Universiteit Leiden
The Netherlands

Optimization of quantum algorithms for near-term quantum computers

Bonet Monroig, X.

7. Calculating energy derivatives for quantum chemistry on a quantum computer

7.1. Introduction

Quantum computers are at the verge of providing solutions for certain classes of problems that are intractable on a classical computer [47]. As this threshold nears, an important next step is to investigate how these new possibilities can be translated into useful algorithms for specific scientific domains. Quantum chemistry has been identified as a key area where quantum computers can stop being science and start doing science [18, 19, 185, 186]. This observation has lead to an intense scientific effort towards developing and improving quantum algorithms for simulating time evolution [187, 188] and calculating ground state energies [39, 86, 87, 189] of molecular systems. Small prototypes of these algorithms have been implemented experimentally with much success [39, 41, 42, 83, 100]. However, advances over the last century in classical computational chemistry methods, such as density functional theory (DFT) [190], coupled cluster (CC) theory [191], and quantum Monte-Carlo methods [192], set a high bar for quantum computers to make impact in the field.

The ground and/or excited state energy is only one of the targets for quantum chemistry calculations. For many applications one also needs to be able to calculate the derivatives of the molecular electronic energy with respect to a change in the Hamiltonian [193, 194]. For example, the energy gradient (or first-order derivative) for nuclear displacements is used to search for minima, transition states, and reaction paths [195] that characterize a molecular potential energy surface (PES). They also form the basis for molecular dynamics (MD) simulations to dynamically explore the phase space of the system in its electronic ground state [196] or,

after a photochemical transition, in its electronically excited state [197]. While classical MD usually relies on force-fields which are parameterized on experimental data, there is a growing need to obtain these parameters on the basis of accurate quantum chemical calculations. One can easily foresee a powerful combination of highly accurate forces generated on a quantum computer with machine learning algorithms for the generation of reliable and broadly applicable force-fields [198]. This route might be particularly important in exploring excited state PES and non-adiabatic coupling terms, which are relevant in describing light-induced chemical reactions [199–201]. Apart from these perturbations arising from changing the nuclear positions, it is also of interest to consider the effect that small external electric and/or magnetic fields have on the molecular energy. These determine well-known molecular properties, such as the (hyper)polarizability, magnetizability, A- and g-tensors, nuclear magnetic shieldings, among others.

Although quantum algorithms have been suggested to calculate derivatives of a function represented on a quantum register [184, 202–205], or of derivatives of a variational quantum eigensolver (VQE) for optimization purposes [115, 206], the extraction of molecular properties from quantum simulation has received relatively little focus. To the best of our knowledge only three investigations; in geometry optimization and molecular energy derivatives [207], molecular vibrations [116], and the linear response function [208]; have been performed to date.

In this chapter, we perform a geometry optimization of the H_2 molecule on a superconducting quantum processor, as well as its response to a small electric field (polarizability), and find excellent agreement with the full configuration interaction (FCI) solution.

7.2. Background

Let $\hat{H}$ be a Hamiltonian on a $2^{N_{\text{sys}}}$-dimensional Hilbert space (e.g. the Fock space of an N_{sys}-spin orbital system), which has eigenstates

$$\hat{H}|\Psi_j\rangle = E_j|\Psi_j\rangle, \tag{7.1}$$

ordered by the energies E_j. In this definition, the Hamiltonian is parametrized by the specific basis set that is used and has additional coefficients $\lambda_1, \lambda_2, \ldots$, which reflect fixed external influences on the electronic energy (e.g. change in the structure of the molecule, or an applied magnetic or electric field). An dth-order derivative of the ground state energy with respect to the

parameters λ_i is then defined as:

$$D^{d_1,d_2,\ldots}_{\lambda_1,\lambda_2,\ldots} = \frac{\partial^d E_0(\lambda_1,\lambda_2,\ldots)}{\partial^{d_1}\lambda_1,\partial^{d_2}\lambda_2,\ldots}, \tag{7.2}$$

where $d = \sum_i d_i$. As quantum computers promise exponential advantages in calculating the ground state E_0 itself, it is a natural question to ask how to efficiently calculate such derivatives on a quantum computer.

