality of the results. This parameter optimization is discussed in Appendix D.

D. Is quantum annealing needed for this problem?

For the graphene structure, we are interested in studying, for relatively small unit cells, an exhaustive search can be performed. In such an approach, the energy of all configurations containing a fixed number (N_{vac}) of vacancies is calculated. This can be achieved by building all the permutations of the **x** vector containing N_{vac} zeros and calculating their energy using the QUBO model. In an even larger exercise, the whole vacancy space could be explored. All the **x** vectors for all configurations with the number of vacancies going from 0 to N_{sites} are built and their energy is calculated. It is unlikely that structures with a large number of vacancies will be of interest since they correspond to non-chemically stable structures. However, if instead of studying vacancies, we were interested in solid solutions, then we might be interested in simulating high concentrations of both species.[58,59] The total number of possible permutations is $2^{N_{sites}}$, while for a fixed number of vacancies N_{vac}, the number of configurations can be calculated by using the binomial factor

$$\binom{N_{sites}}{N_{vac}} = \frac{N_{sites}!}{N_{vac}!(N_{sites} - N_{vac})!}. \tag{4}$$

The number of configurations to evaluate for a reasonable-sized cell and number of vacancies quickly becomes too large to be explored via an exhaustive search. For example, more than 10^{17} configurations exist for a 60-atom cell with a 40% level of substitution. For highly symmetrical structures, symmetry reduction can be used to drastically reduce the number of configurations.[60] In the following, we focus on a generic approach that does not require the system to belong to a particular symmetry group.

This kind of large configuration space can be explored by using classical global optimization heuristics, such as Monte–Carlo sampling,[32,33] and simulated annealing.[38–41] Currently, these heuristic approaches outperform QA. This is a hardware limitation and, as QPUs containing a larger number of more connected qubits are being developed, QA is expected to deliver much-improved performance than classical heuristics. This is because, as discussed in Sec. I, heuristic approaches suffer from several limitations. These limitations become more relevant for large configurational spaces characterized by relatively flat energy surfaces. Therefore, new optimization methods, such as QAs, are needed for the classes of problems encountered in materials science.

E. Objectives

1. Number of bonds

In this section, we will focus on encoding the energy model for the graphene structure as a QUBO problem. The model we are using is based on the total number of bonds. Therefore, we define an arbitrary energy for the presence/absence of a bond caused by the introduction of a vacancy. The contributions to the energy of the system are based on the following assumptions:

- the energy of the system only depends on the total number of bonds in the system;
- if site i and site j are nearest neighbors and they are both occupied by a C atom, the energy of the system decreases by 1 arbitrary unit;
- if site i and site j are nearest neighbors and one or both of them is vacant, the energy of the system is unchanged;
- only nearest neighbors are included in the model. The interaction between sites i and site j does not contribute to the energy of the system if they are more than one bond length apart.

The model outlined above can be expressed as a QUBO matrix

$$Q \begin{cases} Q_{i,j} = -A_{i,j} & \text{for } j > i, \\ Q_{i,j} = 0 & \text{for } j \leq i, \end{cases} \tag{5}$$

where **A** is the adjacency matrix of the structure (**A** and **Q** for the structure displayed in Fig. 2 is reported in the the Jupyter Notebook available at Ref. 47). Equation (5) defines **Q** as a sparse matrix with only $3N_{sites}$ non-zero elements. This is an important aspect to keep in mind when developing the energy model as explained in detail in Appendix B.

2. Vacancy-free case

As a starting point, no constraint on the number of vacancies in the system will be imposed. Therefore, the lowest energy configuration is the one where the **x** vector satisfies the following expression:

$$\underset{\mathbf{x}}{\operatorname{argmin}} \left(\sum_{i}^{N_{sites}} \sum_{j \geq i}^{N_{sites}} -A_{i,j} x_i x_j \right). \tag{6}$$

The result, in a Pandas dataframe[61] form, as returned by the Ocean API is reported in Fig. 3. The first 18 elements represent the

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.0	-27.0	100.0

FIG. 3. The quantum annealing output for the QUBO problem reported in Eq. (5) (vacancy-free structure) using the parameters discussed in Appendix D. The values in the columns labeled 0–17 are the 18 components of **x**. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed 1000 times.

component values of **x**. The QUBO matrix defined in Eq. (6) returns only one configuration: the one where no vacancies are present. This is an intuitive and reassuring result since the presence of vacancies increases the energy of the system. The average percentage of broken chains is also reported in Fig. 3. This is a measure of the quality of the annealing that depends on the way the problem is mapped to the hardware. This topic is discussed in detail in Appendix B.

F. Constraints and penalties

In the previous section, we observed how an unconstrained model (i.e., one that minimizes only objectives and is not subject to constraints) results in only one solution for the model problem. In this section, we want to model systems with a well-defined number of vacancies. Only vectors, **x**, that respect such a constraint will be considered acceptable solutions to the problem. If we are interested in an **x** vector that contains N_{vac} vacancies, we can formulate the constraint as

$$\sum_i^{N_{sites}} x_i = N_C, \qquad (7)$$

where the number of atoms is $N_C = N_{sites} - N_{vac}$. Equation (7) can be transformed into Eq. (8),

$$\sum_i^{N_{sites}} x_i - N_C = 0. \qquad (8)$$

To express the constraint as a penalty function, we look for a function of **x** whose minimum value is obtained when the constraint is respected. Results corresponding to configurations that do not respect the constraint may also be returned. However, if the model (objective and constraints) is well designed, these should be at noticeably higher energy and therefore both less likely to be returned by the annealer and easy to identify. By squaring Eq. (8), we obtain a function whose minimum value of zero occurs when the condition $\sum_i^{N_{sites}} x_i = N_C$ is satisfied,

$$\left(\sum_i^{N_{sites}} x_i - N_C\right)^2 \qquad (9a)$$

$$= \sum_i^{N_{sites}} x_i^2 + \sum_i^{N_{sites}} \sum_{j>i}^{N_{sites}} 2x_i x_j - \sum_i^{N_{sites}} 2x_i N_C + N_C^2 \qquad (9b)$$

$$= \sum_i^{N_{sites}} (1 - 2N_C)x_i + \sum_i^{N_{sites}} \sum_{j>i}^{N_{sites}} 2x_i x_j. \qquad (9c)$$

The expression in Eq. (9b) can be simplified to that shown as (9c) because:

- the term N_C^2 is constant, so does not change the relative energy of solutions corresponding to different configurations;

- the x_i^2 term can be replaced by the linear equivalent, x_i as in Eq. (2).

By combining the objective defined in Eq. (6) and the constraint defined in Eq. (9c), the following equation for the energy (**U**) of the system is Obtained:

$$\mathbf{U} = \alpha \underbrace{\sum_i^{N_{sites}} \sum_{j>i}^{N_{sites}} -A_{i,j} x_i x_j}_{\text{objective}} + \lambda \underbrace{\left(\sum_i^{N_{sites}} (1 - 2N_C)x_i + \sum_i^{N_{sites}} \sum_{j>i}^{N_{sites}} 2x_i x_j \right)}_{\text{constraint}}$$

$$= \lambda \sum_i^{N_{sites}} (1 - 2N_C)x_i + \sum_i^{N_{sites}} \sum_{j>i}^{N_{sites}} \left(2\lambda - \alpha A_{i,j}\right) x_i x_j. \qquad (10)$$

The elements of the resulting QUBO matrix can be expressed as

$$Q \begin{cases} Q_{i,j} = \lambda(1 - 2N_C) & \text{for } i = j, \\ Q_{i,j} = \left(2\lambda - \alpha A_{i,j}\right) & \text{for } j > i, \\ Q_{i,j} = 0 & \text{for } j < i. \end{cases} \qquad (11)$$

The factor λ, also called the Lagrange parameter, determines the weight of the constraint with respect to the objective. The α/λ ratio defines how much the final energy should be affected by not respecting the constraint. There is no simple *a priori* formula for selecting the α and λ values and a tuning procedure is required (for details see Appendix D).

The QUBO model defined in Eq. (11) corresponds to an upper triangular matrix where the total number of off diagonal elements is $\frac{N_{sites}^2 - N_{sites}}{2}$. For an 18-site model, this results in 135 off diagonal non-zero elements out of a total of 324 elements. Off diagonal terms introduce coupling between qubits. Due to the limited connectivity of the physical qubits in the D-Wave quantum annealer, the increased number of couplings will have an impact on the percentage of broken chains and the ratio of valid solutions (i.e., solutions that respect the constraint). This is discussed in Sec. II F 1 and in Appendix B.

1. Single-vacancy case

In this section, the QUBO model derived above will be used to simulate graphene structures containing one vacancy. The values of $\alpha = 1$ and $\lambda = 2$ for Eq. (10) were selected following the tuning procedure outlined in Appendix D. The results of the quantum annealing are reported in Fig. 4. Only the configurations that respect the constraint are included.

Comparing the results reported in Figs. 3 and 4, we can draw some interesting conclusions regarding the effect that the constraint (and the increased number of couplings) has on the results. First, 18 different configurations, corresponding to the vacancy in each of the different sites are obtained. All these configurations result in the same energy. An average value of 1.06% broken chains was obtained in the results reported in Fig. 4 compared to 0.0% found in Fig. 3. The reason for this increase in broken chains is explained in detail in Appendix B. By summing the multiplicity of the different

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.90	-24.0	4.1
p	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.26	-24.0	2.1
o	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2.87	-24.0	2.9
n	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.96	-24.0	5.2
m	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0.29	-24.0	3.8
l	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0.51	-24.0	2.2
k	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0.30	-24.0	3.7
j	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1.15	-24.0	2.9
i	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0.00	-24.0	2.2
h	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1.05	-24.0	10.1
g	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0.40	-24.0	1.4
f	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	2.74	-24.0	6.9
e	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0.45	-24.0	4.9
d	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1.03	-24.0	6.5
c	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	2.02	-24.0	3.3
b	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0.26	-24.0	4.2
q	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0.00	-24.0	1.6
r	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0.58	-24.0	5.7

FIG. 4. The quantum annealing output for the QUBO problem reported in Eq. (11) (single-vacancy structure) using the parameters discussed in Appendix D. The values in the columns labeled 0 to 17 are the 18 components of **x**. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed 1000 times.

configurations, we find that only 73.7% of the times the annealing was run a configuration containing one vacancy was returned. These are the configurations that respect the constraint imposed in Eq. (7) and are defined as feasible. The vectors, **x**, representing the other configurations (not reported in the table) would contain either zero or more than one zeros and are discarded in the post-processing.

When only one vacancy is introduced, we know that all the 18 configurations will be symmetry equivalent and, therefore, the configurations reported in Fig. 4 can be reduced by symmetry to the structure reported in Fig. 5. In this straightforward example, this can be done manually based on the knowledge we have about the symmetry of the system. For more complex configurations, we used the spglib library[62] for symmetry analysis of periodic structures, as implemented in pymatgen.[57]

2. Two and three vacancies

Graphene structures including two and three vacancies were studied using the same method described above for the one-vacancy structure. The QUBO matrix for this model is the same as the one

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.063	-24.0	73.7

FIG. 5. The quantum annealing output for the QUBO reported in Eq. (11) (single-vacancy structure) using the parameters discussed in Appendix D. Only symmetry non-equivalent structures are reported.

