scientific reports

OPEN

Electronic structure with direct diagonalization on a D-wave quantum annealer

Alexander Teplukhin[1], Brian K. Kendrick[1]✉, Sergei Tretiak[1] & Pavel A. Dub[2]✉

Quantum chemistry is regarded to be one of the first disciplines that will be revolutionized by quantum computing. Although universal quantum computers of practical scale may be years away, various approaches are currently being pursued to solve quantum chemistry problems on near-term gate-based quantum computers and quantum annealers by developing the appropriate algorithm and software base. This work implements the general Quantum Annealer Eigensolver (QAE) algorithm to solve the molecular electronic Hamiltonian eigenvalue-eigenvector problem on a D-Wave 2000Q quantum annealer. The approach is based on the matrix formulation, efficiently uses qubit resources based on a power-of-two encoding scheme and is hardware-dominant relying on only one classically optimized parameter. We demonstrate the use of D-Wave hardware for obtaining ground and excited electronic states across a variety of small molecular systems. The approach can be adapted for use by a vast majority of electronic structure methods currently implemented in conventional quantum-chemical packages. The results of this work will encourage further development of software such as *qbsolv* which has promising applications in emerging quantum information processing hardware and has expectation to address large and complex optimization problems intractable for classical computers.

The practical difficulties in simulating many-body quantum systems such as molecules on classical computers based on central or graphics processing units, have been widely recognized in the quantum physics and quantum chemistry communities[1,2]. The computational cost of the numerical solution of the time-independent electronic Schrödinger equation in a given chemical basis set or grid scales factorially with the number of electrons. Since the invention of digital computers in the early 1940s, the exact solution of this central quantum mechanical equation in an appreciable basis set remains unfeasible for molecules having roughly more than 20 electrons[3]. By manipulating quantum states of matter and taking advantage of their unique features, such as superposition, entanglement and quantum tunneling, quantum computers promise to revolutionize quantum simulations of molecules and solids by bringing down the intractable cost to polynomial scaling[1,2]. This can be achieved by using two mathematically equivalent forms of quantum computation[4]: gate-based quantum computing and adiabatic quantum computing, which currently have prototype hardware platforms. Although gate-based quantum computers are currently available at the scale of 50–70 noisy qubits, true adiabatic quantum computers are yet to be technologically available. Existing quantum annealers (e.g. D-Wave 2000Q and recently launched D-Wave Advantage), while naturally suited to perform adiabatic quantum computations, do not currently implement the so-called non-stoquastic Hamiltonian technology[5–8] to resolve the scaling issue.

Unsurprisingly, most of the effort to solve quantum chemistry problems on the near-term quantum architectures has been dedicated to gate-based quantum computers. Examples include the Variational Quantum Eigensolver (VQE)[9–15] and the Full Quantum Eigensolver (FQE)[16] applied to small "toy" molecules (e.g. H_2, LiH, H_2O, NH_3, etc.). Much less effort has been put toward to the use of quantum annealers. To the best of our knowledge, there are only two methods where a modern D-Wave quantum annealer was used to solve electronic structure problems to find the ground state energy[17–19]. These two methods could be potentially extended to the future non-stoquastic Hamiltonians. Alternatively, specifically designed techniques[20] might be used instead.

The main problem in designing a method to solve the electronic structure on today's quantum annealers is to find a mapping between the electronic Hamiltonian and classical Ising model[21,22], i.e. a model of interacting spins, familiar to many physicists. The Ising model is a problem type that is accepted by today's D-Wave quantum annealers. Alternatively, one can search for a mapping to an equivalent formulation, namely Quadratic

[1]Theoretical Division (T-1, MS B221), Los Alamos National Laboratory, Los Alamos, NM 87545, USA. [2]Chemistry Division (C-IIAC, MS K558), Los Alamos National Laboratory, Los Alamos, NM 87545, USA. ✉email: bkendric@lanl.gov; pdub@lanl.gov

Unconstrained Binary Optimization (QUBO), which is the same as the classical Ising formulation, except that the spins are replaced with binary variables. Finding a mapping of the electronic Hamiltonian to either of these two problem types is a very challenging task.

The two recently reported mappings[17,19] were applied to small molecules, such as H_2 and LiH, and constitute the current state-of-the-art in solving the electronic structure on modern D-Wave quantum annealers. Both methods start with an electronic Hamiltonian in the second-quantized form and convert it to the qubit Hamiltonian using either Jordan–Wigner (JW)[23] or Bravyi–Kitaev (BK)[24] transformations. Then, they require either extensive copying of qubits[17] or optimizing a large number of variables classically[19]. Moreover, both approaches have to add extra constraints (with corresponding classical strengths) and sacrifice precious qubits to lower the rank of high-order terms in their optimization problems, making those problems QUBO treatable (i.e. D-Wave compatible). Importantly, both methods were shown to work on the D-Wave 2000Q system[18,19] and their relative effectiveness was studied in detail recently[25].

In our previous study[26] we reported the Quantum Annealer Eigensolver (QAE), to solve the vibrational quantum problem for diatomic and triatomic molecules (nuclear Schrödinger equation). Here we generalize this method to the ubiquitous electronic quantum problem and calculate the ground and excited electronic states of a number of small molecules as a function of the nuclear positions. In contrast to the methods discussed above, the QAE is Hamiltonian and basis agnostic, uses an efficient wave function encoding scheme (powers of two instead of the extensive copying of qubits[17]) and is quantum hardware dominant (one classical variable instead of many[19]). Also, the QAE does not have any conversions to pairwise form, because all terms are quadratic by construction. However, the QAE scales exponentially with the number of electrons or spin-orbitals, similar to the two previous approaches. The QAE algorithm is also redesigned in this work, reducing the computation time and increasing accuracy and usability. The D-Wave 2000Q quantum annealer is used in all representative calculations.

Approach

In this section, we present a new method to solve the electronic structure using the D-Wave quantum annealer. The method has two steps, see Fig. 1a. In the first step, one constructs an electronic Hamiltonian matrix in the basis of Slater Determinants (SDs), which in turn, are constructed from one-electron self-consistent Hartree-Fock molecular orbitals optimized in a finite chemical basis set. In the present study, we generate Full Configuration Interaction (FCI) and Complete Active Space Self-Consistent Field (CASSCF) matrices[27] using an in-house modified Psi4 code[28] to account for full or a selected active space of molecular orbitals. In the second step, the QAE is used to solve the matrix for a few eigenvalues and eigenvectors. The resulting eigenvalues are the electronic energies, whereas the eigenvectors are the electronic wave functions. Since the QAE is a general-purpose method, the matrix can be constructed using any operator, basis and software.

Internally, the QAE maps the eigenvalue problem to the QUBO problem, solvable by D-Wave annealers. The mapping is based on minimization of the Rayleigh-Ritz quotient (RRQ) $R_A = (v, Av)/(v, v)$, where A is an input matrix and v is a vector. The minimum of the RRQ is the smallest eigenvalue, whereas the optimal vector v is the corresponding eigenvector of the matrix A. Using an efficient power-of-two scheme, the vector v is encoded in binary variables $x_i \in \{0, 1\}$ to convert the scalar product (v, Av) to the QUBO expression. The norm $\|v\| = 1$ is enforced by adding a constraint with a classically-optimized Lagrange multiplier λ to the QUBO. The final form of the optimized objective function is $F(v) = (v, Av) + \lambda \cdot (v, v)$. The RRQ minimization is also known as the variational method in quantum chemistry[27].

