

ARTICLE OPEN



Gate-free state preparation for fast variational quantum eigensolver simulations

Oinam Romesh Meitei¹, Bryan T. Gard², George S. Barron², David P. Pappas^{3,4}, Sophia E. Economou², Edwin Barnes² and Nicholas J. Mayhall¹✉

The variational quantum eigensolver is currently the flagship algorithm for solving electronic structure problems on near-term quantum computers. The algorithm involves implementing a sequence of parameterized gates on quantum hardware to generate a target quantum state, and then measuring the molecular energy. Due to finite coherence times and gate errors, the number of gates that can be implemented remains limited. In this work, we propose an alternative algorithm where device-level pulse shapes are variationally optimized for the state preparation rather than using an abstract-level quantum circuit. In doing so, the coherence time required for the state preparation is drastically reduced. We numerically demonstrate this by directly optimizing pulse shapes which accurately model the dissociation of H₂ and HeH⁺, and we compute the ground state energy for LiH with four transmons where we see reductions in state preparation times of roughly three orders of magnitude compared to gate-based strategies.

npj Quantum Information (2021)7:155; <https://doi.org/10.1038/s41534-021-00493-0>

INTRODUCTION

Molecular modeling stands in the juncture of key advances in many important fields including and not limited to energy storage, material designs, and drug discovery. For more than half a century, many seminal works have been reported on the development of theories and methods to enable molecular modeling with high accuracies. Approximate numerical methods which are built on a single Slater determinant reference state, such as density functional theory (DFT), perturbation theory, or coupled cluster, perform well when the amount of electron correlation, ranges from minimal to moderate. However, for systems which are qualitatively governed by electron correlation effects (strongly correlated systems), such approximate methods fail to be sufficiently accurate. While alternative strategies exist, such as density matrix renormalization group (DMRG)^{1–3} or selected configurational interaction (SCI) methods^{4–13}, which can handle strong correlation, these approaches assume that the correlations are either low-dimensional, or limited in scale. Currently, no polynomially scaling classical algorithm exists which can solve for arbitrary molecular ground states.

High dimensionality of the wavefunction is ultimately the core reason for the exponential cost of modeling electronic structure. Even before optimization, simply storing the wavefunction on classical hardware quickly becomes a bottleneck. This is because one typically represents the wavefunction in a basis of “classical” states, or basis states which have a direct product structure (Slater determinants, occupation number vectors, or even tensor product states¹²). In this classical basis, the exact (and generally entangled) state of the system is represented as an exponentially (or factorially, if a basis of Slater determinants is used to span a target projected spin subspace) large vector of coefficients weighting the corresponding classical basis states.

Quantum computing offers a radical departure from this computational strategy. As quantum systems themselves, the state of a quantum processing unit (QPU) is also a vector in a Hilbert space of identical dimension to the molecular problem.

This ability to perform a one-to-one mapping between vectors in the Hilbert space containing the molecule’s electronic wavefunction and those in the state space accessible to a QPU means that with enough control over the QPU, it should be possible to take the vector corresponding to the molecular wavefunction and realize it on the QPU, avoiding altogether the requirement to work with an exponentially large vector of coefficients. Once the QPU is prepared into the state corresponding to the target molecule, any molecular observable (energy, dipole moment, etc) can be obtained by measuring the corresponding operator on the QPU.

In order to turn this strategy into an algorithm, one needs a procedure for realizing the target molecular wavefunction on the QPU. As the leading quantum algorithm for molecular simulation on near-term devices, the variational quantum eigensolver (VQE)¹⁴ provides an efficient procedure for this purpose. In VQE, one defines a parameterized quantum circuit comprised of tunable gates, and then optimizes these gates using the variational principle, minimizing the energy of the molecular Hamiltonian. This parameterized quantum circuit (referred to as an ansatz) defines the variational flexibility (and thus the subspace reachable on the QPU) of the algorithm.

State-preparation circuits with more parameters generally have more variational flexibility, but come with the cost of having deeper quantum circuits and more difficult optimization. This cost can be significant. Current and near-term quantum computers are classified as noisy intermediate scale quantum (NISQ) devices due to the presence of short coherence times, system noise, and frequent gate errors¹⁵. Because each gate has limited fidelity, the success probability for a sequence of gates decreases exponentially with circuit depth. Even if gates could reach unit fidelity, the finite coherence times of a NISQ device still limits the number of gates that one can apply in a circuit which, in turn, limits the accuracy of the molecular VQE simulation. Although VQE is relatively robust in the presence of noise and errors in certain cases¹⁶, the critical limitation preventing larger scale experiments is the accurate implementation of deep circuits. The goal of

¹Department of Chemistry, Virginia Tech, Blacksburg, VA 24061, USA. ²Department of Physics, Virginia Tech, Blacksburg, VA 24061, USA. ³Physics Department, University of Colorado, Boulder, CO 80309, USA. ⁴National Institute of Standards and Technology, Boulder, CO 80305, USA. ✉email: nmayhall@vt.edu

finding parameterized circuits which minimize the circuit depth and maximize the accuracy has led to a number of approaches such as hardware efficient ansätze¹⁷, physically motivated fixed ansätze^{18–22}, and adaptive ansätze^{23–25}.

Each of these mentioned approaches essentially aims to find the shortest quantum circuit for preparing a state to within an acceptable error. However, even if one succeeds at finding the “best” circuit, it is likely that its execution time will still exceed the coherence time of a NISQ device. In order to make meaningful progress toward larger problems, one can imagine three strategies: (i) device improved error mitigation techniques, (ii) significantly increase the ratio of coherence to gate time, or (iii) simply abandon gates and directly optimize the electronic controls. Of these, the latter strategy appears to be the most feasible on near-term NISQ devices.

In this paper, we explore the possibility of performing gate-free VQE simulations by replacing the parameterized quantum circuit with a direct optimization of the laboratory-frame analog control settings. In the following sections, we argue that quantum control techniques are likely to be better suited for fast VQE state preparation than the more conventional circuit-based approaches on NISQ devices. We first provide a detailed overview of circuit-based VQE, then introduce our proposed strategy, ctrl-VQE, then discuss initial results along with a strategy to avoid over-parameterization, and finally compare to gate-based ansätze. Several technical aspects, numerical results, and additional discussion are provided in the Supplementary Methods and Supplementary Discussion.

RESULTS AND DISCUSSION

Dissociation of diatomic molecules

In this section, we explore the ability of ctrl-VQE to reproduce full configuration interaction (FCI) (exact diagonalization) bond dissociation energy curves for two small example diatomic molecules. Here, we demonstrate the performance of our approach by computing the ground state molecular electronic energy along the bond dissociation of the H_2 molecule and the HeH^+ molecular ion. Although small, these molecules have been used as benchmarks for quantum algorithms in recent years^{17,20,21,26–35}. These two molecular examples are chosen because they exhibit different behaviors during bond dissociation in terms of their electronic structure. As two-orbital, two-electron problems, these would naturally be modeled on four qubits following the commonly used Jordan-Wigner transformation³⁶. However, to make the pulse simulations more computationally efficient, we have used the parity mapping instead in which two qubits are diagonal and can be removed from consideration. Details are given in Bravyi et al.³⁷, and we use the implementation for mapping and pruning in Qiskit software³⁸.

As a prototypical example of a homolytic dissociation,



Dynamic correlation \rightarrow Static correlation

as H_2 dissociates, the ground state moves from being dominated by a single Slater determinant to increasingly multiconfigurational, due to the shrinking HOMO-LUMO gap. As a result, the accuracy of a mean-field treatment (e.g., HF) diminishes as the bond length is stretched.