7.3. The quantum chemical Hamiltonian

A major subfield of computational chemistry concerns solving the electronic structure problem. Here, the system takes a second-quantized *ab* initio Hamiltonian, written in a basis of molecular spinors $\{\ \phi_p(\mathbf{r})\ \}$ as follows:

$$\hat{H} = \sum_{pq} h_{pq}\hat{E}_{pq} + \frac{1}{2}\sum_{pqrs} g_{pqrs}\Big(\hat{E}_{pq}\hat{E}_{rs} - \delta_{q,r}\hat{E}_{ps}\Big), \tag{7.3}$$

where $\hat{E}_{pq} = \hat{c}_p^\dagger \hat{c}_q$ and $\hat{c}_p^\dagger$ ($\hat{c}_p$) creates (annihilates) an electron in the molecular spinor ϕ_p. With equation (7.3) relativistic and non-relativistic realizations of the method only differ in the definition of the matrix elements h_{pq} and g_{pqrs} [209]. A common technique is to assume pure spin-orbitals and integrate over the spin variable. As we want to develop a formalism that is also valid for relativistic calculations, we will remain working with spinors in this chapter. Adaptation to a spinfree formalism is straightforward, and will not affect computational scaling and error estimates.

The electronic Hamiltonian defined above depends parametrically on the nuclear positions, both explicitly via the nuclear potential and implicitly via the molecular orbitals that change when the nuclei are displaced.

7.4. Energy derivative estimation using eigenstate truncation approximation

In this section, we present a method for calculating energy derivatives on a quantum computer. For wavefunctions in which all parameters are variationally optimized, the Hellmann–Feynman theorem allows for ready calculation of energy gradients as the expectation value of the perturbing

operator [207, 211]:

$$\frac{\partial E_0}{\partial \lambda} = \langle \Psi_0 | \frac{\partial \hat{H}}{\partial \lambda} | \Psi_0 \rangle. \tag{7.4}$$

This expectation value may be estimated by repeated measurement of a prepared ground state on a quantum computer, and classical calculation of the coefficients of the Hermitian operator $\partial \hat{H}/\partial \lambda$. If state preparation is performed using a VQE, estimates of the expectation values in Eq. 7.4 will often have already been obtained during the variational optimization routine.

The Hellmann–Feynman theorem cannot be so simply extended to higher-order energy derivatives. We may write an energy derivative via perturbation theory as a sum of products of path amplitudes A and energy coefficients f_A. For example, a second order energy derivative may be written as

$$\begin{aligned}\frac{\partial^2 E_0}{\partial \lambda_1 \partial \lambda_2} &= \langle \Psi_0 | \frac{\partial^2 \hat{H}}{\partial \lambda_1 \partial \lambda_2} | \Psi_0 \rangle \\ &+ \sum_{j \neq 0} 2 \,\mathrm{Re}\left[\langle \Psi_0 | \frac{\partial \hat{H}}{\partial \lambda_1} | \Psi_j \rangle \langle \Psi_j | \frac{\partial \hat{H}}{\partial \lambda_2} | \Psi_0 \rangle \right] \frac{1}{E_0 - E_j},\end{aligned} \tag{7.5}$$

allowing us to identify two amplitudes

$$\mathcal{A}_1(j) = \langle \Psi_0 | \frac{\partial \hat{H}}{\partial \lambda_1} | \Psi_j \rangle \langle \Psi_j | \frac{\partial \hat{H}}{\partial \lambda_2} | \Psi_0 \rangle, \tag{7.6}$$

$$\mathcal{A}_2 = \langle \Psi_0 | \frac{\partial^2 \hat{H}}{\partial \lambda_1 \partial \lambda_2} | \Psi_0 \rangle, \tag{7.7}$$

and two corresponding energy coefficients

$$f_1(E_0; E_j) = \frac{2}{E_0 - E_j}, \quad f_2 = 1. \tag{7.8}$$

The generic form of a d-th order energy derivative may be written as

$$\begin{aligned}D = &\sum_{\mathcal{A}} \sum_{j_1, \ldots, j_{X_{\mathcal{A}}-1}} \mathrm{Re}[\mathcal{A}(j_1, \ldots, j_{X_{\mathcal{A}}-1})] \\ &\times f_{\mathcal{A}}(E_0; E_{j_1}, \ldots, E_{j_{X_{\mathcal{A}}-1}}),\end{aligned} \tag{7.9}$$

where $X_{\mathcal{A}}$ counts the number of excitations in the path.

