Los Alamos NATIONAL LABORATORY EST. 1943

LA-UR-19-26724 (Accepted Manuscript)

Quantum isomer search

Mniszewski, Susan M.
Terry, Jason Patrick
Akrobotu, Prosper D.
Negre, Christian Francisco Andres

Provided by the author(s) and the Los Alamos National Laboratory (2020-05-21).

To be published in: PLOS ONE

DOI to publisher's version: 10.1371/journal.pone.0226787

Permalink to record: http://permalink.lanl.gov/object/view?what=info:lanl-repo/lareport/LA-UR-19-26724

Disclaimer:
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

RESEARCH LIBRARY

PLOS ONE

RESEARCH ARTICLE

Quantum isomer search

Jason P. Terry[1,2,3☯], **Prosper D. Akrobotu**[4,5☯], **Christian F. A. Negre**[5*], **Susan M. Mniszewski**[6*]

1 Department of Physics and Astronomy, University of Georgia, Athens, Georgia, United States of America, **2** Data Science Initiative, Brown University, Providence, Rhode Island, United States of America, **3** Center for Nonlinear Studies (T-CNLS), Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico, United States of America, **4** Department of Mathematical Sciences, The University of Texas at Dallas, Richardson, Texas, United States of America, **5** Physics and Chemistry of Materials (T-1), Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico, United States of America, **6** Information Sciences (CCS-3), Computer, Computational and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, New Mexico, United States of America

☯ These authors contributed equally to this work.
* cnegre@lanl.gov (CFAN); smm@lanl.gov (SMM)

Check for updates

OPEN ACCESS

Citation: Terry JP, Akrobotu PD, Negre CFA, Mniszewski SM (2020) Quantum isomer search. PLoS ONE 15(1): e0226787. https://doi.org/10.1371/journal.pone.0226787

Editor: Itay Hen, University of Southern California, UNITED STATES

Received: August 8, 2019

Accepted: December 3, 2019

Published: January 15, 2020

Copyright: This is an open access article, free of all copyright, and may be freely reproduced, distributed, transmitted, modified, built upon, or otherwise used by anyone for any lawful purpose. The work is made available under the Creative Commons CC0 public domain dedication.

Data Availability Statement: All relevant data are within the manuscript.

Funding: SMM and CFAN were funded by Los Alamos National Laboratory (LANL) Information Science and Technology Institute (ISTI) and Laboratory Directed Research and Development (LDRD). JPT was funded by the Los Alamos National Laboratory Quantum Computing Summer School 2019. PDA was funded by the National Science Foundation (NSF) Mathematical Sciences Graduate Internship (MSGI) Program sponsored by the NSF Division of Mathematical Sciences. The funders had no role in the study design, data

Abstract

Isomer search or molecule enumeration refers to the problem of finding all the isomers for a given molecule. Many classical search methods have been developed in order to tackle this problem. However, the availability of quantum computing architectures has given us the opportunity to address this problem with new (quantum) techniques. This paper describes a quantum isomer search procedure for determining all the structural isomers of alkanes. We first formulate the structural isomer search problem as a quadratic unconstrained binary optimization (QUBO) problem. The QUBO formulation is for general use on either annealing or gate-based quantum computers. We use the D-Wave quantum annealer to enumerate all structural isomers of all alkanes with fewer carbon atoms ($n < 10$) than Decane ($C_{10}H_{22}$). The number of isomer solutions increases with the number of carbon atoms. We find that the sampling time needed to identify all solutions scales linearly with the number of carbon atoms in the alkane. We probe the problem further by employing reverse annealing as well as a perturbed QUBO Hamiltonian and find that the combination of these two methods significantly reduces the number of samples required to find all isomers.

Introduction

As quantum computers with more qubits and increased accuracy become available, interest in solving useful problems in the near-term has increased. The Ising problem is among this group. This problem is well-known to be NP-complete and is therefore efficiently mappable to all other NP-complete problems, such as the graph-coloring problem [1]. Since both gate-based and annealing quantum computers can solve the Ising problem [2, 3], it has been of particular interest to the quantum computing community for both near-term and long-term applications. It has already been successfully used to solve several problems such as graph partitioning and community detection [4, 5].

collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Given the structural formula of a molecule, one can always construct a graph defining the connectivity of the atoms. Thus the corresponding graph of a molecule, called a molecular graph, is defined as a labeled graph with a vertex set consisting of the atoms of the molecule and edges representing the chemical bonds existing between the atoms [6]. For the case of hydrocarbons, we refer to the molecule as saturated if the bond between atoms is single and unsaturated if the bond is a double bond. In this vain, a saturated hydrocarbon is defined as a simple molecular graph whose vertices represent hydrogen and carbon atoms, and edges represent single bonds between the atoms [6]. These simple molecular graphs could be either cyclic or acyclic. The acyclic saturated hydrocarbons are known as alkanes with molecular formula C_nH_{2n+2} while the cyclic saturated hydrocarbons are referred to as the cycloalkanes with molecular formula C_nH_{2n}. A general formula for the saturated hydrocarbons is given as $C_n H_{2n-2(k-1)}$, where k is the number of independent loops. A given molecular formula could correspond to different molecules with distinct structural arrangements. The molecules with identical formulas but distinct structures are called isomers. They are classified as structural isomers if their bonding patterns and atomic organization is distinct, or as stereoisomers if the bonding patterns are fixed while the spatial arrangement is distinct [7]. The goal of this work is to enumerate the structural isomers of any given molecular formula for alkanes by encoding this problem into a quantum computing framework.

Isomer search, or molecule enumeration, is the process of searching for all isomers of a given molecule. The search space could be structural (2D) or spatial (3D), but for our purposes the focus will be on structural isomer search. The enumeration of structural isomers is of interest to numerous fields. Examples include pharmaceutical applications as proper identification of isomers "would facilitate resource reduction, including animal usage, and may benefit other areas of pharmaceutical structural characterization including impurity profiling and degradation chemistry" [8]. Moreover, the oil and gas industry relies on knowledge of the structures of isomers of hydrocarbons and alkanes for processes related to refinement [9, 10]

Over the years, many search procedures have been developed for enumerating molecules. Most of these procedures incorporate various techniques such as the labeled enumeration (a procedure for enumerating all $2^{\frac{(n(n-1))}{2}}$ labeled graphs of n nodes), orderly generation method (a procedure that dwells on a so-called canonical representation of graphs such that the canonization process induces an ordering on the edges of the graph), random sampling (techniques that generate structures from randomly selected branches of the construction trees of deterministic structural generation algorithms), Monte-Carlo and simulated annealing (a procedure that focuses on minimizing random displacements on atoms by performing a bond order switch), and genetic algorithms (a procedure for which mutations are carried out using bond perturbations, crossover operations executed using a generated n-tuple code, and selection operators defined by root-mean-square deviation between experimental chemical shifts, and predicted chemical shifts from neural network technology) [11].

The work of Nobel Laureate J. Lederberg [12] on the topology of molecules in 1969 is considered to be the genesis of algorithmic and computational approaches to the field of molecular structure enumeration. A particularly large step was when the algorithm was finally incorporated into the DENDRAL (Dendritic Algorithm) code [13] for enumerating isomers of acyclic compounds containing carbon (C), hydrogen (H), oxygen (O), and nitrogen (N) atoms [14]. Since then, the field has seen the development of several distinct types of algorithms (exhaustive, automated and stochastic) and codes for structural elucidation using different classical approaches with diverse input criteria [11]. In order to generate the isomers, most of these codes, including SKELGEN, CAMGEC, AEGIS, ISOGEN, GI, and MOLGEN [15–20] require only the molecular formula as input. Others, such as DENDRAL, GalvaStructures, CONGEN,

GENOA, ASSEMBLE 2.0, CHEMICS, EPIOS, GEN, StructEluc, COCON, SpecSolv, ESESOC, SIGNATURE, SENECA, COCOA, SESAMI, CISOC-SES, and X-PERT take as input any combination of molecular formula, spectral information and spectroscopic data, fragments, molecular weight, molar mass, and constraints such as long-range distance constraints [13, 21–39].

Here we introduce, formulate, and apply quantum isomer searching. Based on the graph-coloring problem, and formulated as a QUBO/Ising model, our approach is able to identify structural isomers of a given molecule in a way that can be implemented on both quantum annealers and gate-based quantum computers. This particular model is able to search for structural isomers of alkanes.

In order to validate our formulation, the search is implemented on the D-Wave 2000Q machine, a state-of-the-art quantum annealer with 2048 superconducting qubits arranged in a sparse chimera graph [40]. It is a quantum computing device that works using quantum annealing, a method that makes use of quantum tunneling and quantum entanglement in order to solve combinatorial optimization problems through minimizing the Ising objective function:

$$f(\boldsymbol{\sigma}) = \sum_i h_i \sigma_i + \sum_{i<j} J_{i,j} \sigma_i \sigma_j \tag{1}$$

for which $\sigma_i \in \{-1, 1\}$ are magnetic spin variables subject to local fields h_i and nearest neighbor interactions with coupling strength $J_{i,j}$. Any problem to be solved on a D-Wave system is modeled as a search for the minimal energy of the Ising Hamiltonian. When the variables σ_i in Eq 1 are restricted to take values from the set {0, 1}, then the minimization problem is said to be a quadratic unconstrained binary optimization (QUBO). A typical QUBO model can be transformed into an Ising model with the transformation $\sigma = 2\boldsymbol{x} - \mathbb{1}_n$, where entries of $\boldsymbol{x}$ represent the n QUBO variables and $\mathbb{1}_n$ is a vector of ones. In the following sections, we formulate the quantum isomer search problem as a QUBO, then describe and present its implementation.

In our formulation, an alkane with n carbons requires $4(n - 2)$ logical qubits that are fully connected. However, D-Wave 2000Q's chimera graph is sparse, and therefore may require a logical qubit to be represented by a chain of physical qubits [41]. This architecture can limit the complexity of possible problems and creates the difficult task of mapping the necessary connections of the logical qubits onto the possible connections of the physical qubits in a process known as "minor embedding" [41]. This leads to the D-Wave 2000Q being capable of representing up to 64 fully connected logical qubits or variables, meaning that our method can find isomers of alkanes with up to $n \leq 18$ carbon atoms (Octadecane).