In stark contrast, HeH^+ becomes easier to model as the bond distance increases. The reason is that, being the strongest acid and, interestingly, the first molecule formed in the universe³⁹, dissociation is a heterolytic deprotonation,



Dynamic correlation \rightarrow No correlation*

such that both the reactants and the products are both closed

shell and well represented by a single Slater determinant. If fact, in a minimal basis set (i.e., the STO-3G basis set used in this paper), the products have exactly zero electron correlation energy because the H^+ ion has no electrons, while the He atom has no empty virtual orbitals into which electrons can be promoted. As a result, HF becomes exact at dissociation. Of course, in larger basis sets, the He atom would have some dynamic correlation, hence the * in Eq. (2).

The dissociation curves of the H_2 and HeH^+ molecular systems produced with ctrl-VQE using the simple square pulses are presented in Figs. 1 and 2 respectively. Consistent with the physical descriptions above, the HF state moves quickly away from the FCI ground state with bond dissociation for H_2 , while the opposite is true for HeH^+ , where the HF state gradually converges to the exact FCI ground state with increasing bond distance. ctrl-VQE reproduces the FCI bond dissociation curve of H_2 and HeH^+ with high accuracies. The maximum difference from the FCI energy along the dissociation curve is 0.03 mHa for both H_2 and HeH^+ , and the average error is 0.002 mH in both the cases. More detailed information including the pulse parameters, characteristics, and molecular energies along the dissociation curves are provided in the SI. The resulting states have a good overlap with the exact FCI ground states (99% in all cases). For H_2 , it is possible for the overlap to deviate at long bond distances due to degenerate singlet and triplet states. As the bond distance increases, the singlet ground state of H_2 becomes degenerate with the lowest triplet state, making it possible to generate a superposition of a singlet and triplet. It is possible to converge to the exact ground singlet state by supplementing the objective function with a penalty term proportional to the total spin operator (see SI for details).

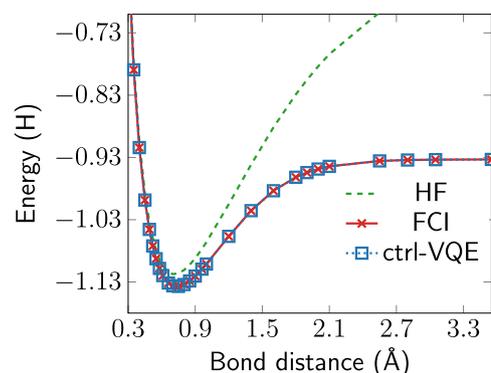


Fig. 1 Bond dissociation curve of H_2 molecule. ctrl-VQE energies are computed using square pulses with two time segments.

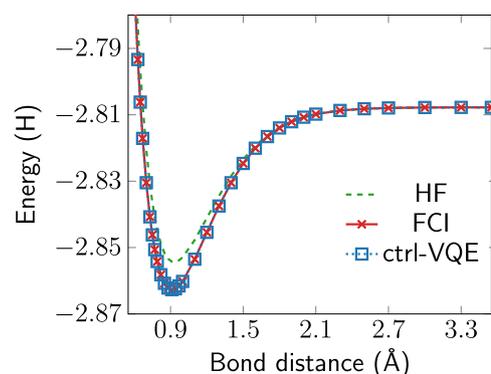


Fig. 2 Bond dissociation curve of HeH^+ molecule. ctrl-VQE energies are computed using square pulses with two time segments.

Effect of electron correlation on pulse duration

In any VQE, one is typically interested in finding a useful balance between accuracy (needing deep circuits) and noise (needing shallow circuits). As such, molecular simulations of strongly correlated molecules are intrinsically more difficult as deeper circuits are required, which increases problems from noise and gate errors. An analogous balance is targeted in ctrl-VQE, where one would hope to obtain sufficiently accurate results with as short of a pulse time as possible. Because entanglement cannot be created instantly⁴⁰, we expect molecules with strong correlation to require longer pulses than simpler molecules.

In order to examine this relationship, in Fig. 3 we plot the duration of the shortest pulses we were able to obtain at each molecular geometry, with H₂ (HeH⁺) shown in red (purple). Referring back to Eqs. (1) and (2), one would expect that as the bond distance is increased, H₂, needing more entanglement, would in turn require increasing pulse durations, whereas pulses for HeH⁺ would get increasingly fast as the bond is stretched. This trend is indeed observed.

The total pulse duration significantly decreases for the dissociation of the HeH⁺ molecular ion, whereas it significantly increases for the H₂ molecule. Note that the initial state and the final target state for HeH⁺ become degenerate with increasing bond distance (above 2.0 Å). Thus, the Hartree Fock state ($|01\rangle$) is a good approximation to the exact FCI ground state, and only a slight modification to the initial state is required to well approximate the ground state. With ctrl-VQE, the total pulse duration at the far bond distances of HeH⁺ are only 1.0 ns. This directly reflects the efficiency of the method presented in this work. In a gate-based compilation method, generally one would still construct an ansatz comprised of costly two-qubit gates, even though the target state is very nearly approximated by the initial state.

The total pulse duration at the dissociation limit for H₂ is significantly longer than near the equilibrium bond distance. Although the HF molecular energy increases monotonically beyond the equilibrium point, the same is not observed for the pulse duration. This is suggestive of the pulse durations reflecting the different dynamics along the bond dissociation of the two molecular systems (see SI for a more detailed analysis).

Leakage to higher energy states

In Figs. 4 and 5, we show the energy and leakage as a function of evolution time for various choices of the hyper-parameter, T , (the total evolution time). In each of the two figures, the top panel displays the behavior of the molecular energy as the state evolves

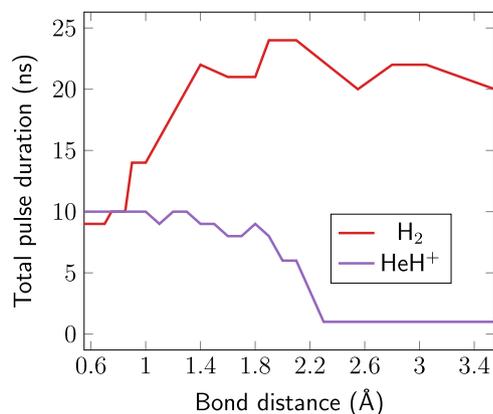


Fig. 3 Total pulse duration. Shown as a function of bond distance of H₂ and HeH⁺ molecular systems. Square pulses were used to generate the ctrl-VQE molecular energies in Figs. 1 and 2. The shortest pulse duration at each geometry is plotted here.

from the reference Hartree-Fock state to the converged trial state, and the bottom panel displays the total population outside of the $|0\rangle$ and $|1\rangle$ computational states for each qubit (leakage).

We see that for each value of T , the molecular energy of both H₂ and HeH⁺ never decreases toward the exact energy monotonically, but rather increases initially, and then either decreases or oscillates before rapidly converging to the exact energy. While each of the optimized pulses do indeed generate suitable transmon dynamics which accurately produce the molecular ground state at the specified time T , the different pulses are not each equally favorable. From the bottom panels of Figs. 4 and 5, we see that some pulses create much more leakage than other pulses. High levels of leakage will likely require a larger number of shots (pulse executions) to get precise determinations of the expectation values, since a higher portion of the shots end up in excited states which are either discarded, or collected for normalizing the results. However, with leakage of only 10%, we anticipate only needing a commensurate increase in the shot count to compensate for post-selection, and so we do not expect this to be a fundamental limitation of the approach. Alternatively, one can use an unnormalized energy in the objective function. This approach naturally constrains the solutions to simultaneously minimize leakage, as any leakage necessarily penalizes the associated pulse. Results using an unnormalized energy in the objective function are comparable to those shown here, and are given in the SI.