28 June 2023 13:11:23

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0.884	-22.0	24.5
b	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.561	-21.0	20.8
c	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0.688	-21.0	12.1
d	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1.054	-21.0	11.6
e	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0.383	-21.0	5.8

FIG. 6. The quantum annealing output for the QUBO problem reported in Eq. (11) ($N_{atoms} = 16$) using the parameters discussed in Appendix D. The values in the columns labeled 0 to 17 are the 18 components of **x**. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed 1000 times.

reported in Eq. (11), where a different number of vacancies is used in the definition of the constraint. For the two-vacancy structure, the N_C term is set equal to 16 (2 of the 18 sites are replaced by vacancies) and for the three-vacancy structure, the N_C term is set equal to 15 (3 of the 18 sites are replaced by vacancies). In Figs. 6 and 8, the configurations of two-and three-vacancy structures, respectively, are reported. Only the symmetry non-equivalent structures (i.e., the ones returned by a symmetry analysis) are included.

The presence of more than one vacancy introduces an interesting effect on the energy distribution. Unlike the one-vacancy structure, where all the feasible configurations resulted in the same energy of -24.0 arbitrary units, in the two-vacancy structures, two energies and five symmetry non-equivalent structures are returned. These are summarized in Fig. 6 and depicted in Fig. 7. Structure **a** is the lowest energy structure. In this structure, the two vacancies are found in neighboring sites. This can be explained by using the model described in Sec. II E 1. When the two vacancies are next to each other, they lead to a total of five broken bonds. On the other hand, the vacancies in structures **b**–**e** are found in non-neighboring sites, which leads to a total of six broken bonds in these structures.

All these structures have the same energy because the energy model used only accounts for nearest-neighbor interactions. Therefore, whether the vacancies are next-nearest neighbors (as in structure **b**) or three bond lengths apart (as in structures **d** and **e**), the energy is unchanged according to the energy model used. Interestingly, structure **b** contains some single-coordinated atoms, which, from a chemical point of view, would be expected to destabilize the system and increase its energy. In Sec. II G, we address this aspect by adding a new objective to the energy model.

When the QUBO model is constrained to return results containing three vacancies, 14 symmetry non-equivalent configurations and three energy values are returned. These are reported in Fig. 8. From the lowest to the highest energy, the corresponding structures have three vacancies in neighboring sites, two adjacent vacancies and one removed and three non-neighboring vacancies. One example per structure is depicted in Fig. 9.

In Fig. 10, three structures predicted by the model to have the same energy as structure **f** in Fig. 9 (-18.0 arbitrary units) are depicted. These configurations were selected specifically to highlight that some of the structures returned by the model will be

a -22.0   **b** -21.0   **c** -21.0   **d** -21.0   **e** -21.0
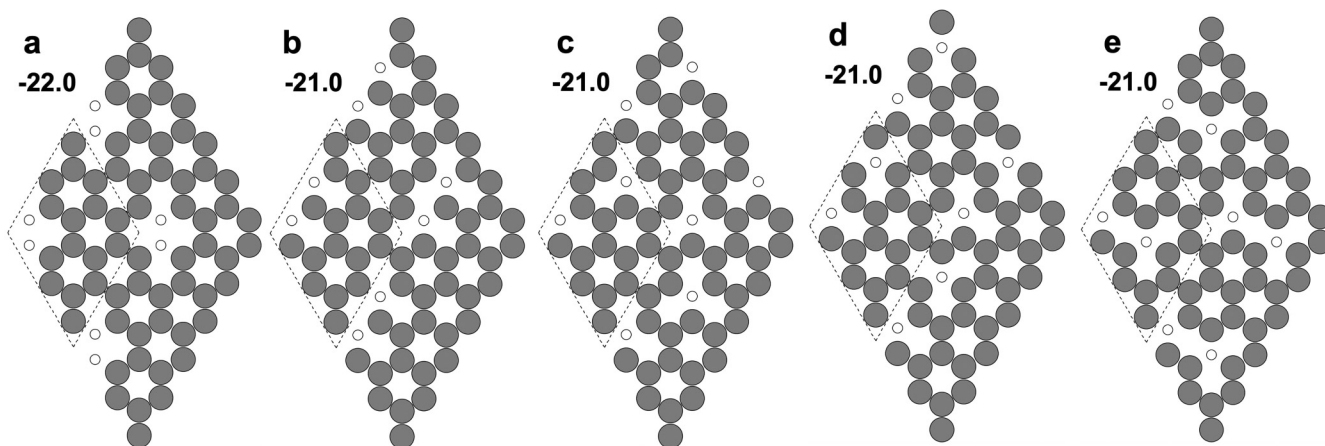
FIG. 7. 2×2 unit cells displaying the structure of the symmetry non-equivalent configurations reported in Fig. 6. Gray-filled circles and smaller open circles represent C atoms and vacancies, respectively. The labeling is the same used in Fig. 6.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0.681	-20.0	10.6
b	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1.651	-19.0	10.1
c	0	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0.947	-19.0	8.8
d	0	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	2.726	-19.0	5.3
e	0	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	0.811	-19.0	4.8
f	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.390	-18.0	0.8
g	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1.853	-18.0	0.6
h	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2.527	-18.0	1.1
i	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	2.829	-18.0	5.7
j	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	2.911	-18.0	2.1
k	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	2.471	-18.0	6.3
l	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	2.869	-18.0	3.1
m	0	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	3.969	-18.0	2.8
n	0	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	2.780	-18.0	0.4

FIG. 8. The quantum annealing output for the QUBO problem reported in Eq. (11) (two-vacancy structure) using the parameters discussed in Appendix D. The values in the columns labeled 0 to 17 are the 18 components of $\mathbf{x}$. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed 1000 times.
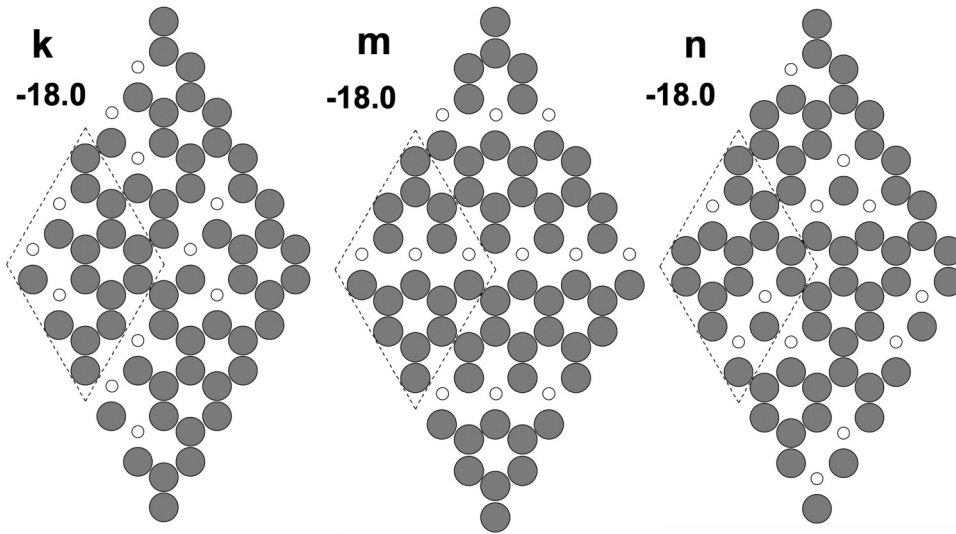
highly unstable from a chemical point of view. Structure **k** contains singly bonded C atoms, structure **m** can be seen as broken into five-atom wide strips and structure **n** contains isolated carbon atoms. In this context, unstable refers to a configuration where one or more atoms, upon geometry relaxation, would move to occupy a neighboring vacancy site and result in a different configuration. For example, in structures **n**, the displacement of the isolated carbon atom results in a structure that is equivalent by symmetry to structure **a** (see Sec. F for more details). This shows the need for an improved energy model that is able to identify these types of

a
-20.0

b
-19.0

f
-18.0
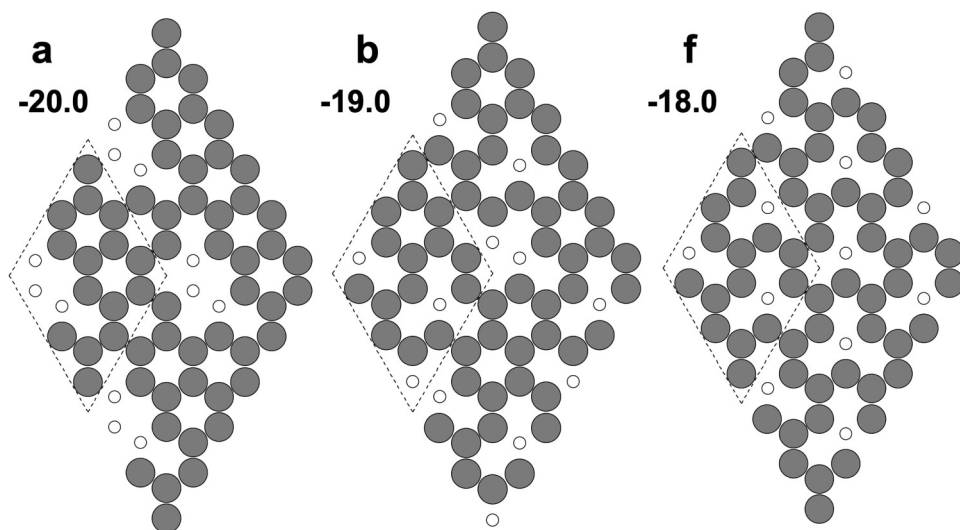
FIG. 9. Structure of the symmetry non-equivalent configurations reported in Fig. 8. Gray circles and white circles represent C atoms and vacancies, respectively. The labeling is the same used in Fig. 8.

k
-18.0

m
-18.0

n
-18.0

structures. This topic is discussed in Sec. II G where the coordination of the atoms is included as an objective in the model.

G. Coordination

In this section, we refine the model described above by adding an extra term to the QUBO formulation in Eq. (11). The model used for the results reported in Secs. II E and II F focused only on the total number of broken bonds. However, as we have seen for the three-vacancy structures, it does not take into account the local character of the vacancies allowing for non-physical structures to be returned.

From a chemical point of view, we know that breaking a bond from a lower coordination atom will be more energy expensive than breaking a bond from the same atom in a higher coordination environment. In order to include this effect in the energy model, we need to start looking at whether the sites surrounding atom i are occupied by atoms or vacancies. This can be achieved by analyzing the next-nearest neighbors for each site. We, therefore, define the **B** matrix, which is analogous to the adjacency matrix **A**, but where the element $B_{i,j} = 1$ if atoms i and j are next-nearest neighbors instead of nearest neighbors. The objective that we are interested in minimizing is

$$\underbrace{-\beta \sum_{i}^{N_{sites}} \sum_{j>i}^{N_{sites}} B_{i,j} x_i x_j}_{\text{coordination objective}}, \qquad (12)$$

where β is a scaling factor. The ratio between α and β will define how much the result is influenced by the coordination objective with respect to the number of broken bonds objective.