One may approximate the sum over (exponentially many) eigenstates $|\Psi_j\rangle$ in Eq. 7.9 by taking a truncated set of (polynomially many) approximate eigenstates $|\tilde{\Psi}_j\rangle$. We call such an approximation the eigenstate

truncation approximation, or ETA for short. However, on a quantum computer, we expect both to better approximate the true ground state $|\Psi_0\rangle$, and to have a wider range of approximate excited states [82, 100, 214–216]. Here, we focus on the quantum subspace expansion (QSE) method of [82]. This method proceeds by generating a set of N_E vectors $|\chi_j\rangle$ connected to the ground state $|\Psi_0\rangle$ by excitation operators $\hat{E}_j$,

$$|\chi_j\rangle = \hat{E}_j|\Psi_0\rangle. \tag{7.10}$$

This is similar to truncating the Hilbert space using a linear excitation operator in the (classical) equation of motion coupled cluster (EOMCC) approach [217]. The $|\chi_j\rangle$ states are not guaranteed to be orthonormal; the overlap matrix

$$S^{(\mathrm{QSE})}_{j,k} = \langle\chi_j|\chi_k\rangle, \tag{7.11}$$

is not necessarily the identity. To generate the set $|\tilde{\Psi}_j\rangle$ of orthonormal approximate eigenstates, one can calculate the projected Hamiltonian matrix

$$H^{(\mathrm{QSE})}_{j,k} = \langle\chi_j|\hat{H}|\chi_k\rangle, \tag{7.12}$$

and solve the generalized eigenvalue problem:

$$\hat{H}^{(\mathrm{QSE})}\vec{v}^{(j)} = \tilde{E}_j\hat{S}^{(\mathrm{QSE})}\vec{v}^{(j)} \rightarrow |\tilde{\Psi}_j\rangle = \sum_l \vec{v}^{(j)}_l|\chi_l\rangle. \tag{7.13}$$

Regardless of the method used to generate the eigenstates $|\tilde{\Psi}_j\rangle$, the dominant computational cost of the ETA is the need to estimate N_E^2 matrix elements. Furthermore, to combine all matrix elements with constant error requires the variance of each estimation to scale as N_E^{-2} (assuming the error in each term is independent). Taking all single-particle excitations sets $N_\mathrm{E} \propto N_\mathrm{sys}^2$. However, in a lattice model one might consider taking only local excitations, setting $N_\mathrm{E} \propto N_\mathrm{sys}$. Further reductions to N_E will increase the systematic error from Hilbert space truncation, although this may be circumvented somewhat by extrapolation.

For the sake of completeness, we also consider here the cost of numerically estimating an energy derivative by estimating the energy at multiple points;

$$\frac{\partial^2 E}{\partial\lambda^2} = \frac{1}{\delta\lambda}\left(\frac{\partial E}{\partial\lambda}(\lambda+\delta\lambda/2) - \frac{\partial E}{\partial\lambda}(\lambda-\delta\lambda/2)\right) + O(\delta\lambda^2). \tag{7.14}$$

In this equation, the derivatives can be computed using VQE via the Hellmann–Feynman theorem. One can see that the sampling noise is amplified by the division of $\delta\lambda$.

7.5. Geometry optimization on a superconducting quantum device

To demonstrate the use of energy derivatives directly calculated from a quantum computing experiment, we first perform geometry optimization of the diatomic H_2 molecule, using two qubits of a superconducting transmon device. (Details of the experiment are given in Sec. 7.B.) Geometry optimization aims to find the ground state molecular geometry by minimizing the ground state energy $E_0(\mathbf{R})$ as a function of the atomic co-ordinates R_i. In this small system, rotational and translational symmetries reduce this to a minimization as a function of the bond distance $R_{\mathrm{H-H}}$ In Fig. 7.1, we illustrate this process by sketching the path taken by Newton's minimization algorithm from a very distant initial bond distance ($R_{\mathrm{H-H}} = 1.5$Å). At each step of the minimization we show the gradient estimated via the Hellman–Feynman theorem. Newton's method additionally requires access to the Hessian, which we calculated via the ETA (details given in Sec. 7.B). The optimization routine takes 5 steps to converge to a minimum bond length of 0.749Å, within 0.014Å of the target FCI equilibrium bond length (given the chosen STO-3G basis set). To demonstrate the optimization stability, we performed 100 simulations of the geometry optimization experiment on the *quantumsim* density-matrix simulator [92], with realistic sampling noise and coherence time fluctuations (details given in Sec. 7.C). We plot all simulated optimization trajectories on Fig. 7.1, and highlight the median $(R_{\mathrm{H-H}}, E(R_{\mathrm{H-H}}))$ of the first 7 steps. Despite the rather dramatic variations between different gradient descent simulations, we observe all converging to within similar error bars, showing that our methods are indeed stable.