Noise analysis: imprecise control

All the analysis thus far has assumed perfect control over the driving parameters used to generate approximate ground states of chemical Hamiltonians. However, there are always uncertainties

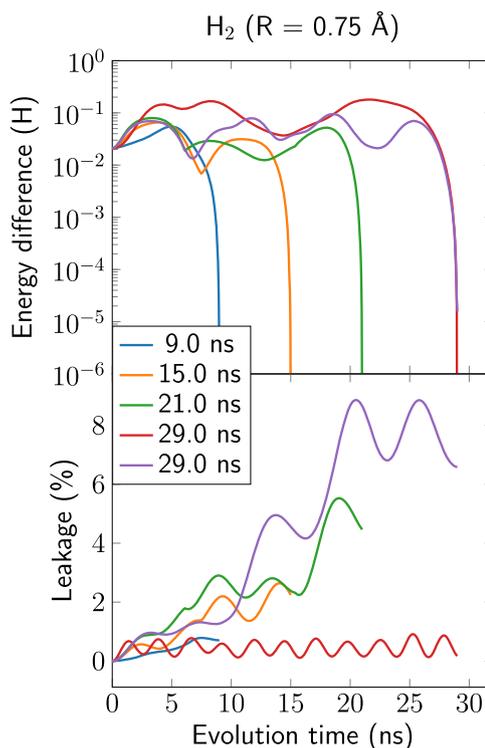


Fig. 4 Error in energy and leakage for H₂. Energy difference shown between ctrl-VQE and FCI (top) and leakage (bottom) along the time evolution steps with optimal pulses of different durations for H₂ with a bond distance of 0.75 Å. The red and purple lines with $T = 29$ ns are distinct solutions to the same optimization. Optimized pulse parameters are provided in the SI.

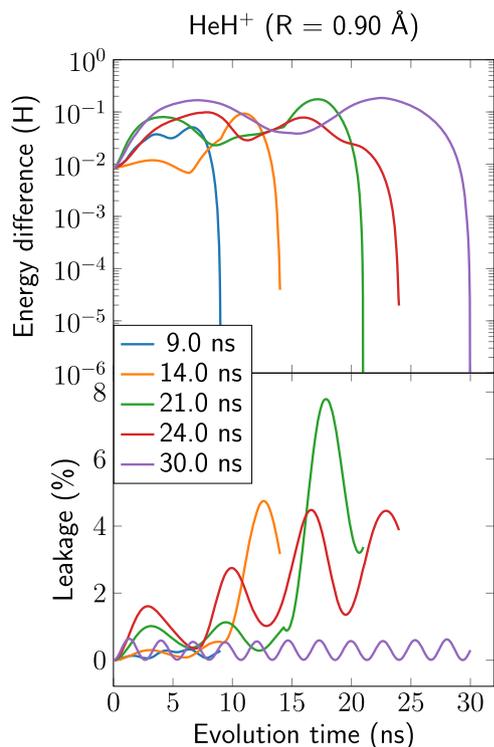


Fig. 5 Error in energy and leakage for HeH⁺. Energy difference shown between ctrl-VQE and FCI (top) and leakage (bottom) along the time evolution steps with optimal pulses of different durations for HeH⁺ with a bond distance of 0.90 Å. Optimized pulse parameters are provided in the SI.

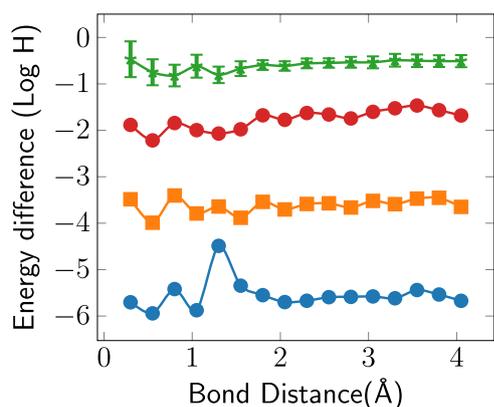


Fig. 6 Effect of imprecise control parameters. Energy error for the dissociation of H₂ is shown. Instead of using the exact, optimized parameters, we use parameters which have added Gaussian uncertainty. For each optimal parameter, we randomly sample 100 points each from a Gaussian distribution centered on its optimal setting and with a standard deviation of $\sigma = \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$ which correspond to the blue, orange, red and green curves respectively. The resulting energies for these 100 trials are then averaged. Error bars are given for each point but are smaller than some markers.

when attempting to implement the optimal driving parameters. In Fig. 6 we simulate the effect of imprecisely implementing these parameters. For each set of optimal drive parameters found by the ctrl-VQE protocol, we produce 100 samples of noisy parameters taken from Gaussian distributions that are centered about the

optimal parameter values:

$$\theta_n \leftarrow \exp\left(-\frac{(\theta - \theta_o)^2}{2\sigma^2}\right), \quad (3)$$

where θ_n is a noisy version of the optimal parameter θ_o . These in turn are used to produce 100 noisy energy samples. We then average these 100 samples and find the resulting energy difference relative to the target ground state energy. With this applied noise model, we can see that even imprecise settings can still be used to achieve accuracies below a 10^{-4} energy error. As shown in the red and green curves in Fig. 6, only when the noise becomes higher than $\sigma = 0.01$ (where errors reach a magnitude of 10 MHz or 1 ns) do the energy differences become significant enough to affect the performance of ctrl-VQE. Since these errors are of the same order as the parameters themselves, they do not hinder the realistic implementation of ctrl-VQE. We do not include any explicit noise due to finite decoherence (T_1) or dephasing (T_2^*) since the pulses we find are many orders of magnitude shorter than the typical time scales for these effects⁴¹. To confirm this, we ran simulations using a Lindblad master equation that includes these effects. Our objective function did not change significantly due to our short time scales.

Comparison with circuit compilation techniques

Although several ansätze have been proposed to achieve shorter circuits, even the most compact approaches involve too many gates to implement on current hardware for all but the smallest molecules. In order to reduce the time spent during the state-preparation stage, and thus the coherence time demands on the hardware, circuit compilation techniques have been designed to take a given quantum circuit and execute it either with fewer gates or by reoptimizing groups of gates for faster execution.

To execute the gate-based VQE (described in Section III A) experimentally, the gates in a circuit are compiled to sequences of analog control pulses using a look-up table that maps each elementary gate to the associated predefined analog pulse. The sequence of control pulses corresponding to the elementary gates in the quantum circuit are then simply collected and appropriately scheduled to be executed on the hardware. As such the compilation is essentially instantaneous, making the gate-based compilation technique well suited for VQE algorithms where numerous iterations are performed.

From a compactness perspective however, this gate-based compilation is far from ideal, resulting in total pulse durations which are much longer than what might be obtained with optimized compilation techniques. As is obvious from the one-to-one translation of gates to pulses, the overall circuit structure is typically not considered in gate-based compilations. Thus one may naturally be inclined to seek an optimal pulse sequence for the entire circuit. This has motivated compilation algorithms where control pulses are optimized for the target circuit, using numerical optimal control techniques such as gradient ascent pulse engineering (GRAPE)⁴² for partially optimal compilation⁴³. However, because the GRAPE algorithm itself is highly non-trivial, the compilation latency for each iteration is of critical concern. Two of the GRAPE-based techniques are briefly described below; see Gokhale et al.⁴³ for a detailed discussion.

The GRAPE compilation technique employs an optimal control routine which compiles to machine-level sequences of analog pulses for a target quantum circuit. This is achieved by manipulating the sets of time-discrete control fields that are executed on the quantum system. The control fields in the optimal routine are updated using gradients (see Boutin et al.⁴⁴ for use of analytical gradients) of the cost function with respect to the control fields. GRAPE compilation achieves up to 5-fold speedups compared to standard gate-based compilation. However, such

speedup comes with a substantial computational cost. This amounts to long compilation latency, making this approach impractical for VQE algorithms in which multiple iterations of circuit-parameter optimizations are performed. The GRAPE-based compilation also suffers from limitations in the circuit sizes it can handle^{45–47}.