Figure 1b shows the overall QAE workflow. The input matrix A and a λ guess define the effective objective function $F(v)$. The function is then converted to the QUBO expression, which is minimized on the D-Wave quantum annealer, see Methods. Once an optimal binary string x is obtained from the hardware, the vector v is constructed using the power-of-two scheme, normalized and then used to evaluate (v, Av). If none of the stopping conditions are met, then a new λ guess is generated and the iterations continue. While iterating, all v values are stored, and when the algorithm stops, the smallest (v, Av) value together with the associated v are returned to the user as a sought eigenpair.

The excited states are computed in a similar fashion, except that a number of eigenvalue shifts are applied to the input matrix A to move the previously computed eigenvalues higher in the eigenspectrum, based on Brauer's theorem[29]. Thus, to compute n eigenpairs, the QAE performs n serial runs, each time with an appropriately modified matrix A.

In the original QAE implementation[26], the normalization penalty λ was scanned within a user-specified range with a user-specified step size. In the present study, the λ-search was completely automated, see Methods. This made the calculations faster, more accurate and easier to run, as only a matrix needs to be provided on input.

The dimensionality of the problem does not explicitly manifest anywhere in the QAE formalism, because it is part of the input matrix. If the matrix represents an operator in a direct product basis set, then the matrix dimension, the number of qubits and the QUBO size grow exponentially with the number of dimensions. We refer the reader to the original QAE paper for more details[26]. In that paper, QAE was used to diagonalize a matrix for the molecular vibrational operator in a direct-product Fourier basis set.

As a final note in this section, a typical run of QAE requires (1) a large number of QUBO variables or qubits and (2) all-to-all connectivity between them, so that every product of two QUBO variables must be supported by the QUBO solver. These two requirements are satisfied for the classical QUBO solvers. However, they present a problem for the modern quantum annealers. The annealers have a small number of fully-connected qubits, emulated on top of a larger number of loosely-connected physical qubits. To overcome this limitation of the modern hardware, the QAE is using a QUBO software called qbsolv[30], which divides a large QUBO problem into smaller chunks (subQUBOs) and minimizes them individually. SubQUBOs are solved either classically using the Tabu search or the D-Wave quantum annealer. The Tabu search is an efficient local search technique that

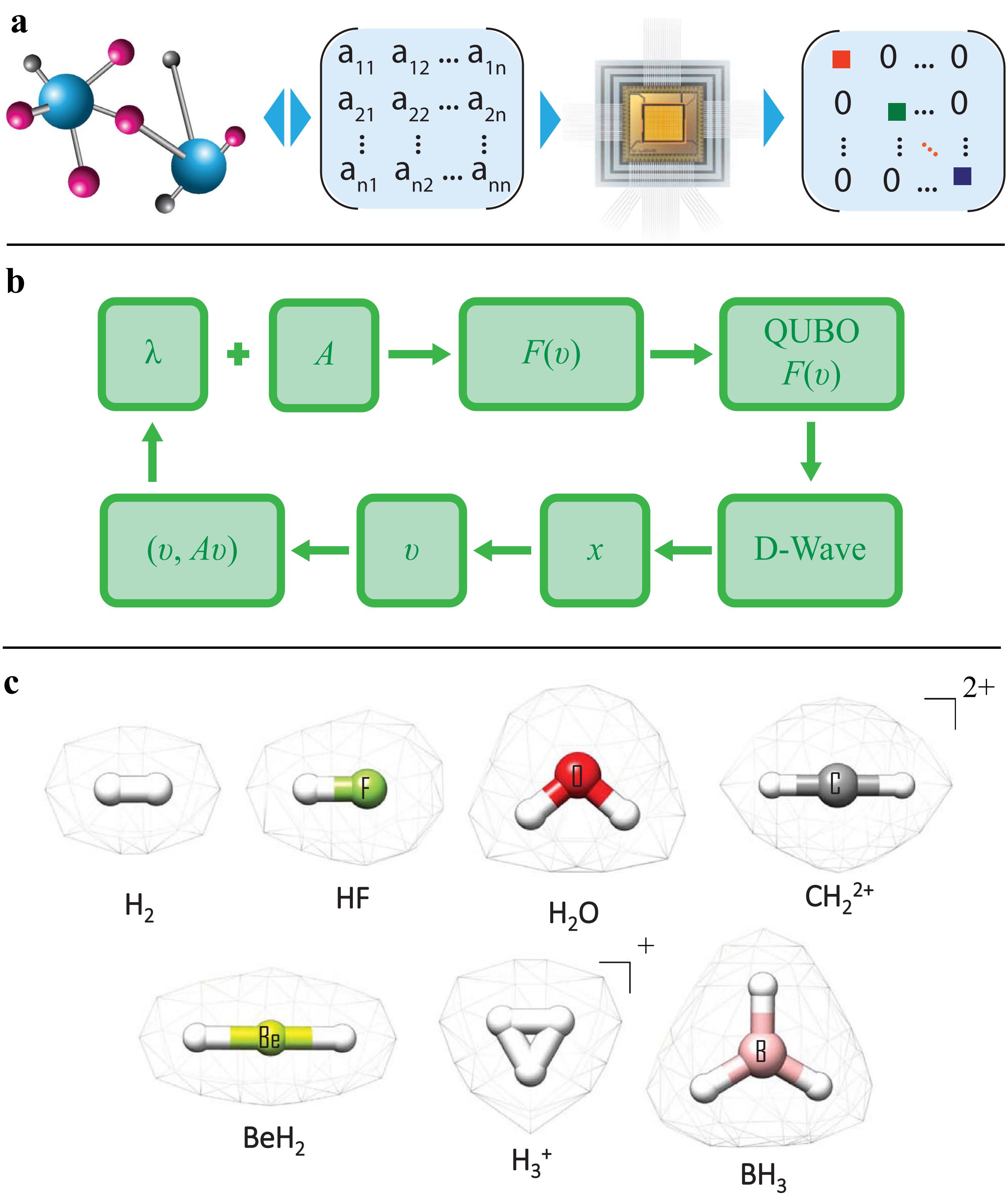

Figure 1. (**a**) Solving the electronic structure problem on a quantum annealer. Starting from Cartesian coordinates of atoms, atomic numbers and a basis set, an electronic Hamiltonian matrix is constructed in the basis of Slater Determinants of spin-orbitals by using classical quantum chemistry codes. The matrix is further diagonalized on a D-Wave 2000Q system using the QAE algorithm. (**b**) Diagram of the QAE iterative workflow. The objective function $F(v)$, defined by the normalization penalty λ and the input matrix A, is converted to QUBO and minimized on a D-Wave quantum annealer (or, alternatively, can be minimized classically using Tabu search). The optimal string of binary variables x is then used to construct the vector v, evaluate the eigenvalue as (v, Av) and guide the next choice of λ. (**c**) Molecular species used in the present work to demonstrate the QAE based approach.

discourages the search from coming back to previously-visited solutions[31]. In this way, qbsolv and, as a consequence the QAE, can be used in two modes: classical or hardware. While processing the multiple subQUBOs, qbsolv also does extra refinement steps to improve the quality of a QUBO solution.