An important aspect of this problem is that there are multiple correct answers for a given QUBO, i.e. more than one answer satisfies all given constraints. In the case of D-Wave 2000Q, this means that all of these answers have the global minimum energy. In this way, the ground state is degenerate, and to fully answer a given problem, all answers with that energy must be found. This degeneracy is an essential, and necessary part of the problem since the purpose is to find all valid solutions. Quantum annealers are ideal for sampling degenerate solutions because of their ability to introduce some randomness in their exploration of the search space. However, this creates complications because it requires the search space to be explored to an exhaustive degree, which quickly becomes more difficult as the problem size increases. This is a well-known issue with annealing devices, and previous results have found that it can be difficult to sample all degenerate solutions in a fair way [42]. To this end, we apply techniques in an attempt to encourage the search space to be more fully explored than it is with a typical anneal.

Methods

In graph theory, a tree is defined as an acyclic connected graph and it is known that a tree with $n \geq 2$ vertices has at least two vertices of degree 1. Trees with vertices of maximum degree k are said to be k-trees. As our problem formulation considers trees with maximum degree constraints, we recall two theorems (see [43]) about the degree sequence of any given tree to aid in the understanding of the formulation.

Theorem 0.1. *Let* $d_1 \geq d_2 \geq \ldots \geq d_n \geq 0$ *be a sequence of integers. Then* $d_1, d_2, \ldots, d_n$ *is a degree sequence of a tree if and only if*

1. $d_i \geq 1 \forall i = 1, 2, \ldots, n$
2. $\sum_i^n d_i = 2(n-1)$

Theorem 0.2. *Let* $d_1, d_2, \ldots, d_n$ *be a degree sequence of a tree. Then there are*

$$\frac{(n-2)!}{\prod_{i=1}^{n}(d_i - 1)!}$$

labelled trees with the degree sequence $d_1, d_2, \ldots, d_n$ $n \geq 2$.

Theorem 0.1, establishes a necessary and sufficient condition for a degree sequence to be a tree, however Theorem 0.2 shows that this tree is not unique, that there are several labeled trees on n vertices with the same ordered degree sequence. In other words, there is surjective relationship between any disordered degree sequence and the chemical isomers structure set. To make the relationship between structures and sequence more clear, in Fig 1, we show that two degree sequences can give the same chemical structure.

QUBO formulation

Employing acyclic molecular tree graphs to represent alkanes, as well as considering the specific properties of these graphs, we formulate isomer search as a quadratic unconstrained binary optimization (QUBO) problem that can be solved via quantum annealing or gate-based quantum computers. We start off this section by constructing the QUBO objective function for searching for the isomers based on their degree sequences.

Given a molecular formula for an alkane, C_nH_{2n+2}, we consider the carbon-carbon connectivity and set up a degree sequence $(d_1, d_2, \ldots, d_n)$ of the corresponding acyclic molecular tree graph. In this case d_i is the degree of the carbon atom—i.e, if its a primary, secondary or

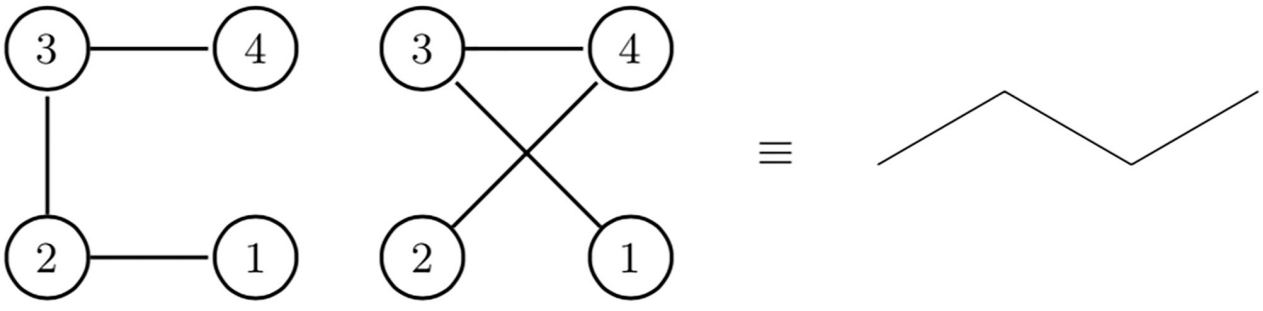

Fig 1. Isomorphism of trees. Two isomorphic tree graphs with degree sequences (1,2,2,1) (left) and (1,1,2,2) (middle) that both correspond to the straight chain isomer (right) of C_4H_{10}.

https://doi.org/10.1371/journal.pone.0226787.g001

tertiary carbon. For our purposes, the hydrogen atoms are irrelevant because they can be inferred from the arrangement of the carbons and can therefore be dropped from the graph. Using the constraints on degree sequences of tree graphs we need to construct the QUBO objective function of the form

$$\boldsymbol{y}^T Q\boldsymbol{y} = \sum_{i=1}^{n} y_i Q_{ii} + \sum_{i\neq j} y_i y_j Q_{ij}$$

where each element y_i of the vector $\boldsymbol{y}$ belongs to the set {0, 1}. Recall that trees with $n \geq 2$ nodes are such that at least two nodes are of degree 1. Without loss of generality, these nodes can be reordered such that they are located at the first and last positions of the corresponding degree sequence. This enables us to set a constraint as $d_1 = d_n = 1$. These nodes correspond to carbons that are in methyl groups (CH_3). Furthermore, the carbon-carbon bond in alkane is such that each carbon atom is bonded to at most 4 other carbon atoms. This implies that alkanes are 4-trees which gives rise to another constraint $1 \leq d_j \leq 4$ for $j = 2,\ldots, n-1$. By the properties of trees, we establish that the sum of the degree sequences must be $2(n-1)$, i.e. $d_1 + d_2 + \ldots + d_n = 2(n-1)$. Thus by Theorem 0.1 we obtain a necessary and sufficent condition that will enable us obtain the structures of the 4-trees (alkanes):

$$d_1 = d_n = 1 \tag{2}$$

$$1 \leq d_j \leq 4 \quad \text{for } j = 2,\ldots,n-1 \tag{3}$$

$$\sum_{j=2}^{n-1} d_j = 2(n-2) \tag{4}$$

To convert the d_i to binary, we define decision variable y_{ij} based on the graph-coloring idea for which a node i is assigned a color j, by considering degrees as colors, that is $j = 1,\ldots, 4$ since the maximum degree is 4. In other words, the number of carbon bonds for an individual atom is one hot encoded in a bit string of length 4.

$$y_{ij} = \begin{cases} 1 & \text{if node } i \text{ is of degree } j (d_i = j) \\ 0 & \text{otherwise} \end{cases}$$

For these variables, we can establish the constraints:

$$\sum_{j=1}^{4} y_{ij} = 1 \qquad i = 1,\ldots,n \tag{5}$$

$$y_{11} = y_{n1} = 1$$

$$\sum_{i=1}^{n}\sum_{j=1}^{4} j y_{ij} = 2(n-1) \tag{6}$$

Fig 2 shows, using 2-methylbutane (an isomer of pentane C_5H_{12}) as an example, how a given alkane can be represented as a molecular graph, a tree graph, a degree sequence, and a one hot encoded bit string. For alkanes with $n \geq 6$ carbons, there are often different

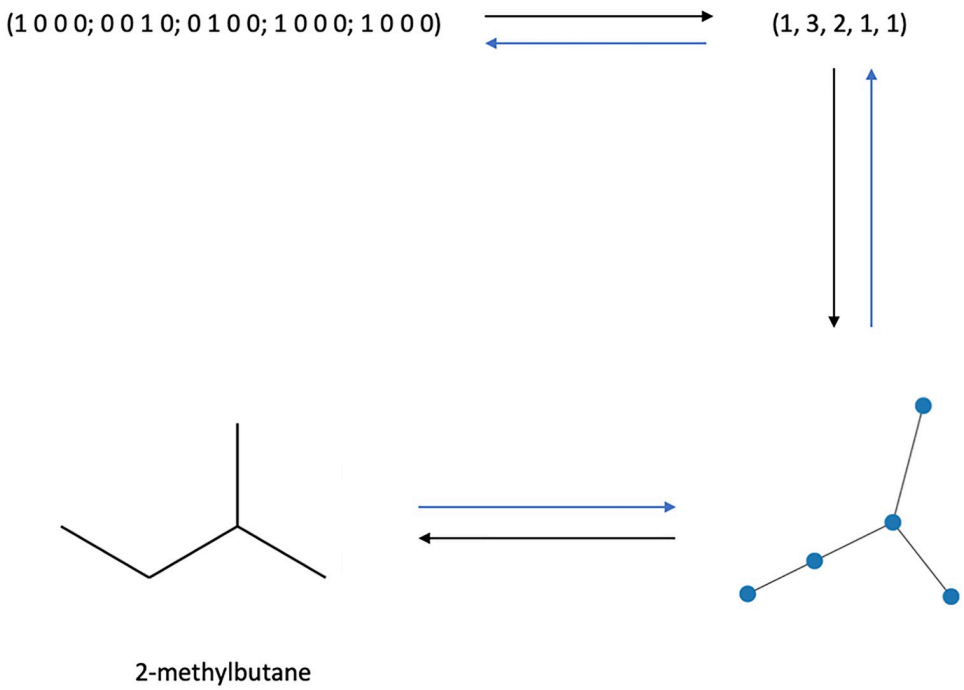

Fig 2. Encoding of 2-methylbutane. Representation of 2-methylbutane (an isomer of pentane C_5H_{12}) as a one hot encoded bit string, degree sequence, graph, and molecular graph (clockwise from the top left).

https://doi.org/10.1371/journal.pone.0226787.g002

permutations that create equivalent isomers. Alternatively, some degree sequence can lead to disconnected structures that needs to be cleaned up in a post-processing step.

In this new formulation, there are $4n$ variables, 8 of which are already predetermined as a result of the constraint $y_{11} = y_{n1} = 1$. Thus, we can restrict the problem to only $M = 4(n - 2)$ variables. This, not only reduces computational complexity but also enables us to explore larger alkanes due to the restrictions on the number of variables or qubits on current machines. For the D-Wave 2000Q machine, this means the ability to explore alkanes with up to 18 carbon atoms instead of 16 carbon atoms.