On the other hand, partial compilation techniques achieve significant pulse speedups by leveraging the fast compilation of standard gate-based techniques together with the pulse speedups of GRAPE-based compilations. Two flavors of such an approach were reported in Gokhale et al.⁴³. Both divide the whole circuit into blocks of subcircuits. In the so-called strict partial compilation, the structure of quantum circuits used in quantum variational algorithms are exploited to only perform GRAPE-based compilation on fixed subcircuits that are independent of the circuit parametrization. The optimal pulses using the GRAPE compilation techniques for the fixed blocks are pre-computed and simply concatenated with the control pulses from the gate-based compilations for the remaining blocks of the circuit. Thus, the compilation speed is comparable to the gate-based compilations in each iteration, but this method also takes advantage of pulse speedups from GRAPE-based compilations. As one may expect, the pulse speedups heavily depend on the circuit depth of the fixed blocks.

In the other flavor, flexible partial compilation, circuits are blocked into subcircuits, each with a single variational parameter, ensuring the subcircuit blocks have low depth. Hyperparameter optimizations are performed for each subcircuit, and they are utilized to find optimal pulses for the circuit blocks. It is noteworthy to mention that in the flexible partial compilation, compilation latency is reduced significantly ($\sim 80\times$, see Gokhale et al.⁴³) by tuning the hyperparameters of the circuit blocks to speed up the convergence of optimal control in GRAPE. Flexible partial compilations achieve significant pulse speedups as compared to strict partial compilations. However, in spite of the high pulse speedups, the flexible partial compilation technique still diverges from the extremely fast compilation time of the standard gate-based methods. Thus, the algorithm still suffers from compilation latency, albeit to a lesser degree compared to GRAPE-based compilation.

Although both of these GRAPE-based compilation techniques and ctrl-VQE share a direct pulse-shaping step, they fundamentally differ because ctrl-VQE is not a compiler. In contrast to full or partial compiling, we make no reference to a circuit or to implementing any specific unitary. Of course, the control pulses we find do implement some equivalent circuit or unitary (and we analyze these in Sec. II F), but we have no need to ever

characterize the unitary. In fact, because ctrl-VQE only targets a single state, a large family of unitaries (defined by the behavior in the spaces orthogonal to the initial and final states) exist which minimize our objective function. As such, many possible solutions exist, with no immediate preference given to any one.

Decompiled control pulses

In the previous sections, the efficiency of ctrl-VQE using optimized control pulses at the device level was demonstrated. The short pulse durations imply that applications to larger molecules might have even more significant speedup since the number of CNOTs in most VQE ansätze increases quickly.

Although ctrl-VQE is performed using no state preparation circuit, the unitary created by the time dynamics of the applied pulse can be decomposed into gates, allowing one to analyze the circuit. This is essentially running a compiler in reverse, or “decompiling” the pulse to yield a state preparation circuit. With this decompiled circuit at hand, we can evaluate the time it would take to execute the optimized pulse as a traditional circuit. By comparing this time to that of the pulse duration, one has a clean benchmark for quantifying the overhead associated with gate-based state preparation. This decompilation can be done in two ways. In the first approach, we simply compute the matrix representation of the evolution operator generated by the optimized ctrl-VQE pulse. For the two-qubit case in focus here, a quantum circuit is then constructed using the KAK decomposition technique. For a detailed description of the technique, we refer the reader to Williams⁴⁸.

In the second approach, an arbitrary circuit corresponding to the state vector from ctrl-VQE is constructed and then transpiled to obtain a shorter circuit depth. In this way we obtained gate durations on IBMQ hardware (mock Johannesburg device available in Qiskit software). The KAK decomposition, arbitrary circuit construction, transpilation, and the mock circuit compilations were performed using Qiskit software³⁸. The quantum circuits are illustrated in Fig. 7. The corresponding pulse durations are 1202 ns for the circuit obtained using the KAK decomposition and 825 ns for the circuit obtained from transpilation. The state vector used in this study was for the H_2 molecule at 0.75 Å, and the corresponding pulse duration was 9 ns with ctrl-VQE. This clearly demonstrates that state preparation circuits can be much deeper than is necessary.

Adaptive update of pulse parameterization

The manner in which a pulse is parameterized (in particular, the number of variational parameters) will significantly impact

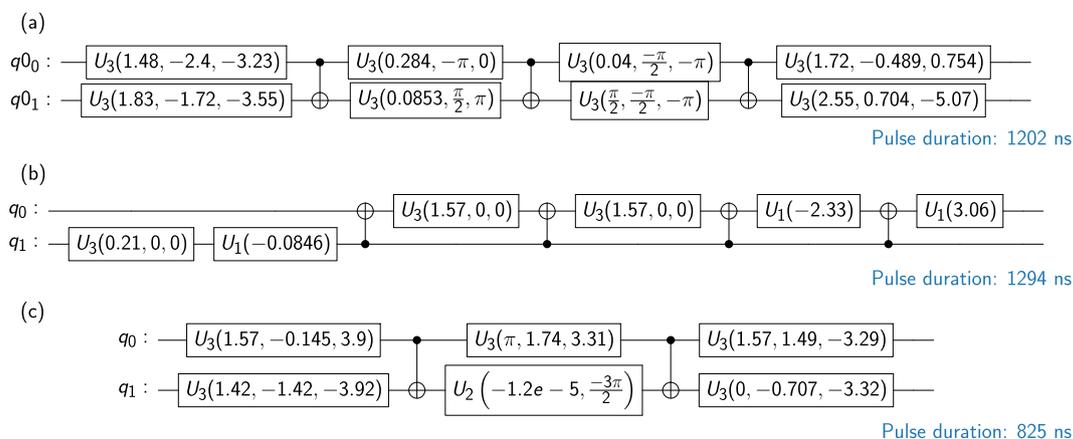


Fig. 7 Circuits from decompiled control pulses. Illustration of circuits constructed using the unitary and state vectors from ctrl-VQE. **a** Circuit corresponding to unitary obtained from a KAK decomposition. **b** An arbitrary circuit corresponding to the state vector. **c** Transpiled version of circuit (b).

experimental performance. An over-parameterization of the pulse leads to difficulties in experimental optimization. In response, we seek a means to limit the number of parameters. In this section, we describe a scheme to avoid over-parameterizations and arrive at the minimal number of pulse parameters to achieve a target accuracy.

Unlike in the previous sections where we chose a fixed number of time segments, in this section we propose an adaptive algorithm which slowly grows the number of segments (and thus parameters). We begin with a single time-segment square pulse (constant amplitude throughout the time evolution), the pulse is then iteratively sliced at random intervals such that the number of time segments systematically increases. This adaptive scheme is outlined below:

1. Initialize a parameterized square pulse with $n=1$ time segment and perform the variational pulse optimization.
2. Divide the pulse obtained from the previous step at a randomly chosen time. The square pulse now has $n=2$ time segments.
3. Perform pulse optimization using the pulse shape from the previous step as an initial guess.
4. Divide the largest time segment in the square pulse obtained from the previous step into two at a randomly chosen time. This increases the number of segments from n to $n+1$.
5. Perform pulse optimization using the pulse shape from the previous step as an initial guess.
6. Repeat Steps 4 and 5 until the energy obtained is sufficiently converged.

Note that drive frequencies for the pulses are also optimized in the above procedure. Irrespective of the pulse shape used, only a single drive frequency is used for each qubit. The switching times for the square pulses with more than one time segment are not optimized and remain fixed in Steps 3 and 5. For N transmons with n pulse time segments each, the total number of parameters to optimize is $N(n+1)$. In the following simulations, we constrain the amplitudes and drive frequencies within the ranges ± 40 MHz and $\omega_k \pm 3\pi$ GHz, respectively.