Results

Any electronic structure problem, formulated as an eigenvalue problem, can be solved on a quantum annealer using the QAE. For example, the matrices, generated using the FCI method in the basis of Slater determinants, can be readily solved by the QAE. This will give the ground state energy and wave function, the excited states energies and wave functions and even potential energy surfaces, if the QAE is applied to multiple molecular geometries. We discuss examples of all these QAE applications for the representative molecular species shown in Fig. 1c.

In what follows we will be comparing the energy errors to the chemical accuracy, generally considered to be less than 1 kcal/mol. This level of accuracy of quantum chemistry methods has been christened twenty years ago and advocated by John Pople in his Nobel lecture[32]. The main motivation behind this target is that data has to be reproduced and predicted to within experimental accuracy, being ~1 kcal/mol for energies such as heats of formation or ionization potentials. Another motivation is that at room temperature the energy change of 1 kcal/mol corresponds to an order of magnitude change in a rate constant, making this a convenient standard in kinetics studies as well.

Ground state calculations. The list of molecules chosen to perform proof-of-concept simulations is shown in Fig. 1c. All molecular geometries were first pre-optimized at the restricted Hartree-Fock level (gas phase) by using one of the tabulated chemical basis sets as specified below.

For the simplest H_2 molecule we generated multiple matrices using FCI and different basis sets. The basis set convergence is shown in Fig. 2a, while the full table with absolute electronic energies is given in the Supplementary Information (SI), Table S1. In total, we used 14 basis sets, ranging from the smallest STO-3G basis set to the largest aug-cc-PVQZ basis set. The respective matrix sizes for H_2 range from 2×2 to 1256×1256. The number of qubits, or QUBO size, depends on the desired level of accuracy. We found that using 10 qubits per eigenvector element is sufficient, as will be discussed later. With this choice, all QUBO sizes were, therefore, 10 times larger than the matrix sizes, as shown in the third column of Table S1. To benchmark the QAE, we diagonalized all matrices using a standard linear algebra routine (reference diagonalization via SciPy) and collected the lowest energy E_{ref}. As expected, larger basis sets give lower ground state energy, except for aug-cc-PVDZ, which causes a kink in the Fig. 2a. Thus for H_2, this basis, while being larger than all others before it, is not representative. Once we obtained the true energies from the classical SciPy calculations, we applied the QAE in both modes of operation, classical and hardware, which resulted in two energies, E_{cl} and E_{hw}, respectively. In practice, we find that the ground state energies obtained using both modes are not very different, with the maximum difference being about 1 kcal/mol. The largest discrepancy between the results of the two modes (green and blue curves) is observed for the 6–311++G2d,2p basis set. However, this discrepancy is fortuitous, since the curves may change slightly from run to run. This non-deterministic behavior is found to be caused by the qbsolv software which solves large QUBOs only approximately (by solving small subQUBOs classically or on an annealer). A scrutinized test of the qbsolv performance is given in the Discussion section. Comparison with the reference diagonalization energies E_{ref}, shows that the QAE error may reach 4.8 kcal/mol for large matrices, however, it is less than 1 kcal/mol for the small matrices, namely those that have a size less than 60 (i.e., the first 6 matrices in Table S1). The $E_{hw} - E_{ref}$ error does not exceed 0.01 kcal/mol for the first 3 smallest matrices with sizes under 10. The aug-cc-PVDZ kink is properly reproduced by the QAE. We also found that the hardware mode fails for the problems approaching 10^4 QUBO variables, which is probably due to the lack of memory. Because of that we could not obtain the E_{hw} energy for the aug-cc-PVQZ basis set. In contrast, this particular case can be addressed with the classical mode. In addition to the problem size limit, the hardware mode runs much slower and seems to have a higher chance to fail than the classical mode (probably due to internal hardware interface library issues). To cope with the fragile and slow hardware mode, we added a checkpoint mechanism to the QAE which allows it to restart a calculation if it has stopped.

The electronic ground state energy errors for all molecules, studied in the present work, are collected in Table 1. This time we address a broader chemical space, rather than basis set expansion. For this reason, most molecules in the table were computed using the FCI method in the minimal STO-3G basis set, except a few exceptions as specified in the table. The number of qubits per eigenvector element remains 10. Similar to H_2, the difference between the modes E_{cl} and E_{hw} is less than 1 kcal/mol, except H_2O, for which it reaches 1.7 kcal/mol (likely due to the qbsolv as will be discussed later). The deviation from the reference energy ranges from 0.0 to 7.3 kcal/mol. Again, it was not feasible to solve a large 1256×1256 FCI problem for BH_3 in the hardware mode (the calculation requires a large amount of memory and more than 4 days to finish). As we have found, the hardware implementation of the QAE is limited to a matrix size of about 600, and matrices of a larger size can only be solved in the classical mode. The absolute energies are given in Table S2, whereas the choice of active space for H_2O and BH_3 is shown in Figs. S1 and S2.

In summary, the chemical accuracy of 1 kcal/mol is reached for small molecules and small basis sets, which demonstrates the applicability of our QAE-based approach for the ground state calculations. However, the errors increase for larger molecules and basis sets, up to 7 kcal/mol in the present study, suggesting that the overall approach still needs to be improved. The qbsolv software likely introduces the largest errors as will be discussed in detail in the Discussion section.