For simplicity, let us re-number the indices as

$$\begin{aligned} \boldsymbol{y} &= (y_{11}, y_{12}, y_{13}, y_{14}, y_{21}, y_{22}, \ldots, y_{(n-2)1}, y_{(n-2)2}, y_{(n-2)3}, y_{(n-2)4}) \\ &= (y_1, y_2, \ldots, y_M) = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} \in \mathbf{R}^M \end{aligned}$$

We introduce positive penalty constants P_i and proceed with the construction of the QUBO by following the methods in [44]. We penalize the n constraints in Eq 5 with penalty constant

P_1 and the constraint in Eq 6 with penalty constant P_2 as follows:

$$\begin{aligned}
P_1\left(\sum_{i=1}^{4} y_i - 1\right)^2 &= P_1\left(-\sum_{i=1}^{4} y_i + 2\sum_{1\le i<j\le 4} y_i y_j\right) + P_1 \\
P_1\left(\sum_{i=1}^{4} y_{4+i} - 1\right)^2 &= P_1\left(-\sum_{i=1}^{4} y_{4+i} + 2\sum_{1\le i<j\le 4} y_{4+i} y_{4+j}\right) + P_1 \\
P_1\left(\sum_{i=1}^{4} y_{8+i} - 1\right)^2 &= P_1\left(-\sum_{i=1}^{4} y_{8+i} + 2\sum_{1\le i<j\le 4} y_{8+i} y_{8+j}\right) + P_1 \\
&\vdots \\
P_1\left(\sum_{i=1}^{4} y_{4(n-3)+i} - 1\right)^2 &= P_1\left(-\sum_{i=1}^{4} y_{4(n-3)+i} + 2\sum_{1\le i<j\le 4} y_{4(n-3)+i} y_{4(n-3)+j}\right) + P_1 \\
\Rightarrow P_1\sum_{j=0}^{n-3}\left(\sum_{i=1}^{4} y_{4j+i} - 1\right)^2 &= P_1\left[-\sum_{i=1}^{M} y_i + 2\sum_{k=0}^{n-3}\sum_{1\le i<j\le 4} y_{4k+i} y_{4k+j} + (n-2)\right]
\end{aligned} \tag{7}$$

$$P_2\left(\sum_{i=1}^{n-2}\sum_{j=1}^{4} j y_{ij} - 2(n-2)\right)^2 = P_2\left[\sum_{i=1}^{M}[(i-1)(\mod 4)+1]y_i - 2(n-2)\right]^2 \tag{8}$$

Let $\mathbb{1}$ be the $M \times M$ matrix of ones and $\mathbb{1}_M$ be the M column vector of ones. Let $U = [u_{ij}]_{4\times 4}$ be a 4×4 upper triangular matrix with entries defined by

$$u_{ij} = \begin{cases} 1 & \text{if } i < j \\ 0 & \text{otherwise} \end{cases}.$$

That is,

$$U = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

We define a $M \times M$ (block) diagonal matrix D_U with each diagonal block consisting of the matrix U as

$$D_U = diag(U, \ldots, U).$$

Eq 7 can then be rewritten as

$$\begin{aligned}
P_1\sum_{j=0}^{n-3}\left(\sum_{i=1}^{4} y_{4j+i} - 1\right)^2 &= P_1\left[-\sum_{i=1}^{M} y_i + 2\sum_{k=0}^{n-3}\sum_{1\le i<j\le 4} y_{4k+i} y_{4k+j} + (n-2)\right] \\
&= P_1[-\mathbb{1}_M^T \boldsymbol{y} + 2\boldsymbol{y}^T D_U \boldsymbol{y} + (n-2)].
\end{aligned} \tag{9}$$

To rewrite Eq 8 in matrix form, we first define $\alpha_i = (i-1) \bmod 4 + 1$. Then

$$\begin{aligned} & P_2\left[\sum_{i=1}^{M}[(i-1) \mod 4+1]y_i - 2(n-2)\right]^2 \\ &= P_2\left[\sum_{i=1}^{M}\alpha_i y_i - 2(n-2)\right]^2 \\ &= P_2\left[(\sum_{i=1}^{M}\alpha_i y_i)^2 - 4(n-2)\sum_{i=1}^{M}\alpha_i y_i + 4(n-2)^2\right] \\ &= P_2\left[\sum_{i=1}^{M}\alpha_i^2 y_i^2 + 2\sum_{1\le i<j\le M}\alpha_i\alpha_j y_i y_j - 4(n-2)\sum_{i=1}^{M}\alpha_i y_i + 4(n-2)^2\right]. \end{aligned} \tag{10}$$

Now, define the following $M \times M$ matrices and M column vectors

$$D_\alpha = diag(\alpha_1, \ldots, \alpha_M) \tag{11}$$

$$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_M) = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_M \end{bmatrix} \tag{12}$$

$$U_\alpha = [u_{ij}]_{M\times M} \quad \text{with } u_{ij} = \begin{cases} \alpha_j & \text{if } i < j \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Then

$$\begin{aligned} & P_2\left[\sum_{i=1}^{M}\alpha_i^2 y_i^2 + 2\sum_{1\le i<j\le M}\alpha_i\alpha_j y_i y_j - 4(n-2)\sum_{i=1}^{M}\alpha_i y_i + 4(n-2)^2\right] \\ &= P_2[\boldsymbol{y}^T D_\alpha^2 \boldsymbol{y} + 2\boldsymbol{y}^T D_\alpha U_\alpha \boldsymbol{y} - 4(n-2)\boldsymbol{\alpha}^T\boldsymbol{y} + 4(n-2)^2] \end{aligned} \tag{14}$$

Adding all these equations, Eqs 9 + 14, gives

$$\begin{aligned} & P_1[-\mathbb{1}_M^T\boldsymbol{y} + 2\boldsymbol{y}^T D_U\boldsymbol{y} + (n-2)] + P_2[\boldsymbol{y}^T D_\alpha^2\boldsymbol{y} + 2\boldsymbol{y}^T D_\alpha U_\alpha\boldsymbol{y} - 4(n-2)\boldsymbol{\alpha}^T\boldsymbol{y} + 4(n-2)^2] \\ &= \boldsymbol{y}^T[P_2 D_\alpha^2 + 2(P_2 D_\alpha U_\alpha + P_1 D_U)]\boldsymbol{y} + [-4(n-2)P_2\boldsymbol{\alpha} - P_1\mathbb{1}_M)]^T\boldsymbol{y} \\ &+[4(n-2)^2 P_2 + (n-2)P_1] \end{aligned} \tag{15}$$

$$= \boldsymbol{y}^T A\boldsymbol{y} + \boldsymbol{b}^T\boldsymbol{y} + c, \tag{16}$$

where

$$A = P_2 D_\alpha^2 + 2(P_2 D_\alpha U_\alpha + P_1 D_U), \tag{17}$$

$$\boldsymbol{b} = -[4(n-2)P_2\boldsymbol{\alpha} + P_1\mathbb{1}_M], \tag{18}$$

$$c = 4(n-2)^2 P_2 + (n-2)P_1. \tag{19}$$

Eq 16 with A, $\boldsymbol{b}$, c defined by Eqs 17–19 respectively, gives us the objective function for our problem and hence we need to solve the minimization problem

$$\min_{\boldsymbol{y}\in\{0,1\}^M} \boldsymbol{y}^T A\boldsymbol{y} + \boldsymbol{b}^T\boldsymbol{y} + c. \tag{20}$$

Let us define a diagonal matrix $D_{\boldsymbol{b}}$ out of the vector $\boldsymbol{b}$ above, that is

$$D_{\boldsymbol{b}} = diag(b_1, b_2, \ldots, b_M)$$

where the b_i are the vector components of $\boldsymbol{b}$ in Eq 18. Then we solve the QUBO problem

$$\begin{aligned} \min_{\boldsymbol{y}\in\{0,1\}^M} \boldsymbol{y}^T A\boldsymbol{y} + \boldsymbol{b}^T\boldsymbol{y} &= \min_{\boldsymbol{y}\in\{0,1\}^M} \boldsymbol{y}^T A\boldsymbol{y} + \boldsymbol{y}^T D_{\boldsymbol{b}}\boldsymbol{y} \quad \text{since } y_i^2 = y_i \in \{0,1\} \\ &= \min_{\boldsymbol{y}\in\{0,1\}^M} \boldsymbol{y}^T Q\boldsymbol{y} \qquad \text{where } Q = A + D_{\boldsymbol{b}} \end{aligned} \tag{21}$$

In terms of the individual variables, the full QUBO objective function can be expressed as:

$$f(\boldsymbol{y}) = \min_{\boldsymbol{y}\in\{0,1\}^M} P_1 \sum_{i=2}^{n-1} \left(\sum_{j=1}^{4} y_{i,j} - 1 \right)^2 + P_2 \left(\sum_{i=2}^{n-1} \sum_{j=1}^{4} j y_{i,j} - 4(n-2) \right)^2. \tag{22}$$

However, D-Wave 2000Q only accepts h_i values between -2 and 2 and $J_{i,j}$ values between -1 and 1, so the resulting coefficients must be scaled to fit within these ranges. This can help increase the energy gap between the ground state and the first excited state, making it less likely that excited states will be sampled. While the `D-Wave Ocean` software [45] can do this on its own, it was done manually since this allowed some experimentation with the range used as it has been observed that it may sometimes be useful to restrict $J_{i,j}$ values to be greater than -0.8 [46]. This was done, although no significant benefit was observed.

It is important to note that, as a result of the constraint relating to the sum of the degrees of the atoms, the resulting QUBO is fully connected. Each atom must take into account the number of carbon bonds of all other atoms in order to determine if its coloring violates this constraint. This can be visualized using the graph representations of the QUBOs (which is what would be embedded into the D-Wave chimera graph) for Butane ($n = 4$) and Heptane ($n = 7$) given in Fig 3.