By adding the coordination objective to Eq. (10) we obtain the following equation:

$$\mathbf{U} = \underbrace{\alpha \sum_{i}^{N_{sites}} \sum_{j>i}^{N_{sites}} -A_{i,j} x_i x_j}_{\text{broken bonds objective}} + \underbrace{\beta \sum_{i}^{N_{sites}} \sum_{j>i}^{N_{sites}} -B_{i,j} x_i x_j}_{\text{coordination objective}} + \underbrace{\lambda \left(\sum_{i}^{N_{sites}} (1 - 2N_C) x_i + \sum_{i}^{N_{sites}} \sum_{j>i}^{N_{sites}} 2 x_i x_j \right)}_{\text{constraint}}$$

$$= \lambda \sum_{i}^{N_{sites}} (1 - 2N_C) x_i + \sum_{i}^{N_{sites}} \sum_{j>i}^{N_{sites}} \left(2\lambda - \alpha A_{i,j} - \beta B_{i,j} \right) x_i x_j \qquad (13)$$

and the corresponding QUBO matrix is

$$Q \begin{cases} Q_{i,j} = \lambda(1 - 2N_C) & \text{for } i = j, \\ Q_{i,j} = \left(2\lambda - \alpha A_{i,j} - \beta B_{i,j}\right) & \text{for } j > i, \\ Q_{i,j} = 0 & \text{for } j < i. \end{cases} \qquad (14)$$

In this work, we decided to define coordination as an objective rather than a constraint. This means that all the structures that respect the constraint on the number of vacancies defined in Eq. (11) will be considered acceptable. The coordination terms will only change the relative energy values to take into account the different coordination of the structure. Therefore, we expect that the

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0.602	-18.15	8.3
b	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0.427	-17.20	3.9
c	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1.234	-17.20	4.5
d	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	3.133	-17.15	11.7
e	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	3.488	-17.15	13.7
f	0	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1.587	-16.20	2.1
g	0	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	0.000	-16.20	0.2
h	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	2.862	-16.15	3.3
i	0	1	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	3.731	-16.15	6.4
j	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	2.964	-16.15	1.5
k	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.852	-16.10	4.5
l	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0.000	-16.05	0.7
m	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	2.779	-16.05	1.4
n	0	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1.112	-16.05	0.5

FIG. 11. Output of the quantum annealing for the QUBO reported in Eqs. (13) and (14) (three-vacancy structure) after symmetry reduction of the returned structures. The details of the parameters used can be found in Appendix D. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed 1000 times.

chemically unstable structures, such as the ones displayed in Fig. 10 will still be found, but they will have higher energy, for example, than structures **b–e** in Fig. 9.

Since in the QUBO formulation constraints are defined as penalty functions, the coordination objective can be transformed into a constraint by using the same procedure outlined in Sec. II F. Ensuring the ratio α/β is small enough and only considering valid the results whose energy is below a fixed threshold will result in structures not having, for example, single coordinated atoms. Whether to treat the coordination as an objective or constraint depends on the application and what is the goal of the simulation.

The results reported in Fig. 11 were obtained by using the coefficients $\alpha=1.0$, $\beta=0.05$, and $\lambda=2.0$. The quantum annealing still returns 14 structures, but, unlike the results reported in Fig. 8, 7 different energy values are found. In Fig. 12 the 6 structures reported in Figs. 9 and 10 are depicted together with the energy calculated using Eq. (13). When including coordination as an objective, the model is able to identify structures based on the local environment of the vacancy and, therefore, allows for easy identification of the non-physical structures.

III. CONCLUSIONS

In this work, we drew the analogy between the classical and the quantum materials chemistry workflows. They both rely on the abstraction of the structure onto a set of functions and the calculation of the energy of the system by means of a Hamiltonian. The identification of the lowest energy configuration of vacancies in graphene structures was used as a test model. The mapping of the problem to a set of qubits was discussed and the energy model was built in terms of objectives to minimize and constraints to respect. This process was broken down into its different components in order to help the computational chemistry community to build energy Hamiltonians suitable for quantum annealing, i.e., running on a quantum computer. The model we discussed can be used to identify the lowest energy state of a system characterized by several states close in energy. These types of problems are often found in chemistry and materials science. In Appendix D we showed how the parameters used to build the model (α, λ and β values) and in particular, their ratios are crucial in obtaining significant results. These parameters may require some "chemical intuition" or other theoretical input to choose values in more complex problems where the ground-state solution is not immediately known or
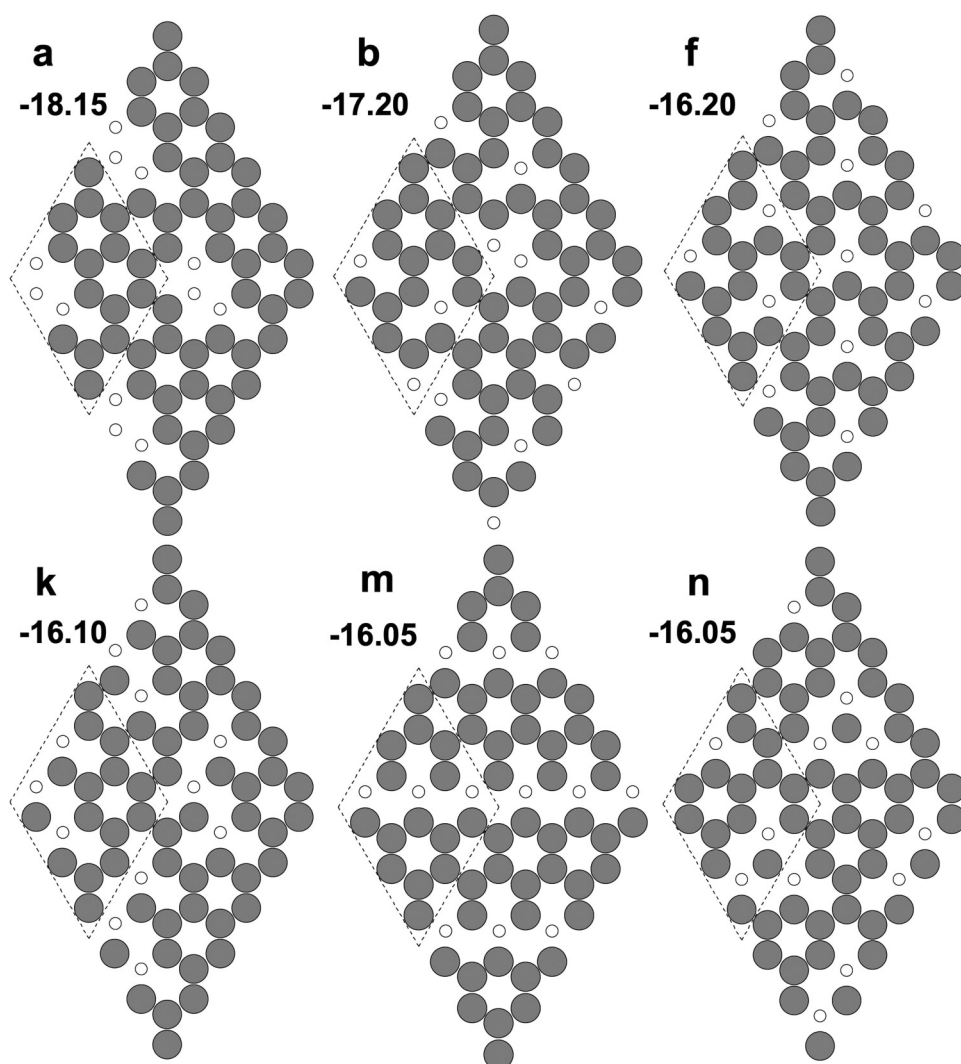
FIG. 12. Structure of six symmetry non-equivalent structures reported in Fig. 11, which were reported in Figs. 9 and 10 using the same labels. The energy is calculated using the QUBO model reported in Eq. (14). Gray-filled circles and smaller white circles represent C atoms and vacancies, respectively.

obvious. The results discussed in this paper aim at showing in a simple manner how quantum annealing is becoming an extra resource in the computational chemist's scientific toolbox. Furthermore, with the development of larger and more connected quantum hardware, these methods are believed to be able to tackle problems that cannot be solved by classical computers. The functions used to map the materials structures onto the D-Wave architecture and to perform the post-processing analysis developed for the results reported in this paper[47] represent the foundation for the development of a python library to be used by the materials chemistry community to directly access quantum annealing simulations much as the ArQTiC code[63] does for gate based quantum computing.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

B. Camino: Formal analysis (equal); Investigation (equal); Methodology (equal); Project administration (equal); Software (equal); Writing – original draft (equal). **J. Buckeridge:** Conceptualization (equal); Funding acquisition (equal);

Investigation (equal); Methodology (equal); Project administration (equal); Supervision (equal); Validation (equal); Writing – review & editing (equal). **P. A. Warburton:** Conceptualization (equal); Funding acquisition (equal); Investigation (supporting); Methodology (equal); Project administration (equal); Supervision (equal); Validation (equal); Writing – review & editing (equal). **V. Kendon:** Conceptualization (equal); Funding acquisition (lead); Methodology (supporting); Project administration (equal); Supervision (supporting); Validation (equal); Writing – review & editing (equal). **S. M. Woodley:** Conceptualization (equal); Funding acquisition (lead); Investigation (supporting); Methodology (equal); Project administration (equal); Resources (equal); Supervision (lead); Validation (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data and software that support the findings of this study are openly available at https://github.com/cmc-ucl/quantum_computing/tree/main/Quantum_computing_and_materials_science.

APPENDIX A: QUBITS AND THE BLOCH SPHERE

Classical bits of information are defined by a two-level classical state. On the other hand, quantum bits of information, called qubits, can be thought of as a two-state quantum-mechanical system. From a vectorial point of view, qubits are represented by a two-dimensional vector whose basis is

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{A1}$$

and any other state can be obtained as a linear combination of $|0\rangle$ and $|1\rangle$.

The Bloch sphere[64,65] is used as a graphical representation of a two-state quantum-mechanical system. In the context of quantum computing, the Bloch sphere is used to represent the state of a qubit, by defining the north and south poles (aligned along the z axis) as corresponding to the standard basis vectors $|0\rangle$ and $|1\rangle$.

During the calculation, the qubit can take any value on the surface of the Bloch sphere. Its projection on the z axis is proportional to the probability of measuring the state $|0\rangle$ or $|1\rangle$. The x axis and y axis components are the phase of the qubit. This phase is responsible for some of the unique properties of qubits with respect to classical bits.

Throughout this paper, when talking about the value of the qubit measured at the end of the quantum annealing, we are referring to the projection of the qubit along the z axis, which is returned as a $|0\rangle$ or $|1\rangle$ vector. For the sake of simplicity, when reporting results, such as different configurations, the braket notation is dropped and the states $|0\rangle$ and $|1\rangle$ are reported as 0 and 1, respectively.

APPENDIX B: MINOR EMBEDDING AND MAPPING OF THE PROBLEM TO THE QPU

Once we have formulated a problem in terms of a binary quadratic model, this needs to be mapped to the quantum annealer hardware. For small problems, each variable of the model can be mapped to a single physical qubit. The term physical qubit refers to the physical quantum system used by the quantum processing unit (QPU) to perform the calculation.