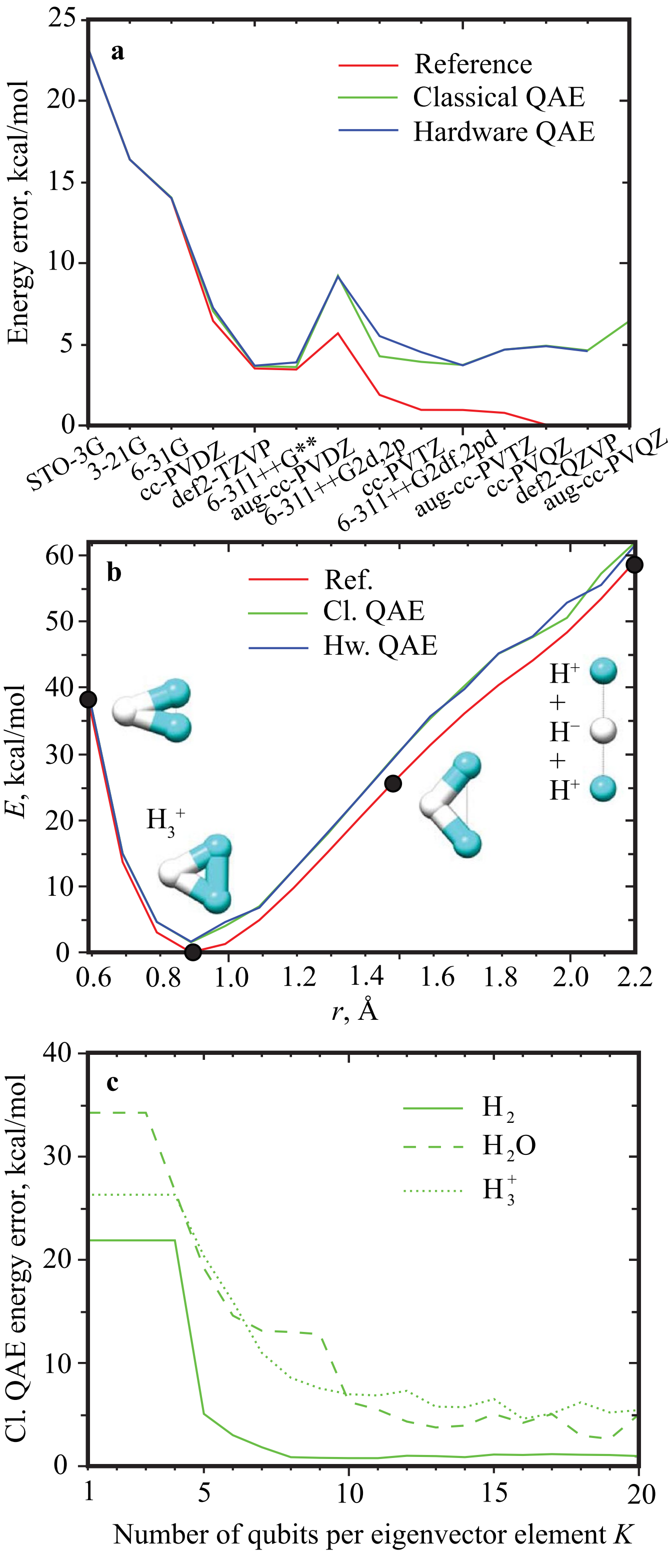

Figure 2. (**a**) Convergence of the H_2 ground state energy with basis set. The results are ordered with increasing Hamiltonian matrix size. The matrices are diagonalized directly (red) and using the QAE in both classical (green) and hardware (blue) modes. The error is given relative to the energy of def2-QZVP reference diagonalization, which is the lowest among all calculations. The QAE error closely follows the reference diagonalization error for small basis sets, but exceeds it by about 5 kcal/mol for larger basis sets. (**b**) Potential energy curve of H_3^+ computed at FCI/cc-PVDZ level using the same methods as in the top panel (same colors). Molecular images (taken at black points) show how the minimum geometry evolves as a function of the distance between two terminal hydrogens. (**c**) Convergence of the ground state energy with respect to the number of qubits per eigenvector element *K*. The convergence study is done for H_2 (solid), H_2O (dashed) and H_3^+ (dotted). The QAE is running in the classical mode. Ten-qubit discretization is sufficient for all three molecules.

Molecule	Method	Basis	Mat. size	QUBO size	$E_{cl} - E_{hw}$ [a]	$E_{hw} - E_{ref}$ [b]
H_2	FCI	STO-3G	2×2	20	0.000	0.000
HF	FCI	STO-3G	18×18	180	− 0.063	0.214
H_2O	FCI	STO-3G	133×133	1330	0.085	5.399
H_2O	CAS(8e,7o)SCF	cc-PVDZ	321×321	3210	1.660	7.252
CH_2^{2+}	FCI	STO-3G	169×169	1690	0.293	4.972
BeH_2	FCI	STO-3G	169×169	1690	− 0.301	2.380
H_3^+	FCI	cc-PVTZ	532×532	5320	− 0.071	7.005
BH_3	CAS(6e,6o)SCF	6-311++G**	208×208	2080	− 0.473	7.145
BH_3	FCI	STO-3G	1250×1250	12500[c]	–	–

Table 1. Electronic ground state energy errors in kcal/mol. [a]Energy difference between the classical (E_{cl}) and hardware (E_{hw}) QAE modes. [b]Energy difference between the QAE in the hardware mode (E_{hw}) and the reference diagonalization (E_{ref}). [c]The QAE failed in the hardware mode (see text).

Transition	T_{ref} [a]	T_{cl} [b]	T_{hw} [c]	$T_{cl} - T_{hw}$	$T_{hw} - T_{ref}$
$S_0 \rightarrow S_1$	303.056	300.563	302.125	− 1.563	− 0.930
$S_0 \rightarrow S_2$	369.233	373.585	379.837	− 6.252	10.603
$S_0 \rightarrow S_3$	441.058	437.217	444.802	− 7.585	3.744
$S_0 \rightarrow S_4$	590.407	606.617	612.352	− 5.735	21.945

Table 2. Electronic transition energies of the H_2O molecule computed using the QAE and FCI/STO-3G matrix in kcal/mol. [a]Transition energy obtained using the reference diagonalization. [b]Transition energy obtained using the QAE in the classical mode. [c]Transition energy obtained using the QAE in the hardware mode.

Excited state calculations.

We also calculate the lowest five singlet state energies (including the ground state) for the water molecule, labeled as S_0, $S_1, \ldots$ and report four ground-to-excited state transition energies in Table 2. In this set of calculations, the FCI/STO-3G matrix (133×133) was solved for eigenvalues using the reference diagonalization and the QAE in both classical and hardware modes. For the excited states, the QAE modifies the matrix after each energy is computed according to the Brauer's theorem, as described in "Approach". Since a QAE eigenvector is not calculated exactly (due to limitations of qbsolv and the finite number of qubits), each spectral transformation that is based on a computed eigenvector outer product, adds a random noise to the modified matrix, causing the higher states to be calculated with less precision. This trend is seen in the last column of the table, $T_{hw} - T_{ref}$. The smaller error for the $S_0 \rightarrow S_3$ transition (when compared to the S_2 and S_4 transitions) seems to be fortuitous, as rerunning the QAE gives somewhat different energies each time, due to the limitations of the qbsolv (see "Discussion and Methods"). Similarly, the energy difference between the two QAE modes for the $S_0 \rightarrow S_1$ transition is larger than the $T_{hw} - T_{ref}$ error. These inaccuracies indicate that more runs and a less noisy implementation of the entire approach are needed to obtain reliable and representative statistics.

Notably, the QAE calculates not only the excited state energies but also the corresponding wave functions represented by the respective eigenvectors (not shown). When compared to the results of the reference diagonalization T_{ref}, the QAE transition energy errors $T_{hw} - T_{ref}$ are about 3%. The absolute energies of all states can be found in Table S3.

Potential energy curves.

To test the QAE capacity to generate potential energy surfaces we calculate the potential energy curve of the H_3^+ system, as shown in Fig. 2b. The FCI/cc-PVDZ matrices were generated for 17 distinct geometries, each pre-optimized at the restricted Hartree-Fock level, where the distance between the two terminal hydrogens was varied from 0.59 to 2.19 Å following a step size of 0.1 Å (i.e. constrained optimization). The errors behave similarly to those reported in Table 1. There is not much difference between the two QAE modes. The difference with the reference energy varies from 1.1 kcal/mol at 0.59 Å to 4.5 kcal/mol at 1.99 Å. All of the matrices were 153×153, except for the last two geometries near 2.19 Å which were 51×51 due to the increasing symmetry of the system.

As with the ground and excited states calculations reported above, the QAE is shown to work for potential energy surfaces as well. However, the method results in a larger 5 kcal/mol error along the H_3^+ dissociation curve compared to the reference simulations.

Number of qubits.