To study the advantage in geometry optimization from direct estimation of derivatives on a quantum computer, we compare in Fig. 7.2 our performance with gradient-free (Nelder-Mead) and Hessian-free (conjugate gradient, or CG) optimization routines. We also compare the performance of Newton's method with an approximate Hessian from Hartree-Fock (HF) theory. All methods converge to near-identical minima, but both Newton methods converge about twice as fast as the raw CG method, which in turn converges about twice as fast as Nelder-Mead. The density-matrix simulations predict that the ETA method Hessians provide less stable convergence than the HF Hessians; we attribute this to the fact that the HF Hessian at a fixed bond distance does not fluctuate between iterations. The density-matrix simulations also predict the CG method performance to be on average much closer to the Newton's method perfor-

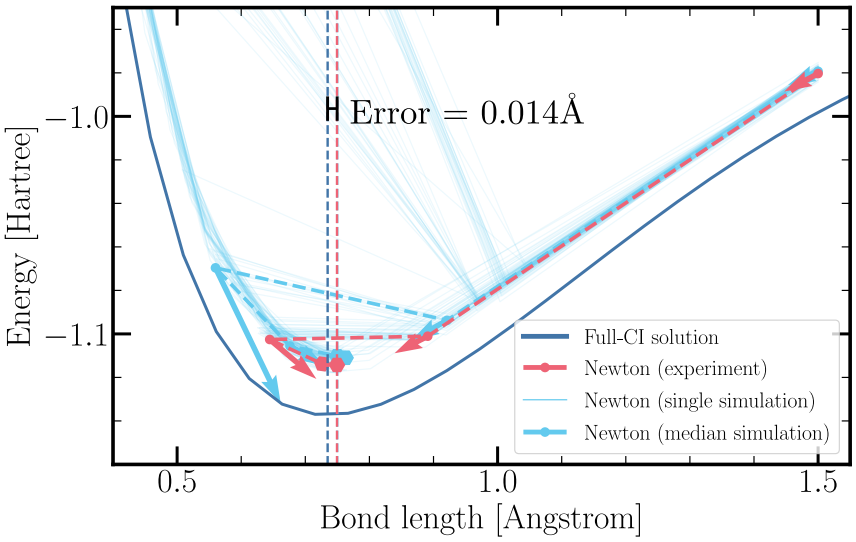

Figure 7.1.: Illustration of geometry optimization of the H_2 molecule. A classical optimization algorithm (Newton) minimizes the estimation of the true ground state energy (dark blue curve) on a superconducting transmon quantum computer (red crosses) as a function of the bond distance R_{H-H}. To improve convergence, the quantum computer provides estimates of the FCI gradient (red arrows) and the Hessian calculated with the response method. Dashed vertical lines show the position of the FCI and estimated minima (error 0.014Å). Light blue dashed lines show the median value of 100 density matrix simulations (Sec. 7.C) of this optimization, with the shaded region the corresponding interquartile range.