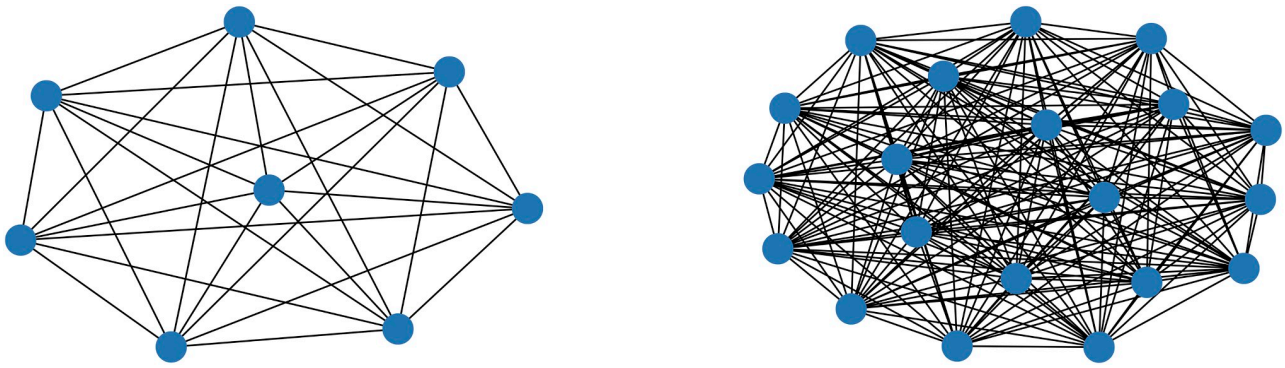

Fig 3. Graphs of QUBOS. A: Butane (C_4H_{10}), B: Heptane (C_7H_{16}).

https://doi.org/10.1371/journal.pone.0226787.g003

Implementation

The QUBO for a given alkane is embedded into the D-Wave 2000Q_5 chimera graph. This is the newly released lower-noise machine that is available via D-Wave's Leap™ quantum cloud service [47]. It was also implemented on the D-Wave 2000Q_LANL machine at Los Alamos National Laboratory [48], but no significant difference in performance were noted. Once the QUBO is embedded using `D-Wave Ocean` [45], the annealer attempts to find the lowest energy solutions, i.e. the bit strings that violate the fewest constraints. This was done using the standard 20μs anneal time. It is important to remember that the lowest energy solutions found by annealing are the valid isomers. Note that this is different from finding the most chemically stable isomer.

The sampled results are then filtered such that only the lowest energy solutions (the isomers) are returned. These one hot encoded results are decoded into the degree sequences and graphs in the method described previously, checked and filtered for redundancy, and returned. As n increases, the relative number of possible results ($2^{4(n-2)}$) grows more quickly than the number of isomers. Furthermore, it is known that larger problems on imperfect quantum annealers have lower probabilities of sampling a ground state solution [49]. Therefore, it becomes necessary to increase the number of samples taken from the embedded QUBO.

To address this problem, an increasingly perturbed QUBO was also used. In this formulation, after every 10,000 samples, the outer product of the ground state result with the most counts, $|\psi\rangle$, with itself was added to the QUBO in an attempt to impose a penalty on returning that result in the next iteration. This new QUBO is represented as

$$Q' = Q + \lambda|\psi\rangle\langle\psi|. \tag{23}$$

The idea behind using this perturbed QUBO is that, by penalizing previously returned results, subsequent sampling runs would be encouraged to explore different parts of the search space that may have valid solutions that had not been visited. Because the search space becomes extremely large as n increases, it is possible that this may help facilitate the identification of all isomers of larger molecules. To the best of our knowledge this is the first time that such a technique has been used to boost the solution space exploration in quantum annealers.

Finally, reverse annealing was added. Rather than ending in a classical state after slowly turning down the strength of the transverse field, this method does the opposite by taking a fully classical state as input, which is then stimulated with an increasingly high transverse field until it reaches the pause location, s^* [50]. At this location, $\mathcal{H}(t) = \mathcal{H}(s^*)$, where $\mathcal{H}(0)$ and $\mathcal{H}(1)$ are the starting and ending Hamiltonians of a forward anneal, respectively. Following this step, the system pauses for some pre-determined time and progresses as a typical forward anneal as the transverse field is gradually weakened, eventually ending once again in a classical state [50]. The classical input states were the results given by a typical forward anneal, the pause location was chosen to be $s^* = 0.5$, and the system was paused for $h = 85\mu s$. One of the ideas behind reverse annealing is that it allows the search space surrounding candidate solutions given by a forward anneal to be further explored [50]. If the forward anneal returns a local minimum then reverse annealing may stimulate that solution to an extent that the system settles into a nearby global minimum [50]. Such a result could make it more likely that a given run returns a result with the minimum energy, which may help with the successful enumeration of all isomers and even decrease the number of samples necessary to find them all.

When one expands the QUBO into the Pauli basis, variational methods, such as the variational quantum eigensolver (VQE) [51] or the quantum approximate optimization algorithm (QAOA) [52], employed on gate-based machines can also approximate solutions to the Ising Problem [2]. As a result, we explored the possibility of using IBM Q's `Qiskit` software on the

available QASM simulator [53]. Because of the requirement of $4(n-2)$ qubits, IBM Q's Tokyo, which has 20 qubits and is currently their largest available device, can only handle alkanes with fewer carbon atoms than Octane (C_8H_{18}) [54]. However, Google Bristlecone's 72 qubits will be able to encode 18 carbon atoms, allowing Icosane's ($n = 20$) 366,319 structural isomers to be searched for [55].

Results

Using Python packages `NumPy` [56], `D-Wave Ocean` [45], `Sympy` [57], `NetworkX` [58], and `Matplotlib` [59] and the D-Wave 2000Q hardware, all structural isomers for Butane (C_4H_{10}), Pentane (C_5H_{12}), Hexane (C_6H_{14}), Heptane (C_7H_{16}), Octane (C_8H_{18}), and Nonane (C_9H_{20}) were identified. These molecules have 2, 3, 5, 9, 18, and 35 isomers, respectively. Fig 4 shows the returned graphs and their corresponding isomers for Heptane (C_7H_{16}).

Without using QUBO perturbation and reverse annealing, it was found that 10,000 samples were sufficient to find all isomers for Butane (C_4H_{10}) and Pentane (C_5H_{12}), but the larger alkanes often needed well over 50,000 samples in order to be fully captured. Information evaluating and describing the results is given below.

Fig 5 gives information on the Hamming distances of all of these isomers. The Hamming distance between two isomers is the number of bit flips that must be made in order to turn one isomer into the other. The left figure shows all pairwise Hamming distances for a given n, and the right figure shows the minimum Hamming distance to each isomer for a given n. As can be seen, while the pairwise Hamming distances tend to follow a fairly wide distribution, almost every isomer has another isomer within the minimum possible Hamming distance (4).

Figs 6 and 7 give information on the frequency with which isomers are found for Butane (C_4H_{10}) and Heptane (C_7H_{16}). In both figures, it is easily seen that isomers of Butane, with only 4 carbon atoms, are much more easily found than those of Heptane (C_7H_{16}). Fig 6 (left) demonstrates that ground state results, i.e. those that violate no constraints, are found several

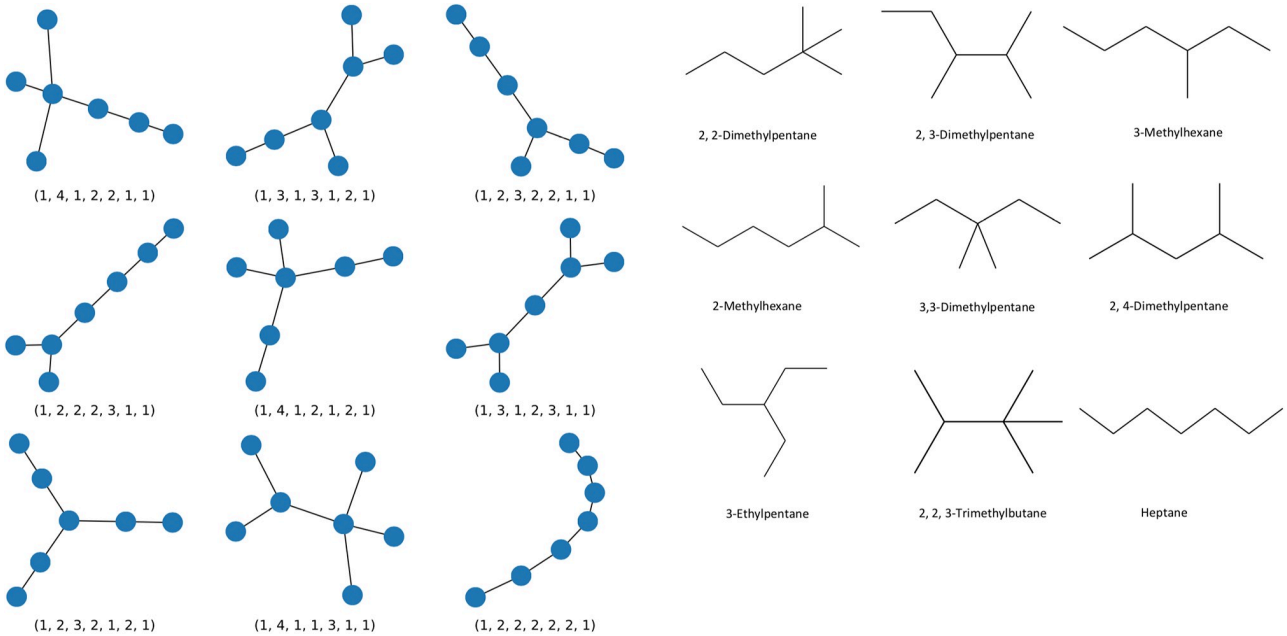

Fig 4. Heptane isomers. Created graphs and corresponding isomers for Heptane (C_7H_{16}) Left: Returned graphs with degree sequences, Right: Isomers of Heptane.