The number of qubits per eigenvector element K is an important and the only QAE convergence parameter. This parameter controls how accurately an eigenvector is represented, when the eigenvalue problem is mapped to the QUBO problem. For example, $K = 10$ translates to $2^{-10} \approx 10^{-3}$ error in the eigenvector elements, whereas $K = 20$ to 10^{-6} error. In order to quantify its influence, we performed a number of calculations with different K.

The ground state energy convergence with respect to K is shown in Fig. 2c for three molecules, H_2, H_2O and H_3^+. The three matrices for these cases were generated using the FCI and cc-PVDZ, STO-3G, and cc-PVTZ basis sets, respectfully. With this choice, the matrices of varying sizes are covered: 22×22 (small), 133×133 (medium) and 532×532 (large). As one can see, increasing the number of qubits improves the resolution of the eigenvector representation and lowers the ground state energy. At the same time, a plateau is reached after $K = 10$ qubits and using more qubits is not necessary. Based on our tests, the noisy plateau is due to the limitations of qbsolv, see Discussion section. The ground state energy has converged to 1.0 kcal/mol for H_2, 4.2 kcal/mol for H_2O and 5.9 kcal/mol for H_3^+ (averaged over K from 11 to 20). The plot was generated using the QAE running in the classical mode, because it is much faster than the hardware mode (also the energy difference between the two modes is smaller than the errors from qbsolv). The H_2O matrix is the same matrix that was used to compute the ground and excited states of water, whereas the H_3^+ matrix corresponds to the minimum geometry matrix in Fig. 2b. Ideally, this convergence study should be done for every specific case, however, verifying convergence for the three typical matrix sizes should suffice.

Discussion and overview

Two overarching statements can be derived from the results reported in the previous section. The first is that the QAE-based approach to the electronic structure problems is viable. The second is that the energy errors are too large (more than 1 kcal/mol), which places the QAE not even close to the capacity of the modern classical quantum chemistry software. The latter motivated us to examine the entire procedure, aiming at identifying a component that is responsible for the inaccurate energies. Ultimately, we found that the qbsolv software[30], while being a great tool to manage large QUBO problems, is the weak spot. This QUBO solver does not solve QUBO problems exactly. Some amount of classical "noise" is always present in the optimal QUBO solutions returned by the qbsolv, inadvertently "polluting" the QAE iterations and causing inaccuracy in the resulting electronic energies. In this section, we first gather statistics on how large a typical energy spread is, then we demonstrate the fluctuating behavior of the qbsolv and discuss possible ways to resolve this issue. Additionally, as we stated earlier, running the QAE on an actual hardware is significantly slower than running it in the classical mode. We outline possible solutions to this performance issue as well.

The energy errors reported in the present study are small and large for small and large matrices, respectfully. To quantify the degree of energy fluctuation, we perform 10 QAE runs for two molecules H_2 and H_2O. For a small 8×8 H_2 FCI/6-31G matrix (80 QUBO variables) the average energy error was 0.004 kcal/mol, when compared to the reference energy E_{ref}, while the spread (σ the standard deviation) was 0.005 kcal/mol. For a larger case, a 133×133 H_2O FCI/STO-3G matrix (1330 QUBO variables) the average error was 6.79 kcal/mol and the standard deviation was 0.47 kcal/mol.

As stated above, we found that qbsolv does not give a reproducible QUBO solution. In order to demonstrate that, we fixed the normalization penalty at $\lambda = -100$ and called the qbsolv 10 times to minimize the same QUBO. For a simple 3×3 matrix (30 QUBO variables), the qbsolv returned 10 identical solutions. For a 9×9 matrix (90 QUBO variables), only 7 solutions were the same, whereas the other 3 were unique. For the H_2O 133×133 matrix all 10 QUBO solutions were different with the ground state energy spread (standard deviation) of 18.7 kcal/mol (for a fixed $\lambda = -100$, which is not too far from $\lambda_{opt} = -83.9$). The fluctuating behavior of qbsolv makes it hard to obtain a reliable ground state energy and subsequently to design a robust technique for finding the optimal normalization penalty λ. On the other hand, exploring the space of 2^{1330} possible configurations, or even of $2^{5320} \approx 3 \times 10^{1601}$ (H_3^+ FCI/cc-PVTZ, the largest QUBO solved using a D-Wave annealer in the present work) is a very challenging task and using a non-exact metaheuristic solver (qbsolv) is the only feasible way to address this task. More details about the qbsolv and its performance in the classical mode are discussed in the corresponding technical report[30]. While the report lacks a comparison of the two modes (classical and hardware), it does say that there is no speed advantage, when subQUBOs are solved on a D-Wave system (i.e., the hardware mode). Since the QAE operates on top of the qbsolv, our method inherits all performance differences between the modes from the qbsolv and potentially from other D-Wave Ocean tools.

There are a number of ways to improve the accuracy of the current implementation. First of all, one can simply run the qbsolv multiple times and always choose the solution with the smallest QUBO value. Alternatively, some of qbsolv's default parameters can be adjusted. The current version of qbsolv has only one parameter that the user can specify, namely, the number of repeats (iterations) of the outer loop before the algorithm stops. Our test for a typical QAE calculation (H_2O 133×133 matrix) showed that the default number of repeats 50 is sufficient. We explored several values in the 5 to 200 range (the error is given in parentheses in kcal/mol): 5 (10.6), 10 (8.3), 20 (7.4), 30 (6.9), 40 (5.6), 50 (5.5), 100 (5.1), 150 (5.7) and 200 (5.2). The error monotonically decreases as the number of repeats is increased and stabilizes at about 40 repeats. Above 40 repeats the error fluctuates with no significant improvement. However, there are other hard-coded parameters in the qbsolv code that may need to be tested as well. Both approaches (multiple qbsolv runs or tightening qbsolv internal parameters) may significantly increase the time-to-solution duration. Second, one may consider using an exact brute-force approach to minimize a QUBO (while somehow interleaving this with the calls to an annealer). In this approach all possible solutions (binary strings) to a given QUBO problem will be tested in a systematic manner, so that the "true" answer will be found eventually. An exact QUBO solver is already implemented by D-Wave and "becomes slow for problems with 18 or more variables"[33], which makes it useful for testing and debugging code only. Alternatively, D-Wave started providing a new Hybrid Solver Service (HSS)[34] via their Leap cloud service[35], which is a combination of classical and hardware solvers working together on QUBO problems up to 10,000 variables. The HSS may help decrease the QAE errors, taking into account that D-Wave strongly recommends using their dwave-hybrid framework (which seems to be the main component of the HSS[34]) instead of qbsolv[33]. At the same time, the original qbsolv was recently upgraded and has started a new life under the "QCI qbsolv" name[36], so it is also worth exploring if this newer version will work better with the QAE.

Running the QAE on an actual hardware has its own issues. Currently, the hardware mode is much slower than the classical mode, which might be due to several reasons. One reason is that every time the qbsolv submits a subQUBO, a mapping between subQUBO variables and the actual physical qubits, called minor-embedding, is recalculated. This takes time, since it presents a separate challenging problem. There is an option to use a lazy embedding, where the mapping computed for the very first submission will be used for all other submission, thus saving some time. We are exploring this possibility. Another piece of the intermediate software, that seems to consume time, is the integrated post-processing system of the Virtual Full-Yield Chimera (VFYC) wrapper of the D-Wave annealer. The main idea behind VFYC solvers is to classically "fix" unrepresented variables in the actual hardware, so that the user will always work with the same number of fully-connected qubits (full-yield), thus realizing a "portable" code. In practice, this increases the time-to-solution lag and may decrease the quality of subQUBO solutions. Multiple users working on a single D-Wave device introduce one more problem, when compared to the purely classical way of running the QAE and qbsolv.