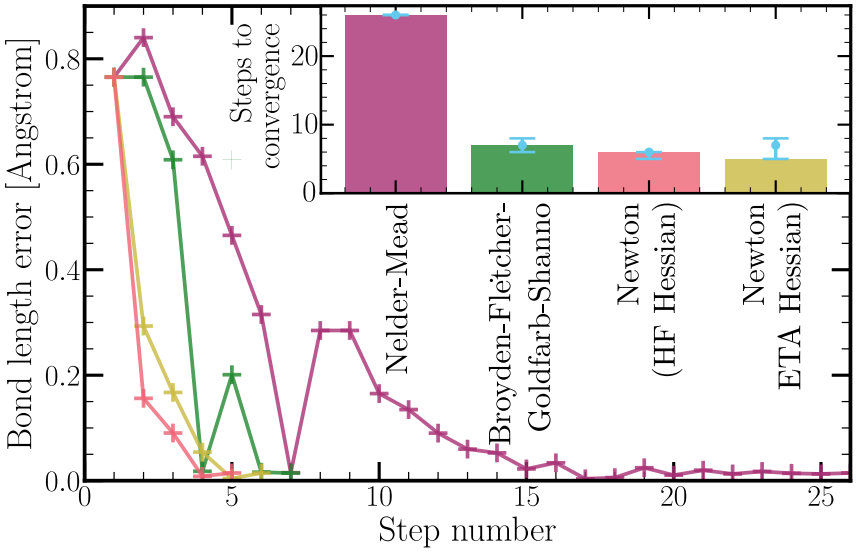

Figure 7.2.: Comparison of geometry optimization via different classical optimization routines, using a quantum computer to return energies and Jacobians as required, and estimating Hessians as required either via the ETA on the experimental device, or the Hartree-Fock (HF) approximation on a classical computer. Each algorithm was run till termination with a tolerance of 10^{-3}, so as to be comparable to the final error in the system. (Inset) bar plot of the number of function evaluations of the four compared methods. Light blue points correspond to median N_{fev} from 100 density-matrix simulations (Sec. 7.C) of geometry optimization, and error bars to the interquartile ranges.

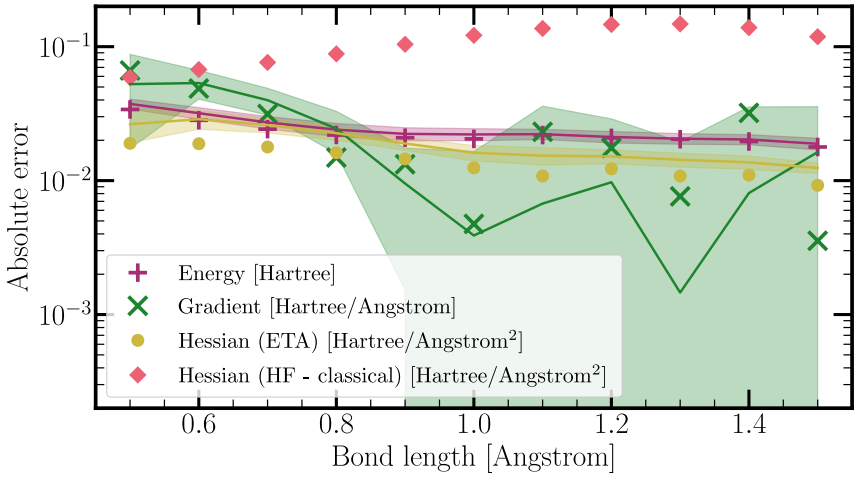

Figure 7.3.: Absolute error in energies and energy derivatives from an experimental quantum computation on 11 points of the bond dissociation curve of H_2. The error is dominated here by experimental sources (in particular qubit decay channels); error bars from sampling noise are smaller than the points themselves. Continuous lines connect the median value of 100 density matrix simulations at each points, with the shaded region corresponding to errors to the interquartile range.

mance. However, we expect the separation between gradient and Hessian-free optimization routines to become more stark at larger system sizes, as is observed typically in numerical optimization [155].

To separate the performance of the energy derivative estimation from the optimization routine, we study the error in the energy E, the Jacobian J and Hessian K given as $\epsilon_A = |A_{\mathrm{FCI}} - A_{\mathrm{expt}}|$, $(A = E, J, K)$. In Fig. 7.3, we plot these errors for different bond distances. For comparison we additionally plot the error in the HF Hessian approximation. We observe that the ETA Hessian is significantly closer than the HF-approximated Hessian to the true value, despite the similar performance in geometry optimization. The accuracy of the ETA improves at large bond distance, where the HF approximation begins to fail, giving hope that the ETA Hessian will remain appropriate in strongly correlated systems where this occurs as well.