Because of the limited connectivity of the current QPU topology in D-Wave annealers, for larger problems, it might not be possible to assign each variable to a single physical qubit. When this happens, two or more physical qubits can be used to represent a single variable. These qubits form a so-called chain. The process of assigning a variable to more than one physical qubit is referred to as minor-embedding and can either be performed manually by the user or heuristically by the D-Wave API through a set of functions called *EmbeddingComposite*, which rely on the minor-miner library.[66] Therefore, users do not need to worry about performing the minor embedding by hand for each problem. Long chains will impact the reliability of the results. One way to reduce the number of chains is to minimize the number of couplings in our Ising model or off diagonal elements in the QUBO formulation of the problem.

Since the physical qubits in the chain represent one variable, they should all return the same value (all spin up or all spin down). When this is not the case, the chain is said to be broken. Several strategies to recover the value of the variable represented by a broken chain are available. In this work, the majority vote technique was used. This technique assigns the value of the physical qubits that was predominant in the chain to all the qubits in the chain.

From a physical point of view, a chain is formed by applying a ferromagnetic coupling among qubits. The strength of such a magnetic field is called chain strength and it is one of the parameters that need to be tuned. If the chain strength is too strong compared to the fields used to implement the *BQM*, the problem part of the Hamiltonian becomes negligible and only trivial solutions are returned. On the other hand, if the chain strength is too small, too many broken chains will be found and the solution might become unreliable. It is, therefore, important to take into account the coupling strength defined by the *BQM* when setting the chain strength. There are several strategies to determine what value will return the best results. The default technique when using the D-Wave Ocean API is the uniform torque compensation (UTC) method. By using the root mean square of the coupling of the *BQM*, this method attempts to compensate for the torque that would break the chain. The chain strength is calculated as

$$C_{UTC} = \sigma \sqrt{\frac{1}{J_{i,j}} \sum_i^{N_C} \sum_{j>i}^{N_C} J_{i,j}^2} \sqrt{\frac{1}{N_C} \sum_i^{N_C} N_i^c} \quad \text{for} \quad J_{i,j} > 0, \tag{B1}$$

where N_i^c is the number of couplings ($J_{i,j}$) for atom i in the model. The prefactor σ can be selected by the user. Values of σ ranging from 0.1 to 1.0 were tested and are reported in Fig. 15 in Appendix D 2.

The D-Wave API allows the user to visualize, on a graphical interface, the mapping of the problem to the QPU. The images
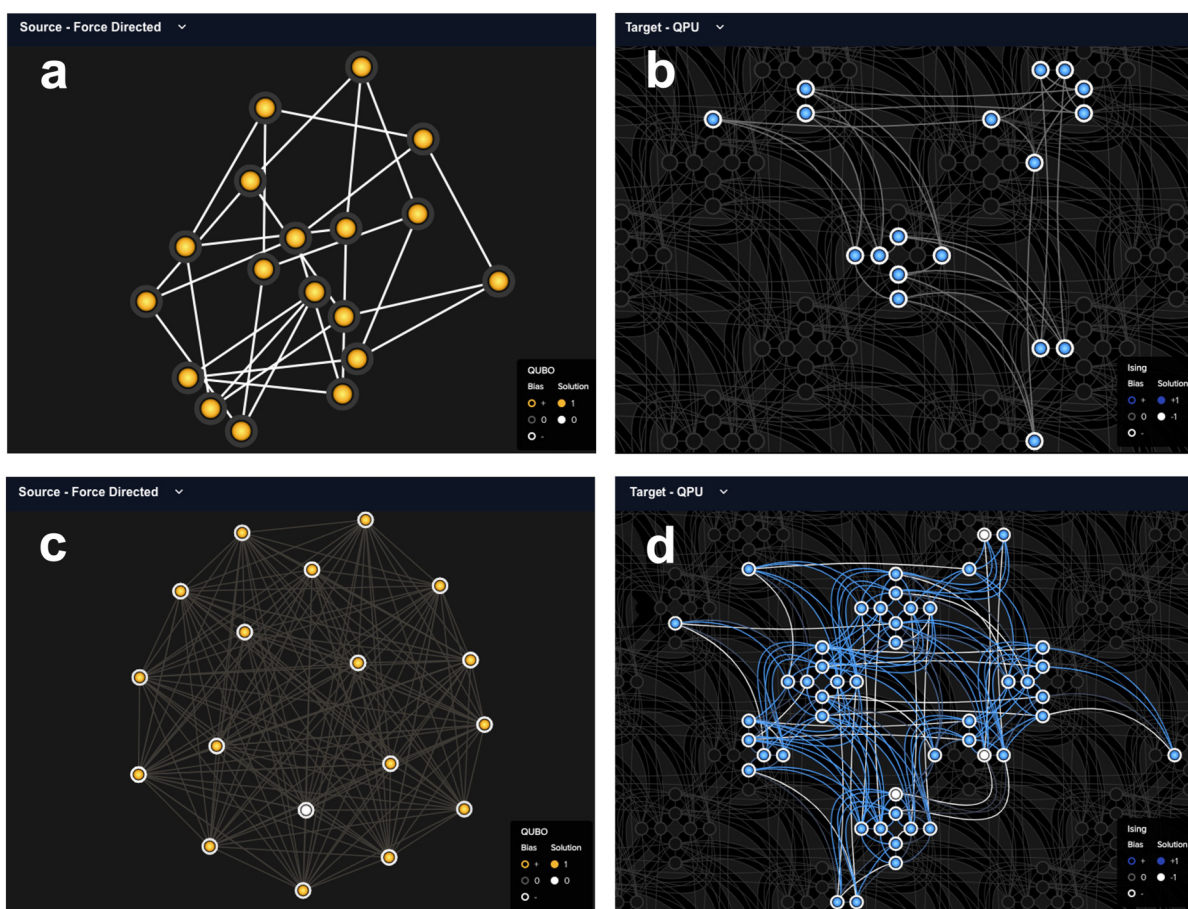
FIG. 13. Panels (a) and (c): mapping of the vacancies in graphene problem to a set of logical qubits. The 1 and 0 states are represented by orange and white circles, respectively. Panels (b) and (d): mapping of the vacancies in the graphene problem to the QPU. The 1 and 0 states are represented by light blue and white circles, respectively. Blue and white lines connecting physical qubits represent positive and negative couplings. Panels (a) and (b) display the results for the vacancy-free structure [energy model summarized in Eq. (5)]. Panels (c) and (d) display the results for the one-vacancy structure [energy model summarized in Eq. (11)].

displayed in Fig. 13 are obtained from the D-Wave Problem Inspector.[67] In the top part of Fig. 13, the problem for the vacancy-free graphene structure, whose QUBO model is defined in Eq. (5) is displayed. The left panel shows the mapping of the problem to a set of qubits. Since this QUBO model only includes coupling between nearest neighbors atoms, this is equivalent to the graph representation of the problem discussed in Appendix C. The right panel in the same figure shows the mapping to a set of physical qubits on the QPU. Because of the limited number of coupled nodes, this problem was mapped to a set of 18 physical qubits and no chains are present.

In the bottom part of Fig. 13, the same mapping described above is displayed for the single-vacancy graphene structure. In the energy model for this structure, all the nodes are coupled to each other. This results in 153 edges (compared to the 27 in the vacancy-free case) in the mapping to the logical qubits (bottom

left panel in Fig. 13). This problem cannot be mapped directly to the QPU architecture and qubit chains are formed. These are represented by the white lines connecting qubits in the bottom right panel in Fig. 13. Despite having the same number of sites in the structure as the vacancy-free case, due to the increased coupling in the model, to solve this problem, 48 physical qubits were used.

APPENDIX C: VACANCIES IN GRAPHENE AS A GRAPH PARTITIONING PROBLEM

In this section, we present a different approach to thinking about the problem of simulating vacancies in graphene. Graph representations are often used in the quantum computing community to represent a problem. A graph is a mathematical structure defined by nodes and edges (a line connecting two nodes). It is, therefore, easy to map such a structure to a set of qubits on a QPU.

TABLE I. Energy model summarized in Eq. (5) expressed in terms of binary variable values x_i and x_j and product of variables x_ix_j. The last column in the table can be useful to find which coupling delivers the wanted result.

x_i	x_j	x_ix_j
0	0	0
0	1	0
1	0	0
1	1	−1

A chemical structure can easily be thought of as a graph where the nodes correspond to the atoms and the edges are the bonds between two atoms. However, this representation is not often used in the computational chemistry community. This is because for most applications, as discussed in Sec. II A, the atom types and coordinates can be directly fed into a computational chemistry code implemented on classical computers. In addition, the graph retains information about the connectivity of the system, but the information about the relative distances of the atoms is lost.

The graphene structure represented in Fig. 2 can be transformed into an 18-node graph (one node per atom) and results in 27 edges ($3N_{sites}/2$ edges since each C atom is connected to three other C atoms). The resulting graph is a 3-regular graph.

When using QA to calculate the energy of a chemical system, the graph representation of the structure can be useful for two reasons:
- it helps build the energy model in terms of pairwise interactions (see Table I);
- it allows for an additional level of abstraction between the chemical structure and the mapping to the QPU. This can help to visualize the coupling required by the model and how that can be optimized to reduce the number of coupled qubits.

The energy model reported in Eq. (5) can be outlined in Table I, where $x_i = 1$ corresponds to site i being occupied by a C atom, $x_i = 0$ corresponds to site i being occupied by a vacancy. The results reported in the third column of Table I can be obtained by using a coupling $J = -1$ and is only needed when two nodes of the graph are connected.

The problem of identifying the lowest energy configuration of C atoms and vacancies can be thought of as a graph partitioning problem. The goal of the partitioning is to assign some nodes to group 1 (the C atoms) and others to group 0 (the vacancies) in such a way that minimizes the energy of the system according to Table I. This representation can be useful when a more complicated energy model is used or when more than two species can occupy the sites in the structure.

APPENDIX D: TUNING THE MODEL PARAMETERS

The effect of different λ/α ratios, chain strength, and annealing time were tested. The metrics analyzed are:

- ratio of broken chains (see Appendix B for details). These are calculated with respect to the number of times the quantum

annealing was run and results that do not respect the constraint were included;
- ratio of feasible solutions: how many solutions respect the constraint?
- ratio states: of the feasible solutions, how many correspond to the ground state and how many correspond to higher energy states?

1. Ratio λ/α

Ratios of λ/α ranging from 1 to 10 were obtained by keeping $\alpha = 1$ and varying the λ value. For these simulations, the chain strength was calculated by using the uniform torque compensation method, as discussed in Sec. D 2 with a prefactor of 0.3 [see Eq. (B1) for details]. This means that a different chain strength is calculated based on the different values of the couplings $J_{i,j}$ to ensure consistency in the results. Each point in the graphs in Fig. 14 represents the mean value of ten different quantum anneals (each run a thousand times). The error bars show the standard deviation for the ten runs.