While there are the hardware sources of errors in the present approach, those are overwhelmed by the classical noise errors, and cannot be tracked easily. For example, parameters like annealing time might be worth reconsidering once the classical noise is eliminated. Previously[26], we found that an increase in the annealing time does not improve the solution quality significantly, rather it is more beneficial to increase the number of reads on a quantum annealer. Since this is an application-focused study, it is also hard to tell if any genuine quantum features (for example, tunneling within the D-Wave QPU) are engaged and play any significant role in the calculations. The interested reader is encouraged to read more specific literature on benchmarking quantum annealers[37,38] and also the discussion section of our earlier QAE paper[26] to see how the QAE may be improved. In addition, while the adiabatic quantum annealing is a viable quantum computing paradigm, it clearly has its own well-defined set of challenges[8].

In summary, a novel methodology was presented for solving electronic structure problems on a quantum annealer by numerically exact diagonalization. The diagonalization was performed using the Quantum Annealer Eigensolver (QAE). Several molecular systems were studied spanning a wide range of complexity from diatomic to tetraatomic systems. The electronic ground and excited state energies (and their wave functions) were computed on a D-Wave quantum annealer. Diagonalizing the electronic structure matrix at multiple molecular geometries was also demonstrated and can be used to generate potential energy surfaces. As an example, we calculated the potential energy curve of the H_3^+ cation. Importantly, the QAE computed potential energy is a smooth function of the nuclear coordinates. The QAE gives accurate molecular energies for small molecules and matrices (energy errors are less than 1 kcal/mol), however, the errors tend to increase and fluctuate for larger matrices. We found that this is primarily due to the noisy output of the underlying QUBO solver called qbsolv, the main goal of which is to reduce a large QUBO problem to a number of small subQUBO tasks (that fit on the D-Wave quantum annealer). Although the fundamentals of the QAE did not change since we applied it to the vibrational problem[26], in this work we generalized the normalization penalty search, which made the method faster and does not require any a priori knowledge about an eigenvalue problem from the user. In contrast to the other works on this subject, our method is general—the only requirement is to formulate the problem as an eigenvalue problem. Also, the power-of-two scheme allows for the efficient use of precious qubits, and having a single classically-optimized parameter λ makes our method hardware dominant.

The electronic structure methods used in the current work are only a small representative subset of methods that can be used to generate the input matrices for the QAE. The solution of the electronic structure problem on a classical computer is ultimately mapped into a diagonalization of a Hermitian Hamiltonian matrix for a variety of approaches of different accuracy ranging from Hartree-Fock (HF) to Density-Functional Theory (DFT) to Coupled Cluster (CC) and to Full Configuration Interaction (FCI) techniques. Due to the eigenproblem representation, solving all these tasks on the upcoming quantum computing hardware is an important and natural challenge for the QAE and similar algorithms. These are just a few examples of possible typical applications of the QAE in the field of quantum chemistry. We also expect that future technological improvements to the quantum annealers will continue to increase the number of qubits and reduce the noise. The results of this work encourage further improvements to the qbsolv software, which is commonly used in the quantum annealing community to address large optimization problems. All these improvements (both hardware and software) will enable practical solutions of larger problems with increased accuracy and may ultimately challenge the capabilities of conventional computing.

Methods

Generation of matrices.

All molecular geometries were first optimized at the restricted Hartree–Fock level (gas) with the Gaussian 09 (E01) code[39], ensuring the highest point group symmetries. For H_3^+ in the cc-PVTZ basis, the optimized geometry has C_{2v} point group symmetry. The restricted FCI and CASSCF matrices were then generated using an in-house modified Psi4 code[28] with the optimized Cartesian coordinates of atoms as input data. The matrices were generated imposing the unitary groups U(1) and SU(2) (spin and particle conservation) and point group symmetries and contain nonzero matrix elements only. For Abelian point groups (e.g. C_{2v}), Psi4 automatically detects the molecular symmetry. If the point group is non-Abelian (e.g. $D_{\infty h}$, $C_{\infty v}$, D_{3h}), the detected symmetry corresponds to one of its subgroups. An energy threshold of 10^{-8} Hartree for diagonal and of 10^{-10} Hartree for off-diagonal elements was used. The nuclear repulsion (NN) term was manually added to all constructed matrices.

Quantum Annealer Eigensolver. The Quantum Annealer Eigensolver (QAE) is a general-purpose variational method to compute eigenvalues and eigenvectors of a given matrix. It is based on the min–max (or variational) theorem, that states that the smallest (largest) eigenvalue of a $n \times n$ Hermitian matrix A is the minimum (maximum) of the Rayleigh–Ritz quotient $R_A = (v, Av)/(v, v)$, where v is a vector. If the vector v is normalized, i.e. its norm $(v, v) = \|v\|$ is 1, then $R_A = (v, Av)$. The vector v that realizes the minimum (maximum), is the eigenvector associated with that eigenvalue. For the QAE, we are interested in the minimum only. Notably, $R_A(v)$ is functionally similar to the polynomials used in the Quadratic Unconstrained Binary Optimization (QUBO) problems solved by D-Wave quantum annealers. A QUBO problem is the problem of minimizing a quadratic polynomial $x^T Qx$ over a vector of binary variables $x = (x_1, x_2, \ldots, x_m)$ for a given $m \times m$ matrix of real weights Q. Although looking similar, $R_A(v) = \sum_{\alpha,\beta}^{n,n} v_\alpha A_{ij} v_\beta$ and $x^T Qx = \sum_{i,j}^{m,m} x_i Q_{ij} x_j$ operate on different kinds of variables: the elements of the normalized vector v are continuous real numbers of maximum magnitude of 1, $v_\alpha \in [-1; 1]$, whereas the elements of x are discrete, $x_i \in \{0, 1\}$. The QAE provides a mapping between v and x using a powers-of-two encoding scheme. In this scheme each element of v is encoded using K binary variables or qubits $q_i^\alpha, 1 \leq i \leq K$. Each qubit is multiplied by the weight of an appropriate power of two, and one more qubit is responsible for the sign of v_α. This encoding scheme is known as the fixed-point number representation and can represent fractional numbers between -1 and $+1$ up to 2^{-K} accuracy. The sum $x^T Qx$ includes both $Q_{i,j}$ and $Q_{j,i}$ terms which in general can be non-symmetric. However, QUBO problems are by definition strictly symmetric and therefore only the $i \leq j$ terms are needed. This requires a symmetric Q matrix and all pairwise terms with $i < j$ are doubled in constructing the QUBO functional.