Following the above strategy of adaptively increasing the number of time segments in the square pulses, we obtained optimal pulses for H_2 , HeH^+ and LiH with total pulse durations of 9, 9, and 40 ns respectively. Figure 8 plots the energy difference relative to FCI with increasing number of time segments in the optimal pulse. For H_2 and HeH^+ , chemical accuracy is obtained with a single-time-segment square pulse on each qubit, while for

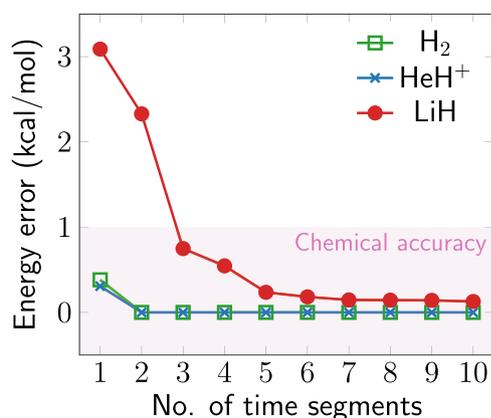


Fig. 8 Convergence of pulse parameterization. Energy difference of ctrl-VQE relative to FCI with increasing number of time-segments in the square pulse used.

LiH, three time segments are required to achieve chemical accuracy. The molecular energy converges for square pulses with two time segments for H_2 and HeH^+ and with five time segments for LiH molecules. The error in ground state energy of LiH molecule (Li–H $R=1.5$ Å) computed with ctrl-VQE along with pulse duration, leakage, and overlap with the exact FCI state is tabulated in Table 1. To illustrate the sequential pulse refinement from iteration to iteration, the square pulse shapes at each adaptive step for LiH are shown in Fig. 9.

In the case of LiH, we see that the accuracy is not of the same order as in the other two diatomic molecules. To further examine this, we perform over 100 pulse optimizations on the square pulse with 10 time segments obtained from the adaptive strategy by retaining only the switching times and starting with random amplitudes and drive frequencies. We use a large number of initial guesses to reduce the likelihood of getting trapped in a local minimum. We find that the highest accuracy is achieved when we use the pulse obtained from the adaptive strategy above. This suggests that we are nearing the accuracy limit of LiH using a 40 ns pulse.

While the above adaptive strategy aims at practical experimental implementation, the cost of classical simulation remains the same irrespective of the number of time-segments when analytical gradients for the pulse amplitudes are used. The cost of evaluating an analytical gradient is only about 2.5 times the cost of an energy evaluation. Here, the analytical gradient is obtained at each trotter step of time evolution and adapted for the pulse shaped considered. Details along with the derivation are provided in the SI.

Comparison to gate-based ansätze

Now we directly compare the results of our pulse-based technique with gate-based variational ansätze. We use calibration data from an IBMQ device (mock Johannesburg device available in Qiskit Terra version 0.14.2) to compute the duration of the circuits used to prepare trial wavefunctions. We consider the RY and UCCSD ansätze, which are capable of producing the exact ground state.

For both H_2 and HeH^+ , the RY ansatz requires 1 CNOT, and the UCCSD requires 2 CNOTs. The RY ansatz has four parameters, and the UCCSD ansatz has three parameters. The total pulse time of the RY ansatz is 519 ns, and the UCCSD ansatz is 825 ns. For the LiH molecule, the RY and UCCSD ansätze require 9 and 245 CNOTs respectively. The RY ansatz has 16 parameters whereas the UCCSD ansatz has 8 parameters. The total pulse duration of the RY ansatz is 3485 ns, and that of the UCCSD ansatz is 82,169 ns for LiH. In each case, the time to execute the circuit is significantly longer than the time required to apply the pulses using ctrl-VQE, which is 9 ns for H_2 and HeH^+ , and 40 ns for LiH. The T_1 and T_2 times of the IBMQ device used range from 45,150 to 115,217 ns (average 71,262 ns) and from 48,173 to 107,025 ns (average 74,490 ns) respectively.

Table 1. Performance of ctrl-VQE in computing the ground state energy of the LiH molecule (Li–H $R=1.5$ Å). The energy error relative to FCI, the leakage to higher energy states, the total pulse duration, and the overlap with the exact FCI state are shown. The data correspond to the pulse with five time segments from Fig. 8.

Method	Energy	Error
Hartree-Fock	−7.8633576	0.0177
ctrl-VQE	−7.8806399	0.0004
FCI	−7.8810157	–
Pulse duration	40 ns	–
Leakage	0.59%	–
Overlap	99.38%	–

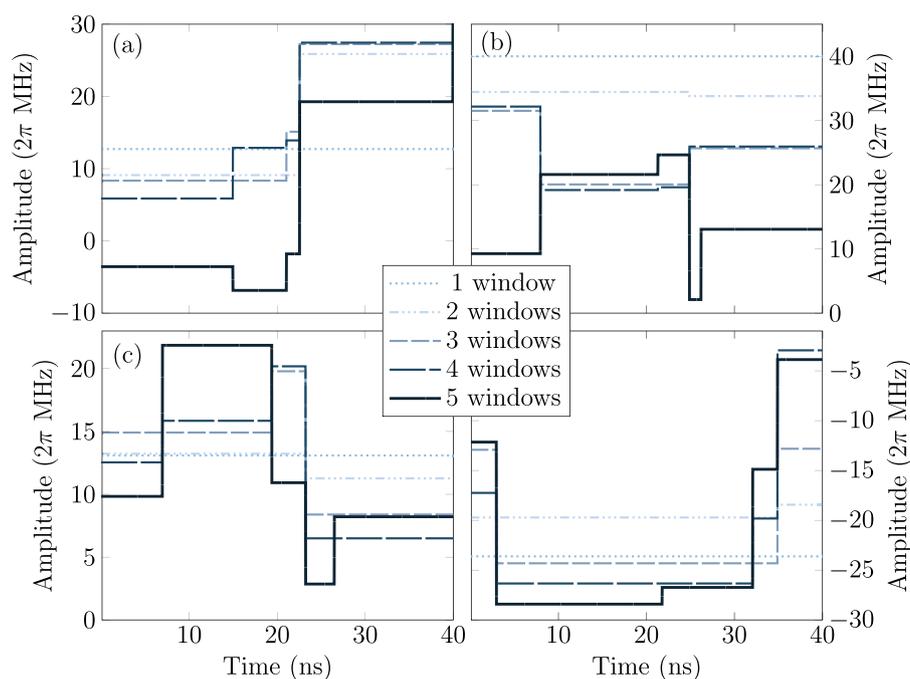


Fig. 9 Adaptive pulse sequences for LiH molecule. Square pulse shapes obtained at each adaptive step, starting from one time window and adaptively dividing the time window up to five-segments. The accuracy increases with each division of the pulse. Pulse shapes are shown for the pulses on **a** qubit 1, **b** qubit 2, **c** qubit 3, **d** qubit 4.

This means that the pulse duration for LiH using the UCCSD ansatz is already reaching the limit set by decoherence times of the device, while the pulse duration obtained with ctrl-VQE is well within the decoherence time.

We note here that this comparison needs to be approached with a bit of caution, as our simulated device has slightly different parameters than the IBM-Q devices used for the circuit timings. However, we do not expect that our results would change significantly if we had access to the full set of exact parameters of the device (including anharmonicities, bus frequencies, and couplings), because previous works using the present device parameter regime^{49,50} have demonstrated universal gate sets with single and two-qubit operations commensurate with those of the IBM-Q devices. As such, we expect the ctrl-VQE pulse times reported in this study to be close to what one would get by directly running the calculations on IBM-Q devices.