Based on the results reported in Fig. 14, the ratio $\lambda/\alpha = 2$ was used for all the simulations reported in this paper. This value was selected because it results in a low percentage of broken chains while delivering high constraints-respecting solutions. In addition, it ensures a high probability of finding the ground state (third column in Fig. 14). For all α/λ values, the probability of finding excited states is higher than the ground state probability. This can be explained in terms of the multiplicity of states. From a symmetry point of view, the states corresponding to excited states (non-neighboring vacancies) result in exponentially more structures than the ground state ones (neighboring vacancies).
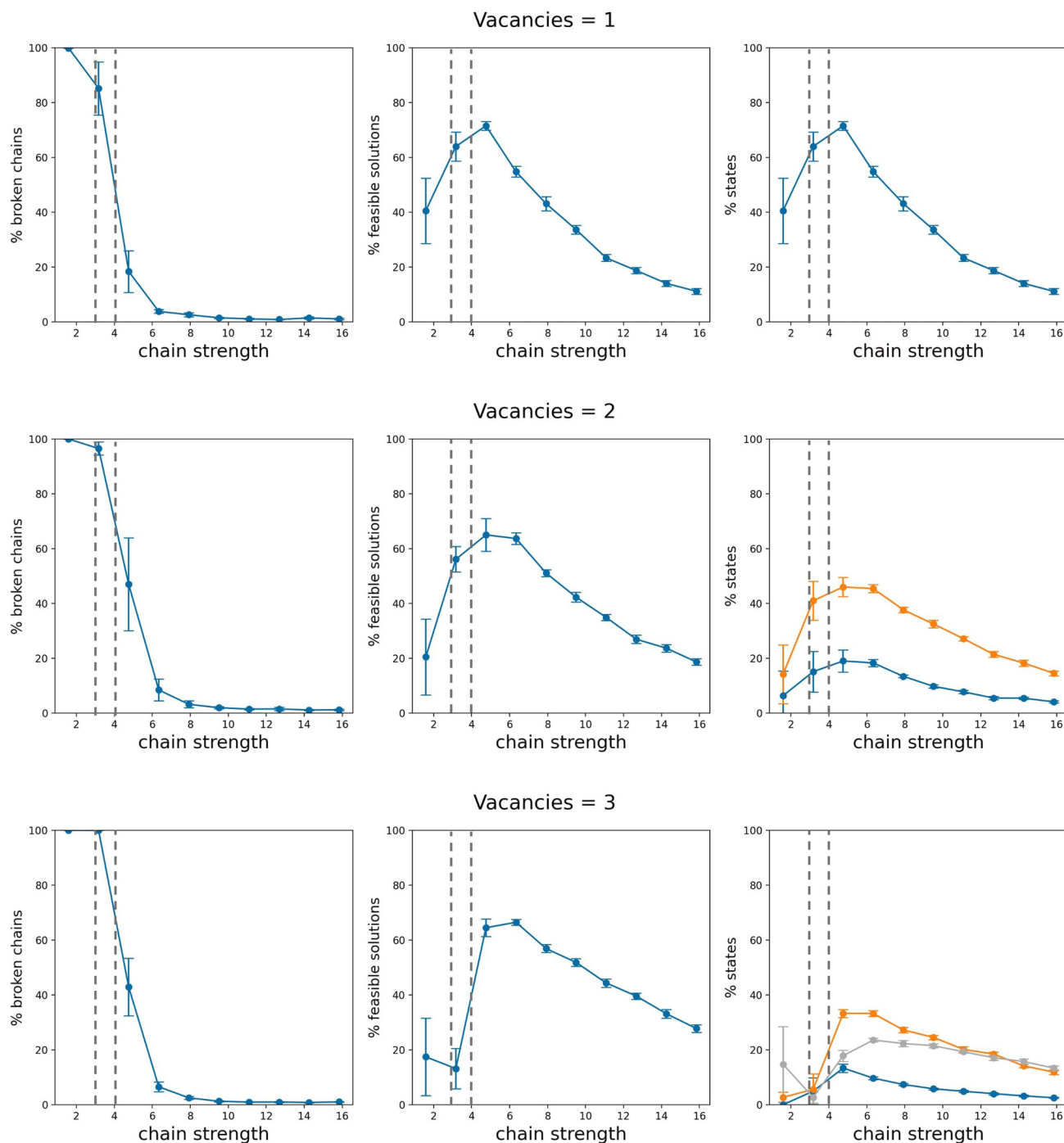
2. Chain strength

When the problem cannot be directly mapped to the QPU architecture, chains of physical qubits are formed. This topic is discussed in Appendix B. The D-Wave Ocean API allows for different ways to select the chain strength. The default is the so-called Uniform Torque Compensation (UTC) method reported in Eq. (B1). This model uses the residue mean square of the coupling values. It is, therefore, important to use the same energy model for all simulations when testing the effect of the chain strength using this model. For the results reported in this section a $\lambda/\alpha=2$ was used. The UTC method allows for the selection of a prefactor, which increases or decreases the chain strength values by a constant amount. Prefactor values ranging from 0.1 to 1.0 were tested. In Fig. 15 the metrics defined above are plotted against values of chain strength. The x axis reports the chain strength value rather than the prefactor that results in such value. The minimum and maximum values of the $J_{i,j}$ terms in the energy model are 3 and 4. The h_i terms depend on the number of vacancies in the model [see Eq. (11)]. Values of $h_i = -66$, $h_i = -64$ and $h_i = -58$ were used for the one, two and three-vacancy structures. Following the tests discussed in this section, the prefactor value of 0.3 was chosen for all the simulations reported in this paper.
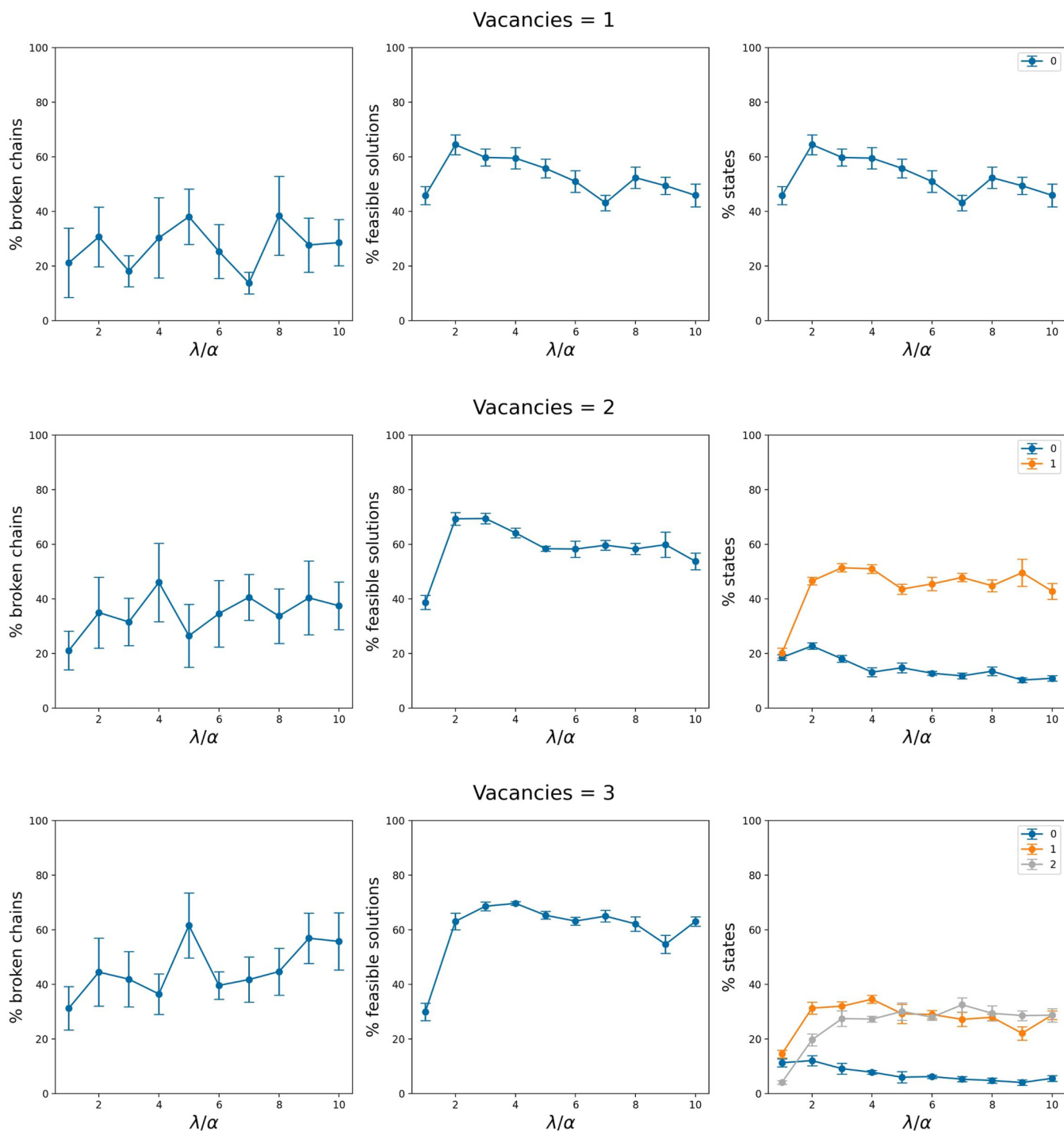
FIG. 14. Tuning of the λ/α ratio. Each row corresponds to a number of vacancies in the model. The columns, going from left to right report the percentage of broken chains, the probability of finding a feasible solution (i.e., a solution that respects the constraint on the number of vacancies) and the percentage of ground states and excited states among. In this context, ground state refers to the configuration corresponding to the lowest energy for the model discussed in Sec. II E. The excited states respect the constraint, but the arrangement of the vacancies leads to a higher energy value. Each data point represents the average of ten quantum anneal runs performed with a fixed set of parameters. The error bars show the standard deviation of the ten runs on the result.
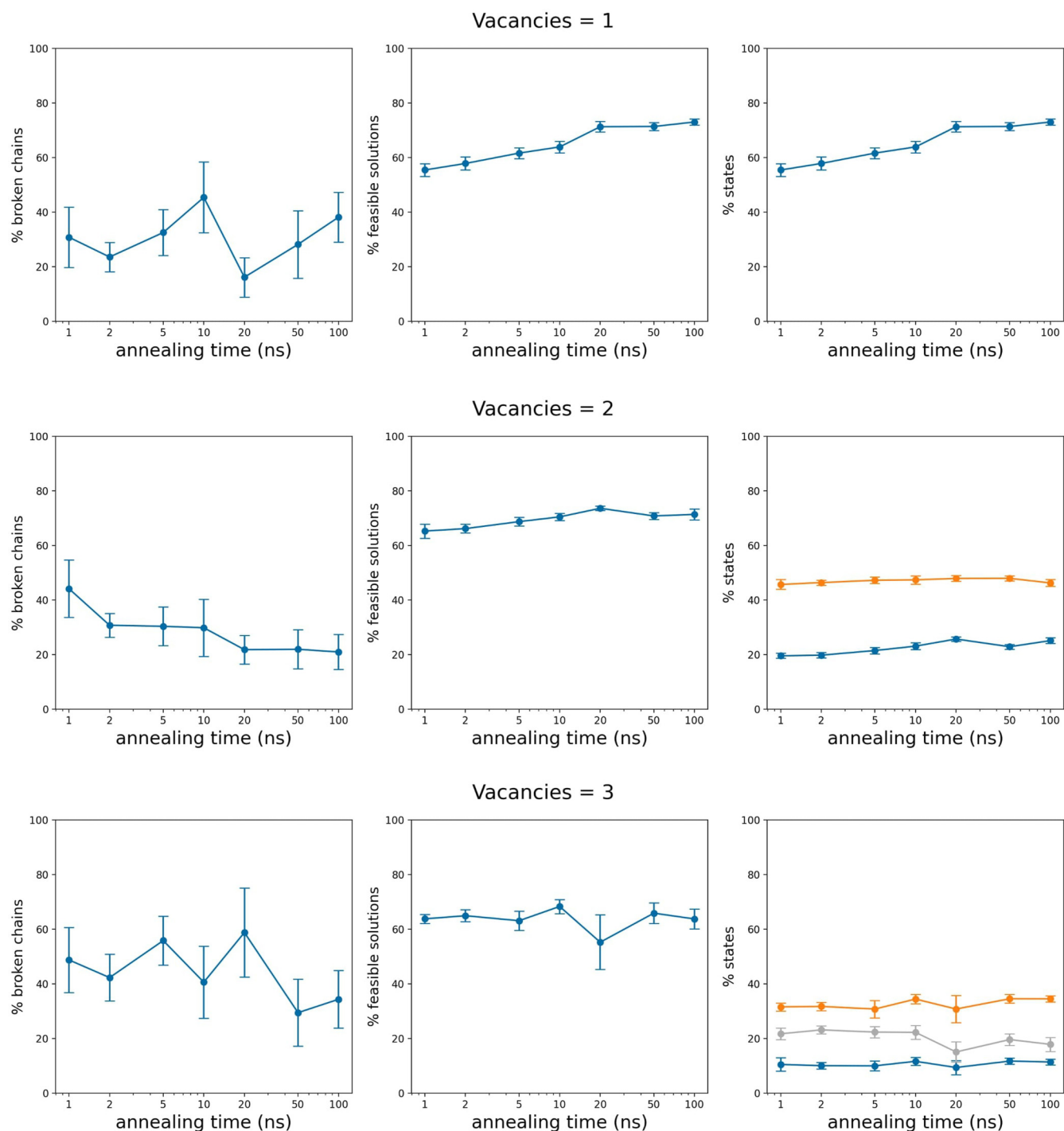
FIG. 15. Tuning of the chain strength. The **x** axis reports the value of the chain strength. The columns, going from left to right report the percentage of broken chains, the probability of finding a feasible solution (i.e., a solution that respects the constraint on the number of vacancies) and the percentage of ground states and excited states among. In this context, ground state refers to the configuration corresponding to the lowest energy for the model discussed in Sec. E. The excited states respect the constraint, but the arrangement of the vacancies leads to a higher energy value. Each data point represents the average of ten quantum anneal runs performed with a fixed set of parameters. The error bars show the standard deviation of the ten runs on the result. The dashed vertical lines show the minimum (3) and maximum (4) values of the $J_{i,j}$ terms in the energy model.