Although the mapping describe above looks natural and promising, it lacks one important component, the vector normalization. Indeed, a trivial solution to the QUBO function constructed as described, is the null vector $\mathbf{0}$ (i.e., the unique vector having zero length). To fix that, the QUBO must be augmented with a normalization constraint that ensures $\|v\|$ is unity or close to unity. As with any constraint in combinatorial optimization, the normalization condition comes with its own penalty or Lagrange multiplier λ. Since the norm can be larger or smaller than 1, a natural choice for the constraint is the quadratic form $\lambda(\|v\| - 1)^2$. However, this form is biquadratic in v or x, which is not a valid QUBO form. The workaround is to decrease the degree of the constraint to the first power, $\lambda(\|v\| - 1)$, which becomes $\lambda\|v\|$ after we drop the constant shift λ. Thus, the final QUBO function is $F(v) = (v, Av) + \lambda \cdot (v, v)$ or $F(x) = x^T Qx + \lambda \cdot (x, x)$.

The optimal normalization penalty λ_{opt} can be found in many ways. In our previous study[26], we implemented a simple scanning in λ, where for each λ a non-trivial solution v_λ with the smallest $R(v)$ value was considered as the best solution. The best of the best v_λ solutions was the final answer. However, this method of λ-searching is inconvenient as it requires a good guess for the λ search range and also depends on a step size $(\Delta\lambda)$, both of which are problem dependent.

In the present study, we made the λ-search completely automatic and iterative. Based on our experience with the scanning technique[26], we noticed that the solution v_λ that minimizes $F(v)$ is a zero vector $\mathbf{0}$ for a large positive λ and is a non-zero vector for a large negative λ. The optimal λ_{opt} value is located somewhere in between, usually around a "phase-transition" point (i.e., on the boundary between trivial and non-trivial solution areas). Therefore, we first construct a wide λ-range, where the best solution at the left end is non-trivial and the solution at the right end is trivial. The matrix element with the largest magnitude serves as a good guess for the range ends (multiplied by -1 for the left end). If a solution of the required type is not found for an end, then the guess is doubled. Once the range in λ is determined, QAE starts iteratively shrinking it using a bisection method, such that the best solutions at the range ends do not change their type. In this way, the range reduces to a small region around the optimal λ_{opt}. Currently, we have two stopping criteria for this iterative scheme: either the change in the expectation value $R(v_\lambda)$, computed classically from optimal x, becomes smaller than a user specified tolerance, or the λ-range shrinks to a single point λ_{opt}. While the first criterion seems to be sufficient, it may never be triggered for inaccurate QUBO solvers (either classical or hardware). The fluctuating output of these solvers leads to a situation where the tolerance can never be reached, causing an infinite loop. Additionally, inaccurate QUBO solvers make some types of searching techniques inapplicable, such as those based on the dependence of $R(v_\lambda)$ on λ. The randomness in the vicinity of λ_{opt} will cause gradient-based methods to fail. In this case, extrapolating techniques, such as Direct Inversion in the Iterative Subspace (DIIS)[40–42] based on previous λ-points, could be of help. The overall workflow is shown in Fig. 1b of the main text.

The eigenpairs with larger eigenvalues are computed by applying a spectrum transformation to the initial matrix and repeating the iterative procedure described above. Specifically, based on Brauer's theorem[29], if v_0 is the eigenvector with the smallest eigenvalue, then one can construct a new matrix $A' = A + \mu_0(v_0 \otimes v_0)$, where $\otimes$ denotes the outer product operation and multiplier μ_0 controls how much higher in the spectrum we want to relocate the computed eigenpair. If it was relocated high enough, then repeating the QAE iterations for A' will give the next eigenpair. Thus, to compute n eigenpairs, the QAE performs n serial runs, each time with an appropriately modified matrix A'. The multipliers μ_i $(i = 0 \ldots n-1)$ could be quite arbitrary, but each has to be larger than all previously computed eigenvalues. Currently, all μ_i shifts are set equal to the largest matrix element multiplied by 16, however, multiplying by 8, 4 or 2 gives essentially the same results, with differences smaller than other errors. Alternatively, the difference between the maximum and minimum elements of A, or any other dynamic range estimate, can be used as a shift. In the present work, the calculation of multiple eigenpairs is not the main focus, as only a few eigenpairs are of interest. However, the QAE can be potentially modified to find "batches" of eigenpairs. The shifting may need a revision or replaced with a smarter technique[43,44]. Linear algebra tricks, such as a correlation of the minimum eigenvector with the matrix rows[45,46], could also be leveraged to aid the QAE.

Software. In the present work, the QAE was implemented in Python 3 and requires D-Wave's Ocean tools to be installed. The access to a D-Wave quantum annealer has to be configured separately. After the QAE has mapped an eigenvalue problem (with a chosen normalization penalty λ) to a QUBO form, the code makes a call to the qbsolv to minimize the QUBO and obtain the best QUBO solution. The qbsolv is included in the Ocean tools, and is capable of running in both classical and hardware modes. For the reference diagonalization, we used SciPy eig() function, which is a wrapper for the geev() family of functions in LAPACK[47].

Quantum annealer. LANL's D-Wave 2000Q was used in all hardware-mode calculations. This machine operates at $T = 0.015K$ and has 2048 qubits and 6,016 couplers. In an actual quantum annealer some qubits and couplers are not active (unrepresented), so the total number of them is a bit smaller. One will need either to find a specific mapping between QUBO variables and physical qubits (minor-embedding) for a given machine or use a VFYC version of a hardware QUBO solver, which postprocess a QUBO solution to fix unrepresented qubits and couplers. Another issue is a connectivity between qubits. In the D-Wave 2000Q, each qubit is connected to 6 neighbors only. To increase the degree of connectivity, the minor-embedding procedure generates a chain of physical qubits to represent one logical qubit. Chains overcome the 6-neighbor limit while sacrificing physical qubits. As a result, the size of the final full graph is only 64 logical qubits. The qbsolv constructs subQUBOs of this particular size.

Code availability

The QAE code is available from the authors upon request.

Received: 16 September 2020; Accepted: 6 November 2020
Published online: 27 November 2020