Summary and future outlook

In summary, we have presented an alternative new variational quantum algorithm which is fundamentally different from the existing quantum algorithms for molecular simulation. The quantum circuit used for state preparation in standard variational algorithms is entirely replaced by a hardware-level control routine with optimized pulse shapes to directly drive the qubit system. As such, we demonstrate that VQE applications do not need to be confined to two level systems (qubits), which are required for general quantum computing. This opens a possibility of faster ansatz preparation within a time window defined by coherence times of NISQ devices.

The efficacy of the presented method is numerically demonstrated by modeling the bond dissociation of two diatomic molecules, H_2 and HeH^+ . The maximum error from the exact FCI energy is well within chemical accuracy (0.02 kcal/mol) for both molecular systems throughout the bond dissociation. ctrl-VQE captures the important electron correlation effects involved in the bond dissociation of the two molecular systems, as reflected in the pulse durations along the bond dissociation.

To demonstrate the application of ctrl-VQE to a larger molecular system, we have also computed the energy of the LiH molecule at a single bond distance. An adaptive scheme which slowly increases the variational flexibility of the pulse parameterization was used to determine the minimal number of pulse parameters required to achieve a target accuracy, ultimately simplifying experimental implementation and avoiding over-parameterization. The results demonstrate that pulse shapes with relatively modest numbers of parameters achieve convergence in the energy even as the problem size increased. The approach yields significant state-preparation speedups for VQE as compared to standard gate-based ansätze. The short pulse durations minimize losses due to decoherence and dephasing, which is a step toward enabling more accurate VQE simulations on larger, strongly correlated systems. ctrl-VQE can be viewed as a lowest-level quantum variational algorithm.

Because ctrl-VQE operates directly on the hardware, classical simulations of the algorithm are even more computationally expensive than typical VQE emulations, as one needs to not only solve the time-dependent Schrödinger equation, but also optimize the driving Hamiltonian. As such, the systems we have been able to study so far are very small. In future work, we will develop an improved implementation to study the behavior of larger systems and with more sophisticated constraints on the pulse shape ($\Omega_n(t)$). Improved software will also make it possible to numerically study the impact that device controllability limitations⁵¹ have on the amount or type of electron correlation able to be described.

METHOD

Variational quantum eigensolver

The VQE algorithm aims to leverage classical resources to reduce the circuit depth required for molecular simulation. The algorithm finds optimal rotation angles for a parameterized quantum circuit of fixed depth by variationally minimizing the energy of a target molecule, which is obtained by repeated state preparation and measurement cycles. In order to account for the distinguishability of qubits, we start by transforming the

second quantized electronic Hamiltonian into an equivalent form involving non-local strings of Pauli spin operators, \hat{o}_i :

$$\begin{aligned} \hat{H}^{\text{molecule}} &= \sum_{pq} h_{pq} \hat{p}^\dagger \hat{q} + \frac{1}{2} \sum_{pqrs} (pq|rs) \hat{p}^\dagger \hat{q}^\dagger \hat{s} \hat{r} \\ &= \sum_i \hat{o}_i h_i, \end{aligned} \quad (4)$$

where h_i is a sum of molecular one or two-electron integrals, and \hat{p} operators are fermionic annihilation operators. Several such transformations exist, such as the Jordan–Wigner³⁶, Bravyi–Kitaev⁵², or parity transformation⁵³. The main steps in VQE are defined as follows:

1. Precompute all h_i values, and transform terms in the Hamiltonian operator into the desired qubit representation.
2. Choose an ansatz for the problem which defines the variables ($\vec{\theta}$) to optimize. Assuming one starts from the Hartree-Fock (HF) reference state, this involves predefining a parameterized circuit which provides enough variational flexibility to describe the molecule's electron correlation effects. Many ansätze have been proposed, several of which are variants of the original proposal using the unitary coupled-cluster (UCC) ansatz^{54,55}.
3. Choose an initial set of parameter values, $\vec{\theta} = \vec{\theta}_0$. These can be initialized to zero, chosen randomly, or if appropriate, chosen based on some classical precomputation such as using MP2 parameters to start a UCCSD optimization.
4. Using current parameters, $\vec{\theta}$, repeatedly execute the circuit, each time performing an individual measurement of one of the operators, \hat{o}_i , in \hat{H} or a set of mutually commuting operators. After a sufficient number of circuit executions (shots), the averages of the resulting data converge to the expectation values of the operators such that the average molecular energy can be obtained by multiplication with the one and two-electron integrals,

$$E(\vec{\theta}) = \sum_i h_i \langle \psi(\vec{\theta}) | \hat{o}_i | \psi(\vec{\theta}) \rangle. \quad (5)$$

5. Check convergence. If the average energy has decreased by a small enough value determined to be converged, exit. If the energy is not yet converged, update $\vec{\theta}$, and return to step 4.

Various approaches have been proposed, each differing in the details of state-preparation^{18,24,56–59}, and ways to reduce the number of circuits required to compute expectation values^{60–63}.

Control variational quantum eigensolver

In this section, we present an alternative to the gate-based VQE algorithm, replacing the parameterized state-preparation circuit with a parameterized laboratory-frame pulse representation, which is optimized in an analogous manner, but with the benefit of a much faster state preparation, opening up the possibility of more accurate simulations on NISQ devices. All other aspects of VQE (i.e., measurement protocols) are essentially the same. Using the molecular energy as the objective function to minimize, the pulse parameters are optimized using the variational principle. This general strategy, which we refer to as control variational quantum eigensolver (ctrl-VQE), is outlined as follows:

1. As done in any regular VQE, compute the one- and two-electron integrals and transform the molecular Hamiltonian into a qubit representation, for example using Jordan–Wigner, Parity or Bravyi–Kitaev mappings. This defines the objective function to minimize, $\langle \hat{H}^{\text{molecule}} \rangle$.
2. Define a fixed pulse representation (e.g. square pulses, sum of Gaussian pulses, etc.). Parametrize the chosen pulse representation, and initialize parameters.
3. Choose an initial state for the qubit(s) system. HF is a good choice for the molecular problems studied here. Controls are assumed to be in the form of direct drives on each qubit.
4. Measure the expectation value of the objective function $\langle \hat{H}^{\text{molecule}} \rangle$ on the quantum device.
5. Using a classical optimization routine, determine the pulse parameters for the next measurement on the quantum device.
6. Repeat until a target convergence threshold is met. If the chosen parameterized pulse can fully specify the target Hilbert-space, then the optimal pulse finds the minimum energy state.

We note here that the optimization used in this work excludes the total pulse duration. This is fixed throughout the optimization routine, and only the pulse parameters such as the amplitudes or the frequencies are directly optimized. As such, the total pulse time enters the algorithm as a “hyperparameter”, which can be optimized in an outerloop if desired. In the square pulses considered in this work, the time segments are also optimized. Unless stated otherwise, optimal pulses correspond to pulse shapes that are optimal with a given fixed total pulse duration.