https://doi.org/10.1371/journal.pone.0226787.g004

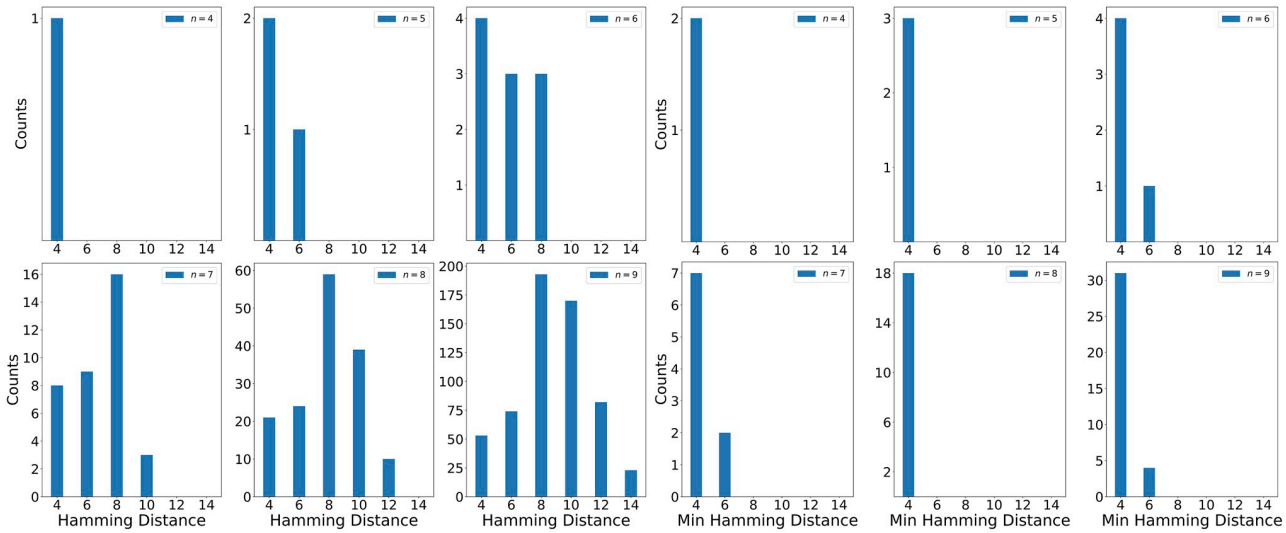

Fig 5. Measures of Hamming distances. Left: All pairwise Hamming distances, Right: Minimum Hamming distance to each isomer.

https://doi.org/10.1371/journal.pone.0226787.g005

hundred times per 10,000 anneals when Butane (C_4H_{10}) is being investigated. However, as the Fig 6 (right) shows, only a handful are sampled for Heptane (C_7H_{16}). Fig 7 (left) shows that each isomer is, on average, found several hundred times per 10,000 samples whereas the right plot shows that isomers of Heptane (C_7H_{16}) are generally found less than once per 10,000 samples.

Our sample reduction methods were also explored and evaluated. Perturbing the QUBO clearly had an effect on the distribution of the returned results. Fig 8 gives the distributions of the returned isomers with (left) and without (right) perturbing the QUBO for Pentane (C_5H_{12}) after 10,000 samples using $\lambda = 5(10^{-5})$. The distributions for the non-perturbed QUBO runs, Fig 8 (left), are somewhat uniform. Every isomer is found during each iteration, and the isomers are roughly returned at the same rate. The randomness of the annealing will always introduce some fluctuations. However, these fluctuations are not too large and tend to settle back to normal by the next iteration. This is starkly contrasted by Fig 8 (right). This shows the distributions when QUBO perturbation is used and as can easily be seen, this method

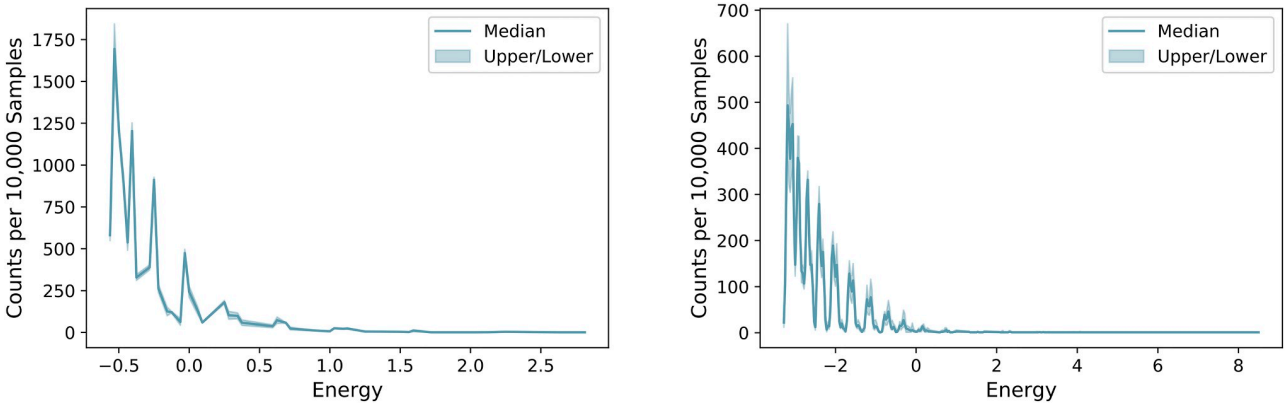

Fig 6. Number of results returned for each energy. Number of results out of 10,000 samples returned at each energy. Left: Butane (C_4H_{10}), Right: Heptane (C_7H_{16}).

https://doi.org/10.1371/journal.pone.0226787.g006

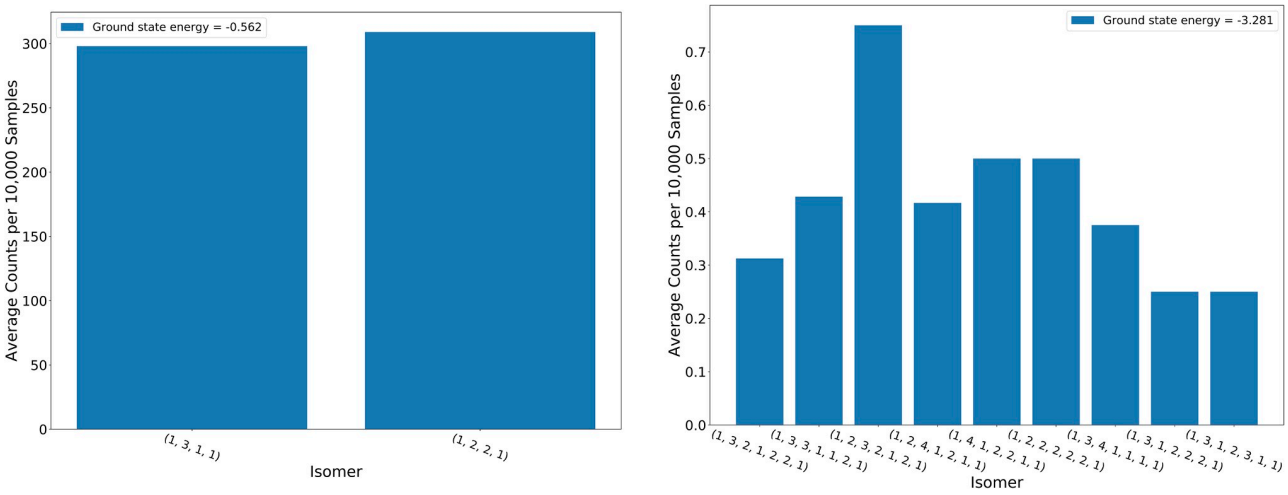

Fig 7. Distribution of returned isomers. Average number of times each isomer was returned per 10,000 samples. Left: Butane (C_4H_{10}), Right: Heptane (C_7H_{16}).

https://doi.org/10.1371/journal.pone.0226787.g007

drastically changes the results for subsequent runs. The isomer that is sampled the most frequently for a given iteration is typically sampled significantly fewer times during the next iteration. Eventually, an isomer that has been sampled the most frequently is penalized to the extent that it is never sampled again. By the final iteration, each isomer has at some point been the most frequently sampled. This seems to drive the QUBO so far away from them that none of them are sampled.

This QUBO perturbation technique and reverse annealing were tried alone and in tandem. It was found that, by themselves, each typically led to a reduction in the number of samples needed, but combining them decreased the number of samples even more significantly. The effect of these methods on the search for isomers of Heptane (C_7H_{16}) was measured by finding the number of iterations of 10,000 samples that were necessary to find all isomers. As this is not a constant number, the experiment was repeated 25 times for each of the four methods (only forward annealing, forward annealing with QUBO perturbation, reverse annealing, and

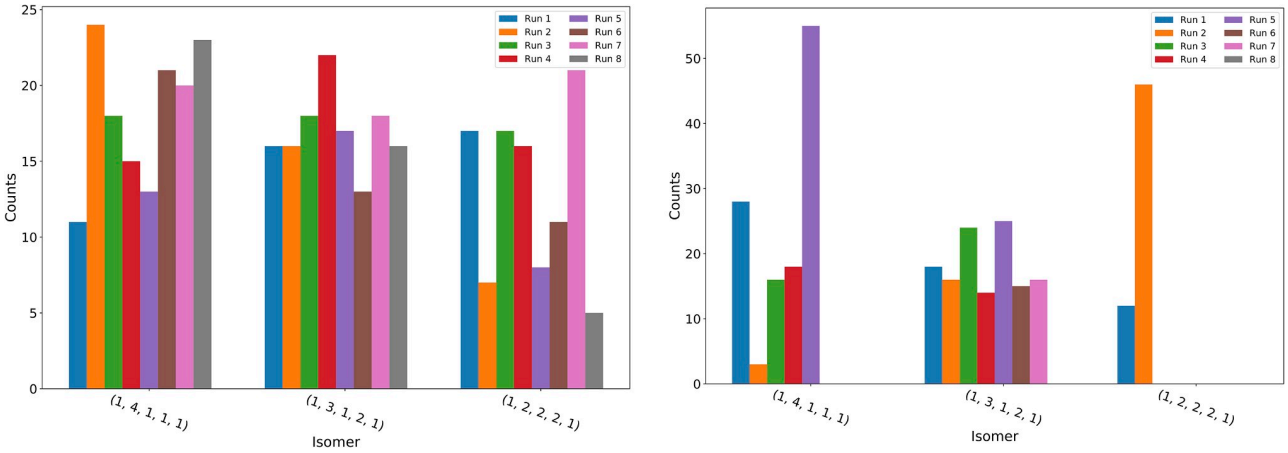

Fig 8. Sequential distributions of results. Distributions of returned isomers of Pentane (C_5H_{12}) after each run of 10,000 samples. Left: Not using QUBO perturbation, Right: Using QUBO perturbation.

https://doi.org/10.1371/journal.pone.0226787.g008

Table 1. Number of runs to find all Heptane isomers.