References

1. Cao, Y. *et al.* Quantum chemistry in the age of quantum computing. *Chem. Rev.* **119**, 10856–10915 (2019).
2. McArdle, S., Endo, S., Aspuru-Guzik, A., Benjamin, S. C. & Yuan, X. Quantum computational chemistry. *Rev. Mod. Phys.* **92**, 015003 (2020).
3. Gan, Z. & Harrison, R. J. Calibrating quantum chemistry: A multi-teraflop, parallel-vector, full-configuration interaction program for the Cray-X1. In *SC'05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing*, 22–22 (IEEE, 2005).
4. Mizel, A., Lidar, D. A. & Mitchell, M. Simple proof of equivalence between adiabatic quantum computation and the circuit model. *Phys. Rev. Lett.* **99**, 070502 (2007).
5. Bravyi, S., Divincenzo, D. P., Oliveira, R. & Terhal, B. M. The complexity of stoquastic local Hamiltonian problems. *Quant. Inf. Comput.* **8**, 361–385 (2008).
6. Vinci, W. & Lidar, D. A. Non-stoquastic Hamiltonians in quantum annealing via geometric phases. *NPJ Quant. Inf.* **3**, 1–6 (2017).
7. Albash, T. Validating a two-qubit nonstoquastic Hamiltonian in quantum annealing. *Phys. Rev. A* **101**, 012310 (2020).
8. Hauke, P., Katzgraber, H. G., Lechner, W., Nishimori, H. & Oliver, W. D. Perspectives of quantum annealing: Methods and implementations. *Rep. Prog. Phys.* **83**, 054401 (2020).
9. Peruzzo, A. *et al.* A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213 (2014).
10. McClean, J. R., Romero, J., Babbush, R. & Aspuru-Guzik, A. The theory of variational hybrid quantum-classical algorithms. *New J. Phys.* **18**, 023023 (2016).
11. O'Malley, P. J. J. *et al.* Scalable quantum simulation of molecular energies. *Phys. Rev. X* **6**, 031007 (2016).
12. Li, Y., Hu, J., Zhang, X.-M., Song, Z. & Yung, M.-H. Variational quantum simulation for quantum chemistry. *Adv. Theor. Simul.* **2**, 1800182 (2019).
13. Quantum, Google AI. Hartree-Fock on a superconducting qubit quantum computer. *Science* **369**, 1084–1089 (2020).
14. McCaskey, A. J. *et al.* Quantum chemistry as a benchmark for near-term quantum computers. *NPJ Quant.Inf.* **5**, 1–8 (2019).
15. Nam, Y. *et al.* Ground-state energy estimation of the water molecule on a trapped-ion quantum computer. *NPJ Quantum Inf.* **6**, 1–6 (2020).
16. Wei, S. *et al.* A full quantum eigensolver for quantum chemistry simulations. *Research* **2020**, 1486935 (2020).
17. Xia, R., Bian, T. & Kais, S. Electronic structure calculations and the Ising Hamiltonian. *J. Phys. Chem. B* **122**, 3384–3395 (2017).
18. Streif, M., Neukart, F. & Leib, M. Solving quantum chemistry problems with a D-Wave quantum annealer. In *International Workshop on Quantum Technology and Optimization Problems*, 111–122 (Springer, 2019).
19. Genin, S. N., Ryabinkin, I. G. & Izmaylov, A. F. Quantum chemistry on quantum annealers. Preprint at arXiv:1901.04715 (2019).
20. Babbush, R., Love, P. J. & Aspuru-Guzik, A. Adiabatic quantum simulation of quantum chemistry. *Sci. Rep.* **4**, 6603 (2014).
21. Ising, E. Beitrag zur theorie des ferromagnetismus. *Z. Phys* **31**, 253–258 (1925).
22. Baxter, R. J. *Exactly solved models in statistical mechanics* (Academic Press, London, 1982).
23. Wigner, E. & Jordan, P. Über das Paulische Äquivalenzverbot. *Z. Phys.* **47**, 631–651 (1928).
24. Bravyi, S. B. & Kitaev, A. Y. Fermionic quantum computation. *Ann. Phys.* **298**, 210–226 (2002).
25. Copenhaver, J., Wasserman, A. & Wehefritz-Kaufmann, B. Using quantum annealers to calculate ground state properties of molecules. Preprint at arXiv:2009.10779 (2020).
26. Teplukhin, A., Kendrick, B. K. & Babikov, D. Calculation of molecular vibrational spectra on a quantum annealer. *J. Chem. Theory Comput.* **15**, 4555–4563 (2019).
27. Szabo, A. & Ostlund, N. S. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory* (Dover Publications, New York, 1996).
28. Parrish, R. M. *et al.* Psi4 1.1: An open-source electronic structure program emphasizing automation, advanced libraries, and interoperability. *J. Chem. Theory Comput.* **13**, 3185–3197 (2017).
29. Brauer, A. Limits for the characteristic roots of a matrix. IV: Applications to stochastic matrices. *Duke Math. J.* **19**, 75–91 (1952).
30. Booth, M., Reinhardt, S. & Roy, A. Partitioning optimization problems for hybrid classical/quantum execution. D-Wave Technical Report Series, 14-1006A-A (2017).
31. Glover, F. Future paths for integer programming and links to artificial intelligence. *Comput. Oper. Res.* **13**, 533–549 (1986).
32. Pople, J. A. Nobel lecture: Quantum chemical models. *Rev. Mod. Phys.* **71**, 1267 (1999).
33. D-Wave Ocean Software Documentation. https://docs.ocean.dwavesys.com/. Accessed: 2020-09-15.
34. D-Wave Hybrid Solver Service: An overview. D-Wave Whitepaper Series, 14-1039A-A (2020).
35. D-Wave Leap. https://www.dwavesys.com/take-leap. Accessed: 2020-09-15.
36. Booth, M. *et al.* QCI Qbsolv delivers strong classical performance for quantum-ready formulation. Preprint at arXiv:2005.11294 (2020).

37. Buffoni, L. & Campisi, M. Thermodynamics of a quantum annealer. *Quant. Sci. Technol.* **5**, 035013 (2020).
38. Gardas, B. & Deffner, S. Quantum fluctuation theorem for error diagnostics in quantum annealers. *Sci. Rep.* **8**, 1–8 (2018).
39. Frisch, M. J. *et al.* Gaussian 09 Revision E.01 (2009). Gaussian Inc. Wallingford CT.
40. Pulay, P. Convergence acceleration of iterative sequences. The case of SCF iteration. *Chem. Phys. Lett.* **73**, 393–398 (1980).
41. Rohwedder, T. & Schneider, R. An analysis for the DIIS acceleration method used in quantum chemistry calculations. *J. Math. Chem.* **49**, 1889 (2011).
42. Garza, A. J. & Scuseria, G. E. Comparison of self-consistent field convergence acceleration techniques. *J. Chem. Phys.* **137**, 054110 (2012).
43. Saad, Y. *Numerical Methods for Large Eigenvalue Problems* (Manchester University Press, Manchester, 1992).
44. Saad, Y. *Iterative Methods for Sparse Linear Systems* (Society for Industrial and Applied Mathematics, USA, 2003), 2nd edn.
45. Pan, W., Wang, J. & Sun, D. A new strategy for directly calculating the minimum eigenvector of matrices without diagonalization. *Sci. Rep.* **10**, 1–8 (2020).
46. Pan, W., Wang, J. & Sun, D. Establishing simple relationship between eigenvector and matrix elements. *Phys. Lett. A* **384**, 126610 (2020).
47. Anderson, E. *et al.LAPACK Users' Guide* 3rd edn. (Society for Industrial and Applied Mathematics, USA, 1999).

Acknowledgements

Research presented in this article was supported by the Laboratory Directed Research and Development (LDRD) program of Los Alamos National Laboratory (LANL) under project number 20200056DR. This work was conducted in part at the Center for Integrated Nanotechnologies, a U.S. Department of Energy, Office of Basic Energy Sciences user facility. The authors would like to thank Yu Zhang for modifying the Psi4 code to suit the project needs, as well as Petr M. Anisimov and Christian F. A. Negre for fruitful discussions.

Author contributions

A.T. developed the algorithm, performed the numerical calculations, analysis and writing of the manuscript. B.K.K. contributed to the development of the algorithm, analysis and writing of the manuscript. P.A.D. selected and optimized molecules/ions with Gaussian code and generated the input matrices with Psi4. P.A.D. and S.T. contributed to the analysis and writing of the manuscript. P.A.D. supervised the project.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41598-020-77315-4.

Correspondence and requests for materials should be addressed to B.K.K. or P.A.D.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.