Unlike universal quantum computing algorithms, ctrl-VQE occurs at the hardware-level, and any simulation must refer to a specific physical platform. For this work, we choose a well-established transmon platform with the following 1D device Hamiltonian⁴¹:

$$\begin{aligned} \hat{H}_D &= \sum_{k=1}^N \left(\omega_k \hat{a}_k^\dagger \hat{a}_k - \frac{\delta_k}{2} \hat{a}_k^\dagger \hat{a}_k^\dagger \hat{a}_k \hat{a}_k \right) \\ &+ \sum_{\langle kl \rangle} g(\hat{a}_k^\dagger \hat{a}_l + \hat{a}_l^\dagger \hat{a}_k). \end{aligned} \quad (6)$$

where $\langle kl \rangle$ indicates that the summation is over pairs of coupled qubits. Here, we assume that the qubits are arranged in a closed linear chain and have always-on direct couplings, where \hat{a}_k is the bosonic annihilation operator for the k th transmon, and ω_k , δ_k , and g are the resonant frequency, anharmonicity, and constant coupling rate, respectively. Furthermore, each transmon in Eq. (6) formally supports an infinite number of states. However, in our simulations we necessarily approximate this system by a finite number of levels (three unless otherwise stated) per transmon. We tested the accuracy of this by adding more levels and found that the results did not change significantly. The parameters used in this work are chosen to be typical of values found in current superconducting transmons, and are provided in Table 2. We find that our results below do not qualitatively depend on the frequency difference between the qubits, and in the SI we provide a comparison of this current device to one with a larger detuning between the transmons. In order to drive the device, we apply a time-dependent field, with separate controls on each qubit such that within the rotating wave approximation, the control Hamiltonian is expressed as:

$$\hat{H}_C = \sum_{k=1}^N \Omega_k(t) (e^{i\nu_k t} \hat{a}_k + e^{-i\nu_k t} \hat{a}_k^\dagger). \quad (7)$$

where $\Omega_k(t)$ is the real-valued, time-dependent amplitude of the drive, and ν_k is the frequency of the field. The system therefore evolves under the total Hamiltonian:

$$\hat{H} = \hat{H}_D + \hat{H}_C(t, \Omega_n(t), \nu_n). \quad (8)$$

By moving to the interacting frame, the final ansatz has the following form:

$$|\psi^{\text{trial}}(\Omega_n(t), \nu_n)\rangle = \mathcal{T} e^{-i \int_0^T dt \hat{H}_C(t, \Omega_n(t), \nu_n)} |\psi_0\rangle. \quad (9)$$

where $|\psi_0\rangle$ is the VQE reference state (e.g., the HF state), $\{\Omega_n(t), \nu_n\}$ are the variational parameters for the ansatz, T is the total pulse time, and \mathcal{T} is the time-ordering operator. Note that although the control Hamiltonian above only has single-qubit terms, the device itself has inter-qubit couplings (Eq. (6) with strength g), which create an entangling control Hamiltonian in the interacting frame:

$$\hat{\tilde{H}}(t)_C = e^{i\hat{H}_D t} \hat{H}_C(t) e^{-i\hat{H}_D t}. \quad (10)$$

Table 2. Device parameters appearing in Eq. (6) for a four-transmon system.

	Transmon			
	1	2	3	4
ω	4.8080	4.8333	4.9400	4.7960
δ	0.3102	0.2916	0.3302	0.2616
	1 \leftrightarrow 2	2 \leftrightarrow 3	3 \leftrightarrow 4	4 \leftrightarrow 1
g	0.01831	0.02131	0.01931	0.02031

For the simulations of H_2 and HeH^+ , which only involve two transmons, device parameters corresponding to transmons 1 and 2 were used. All units are in 2π GHz.

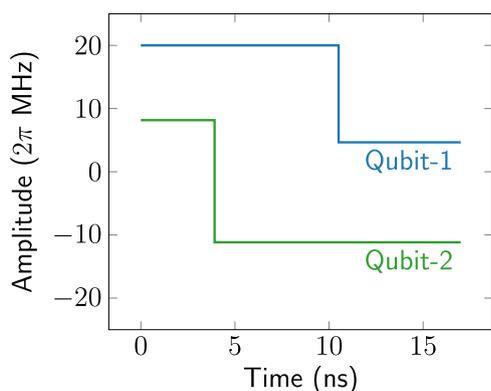


Fig. 10 Illustration of square pulses for a two transmon system. Simple square pulse with two time segments.

As such, the coupling strength g is ultimately responsible for describing electron correlation in the target molecule.

Using this ansatz in Eq. (9), the energy to be minimized in the ctrl-VQE objective function is,

$$E(\Omega_n(t), v_n) = \langle \psi^{\text{trial}} | \hat{H}^{\text{molecule}} | \psi^{\text{trial}} \rangle. \quad (11)$$

where $|\psi^{\text{trial}}\rangle$ is just the $|\psi^{\text{trial}}\rangle$ state above, projected onto the computational basis and normalized. Note that an unnormalized state can also be used; this yields similar results, as shown in the SI.

The ansatz above is completely (and solely) determined by the device and controls, granting enormous flexibility to the ansatz. In fact, any digital quantum circuit ansatz can be compiled into the form above. As such, the ansatz in Eq. (9) does not intrinsically possess any limitation on its potential accuracy beyond the fundamental limitations imposed by quantum speed limits⁴⁰. However, this additional flexibility can make optimization more difficult.

In this work, we have chosen to impose simple constraints on the form of $\Omega_n(t)$ to minimize the number of optimization parameters and simplify experimental implementation. We have considered two examples: (i) piecewise-constant pulses, and (ii) sum of Gaussian pulses. Because these two examples yield similar results, we present only the square pulse data in the main text, and provide the Gaussian pulse data in the SI. An example of a pair of two-time-segment square pulses for a two transmon system is shown in Fig. 10. With a single time segment square pulse, the amplitude is taken as constant throughout the pulse duration. For n time segments, the parameterized square pulse is given by,

$$\Omega_k(t) = \begin{cases} c_1 & 0 \leq t < t_1 \\ c_2 & t_1 \leq t < t_2 \\ \vdots & \\ c_n & t_{n-1} \leq t < T \end{cases}, \quad (12)$$

where c_i are amplitudes of the pulse, t_i are the switching times, and T is the total pulse duration. The amplitudes are constrained to ± 20 MHz unless otherwise mentioned; this is a typical amplitude for RF control signals in superconducting qubit systems⁶⁴. Each transmon drive term also has a frequency modulation of the form $\exp(i v_k t)$, with a driving frequency v_k constrained to $(\omega_k - 2\pi \text{ GHz}) \leq v_k \leq (\omega_k + 2\pi \text{ GHz})$, where ω_k, v_k are in units of 2π GHz. Therefore, the pulse parameters that will be optimized include c_j, t_j , and v_k . With N transmons and n square pulses on each transmon, we then have $2Nn$ parameters to optimize. The pulse parameter optimizations are performed using I-BFGS-b.

We note that a very recent preprint uses similar quantum control considerations to improve VQE⁶⁵. However, in that work, these considerations are used only to define an improved gate-based ansatz, unlike the direct variational pulse shaping described in this work. Our work is also somewhat related to the paper from Chamon and coworkers⁶⁶ in which they find that the optimal pulse for preparing the ground state of a Hamiltonian is of a bang-bang form. Although both ctrl-VQE and the work of Yang et al.⁶⁶ both combine variational quantum algorithms with quantum control, our effort aims to find ground states of non-diagonal Hamiltonians (such as the molecular Hamiltonian in Eq. (4)), preventing the efficient implementation of the time-evolution operator needed for realizing the bang-bang protocol. Because our control Hamiltonian is not

determined by our problem Hamiltonian (Eq. (4)), we should not expect such bang-bang protocols to be optimal. We also note that a preprint was posted soon after this paper which suggests that optimal control would be helpful in VQE applications⁶⁷. While pulse level control presents a challenge for front-end development which has focused on circuits, IBM has already begun making direct pulse access available⁶⁸, and we hope this work will help encourage other platforms to do the same.

Computational details

The numerical results presented in this work were generated with a locally developed program: CtrlQ. The source code for the program is available on Github, codebase: <https://github.com/oimeitei/ctrlq> under a Apache License 2.0. Functionalities from Qiskit³⁸ and Qutip⁶⁹ were used to check and compare the results presented in this work. Molecular integrals were generated using PySCF⁷⁰, and an STO-3G basis set was used throughout. Below, we consider simulations that involve either two-transmon or four-transmon devices depending on the molecule. The parameters ω, δ and g used in the simulations are explicitly given in Table 2. To demonstrate qualitative insensitivity to the device parameters, we provide some results using a device with a larger detuning in the SI.