FIG. 16. Tuning of the anneal time. The **x** axis reports the annealing time in ns on a logarithmic scale. The columns, going from left to right report the percentage of broken chains, the probability of finding a feasible solution (i.e., a solution that respects the constraint on the number of vacancies) and the percentage of ground states and excited states among. In this context, ground state refers to the configuration corresponding to the lowest energy for the model discussed in Sec. E. The excited states respect the constraint, but the arrangement of the vacancies leads to a higher energy value. Each data point represents the average of ten quantum anneal runs performed with a fixed set of parameters. The error bars show the standard deviation of the ten runs on the result.

3. Annealing time

The D-Wave Ocean API allows the user to define the duration of the quantum annealing. Values ranging from $1\,\mu s$ to $100\,\mu s$ were tested. results are reported in Fig. 16. The default value of $20\,\mu s$ was selected for all calculations because it ensures the highest rate of feasible solutions.

APPENDIX E: CONSIDERATIONS ON THE CONFIGURATION MULTIPLICITY

The Boltzmann distribution can be used, in chemistry and statistical mechanics, to calculate the probability (p_i) of finding the system in state i as a function of the state energy and the temperature,

$$p_i = \frac{e^{\frac{-\epsilon_i}{kT}}}{\sum_{j=1}^{n} e^{\frac{-\epsilon_j}{kT}}}, \tag{E1}$$

where ϵ_i is the energy of state i, T is the temperature of the system, n is the total number of states, and k is the Boltzmann constant.

In the approach outlined in Eqs. (2), (5), (11), and (14), the energy of different configurations depends on the α and β values. These are optimized in order to correctly identify the energy order of different configurations. However, the absolute energy value calculated with this approach is not expected to be representative of

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	% broken chains	Energy	% occurrence
a	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.27	-24.0	5.013311
p	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.10	-24.0	5.320163
o	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.09	-24.0	5.254309
n	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.69	-24.0	5.748914
m	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1.75	-24.0	5.461679
l	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	2.02	-24.0	6.097800
k	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	2.33	-24.0	5.687264
j	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1.72	-24.0	5.608799
i	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1.83	-24.0	5.566765
h	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1.90	-24.0	5.786745
g	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1.21	-24.0	5.376208
f	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1.63	-24.0	5.746112
e	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1.55	-24.0	5.264117
d	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0.96	-24.0	5.282332
c	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1.90	-24.0	5.664845
b	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1.28	-24.0	5.421045
q	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1.88	-24.0	5.844192
r	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1.50	-24.0	5.855401

FIG. 17. The quantum annealing output for the QUBO problem reported in Eq. (11) (one-vacancy structure) using the parameters discussed in Appendix D. The values in the columns labeled 0 to 17 are the 18 components of **x**. The percentage of broken chains as a measure of the quality of the annealing is explained in detail in Appendix B. The energy is reported in arbitrary units. The quantum annealing was performed a total of 100 000 times (100 separate quantum anneals run 100 times each). The values reported in the last column only refer to the feasible solutions, i.e., the ones containing one vacancy.
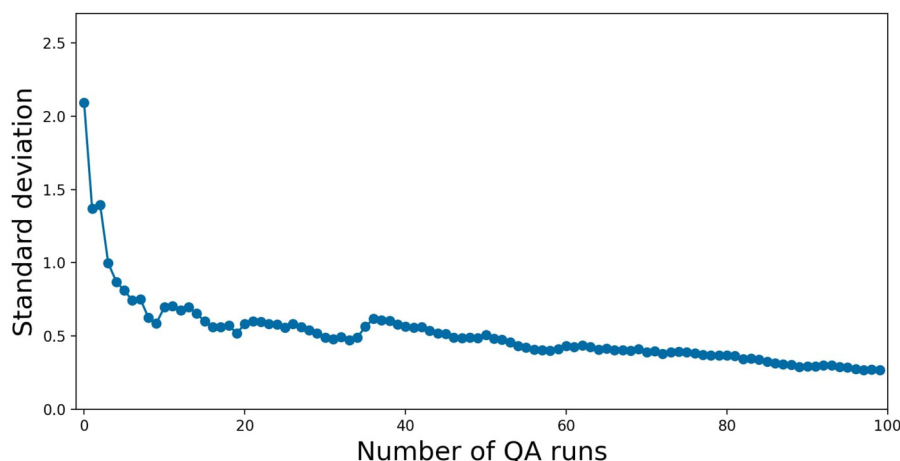
FIG. 18. Dependence of the standard deviation on the distribution of the configuration as a function of the number of quantum anneals. Each quantum anneal run was performed 1000 times.

the real energy of the structure. Therefore, the energies reported above would not yield sensible results when used in (E1).

When comparing the number of occurrences of different structures having the same number of vacancies and within the same energy bracket, the energy dependence of Eq. (E1) can be removed to obtain:

$$p_i^E = \frac{1}{\sum_{j=1}^{n^E} 1} = \frac{1}{n}, \tag{E2}$$

where the superscript E was added to specify we are only interested in states with energy E. We can use Eq. (E2) to study the distribution of symmetry equivalent structures for the one-vacancy graphene structure. For this structure, 18 equivalent configurations are found by the quantum annealer. According to Eq. (E2), these configurations should be found the same number of times (5.55% of the times).

In Fig. 17, the result of 100 different quantum annealings, each of them run 1000 times, resulting in a total of 100 000 runs is reported. The reason why it was not run directly 100 000 times in a single time is that we want to allow the minor embedding to map the logical qubits to different physical qubits. This is important because of the possible presence of unwanted biases within the quantum hardware that might bias the result.

The values reported in the "percentage occurrence" column only refer to the feasible solutions, i.e., the ones containing one vacancy. Therefore, their sum is 100%. This is different from the results in the main body of this paper that reported the % of occurrence with respect to all the solutions returned by the quantum annealer including the ones that do not respect the constraint on the number of vacancies. A standard deviation on the percentage of occurrence values of 0.267 with respect to the ideal 5.55% value was found.

The standard deviation was then calculated for different numbers of QA runs to study the dependence of the standard deviation on the number of runs. Results are reported in Fig. 18 where the standard deviation as a function of the increasing number of QA runs is displayed. The standard deviation decreases with the number of runs from 2.092 to 0.267 going from one single run to 100. After a first steep initial decrease of 1.394 going from 1 to 10 runs, the standard deviation decreases constantly and slowly with the increasing number of runs. These results suggest that the standard deviation tends to zero in the limit of an infinite number of runs and the 18 symmetry equivalent structures would be found with the same frequency.

APPENDIX F: COMPARING THE QA ENERGIES TO DFT

The energy of the structures reported in Figs. 3 to 8 was calculated using Density Functional Theory (DFT). All calculations were performed by using the periodic CRYSTAL17 code, which implements localized Gaussian-type basis functions. Thanks to the use of local basis functions instead of plane waves, on which most solid-state simulation codes rely, the graphene structure can be treated as a real 2D system. The PBE functional and a pob-TZVP basis set[68] were used. The truncation criteria for the Coulomb and exchange infinite lattice series were set to 8 (T1–T4) and 16 (T5). An 8×8 Monkhorst-Pack grid was used to sample the reciprocal space. A 16×16 Gilat-net grid, which is used to determine the Fermi energy for conductors, was used. A tolerance of 10^{-9} Hartree/cell was selected as a criterion for both the Self Consistent Field (scf) and geometry optimization energies.