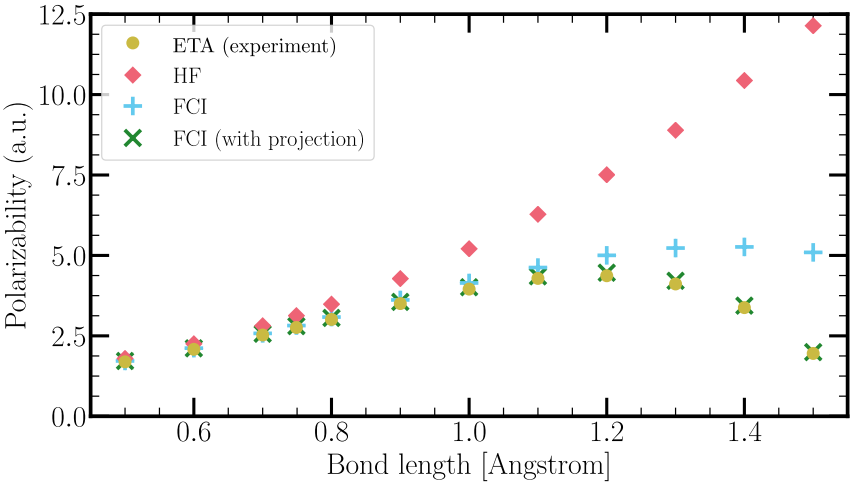

Figure 7.4.: Estimated polarizability of the hydrogen molecule as a function of the bond distance, in atomic units (1 a.u. = 0.14818471 Å^3).

7.6. Polarizability estimation

A key property to model in quantum chemistry is the polarizability, which describes the tendency of an atom or molecule to acquire an induced dipole moment due to a change in an external electric field $\vec{F}$. The polarizability tensor may be calculated as $\alpha_{i,j} = \frac{\partial E(\vec{F})}{\partial F_i \partial F_j}\Big|_{\vec{F}=0}$ [1]. In Fig. 7.4, we calculate the z-component of the polarizability tensor of H_2 in the ETA, and compare it to FCI and HF polarizability calculations on a classical computer. We observe good agreement to the target FCI result at low R_{H-H}, finding a 0.060 a.u. (2.1%) error at the equilibrium bond distance (including the inaccuracy in estimating this distance). However our predictions deviate from the exact result significantly at large bond distance ($R_{H-H} \gtrsim 1.2$ Å). We attribute this deviation to the transformation used to reduce the description of H_2 to a two-qubit device (see Sec. 7.B), which is no longer valid when adding the dipole moment operator to the Hamiltonian. To confirm this, we classically compute the FCI polarizability following the same transformation (which corresponds to projecting the larger operator onto a 2-qubit Hilbert space). We find excellent agreement between this and the result from the quantum device across the entire bond dissocia-

[1] The first-order derivative $\partial E/\partial F_i$ gives the dipole moment, which is also of interest, but is zero for the hydrogen molecule.

tion curve. This implies that simulations of H_2 on a 4-qubit device should match the FCI result within experimental error.

7.7. Conclusion

In this chapter, we have designed a new method for estimating energy gradients on a near-term quantum computer that we named eigenstate truncation approximation. We have demonstrated the use of this method on a small-scale quantum computing experiment, obtaining the equilibrium bond length of the H_2 molecule to 0.014Å (2%) of the target Full-CI value, and estimating the polarizability at this bond length to within 0.060 a.u. (2.1%) of the same target.

Our method do not particularly target the ground state over any other eigenstate of the system, and so can be used out-of-the-box for gradient estimation for excited state chemistry. Further investigation is also required to improve some of the results drawn upon for this work, in particular reducing the number of measurements required during a VQE.