	FA	FA + QP	RA	RA + QP
Mean	9.68	8.44	8.04	6.56
Median	9	7	8	6

Average and median number of samples (in 10,000s) needed to find all isomers of Heptane (C_7H_{16}) using forward annealing (FA), forward annealing and QUBO perturbation (FA + QP), reverse annealing (RA), and reverse annealing and QUBO perturbation (RA + QP). $s^* = 0.5$, $h = 85\mu s$, $\lambda = 5(10^{-6})$.

https://doi.org/10.1371/journal.pone.0226787.t001

reverse annealing with QUBO perturbation). The results are shown in Table 1. This gives the average and median number of runs needed to find all isomers for each different technique over the 25 runs. As it shows, both QUBO perturbation and reverse annealing separately outperform a typical forward anneal, but combining the two gives the largest reduction in the number of samples needed.

When the runtime scaling was evaluated, it was found that the time taken to generate 10,000 samples grows fairly linearly with n, with each additional carbon atom adding roughly 20 microseconds to the total runtime, seen in Fig 9. This happens despite the fact that the number of parameters grows quadratically. QPU access time includes everything that is done on the QPU: QPU programming time and QPU sampling time. QPU programming time measures how long it takes to initialize the problem on the QPU, a procedure that is only done once per 10,000 samples. QPU sampling time includes the time it takes to perform and readout all anneals, with delays in between subsequent samples to allow for the system to return to its initial temperature [60]. After all samples are collected, they then undergo post-processing in an attempt to improve the quality of the solutions [61, 62]. These three times are shown in Fig 9 (left). As can be seen, while QPU access time dominates, post processing time has the

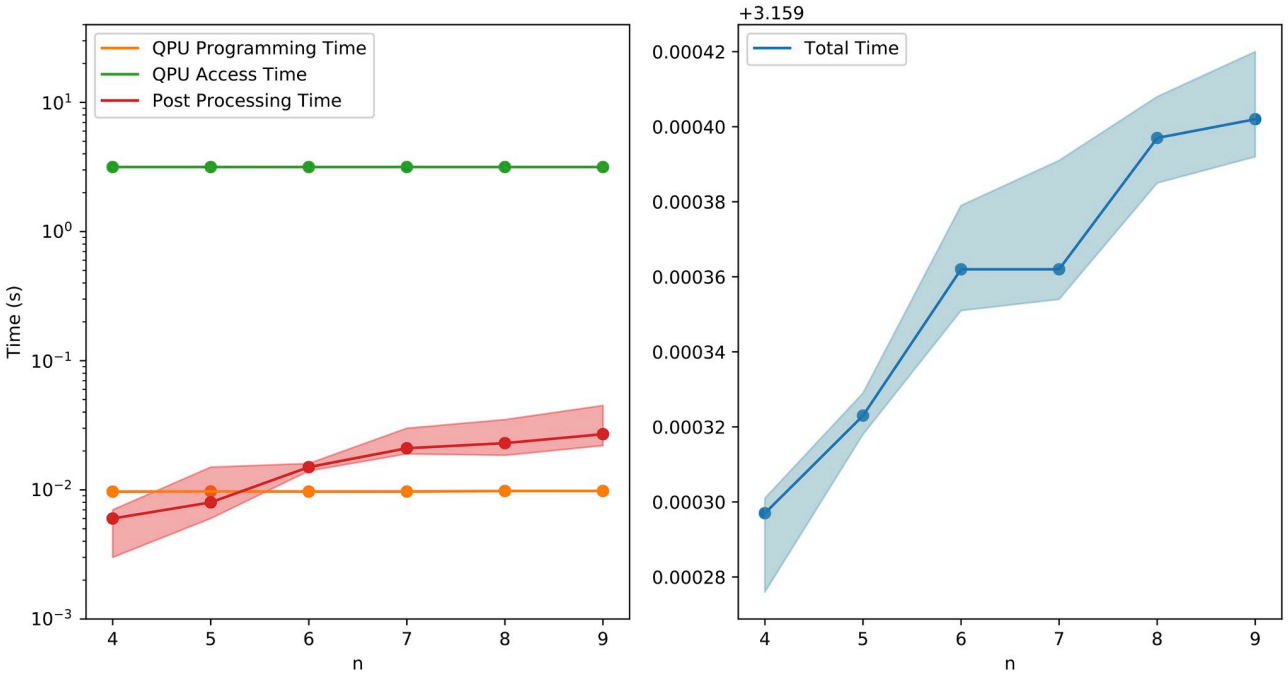

Fig 9. Benchmark times. Time taken per 10,000 samples for $4 \leq n \leq 9$. Left: QPU programming time, QPU access time, and total post processing time. Right: total real time (+3.159 seconds). Note that the left panel is on a log scale.

https://doi.org/10.1371/journal.pone.0226787.g009

largest relative scaling. This is reflected in the right panel of Fig 9, which shows that the total time grows linearly with the number of atoms.

Finally, there was also some attempt to use IBM Q's `Qiskit` to implement the search on their hardware [53]. Current attempts on the IBM Q simulator using this method have been able to identify all isomers of Butane (C_4H_{10}) and calculate the correct ground state energy of the QUBO Hamiltonian. However, 2-methylpropane (an isomer of Butane C_4H_{10}) is found very often, typically well over five hundred counts per 8,192 samples. This generally makes it one of the three most common results. However, unbranched Butane (C_4H_{10}) occurs much more rarely.

Discussion

These results are a proof of concept that quantum isomer search using a QUBO formulation is a valid method. With this approach all isomers for all alkanes with fewer carbon atoms than Decane ($C_{10}H_{22}$) were identified. However, as the number of carbon atoms grows, it becomes more and more essential to take more samples. As shown in Table 1, combining reverse annealing with our method of perturbing the QUBO after every iteration of 10,000 samples decreased the number of samples required to find all isomers. Therefore, we can speculate that perturbing the QUBO and reverse annealing are important methods that may significantly help expand the search space, decrease the runtime, and facilitate the complete identification of isomers for larger molecules.

It is important to note that the necessity of increasing the number of samples as the problem size grows is indicative of imperfect hardware. This scaling is not an inherent part of the quantum isomer search algorithm. Increasing the problem size decreases the probability of finding a successful answer due to annealing error and imperfect hardware [49]. It is not surprising that more runs are needed for larger molecules, especially when the quadratic scaling of parameters and the limited connectivity of the D-Wave 2000Q's chimera graph are taken into account. This is a large contributor to the need for more sampling for larger molecules and comes from imperfect hardware rather than the scaling of the algorithm itself.

Evidence for this can be seen in Fig 10. This figure compares the number of samples returned for each energy when the isomers of Octane (C_8H_{18}) are searched for. As is easily seen, the number of ground state samples is significantly lower for the quantum annealer (left) when compared to the simulated annealer (right). This indicates that the algorithm is working correctly, but the hardware is limiting its performance. Furthermore, every isomer was able to

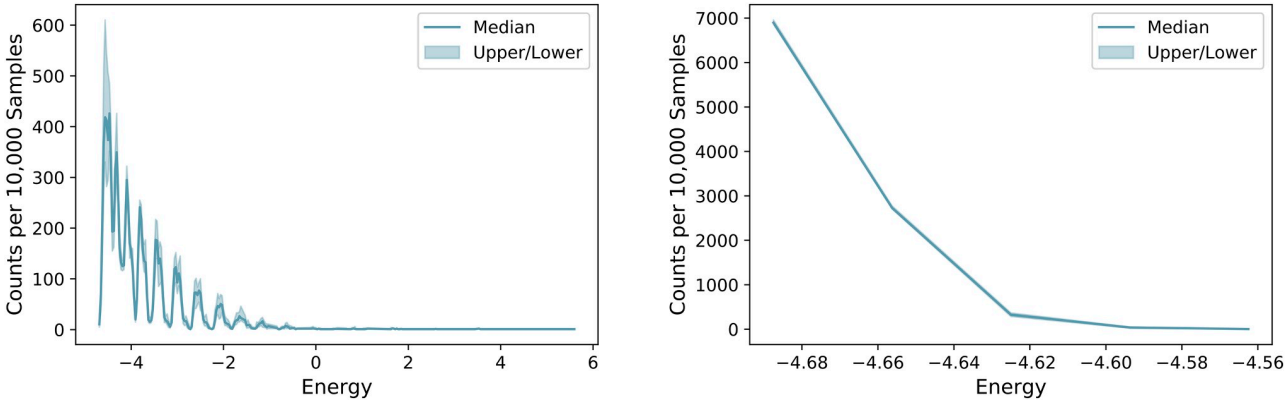

Fig 10. Number of results returned for each energy. Left: using quantum annealing, Right: using simulated annealing.

https://doi.org/10.1371/journal.pone.0226787.g010

be found within 10,000 samples when simulated annealing was used, regardless of the size of the molecule. This also indicates the fact that the imperfect hardware is limiting the performance and is responsible for the increase in the number of samples needed for larger molecules to some extent.

We speculate, however, that as the hardware's performance increases, it may be the case that the quantum annealer will outperform the simulated annealer, particularly in runtime. This is due to the linear scaling of the quantum runtime with problem size, whereas simulated annealing has been found to scale exponentially in some cases [50]. In fact, some problems with 945 variables were found to run over 10^8 times slower when simulated annealing was used [63]. Embedding 13 carbons, i.e. searching for isomers of Pentadecane ($C_{15}H_{32}$), requires 946 variables. Because this molecule has fewer than 18 carbons, it is small enough to already be embedded on D-Wave 2000Q. Therefore, this enormous potential decrease in runtime may occur for isomers that are small enough to be embedded on current hardware. However, as discussed earlier, even though a molecule with this size can be embedded, the hardware's error makes searching for the 4,347 isomers impractical. As a result, this runtime decrease will have to wait for better hardware. We make no claim that the potential decrease in runtime will be of the same magnitude as that found in [63]. However, the complexity of the isomer search problem along with the sheer magnitude of the runtime decrease that was demonstrated suggest that some decrease in runtime should be expected when quantum isomer search is applied to larger molecules.