The structure of the vacancy-free graphene was fully optimized (lattice parameters and atom positions). It was then used as the initial geometry for all the other structures containing vacancies. The lattice parameters were fixed at the vacancy-free values in an attempt to simulate a low concentration of vacancies, which

TABLE II. Comparison between the QA calculated energies and the DFT energy for the graphene structures including vacancies discussed in Sec. E to II G. The energies for the quantum annealing are reported as arbitrary units. The column labeled "E sp" refers to the energy of the non-optimized structure. In column "E optgeom," the energies of the geometry-optimized structures are reported. The values of ΔE are normalized with respect to the lowest energy structure containing the same number of vacancies. The standard deviation on the displacements of the atoms and the maximum displacement reported in the last two columns refer to the structure before and after geometry optimization.

|	E QA	ΔE QA	E sp (eV/atom)	E optgeom (eV/atom)	ΔE sp (eV/atom)	ΔE optgeom (eV/atom)	Std disp	Max disp
			vacancy-free structure					
a	−27.0	0.0	−1036.195 07	−1036.195 07	0.0000	0.0000	0.0000	0.0000
			one-vacancy structure					
a	−24.0	0.0	−1035.710 24	−1035.726 04	0.0000	0.0000	0.0257	0.0532
			two-vacancy structure					
a	−22.0	0.0	−1035.586 97	−1035.630 47	0.0000	0.0000	0.0940	0.1201
b	−21.0	1.0	−1035.152 02	−1035.190 41	0.0272	0.0275	0.0392	0.0724
c	−21.0	1.0	−1035.238 12	−1035.238 12	0.0218	0.0245	0.1784	0.5633
d	−21.0	1.0	−1035.314 04	−1035.400 40	0.0171	0.0144	0.0734	0.1301
e	−21.0	1.0	−1035.307 56	−1035.376 83	0.0175	0.0159	0.0687	0.1247
			three-vacancy structure					
a	−20.0	0.0	−1035.343 22	−1035.374 96	0.0000	0.0000	0.0492	0.1065
b	−19.0	1.0	−1035.070 46	−1034.889 79	0.0182	0.0323	0.1525	0.2776
c	−19.0	1.0	−1035.094 17	−1035.266 82	0.0166	0.0072	0.2092	0.2144
d	−19.0	1.0	−1035.039 95	−1035.125 34	0.0202	0.0166	0.1440	0.2315
e	−19.0	1.0	−1035.107 06	−1035.261 03	0.0157	0.0076	0.3275	0.6058
f	−18.0	2.0	−1034.694 29	−1035.375 03	0.0433	−0.0000	0.5173	1.3846
g	−18.0	2.0	−1034.732 32	−1034.656 41	0.0407	0.0479	1.7347	5.5544
h	−18.0	2.0	−1034.698 64	−1034.698 64	0.0430	0.0451	0.4224	1.0414
i	−18.0	2.0	−1034.875 51	−1035.263 72	0.0312	0.0074	0.8710	1.4490
j	−18.0	2.0	−1034.742 82	−1035.375 03	0.0400	−0.0000	1.6188	5.7270
k	−18.0	2.0	−1034.844 23	−1035.266 85	0.0333	0.0072	0.4115	0.2792
l	−18.0	2.0	−1034.506 75	−1034.506 75	0.0558	0.0579	0.0149	0.0168
m	−18.0	2.0	−1034.771 43	−1034.889 90	0.0381	0.0323	0.3783	0.7055
n	−18.0	2.0	−1034.822 28	−1034.822 28	0.0347	0.0368	0.3625	0.2500

would not be able to distort the lattice. In the geometry optimization calculations, the symmetry of the structure was removed by using the SYMMREMO keyword. The atom positions were not manually changed to break the symmetry. This means that the initial geometry for the single-vacancy structure is close to a local minimum and upon optimization, we did not find a Stone-Wales structure, but only a small displacement (0.05 Å) of the atoms surrounding the vacancy.

In Table II, the "E sp" column reports the single point energy on such initial geometry. Then, the atom coordinates of all structures were allowed to relax and their final energies are reported in the "E optgeom" column. For ease of comparing the relative energies, these were normalized with respect to the lowest energy within the structures containing the same number of vacancies.

The following observations can be drawn from Table II:

- the energy per atom increases with the number of vacancies in the structures;

- for all structures, the relative DFT single point energies agree with the QA calculations. It can, indeed, be observed that the structures where the vacancies are located on neighboring sites have the lowest energy within a fixed number of vacancies group. For example, among the two-vacancy structures, structures **b–e** have higher energy than structure **a**;

- when the geometry is optimized (column "E optgeom"), the observation outlined in the precedent point does not always hold true. For example, among the three-vacancy structures, structures **f** and **j**, which are predicted by the QA to have the highest energy have, according to DFT, the same energy as the lowest energy structure **a**. This can be explained in terms of the heavy reconstruction these structures undergo. Structures **f** and **j** are depicted in Fig. 19 before and after geometry optimization. Both structures start from a sparse initial distribution of vacancies and get to a final structure where the vacancies can be found on three neighboring sites. In order to quantify such reconstruction, the standard deviation (std) on
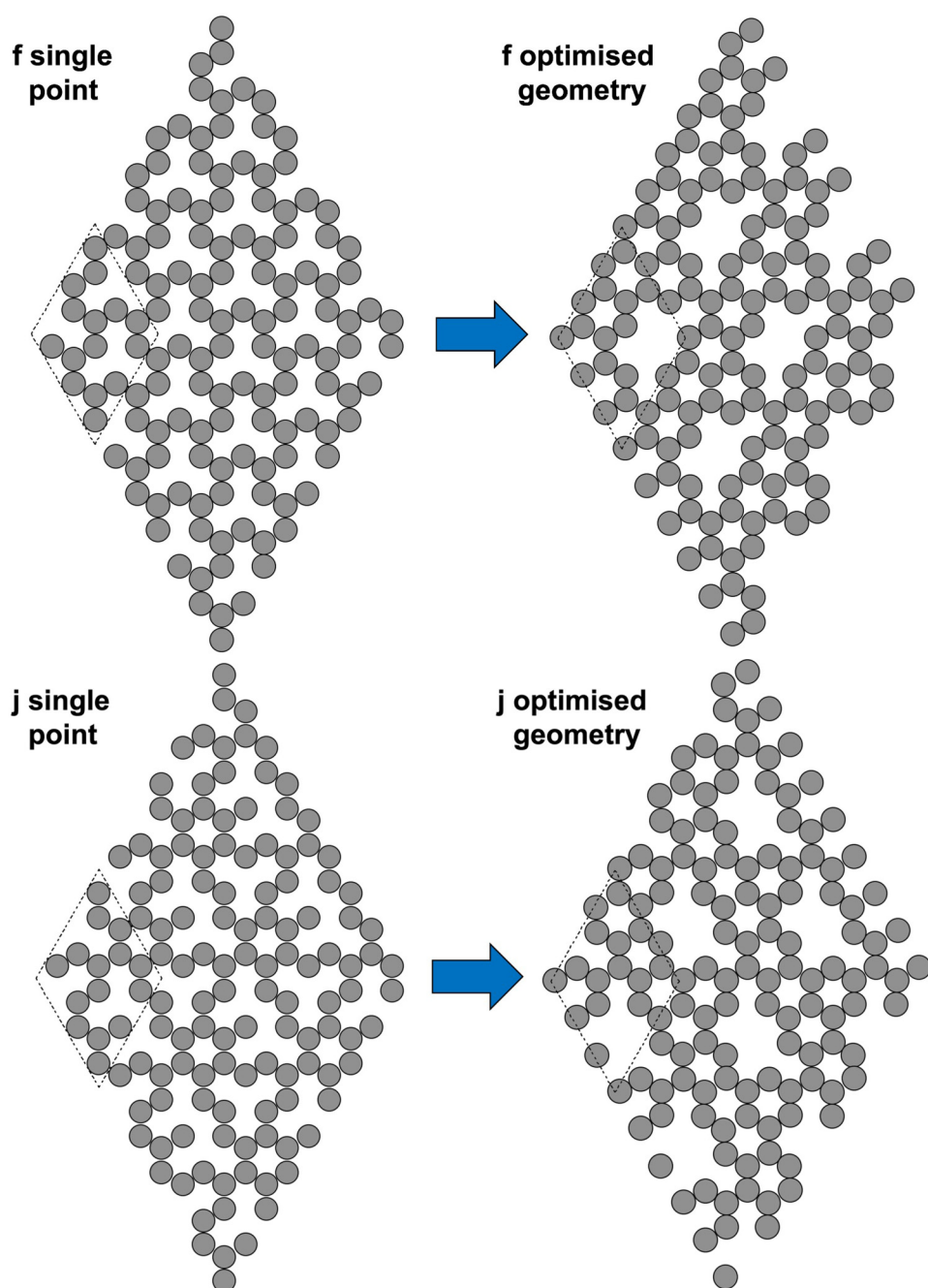
FIG. 19. Geometry of the **f** (top) and **j** (bottom) structures before (left) and after (right) geometry optimization.

the atom displacements and the maximum displacements (in Å) are reported in the last two columns of Table II. Both structures **f** and **j** undergo much larger reconstruction than any other structure. Their std on the atomic displacements is one order of magnitude larger than that for the other structures.

REFERENCES

[1] R. P. Feynman, "Simulating physics with computers," Int. J. Theor. Phys. **21**, 467–488 (1982).

[2] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, "Simulated quantum computation of molecular energies," Science **309**, 1704–1707 (2005).

[3]S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, "Quantum computational chemistry," Rev. Mod. Phys. **92**, 015003 (2020).

[4]Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis, and A. Aspuru-Guzik, "Quantum chemistry in the age of quantum computing," Chem. Rev. **119**, 10856–10915 (2019).

[5]S. T. Stober, S. M. Harwood, D. Trenev, P. K. Barkoutsos, T. P. Gujarati, and S. Mostame, "Considerations for evaluating thermodynamic properties with hybrid quantum-classical computing work flows," Phys. Rev. A **105**, 012425 (2022).

[6]A. Ajagekar, T. Humble, and F. You, "Quantum computing based hybrid solution strategies for large-scale discrete-continuous optimization problems," Comput. Chem. Eng. **132**, 106630 (2019).

[7]I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, "Simulating chemistry using quantum computers," Annu. Rev. Phys. Chem. **62**, 185–207 (2011).

[8]R. Babbush, P. J. Love, and A. Aspuru-Guzik, "Adiabatic quantum simulation of quantum chemistry," Sci. Rep. **4**, 6603 (2014).

[9]A. Teplukhin, B. K. Kendrick, S. Tretiak, and P. A. Dub, "Electronic structure with direct diagonalization on a D-Wave quantum annealer," Sci. Rep. **10**, 20753 (2020).

[10]H. Yu, D. Lu, Q. Wu, and T.-C. Wei, "Geometric quantum adiabatic methods for quantum chemistry," Phys. Rev. Res. **4**, 033045 (2022).

[11]B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, "Quantum algorithms for quantum chemistry and quantum materials science," Chem. Rev. **120**, 12685–12717 (2020).

[12]D. E. Bernal, A. Ajagekar, S. M. Harwood, S. T. Stober, D. Trenev, and F. You, "Perspectives of quantum computing for chemical engineering," AIChE J. **68**, e17651 (2022).

[13]A. Nourbakhsh, M. N. Jones, K. Kristjuhan, D. Carberry, J. Karon, C. Beenfeldt, K. Shahriari, M. P. Andersson, M. A. Jadidi, and S. S. Mansouri, "Quantum computing: Fundamentals, trends and perspectives for chemical and biochemical engineers," arXiv:2201.02823 (2022).

[14]V. Kendon, "Quantum computing using continuous-time evolution," Interface Focus **10**, 20190143 (2020).