Appendix

7.A. Appendix: Classical computation

The one- and two-electron integrals defining the fermionic Hamiltonian in Eq. 7.3 are obtained from a preliminary HF calculation that is assumed to be easily feasible on a classical computer. In non-relativistic theory the one-electron integrals are given by

$$h_{pq} = \int \mathrm{d}\mathbf{r}\phi_p^*(\mathbf{r})\left(-\frac{1}{2}\nabla_{\mathbf{r}} + V(\mathbf{r})\right)\phi_q(\mathbf{r}), \tag{7.15}$$

where $V(\mathbf{r})$ is the electron-nuclear attraction potential from fixed nuclei at positions $\mathbf{R}_i$. The two-electron integrals are given by,

$$g_{pqrs} = \iint \mathrm{d}\mathbf{r}_1\mathrm{d}\mathbf{r}_2 \frac{\phi_p^*(\mathbf{r}_1)\phi_q(\mathbf{r}_1)\phi_r^*(\mathbf{r}_2)\phi_s(\mathbf{r}_2)}{r_{12}}. \tag{7.16}$$

For simplicity we used a finite difference technique to compute the matrix representations of perturbations corresponding to a change in nuclear coordinates and an external electric field

$$\frac{\partial \hat{H}}{\partial \lambda} \approx \frac{\hat{H}(\lambda + \delta\lambda/2) - \hat{H}(\lambda - \delta\lambda/2)}{\delta\lambda}, \tag{7.17}$$

and

$$\frac{\partial^2 \hat{H}}{\partial \lambda^2} \approx \frac{\hat{H}(\lambda + \delta\lambda) + \hat{H}(\lambda - \delta\lambda) - 2\hat{H}(\lambda)}{\delta\lambda^2}, \tag{7.18}$$

where $\delta\lambda = 0.001$ corresponds to a small change in λ. The above (perturbed) quantum chemical Hamiltonians have been determined within the Dirac program [219] and transformed into qubit Hamiltonians using the OpenFermion [90] package. This uses the newly-developed, freely-available [220] OpenFermion-Dirac interface, allowing for the simulation of relativistic quantum chemistry calculations on a quantum computer. While a finite difference technique was sufficient for the present purpose,

such schemes are sensitive to numerical noise and have a high computational cost when applied to larger molecular systems. A consideration of the analytical calculation of energy derivatives can be found in the Supplementary Materials.

7.B. Appendix: Experimental methods

The experimental implementation of the geometry optimization algorithm was performed using two of three transmon qubits in a circuit QED quantum processor. This is the same device used in Ref. [44] (raw data is the same as in Fig.1(e) of this paper, but with heavy subsequent processing). The two qubits have individual microwave lines for single-qubit gating and flux-bias lines for frequency control, and dedicated readout resonators with a common feedline. Individual qubits are addressed in readout via frequency multiplexing. The two qubits are connected via a common bus resonator that is used to achieve an exchange gate,

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & i\sin(\theta) & 0 \\ 0 & i\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{7.19}$$

via a flux pulse on the high-frequency qubit, with an uncontrolled additional single-qubit phase that was cancelled out in post-processing. The exchange angle θ may be fixed to a $\pi/6000$ resolution by using the pulse duration (with a 1 ns duration) as a rough knob and fine-tuning with the pulse amplitude. Repeat preparation and measurement of the state generated by exciting to $|01\rangle$ and exchanging through one of 41 different choices of θ resulted in the estimation of 41 two-qubit density matrices ρ_i via linear inversion tomography of 10^4 single-shot measurements per prerotation [111]. All circuits were executed in eQASM [221] code compiled with the QuTech OpenQL compiler, with measurements performed using the qCoDeS [222] and PycQED [223] packages.

To use the experimental data to perform geometry optimization for H_2, the ground state was estimated via a VQE [39, 87]. The Hamiltonian at a given H-H bond distance $R_{\mathrm{H-H}}$ was calculated in the STO-3G basis using the Dirac package [219], and converted to a qubit representation using the Bravyi-Kitaev transformation, and reduced to two qubits via exact block-diagonalization [41] using the Openfermion package [90] and the Openfermion-Dirac interface [220]. With the Hamiltonian $\hat{H}(R_{\mathrm{H-H}})$ fixed, the ground state was chosen variationally: $\rho(R_{\mathrm{H-H}}) =$

$\min_{\rho_i} \mathrm{Trace}[\hat{H}(R_{\mathrm{H-H}})\rho_i]$. The gradient and Hessian were then calculated from $\rho(R_{\mathrm{H-H}})$ using the Hellmann–Feynman theorem and ETA respectively. For the ETA, we generated eigenstates using the QSE, with the Pauli operator XY as a single excitation. This acts within the number conserving subspace of the two-qubit Hilbert space, and, being imaginary, will not have the real-valued H_2 ground state as an eigenstate. (This in turn guarantees the generated excited state is linearly independent of the ground state.) For future work, one would want to optimize the choice of θ at each distance $R_{\mathrm{H-H}}$, however this was not performed due to time constraints. We have also not implemented the error mitigation strategies studied in Ref. [44] for the sake of simplicity.