Degeneracy is an additional issue that pertains to the need for thorough sampling. As described earlier, the global minimum is degenerate in that there are multiple solutions that satisfy all constraints and all of these need to be sampled in order to ensure that all isomers are found. This necessitates a very thorough exploration of the search space. Such a requirement is the reason that QUBO perturbation was added. The results seem to suggest that this would encourage a wider exploration of the search space. The complication is furthered by the fact that for several isomers, there are multiple permutations of a degree sequence that lead to identical graphs. Therefore, in a way the degeneracy is two-fold. There are multiple valid and distinct global minima, but there are also multiple valid yet identical global minima. The degeneracy will only increase as the problem size increases and is perhaps one of the largest limitations.

The degeneracy is further complicated by the Hamming distances between the isomers. The Hamming distance is a measure of how many modifications need to be made to a result in order to transform it into another result. Because of the one hot encoding of the degrees, changing the degree of one carbon requires 2 bit flips. The constraint pertaining to the sum of the degrees means that if one degree is changed, at least one other degree must be changed and at most $(n - 2)$ degrees. The total number of carbons embedded may be changed. Therefore, the Hamming distance between any two isomers of a given n is strictly within 4 and $2(n - 2)$, inclusive. As can be seen in the left panel of Fig 5, the pairwise distribution of Hamming distances follows this pattern. However, when the minimum Hamming distance to a given isomer is calculated, as is shown in the right panel of Fig 5, it is seen that for almost every isomer for all n another isomer can be made by changing only 2 degrees. Therefore, while any two isomers may be far apart, almost every isomer has another isomer quite close by.

When measured in Hamming distances, the isomers can form very close clumps of ground state results. It is possible that this property helps simulated annealing deal with degeneracy in an effective way. Once a given clump is found, the other isomers can be found by only flipping a handful of bits. The implications for quantum annealing are less clear. It is entirely possible that a clump may be found, but not all isomers within that clump are found because the quantum annealing explores so vastly that it may quickly leave the clump. Alternatively, it is

possible that the annealer would be drawn to larger clumps, i.e. isomers from the clumps that contain many molecules with small Hamming distances would be more likely to be visited and explored. It is possible that this is an issue that can be addressed to some extent by the introduction of reverse annealing. Its ability to do local searches surrounding the candidate solution given by forward annealing may allow it to explore a given clump more fully. Furthermore, QUBO perturbation may help the annealing explore between clumps by driving the search away from clumps with answers that were already visited. This complementary combination of exploring within and between clumps introduced by reverse annealing and QUBO perturbation may be the reason that combining the two methods is so effective in terms of decreasing the number of samples needed to find all isomers. There very well may be other implications that we have not brought up, so this may be an interesting direction for further research on degenerate problems.

On D-Wave 2000Q, the sampling of a QUBO that grows quadratically in the number of parameters can be done in linear time. Despite the fact that more samples are required for larger molecules, the linear scaling of sampling time is an important quality. When combined with the significant reduction in samples needed due to QUBO perturbation and reverse annealing it becomes an encouraging sign for the feasibility of applying this method to larger molecules.

Even though the current results on the IBM Q hardware are not competitive with those from D-Wave 2000Q, the rapidly increasing performance and growing number of qubits on this and other gate-based machines makes this direction a promising avenue for further research. However, variational techniques on gate-based machines may not have the linear runtime scaling that is found when quantum annealing is used. This is because these techniques will likely use QUBO Hamiltonians that come with a quadratically scaling number of terms due to the quadratic scaling of the problem. Because the expectation value of all of these terms must be taken when a technique such as VQE is used, this will likely result in a runtime that scales closer to quadratic than linear. Furthermore, initial results do not seem to indicate that these variational techniques can handle degeneracy as effectively as quantum annealing. However, it seems that this limitation may be addressed to some extent by QUBO perturbation as well as other methods including low-energy subspace sampling using something akin to a subspace-search variational quantum eigensolver [64].

Conclusion

We have demonstrated that quantum isomer search using the QUBO formulation is possible and effective. With our approach, the sampling time grows linearly with the number of carbon atoms. All isomers for all alkanes with fewer carbon atoms than Decane ($C_{10}H_{22}$) were identified and enumerated using this approach on the D-Wave 2000Q system. Alkanes with fewer carbon atoms than Nonadecane ($C_{19}H_{40}$) can be embedded directly into the D-Wave 2000Q for the quantum isomer search. However, the next-generation D-Wave with 5000 qubits is coming soon. Along with its decrease in noise and significant increase in the number of physical qubits, it will also feature a more connected Pegasus graph in which each physical qubit is connected to 15 others rather than only 6 [65]. This combination will allow the isomers of much larger molecules to be searched for. It is likely that as the problem size increases, the importance of the significant sample reduction and wider exploration of the search space made possible by perturbing the QUBO and adding reverse annealing will quickly grow.

The natural next step of this problem is to implement it on a gate-based quantum computer. Variational methods on these computers can also solve the Ising problem, so quantum isomer search is possible on those machines. However, all available gate-based quantum

computers have significantly fewer qubits than D-Wave 2000Q, so they can only search for isomers of relatively small molecules. Regardless, the number of qubits that these machines have is quickly growing as their noise is decreasing, so it is a promising direction of future work.

Acknowledgments

The authors would like to acknowledge the NNSA's Advanced Simulation and Computing (ASC) program at Los Alamos National Laboratory (LANL) for use of their Ising D-Wave 2000Q quantum computing resource. LANL is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218NCA000001). The authors would also like to acknowledge D-Wave Systems Inc. for the use of their lower-noise D-Wave 2000Q from D-Wave Leap. Jason Terry was funded by the LANL Quantum Computing Summer School (QCSS) 2019. The QCSS is sponsored by the LANL Information Science and Technology Institute (ISTI). This research was also supported in part by an appointment by the National Science Foundation (NSF) Mathematical Sciences Graduate Internship (MSGI) Program sponsored by the NSF Division of Mathematical Sciences. Prosper Akrobotu's contribution to this work was funded by NSF-MSGI. This program is administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between the U.S. Department of Energy (DOE) and NSF. ORISE is managed for DOE by ORAU. All opinions expressed in this paper are the author's and do not necessarily reflect the policies and views of NSF, ORAU/ORISE, or DOE. Christian Negre's and Sue Mniszewski's contributions to this research have been funded by the LANL ISTI and Laboratory Directed Research and Development (LDRD). Assigned: Los Alamos Unclassified Report 19-26724.

Author Contributions

Conceptualization: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre, Susan M. Mniszewski.

Funding acquisition: Susan M. Mniszewski.

Investigation: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre, Susan M. Mniszewski.

Methodology: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre, Susan M. Mniszewski.

Project administration: Susan M. Mniszewski.

Resources: Susan M. Mniszewski.

Software: Jason P. Terry, Prosper D. Akrobotu.

Supervision: Christian F. A. Negre, Susan M. Mniszewski.

Validation: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre, Susan M. Mniszewski.

Visualization: Jason P. Terry, Prosper D. Akrobotu.

Writing – original draft: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre.

Writing – review & editing: Jason P. Terry, Prosper D. Akrobotu, Christian F. A. Negre, Susan M. Mniszewski.

References

1. Lucas A. Ising formulations of many NP problems. Frontiers in Physics. 2014; 2:5. https://doi.org/10.3389/fphy.2014.00005
2. Cervera-Lierta A. Exact Ising model simulation on a quantum computer. Quantum. 2018; 2:114. https://doi.org/10.22331/q-2018-12-21-114
3. Bian Z, Chudak F, Macready W, Rose G. The Ising model: teaching an old problem new tricks. D-Wave Systems. 2010;.
4. Ushijima-Mwesigwa H, Negre CFA, Mniszewski SM. Graph Partitioning Using Quantum Annealing on the D-Wave System. In: Proceedings of the Second International Workshop on Post Moores Era Supercomputing. PMES'17. New York, NY, USA: ACM; 2017. p. 22–29. Available from: http://doi.acm.org/10.1145/3149526.3149531.
5. Negre CFA, Ushijima-Mwesigwa H, Mniszewski SM. Detecting Multiple Communities Using Quantum Annealing on the D-Wave System. arXiv preprint arXiv:190109756. 2019;.
6. Minkin VI. Glossary of terms used in theoretical organic chemistry. J Macromol Sci Part A Pure Appl Chem. 1999; 71(10):1919–1981.
7. Moss GP. Basic terminology of stereochemistry (IUPAC Recommendations 1996). J Macromol Sci Part A Pure Appl Chem. 1996; 68(12):2193–2222.
8. Reading E, Munoz-Muriedas J, Roberts AD, Dear GJ, Robinson CV, Beaumont C. Elucidation of Drug Metabolite Structural Isomers Using Molecular Modeling Coupled with Ion Mobility Mass Spectrometry. Analytical Chemistry. 2016; 88(4):2273–2280. https://doi.org/10.1021/acs.analchem.5b04068 PMID: 26752623
9. Mäki-Arvela P, Kaka khel T, Azkaar M, Engblom S, Murzin D. Catalytic Hydroisomerization of Long-Chain Hydrocarbons for the Production of Fuels. Catalysts. 2018; 8(534).
10. Ranzi E, Pierucci S, Dente M, van Goethem M, van Meeuwen D, Wagner E. Correct Molecular Reconstruction of Cracking Feeds: a Need for the Accurate Predictions of Ethylene Yields. Chemical Engineering Transactions. 2015; 43:871–876.
11. Faulon JL, Visco DP, Roe D. Enumerating Molecules. In: Reviews in Computational Chemistry. vol. 21. Hoboken, New Jersey: John Wiley & Sons Inc.; 2005. p. 209–275.
12. Lederberg J. Topology of Molecules. In: The Mathematical sciences; a collection of essays. Cambridge, Mass.: M.I.T. Press; 1969. p. 37–51.
13. Lindsay RK, Buchanan BG, Feigenbaum EA, Lederberg J. Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project. New York: McGraw-Hill; 1980.
14. Lederberg J, Sutherland GL, Buchanan BG, Feigenbaum EA, Robertson AV, Duffield AM, et al. Applications of artificial intelligence for chemical inference. I. Number of possible organic compounds. Acyclic structures containing carbon, hydrogen, oxygen, and nitrogen. Journal of the American Chemical Society. 1969; 91(11):2973–2976. https://doi.org/10.1021/ja01039a025
15. Hendrickson JB, Parks CA. Generation and enumeration of carbon skeletons. Journal of Chemical Information and Computer Sciences. 1991; 31(1):101–107.
16. Contreras ML, Valdivia R, Rozas R. Exhaustive generation of organic isomers. 1. Acyclic structures. Journal of Chemical Information and Computer Sciences. 1992; 32(4):323–330.
17. Luinge HJ. AEGIS, a Structure Generation Program in Prolog. MATCH. 1992; 27:175.
18. Zhu SY, Zhang JP. Exhaustive generation of structural isomers for a given empirical formula—a new algorithm. Journal of Chemical Information and Computer Sciences. 1982; 22(1):34–38.
19. Barone R, Barberis F, Chanon M. Exhaustive Generation of Organic Isomers from Base 2 and Base 4 Numbers. MATCH. 1995; 32:19–25.
20. Kerber A, Laue R, Moser DA. Structure Generator for Molecular Graphs. Analytica Chimica Acta. 1990; 235:2973.
21. Le Bret C. Exhaustive Isomer Generation using the Genetic Algorithm. Match. 2000; 41: 79–97.
22. Carhart RE, Smith DH. Applications of artificial intelligence for chemical inference–XX. Computers & Chemistry. 1977; 1:79–84.
23. Carhart RE, Smith DH, Gray NAB, Nourse JG, Djerassi C. GENOA: A computer program for structure elucidation utilizing overlapping and alternative substructures. Journal of Organic Chemistry—J ORG CHEM. 1981; 46.
24. Badertscher M, Korytko AI, Schulz KP, Madison MS, Munk ME, Portmann P, et al. Assemble 2.0: a structure generator. Chemometrics and Intelligent laboratory Systems. 2000; 51(1):73–79. https://doi.org/10.1016/S0169-7439(00)00056-3