[15]S. S. Gill, A. Kumar, H. Singh, M. Singh, K. Kaur, M. Usman, and R. Buyya, "Quantum computing: A taxonomy, systematic review and future directions," Software Practice Experience **52**, 66–114 (2022).

[16]E. K. Grant and T. S. Humble, *Adiabatic Quantum Computing and Quantum Annealing* (Oxford University Press, 2020).

[17]A. D. King, S. Suzuki, J. Raymond, A. Zucca, T. Lanting, F. Altomare, A. J. Berkley, S. Ejtemaee, E. Hoskinson, S. Huang, E. Ladizinsky, A. J. R. MacDonald, G. Marsden, T. Oh, G. Poulin-Lamarre, M. Reis, C. Rich, Y. Sato, J. D. Whittaker, J. Yao, R. Harris, D. A. Lidar, H. Nishimori, and M. H. Amin, "Coherent quantum annealing in a programmable 2,000 qubit Ising chain," Nat. Phys. **18**, 1324–1328 (2022).

[18]K. Sugisaki, S. Nakazawa, K. Toyota, K. Sato, D. Shiomi, and T. Takui, "Quantum chemistry on quantum computers: A method for preparation of multiconfigurational wave functions on quantum computers without performing post-Hartree–Fock calculations," ACS Cent. Sci. **5**, 167–175 (2019).

[19]H. Liu, G. H. Low, D. S. Steiger, T. Häner, M. Reiher, and M. Troyer, "Prospects of quantum computing for molecular sciences," Mater. Theory **6**, 11 (2022).

[20]R. Xia, T. Bian, and S. Kais, "Electronic structure calculations and the Ising Hamiltonian," J. Phys. Chem. B **122**, 3384–3395 (2018).

[21]M. Streif, F. Neukart, and M. Leib, "Solving quantum chemistry problems with a D-Wave quantum annealer," in *Lecture Notes in Computer Science* (Springer, 2019), Vol. 11413.

[22]S. N. Genin, I. G. Ryabinkin, and A. F. Izmaylov, "Quantum chemistry on quantum annealers," arXiv:1901.04715 (2019).

[23]A. Teplukhin, B. K. Kendrick, S. M. Mniszewski, Y. Zhang, A. Kumar, C. F. Negre, P. M. Anisimov, S. Tretiak, and P. A. Dub, "Computing molecular excited states on a D-Wave quantum annealer," Sci. Rep. **11**, 18796 (2021).

[24]S. Yarkoni, E. Raponi, T. Bäck, and S. Schmitt, "Quantum annealing for industry applications: Introduction and review," Rep. Prog. Phys. **85**, 104001 (2022).

[25]D. Ghamari, P. Hauke, R. Covino, and P. Faccioli, "Sampling rare conformational transitions with a quantum computer," Sci. Rep. **12**, 16336 (2022).

[26]C. Micheletti, P. Hauke, and P. Faccioli, "Polymer physics by quantum computing," Phys. Rev. Lett. **127**, 080501 (2021).

[27]V. K. Mulligan, H. Melo, H. I. Merritt, S. Slocum, B. D. Weitzner, A. M. Watkins, P. D. Renfrew, C. Pelissier, P. S. Arora, and R. Bonneau, "Designing peptides on a quantum computer," BioRxiv 752485 (2019).

[28]J. P. Terry, P. D. Akrobotu, C. F. A. Negre, and S. M. Mniszewski, "Quantum isomer search," PLoS One **15**, e0226787 (2020).

[29]K. Hatakeyama-Sato, T. Kashikawa, K. Kimura, and K. Oyaizu, "Tackling the challenge of a huge materials science search space with quantum-inspired annealing," Adv. Intell. Syst. **3**, 2000209 (2021).

[30]K. Kitai, J. Guo, S. Ju, S. Tanaka, K. Tsuda, J. Shiomi, and R. Tamura, "Designing metamaterials with quantum annealing and factorization machines," Phys. Rev. Res. **2**, 013319 (2020).

[31]V. Carnevali, I. Siloi, R. Di Felice, and M. Fornari, "Vacancies in graphene: An application of adiabatic quantum optimization," Phys. Chem. Chem. Phys. **22**, 27332–27337 (2020).

[32]A. Vitalis and R. V. Pappu, *Methods for Monte Carlo Simulations of Biomacromolecules* (Elsevier, 2009), Chap. 3, pp. 49–76.

[33]J. Lee, H. Q. Pham, and D. R. Reichman, "Twenty years of auxiliary-field quantum Monte Carlo in quantum chemistry: An overview and assessment on main group chemistry and bond-breaking," J. Chem. Theory Comput. **18**(12), 7024–7042 (2022).

[34]D. J. Klein, J. L. Palacios, M. Randić, and N. Trinajstić, "Random walks and chemical graph theory," J. Chem. Infor. Comput. Sci. **44**, 1521–1525 (2004).

[35]X. Wang, Y. Qian, H. Gao, C. W. Coley, Y. Mo, R. Barzilay, and K. F. Jensen, "Towards efficient discovery of Green synthetic pathways with Monte Carlo tree search and reinforcement learning," Chem. Sci. **11**, 10959–10972 (2020).

[36]Y. Ebadi and N. Jafari Navimipour, "An energy-aware method for data replication in the cloud environments using a tabu search and particle swarm optimization algorithm," Concurrency Comput. Practice Experience **31**, e4757 (2018).

[37]R. Rutenbar, "Simulated annealing algorithms: An overview," IEEE Circ. Dev. Mag. **5**, 19–26 (1989).

[38]J. Pannetier, J. Bassas-Alsina, J. Rodriguez-Carvajal, and V. Caignaert, "Prediction of crystal structures from crystal chemistry rules by simulated annealing," Nature **346**, 343–345 (1990).

[39]J. H. Kalivas, *Adaption of Simulated Annealing to Chemical Optimization Problems* (Elsevier, 1995).

[40]J.-L. Faulon, "Stochastic generator of chemical structure. 2. Using simulated annealing to search the space of constitutional isomers," J. Chem. Inform. Comput. Sci. **36**, 731–740 (1996).

[41]F. Guarnieri and M. Mezei, "Simulated annealing of chemical potential: A general procedure for locating bound waters. application to the study of the differential hydration propensities of the major and minor grooves of DNA," J. Am. Chem. Soc. **118**, 8493–8494 (1996).

[42]D. Bertsimas and J. Tsitsiklis, "Simulated annealing," Stat. Sci. **8**, 10–15 (1993).

[43]S. M. Woodley and A. A. Sokol, "From ergodicity to extended phase diagrams," Angew. Chem. Int. Ed. **51**, 3752–3754 (2012).

⁴⁴See https://www.dwavesys.com/ for "D-Wave Systems: The Practical Quantum Computing Company."

⁴⁵See https://github.com/dwavesystems for "D-Wave Systems: Github."

⁴⁶K. Boothby, P. Bunyk, J. Raymond, and A. Roy, "Next-generation topology of D-Wave quantum processors," arXiv:2003.00133 (2020).

⁴⁷B. Camino, https://github.com/cmc-ucl/quantum_computing/blob/main/Quantum_computing_and_materials_science/data_analysis.ipynb for "Jupyter notebook: Quantum computing and materials science" (2022).

⁴⁸R. Dovesi, A. Erba, R. Orlando, C. M. Zicovich-Wilson, B. Civalleri, L. Maschio, M. Rérat, S. Casassa, J. Baima, S. Salustro, and B. Kirtman, "Quantum-mechanical condensed matter simulations with CRYSTAL," WIREs Comput. Mol. Sci. **8**, e1360 (2018).

⁴⁹W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects," Phys. Rev. **140**, A1133–A1138 (1965).

⁵⁰W. Hehre, W. Lathan, R. Ditchfield, M. D. Newton, and J. Pople, "Gaussian 70," Quantum Chemistry Program Exchange No. 236 (1970).

⁵¹C. Froese Fischer, "General Hartree–Fock program," Comput. Phys. Commun. **43**, 355–365 (1987).

⁵²J. C. Slater, "A simplification of the Hartree–Fock method," Phys. Rev. **81**, 385–390 (1951).

⁵³A. D. Becke, "Density-functional exchange-energy approximation with correct asymptotic behavior," Phys. Rev. A **38**, 3098–3100 (1988).

⁵⁴C. Lee, W. Yang, and R. G. Parr, "Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density," Phys. Rev. B **37**, 785–789 (1988).

⁵⁵D. W. Berry, A. M. Childs, Y. Su, X. Wang, and N. Wiebe, "Time-dependent Hamiltonian simulation with less math," Quantum **4**, 254 (2020).

⁵⁶The bias corresponds to the magnetic field applied to the qubit along its z axis (in the Bloch sphere representation) that will make it more or less likely to return the spin up or spin down result. The coupling is the strength that is applied between two qubits to define their probability to return the same or opposite spin state at the end of the annealing.

⁵⁷S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder, "Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis," Comput. Mater. Sci. **68**, 314–319 (2013).

⁵⁸R. Schmitt, A. Nenning, O. Kraynis, R. Korobko, A. I. Frenkel, I. Lubomirsky, S. M. Haile, and J. L. Rupp, "A review of defect structure and chemistry in ceria and its solid solutions," Chem. Soc. Rev. **49**, 554–592 (2020).

⁵⁹R. Rocca, M. F. Sgroi, B. Camino, M. D'Amore, and A. M. Ferrari, "Disordered rock-salt type Li_2TiS_3 as novel cathode for libs: A computational point of view," Nanomaterials **12**, 1832 (2022).

⁶⁰M. J. Akhtar, C. R. A. Catlow, B. Slater, A. M. Walker, and S. M. Woodley, "Bulk and surface simulation studies of $La_{1-x}Ca_xMnO_3$," Chem. Mater. **18**, 1552–1560 (2006).

⁶¹Pandas is an open source data analysis and manipulation tool distributed as a library of the Python programming language.

⁶²A. Togo and I. Tanaka, "Spglib: A software library for crystal symmetry search," arXiv:1808.01590v1 (2018).

⁶³L. Bassman, C. Powers, and W. A. De Jong, "ArQTiC: A full-stack software package for simulating materials on quantum computers," ACM Trans. Quant. Comput. **3**, 1–17 (2022).

⁶⁴R. P. Feynman, F. L. Vernon, and R. W. Hellwarth, "Geometrical representation of the Schrödinger equation for solving Maser problems," J. Appl. Phys. **28**, 49–52 (1957).

⁶⁵O. Gamel, "Entangled Bloch spheres: Bloch matrix and two-qubit state space," Phys. Rev. A **93**, 062320 (2016).

⁶⁶See https://docs.ocean.dwavesys.com/projects/minorminer/en/latest/#documentation for "D-Wave minorminer."

⁶⁷See https://docs.ocean.dwavesys.com/projects/inspector/en/latest/intro.html for "D-Wave systems: Problem inspector."

⁶⁸D. Vilela Oliveira, J. Laun, M. F. Peintinger, and T. Bredow, "BSSE-correction scheme for consistent Gaussian basis sets of double- and triple-zeta valence with polarization quality for solid-state calculations," J. Comput. Chem. **40**, 2364–2376 (2019).

28 June 2023 13:11:23