7.C. Appendix: Simulation methods

Classical simulations of the quantum device were performed in the full-density-matrix simulator *(*quantumsim) [92]. A realistic error model of the device was built using experimentally calibrated parameters to account for qubit decay (T_1), pure dephasing (T_2^*), residual excitations of both qubits, and additional dephasing of qubits fluxed away from the sweet spot (which reduces T_2^* to $T_2^{*,red}$ for the duration of the flux pulse). This error model further accounted for differences in the observed noise model on the individual qubits, as well as fluctuations in coherence times and residual excitation numbers. Further details of the error model may be found in Ref. [44] (with device parameters in Tab.S1 of this reference).

With the error model given, 100 simulated experiments were performed at each of the 41 experimental angles given. Each experiment used unique coherence time and residual excitation values (drawn from a distribution of the observed experimental fluctuations), and had appropriate levels of sampling noise added. These density matrices were then resampled 100 times for each simulation.

7.D. Appendix: Numerical optimization and approximate Hessian calculations

Numerous numerical methods for geometry optimization exist, some gradient-free, some requiring only gradient calculations, and some making use of both gradients and Hessian data [155, 195]. As sampling noise from a quantum computer is typically far larger than the fixed point error on a classical computer, optimization techniques are required to be stable

in the presence of this noise. In particular, common implementations of algorithms that numerically estimate gradients tend to construct approximate derivatives by difference approximations, which (as we investigated above) dramatically enhance sampling noise unless care is taken. The Nelder–Mead gradient-free algorithm [224] is a common choice for optimization in quantum algorithms for this reason; as it does not rely on such an approximation, and implementations in scipy [159] prove relatively stable. Gradient- and Hessian-requiring algorithms do not tend to suffer from such instability as gradient-free methods.

In this work, our geometry optimization was reduced to a one-dimensional problem, removing some of the complexity of the task. With more atoms, one need to choose both the direction and the distance to step towards the minima of the energy landscape. Both the CG and Newton's methods are adjustments to the steepest descent algorithm (which aims to go solely in the direction of the derivative) to account for local curvature. In the absence of any higher order derivatives to assist adjustment, the non-linear CG algorithm weights each direction against traveling in previously-explored directions, and then performs a line-search in this direction (absent additional information that allows an estimation of how far to initially travel). Newton's method, by comparison, benefits from access to the Hessian, allowing us to choose

$$\delta\mathbf{R} = \left[\frac{\partial^2 E_0}{\partial \mathbf{R}^2}\right]^{-1} \frac{\partial E_0}{\partial \mathbf{R}}, \tag{7.20}$$

for the direction. One must compensate here for the fact that we wish to minimize, and not maximise, the energy. For our one-dimensional problem this is achieved by taking the absolute value of $\partial^2 E_0/\partial R_{\mathrm{H-H}}^2$; for a higher-dimensional problem this is slightly more involved [155]. Regardless, such modified Newton's methods tend to provide a far more optimal method for estimating higher dimensional functions than Hessian-free methods [155, 195]. We are further able to bound the minimum bond length in our geometry optimization (in particular to $R_{\mathrm{H-H}} > 0.3$ Å), which can be of importance for stability as classical methods tend to fail when atoms are unrealistically close together.

For large systems when low accuracy is needed (e.g. at the start of a geometry optimization calculation), one may consider calculating the Hessian via the HF Hamiltonian for the same geometry as a low-cost alternative to explicit calculation on the quantum computer. This is a standard technique for geometry optimization in computational chemistry [225]. As the Hessian is not used to determine convergence (which depends instead

on the size of the gradient), the approximation only affects the convergence rate and stability, rather than the final result. This is even more so for quasi-Newton methods, as the Hessian is updated during the geometry optimization by the estimated gradients, which are more accurate. Calculating the HF Hessian is a standard procedure in most computational programs; for further mathematical information, we refer the reader to Ref. [226].