25. Funatsu K, Miyabayashi N, Sasaki S. Further development of structure generation in the automated structure elucidation system CHEMICS. Journal of Chemical Information and Computer Sciences. 1988; 28(1):18–28.
26. Carabedian M, Dagane I, Dubois JE. Elucidation by progressive intersection of ordered substructures from carbon-13 nuclear magnetic resonance. Analytical Chemistry. 1988; 60(20):2186–2192. https://doi.org/10.1021/ac00171a005
27. Bohanec S, Zupan J. Structure Generator GEN. MATCH. 1992; 27:49.
28. Elyashberg ME, Blinov KA, Williams AJ, Martirosian ER, Molodtsov SG. Application of a New Expert System for the Structure Elucidation of Natural Products from Their 1D and 2D NMR Data. Journal of Natural Products. 2002; 65(5):693–703. https://doi.org/10.1021/np0103315 PMID: 12027744
29. Lindel T, Junker J, Köck M. Cocon: From NMR Correlation Data to Molecular Constitutions. Molecular modeling annual. 1997; 3(8):364–368. https://doi.org/10.1007/s008940050052
30. Will M, Fachinger W, Richert JR. Fully Automated Structure ElucidationA Spectroscopist's Dream Comes True. Journal of Chemical Information and Computer Sciences. 1996; 36(2):221–227. https://doi.org/10.1021/ci950092p
31. Hu C, Xu L. Computer Automated Structure Elucidation Expert System, Esesoc. Fenxi Huaxue. 1992; 20:643.
32. Hao J, Xu L, Hu C. Expert System for Elucidation of Structures of Organic Compounds (Esesoc)- Algorithm on Stereoisomer Generation. vol. 43 of B: Chemistry. Science in China; 2000.
33. Faulon JL. Stochastic Generator of Chemical Structure. 1. Application to the Structure Elucidation of Large Molecules. Journal of Chemical Information and Computer Sciences. 1994; 34(5):1204–1218.
34. Faulon JL. Stochastic Generator of Chemical Structure. 2. Using Simulated Annealing To Search the Space of Constitutional Isomers. Journal of Chemical Information and Computer Sciences. 1996; 36(4):731–740. https://doi.org/10.1021/ci950179a
35. Steinbeck C. SENECA: A Platform-Independent, Distributed, and Parallel System for Computer-Assisted Structure Elucidation in Organic Chemistry. Journal of Chemical Information and Computer Sciences. 2001; 41(6):1500–1507. https://doi.org/10.1021/ci000407n PMID: 11749575
36. Christie BD, Munk ME. Structure generation by reduction: a new strategy for computer-assisted structure elucidation. Journal of Chemical Information and Computer Sciences. 1988; 28(2):87–93. https://doi.org/10.1021/ci00058a009 PMID: 3392122
37. Christie BD, Munk ME. The role of two-dimensional nuclear magnetic resonance spectroscopy in computer-enhanced structure elucidation. Journal of the American Chemical Society. 1991; 113(10):3750–3757. https://doi.org/10.1021/ja00010a018
38. Peng C, Yuan S, Zheng C, Hui Y, Wu H, Ma K, et al. Application of Expert System CISOC-SES to the Structure Elucidation of Complex Natural Products. Journal of Chemical Information and Computer Sciences. 1994; 34(4):814–819.
39. Elyashberg ME, Martirosian ER, Karasev YZ, Thiele H, Somberg H. X-PERT: a user-friendly expert system for molecular structure elucidation by spectral methods. Analytica Chimica Acta. 1997; 337:265–286. https://doi.org/10.1016/S0003-2670(96)00391-1
40. D-Wave. The D-Wave 2000Q System;. https://www.dwavesys.com/d-wave-two-system.
41. D-Wave QPU Architecture: Chimera; 2019. Available from: https://docs.dwavesys.com/docs/latest/c_gs_4.html [cited 2019 July 11].
42. Matsuda Y, Nishimori H, Katzgraber H . Quantum annealing for problems with ground-state degeneracy. Journal of Physics: Conference Series. 2009; 143(012003).
43. Bondy JA, Murty USR. Graph Theory. vol. 244 of Graduate Text in Mathematics. Springer-Verlag London; 2008.
44. Glover FW, Kochenberger GA. A Tutorial on Formulating QUBO Models. CoRR. 2018;abs/1811.11538.
45. D-Wave's Ocean Software; 2019. Available from: https://ocean.dwavesys.com/ [cited 2019 July 11].
46. D-Wave. Solving a Problem on the QPU; 2019. https://docs.dwavesys.com/docs/latest/c_handbook_6.html#overcoming-imprecisions-of-qubit-biases-and-coupling-strengths.
47. D-Wave. D-Wave Makes New Lower-Noise Quantum Processor Available in Leap; 2019. https://www.dwavesys.com/press-releases/d-wave-makes-new-lower-noise-quantum-processor-available-leap.
48. D-Wave. Los Alamos National Laboratory Upgrades to D-Wave 2000Q™ Quantum Computer; 2019. https://www.dwavesys.com/press-releases/los-alamos-national-laboratory-upgrades-d-wave-2000q%E2%84%A2-quantum-computer.
49. Pudenz KL, Albash T, Lidar DA. Quantum annealing correction for random Ising problems. Physical Review A. 2015; 91(4):042302. https://doi.org/10.1103/PhysRevA.91.042302

50. King J, Mohseni M, Bernoudy W, Fréchette A, Sadeghi H, Isakov S, et al. Quantum-Assisted Genetic Algorithm. arXiv preprint arXiv:190700707. 2019;.
51. Peruzzo A, McClean J, Shadbolt P, Yung M, Zhou X, Love P, et al. A variational eigenvalue solver on a photonic quantum processor. Nature Communications. 2014; 5 (4213). https://doi.org/10.1038/ncomms5213 PMID: 25055053
52. Farhi E, Goldstone J. A Quantum Approximate Optimization Algorithm. arXiv preprint arXiv:14114028. 2014;.
53. Qiskit; 2019. Available from: https://qiskit.org/ [cited 2019 July 11].
54. IBM Q systems; 2019. Available from: https://www.research.ibm.com/ibm-q/technology/devices/ [cited 2019 July 11].
55. Kelley J. A Preview of Bristlecone, Google's New Quantum Processor; 2018.
56. Numpy; 2019. Available from: https://www.numpy.org/ [cited 2019 July 11].
57. Sympy; 2018. Available from: https://www.sympy.org/en/index.html [cited 2019 July 11].
58. Hagberg AA, Schult DA, Swart PJ. Exploring Network Structure, Dynamics, and Function using NetworkX. In: Proceedings of the 7th Python in Science Conference (SciPy 2008). SciPy 2008. ACM; 2008. p. 11–16.
59. Matplotlib; 2019. Available from: https://matplotlib.org/ [cited 2019 July 11].
60. D-Wave. Breakdown of QPU Access Time; 2019. https://docs.dwavesys.com/docs/latest/c_timing_2.html.
61. D-Wave. Types of Postprocessing; 2019. https://docs.dwavesys.com/docs/latest/c_post-processing_1.html.
62. D-Wave. Sampling Tests and Results; 2019. https://docs.dwavesys.com/docs/latest/c_post-processing_4.html#sampling-tests-and-results.
63. Denchev V, Boixo S, Isakov S, Ding N, Babbush R, Smelyanskiy V, et al. What is the Computational Value of Finite Range Tunneling? arXiv preprint arXiv:151202206. 2016;.
64. Nakanishi K, Mitarai K, Fujii K. Subspace-search variational quantum eigensolver for excited states. arXiv preprint arXiv:181009434. 2019;.
65. Boothby K, Bunyk P, Raymond J, Roy A. Next-Generation Topology of D-Wave Quantum Processors. D-Wave Technical Report Series. 2019;(14-1026A-C).