Although the simulations presented in the following sections use highly restricted forms of control pulses, our in-house code (CtrlQ) supports various forms of pulses. In fact, we have implemented efficient analytic gradients of the molecular energy with respect to pulse amplitudes (see SI), making it possible to optimize pulses with arbitrary shapes. As mentioned in the previous section, here we focus primarily on simple pulse waveforms with few parameters to simplify experimental implementations.

DATA AVAILABILITY

The data for the numerical simulations is available upon reasonable request.

CODE AVAILABILITY

The code for the numerical simulations is available at <https://github.com/mayhallgroup/ctrlq>.

Received: 4 March 2021; Accepted: 28 September 2021;

Published online: 27 October 2021

REFERENCES

- White, S. R. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **69**, 2863–2866 (1992).
- Chan, G. K.-L. & Head-Gordon, M. Highly correlated calculations with a polynomial cost algorithm: A study of the density matrix renormalization group. *J. Chem. Phys.* **116**, 4462–4476 (2002).
- Schollwöck, U. The density-matrix renormalization group in the age of matrix product states. *Ann. Phys.* **326**, 96–192 (2011).
- Huron, B., Malrieu, J. P. & Rancurel, P. Iterative perturbation calculations of ground and excited state energies from multiconfigurational zeroth-order wavefunctions. *J. Chem. Phys.* **58**, 5745 (1973).
- Bender, C. F. & Davidson, E. R. Studies in configuration interaction: The first-row diatomic hydrides. *Phys. Rev.* **183**, 23–30 (1969).
- Buenker, R. J. Ab initio SCF MO and CI studies of the electronic states of butadiene. *J. Chem. Phys.* **49**, 5381 (1968).
- Evangelista, S., Daudey, J.-P. & Malrieu, J.-P. Convergence of an improved CIPSI algorithm. *Chem. Phys.* **75**, 91–102 (1983).
- Tubman, N. M., Lee, J., Takeshita, T. Y., Head-Gordon, M. & Whaley, K. B. A deterministic alternative to the full configuration interaction quantum Monte Carlo method. *J. Chem. Phys.* **145**, 044112 (2016).
- Schriber, J. B. & Evangelista, F. A. Communication: an adaptive configuration interaction approach for strongly correlated electrons with tunable accuracy. *J. Chem. Phys.* **144**, 161106–161106 (2016).
- Holmes, A. A., Tubman, N. M. & Umrigar, C. J. Heat-bath configuration interaction: an efficient selected configuration interaction algorithm inspired by heat-bath sampling. *J. Chem. Theory Comput.* **12**, 3674–3680 (2016).
- Levine, D. S. et al. CASSCF with extremely large active spaces using the adaptive sampling configuration interaction method. *J. Chem. Theory Comput.* **16**, 2340–2354 (2020).
- Abraham, V. & Mayhall, N. J. Selected configuration interaction in a basis of cluster state tensor products. *J. Chem. Theory Comput.* **16**, 6098–6113 (2020).

13. Caffarel, M., Applencourt, T., Giner, E. & Scemama, A. *Recent Progress in Quantum Monte Carlo*, vol. 1234 of *ACS Symposium Series*, Chap. 2, p. 15–46 (American Chemical Society, 2016).
14. Peruzzo, A. et al. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213–4213 (2014).
15. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
16. Sharma, K., Khatri, S., Cerezo, M. & Coles, P. J. Noise resilience of variational quantum compiling. *New J. Phys.* **22**, 043006 (2020).
17. Kandala, A. et al. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**, 242–246 (2017).
18. Huggins, W. J., Lee, J., Baek, U., O’Gorman, B. & Whaley, K. B. A non-orthogonal variational quantum eigensolver. *New J. Phys.* **22**, 073009 (2020).
19. Lee, J., Huggins, W. J., Head-Gordon, M. & Whaley, K. B. Generalized unitary coupled cluster wave functions for quantum computation. *J. Chem. Theory Comput.* **15**, 311–324 (2019).
20. Gard, B. T. et al. Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm. *npj Quant. Inf.* **6**, 1–9 (2020).
21. Barron, G. S. et al. Preserving symmetries for variational quantum eigensolvers in the presence of noise. *Phys. Rev. Applied* **16**, 034002 (2021).
22. Ryabinkin, I. G., Yen, T.-C., Genin, S. N. & Izmaylov, A. F. Qubit coupled cluster method: A systematic approach to quantum chemistry on a quantum computer. *J. Chem. Theory Comput.* **14**, 6317–6326 (2018).
23. Grimsley, H. R., Economou, S. E., Barnes, E. & Mayhall, N. J. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nat. Commun.* **10**, 3007 (2019).
24. Tang, H. L. et al. Qubit-adapt-vqe: an adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor. *PRX Quantum* **2**, 020310 (2021).
25. Ryabinkin, I. G., Lang, R. A., Genin, S. N. & Izmaylov, A. F. Iterative qubit coupled cluster approach with efficient screening of generators. *J. Chem. Theory Comput.* **16**, 1055–1063 (2020).
26. Wang, Y. et al. Quantum simulation of helium hydride cation in a solid-state spin register. *ACS Nano* **9**, 7769–7774 (2015).
27. Shen, Y. et al. Quantum implementation of the unitary coupled cluster for simulating molecular electronic structure. *Phys. Rev. A* **95**, 020501 (2017).
28. Colless, J. I. et al. Computation of molecular spectra on a quantum processor with an error-resilient algorithm. *Phys. Rev. X* **8**, 011021 (2018).
29. McCaskey, A. J. et al. Quantum chemistry as a benchmark for near-term quantum computers. *npj Quant. Inf.* **5**, 1–8 (2019).
30. Xia, R., Bian, T. & Kais, S. Electronic structure calculations and the ising hamiltonian. *J. Phys. Chem. B* **122**, 3384–3395 (2018).
31. Hempel, C. et al. Quantum chemistry calculations on a trapped-ion quantum simulator. *Phys. Rev. X* **8**, 031022 (2018).
32. Sugisaki, K. et al. Quantum chemistry on quantum computers: a method for preparation of multiconfigurational wave functions on quantum computers without performing post-hartree-fock calculations. *ACS Cent. Sci.* **5**, 167–175 (2019).
33. Ritter, M. B. Near-term quantum algorithms for quantum many-body systems. *J. Phys. Conf. Ser.* **1290**, 012003 (2019).
34. Sagastizabal, R. et al. Experimental error mitigation via symmetry verification in a variational quantum eigensolver. *Phys. Rev. A* **100**, 010302 (2019).
35. Armaos, V., Badounas, D. A., Deligiannis, P. & Lianos, K. Computational chemistry on quantum computers. *Appl. Phys. A* **126**, 625 (2020).
36. Jordan, W. P. über das paulische äquivalenzverbot. *Z. Phys.* **47**, 631 (1928).
37. Bravyi, S., Gambetta, J. M., Mezzacapo, A. & Temme, K. Tapering off qubits to simulate fermionic hamiltonians (2017). Preprint at <https://arxiv.org/abs/1701.08213> (2017).
38. Abraham, H. et al. *Qiskit: An Open-source Framework For Quantum Computing* (Zenodo, 2019).
39. Güsten, R. et al. Astrophysical detection of the helium hydride ion HeH⁺. *Nature* **568**, 357–359 (2019).
40. Deffner, S. & Campbell, S. Quantum speed limits: from heisenberg’s uncertainty principle to optimal quantum control. *J. Phys. A Math. Theor.* **50**, 453001 (2017).
41. Koch, J. et al. Charge-insensitive qubit design derived from the Cooper pair box. *Phys. Rev. A* **76**, 042319 (2007).
42. Khaneja, N. et al. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. *J. Magn. Reson.* **172**, 296–305 (2005).
43. Gokhale, P. et al. Partial compilation of variational algorithms for noisy intermediate-scale quantum machines. *MICRO ’52: Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 266. <https://doi.org/10.1145/3352460.3358313> (2019).
44. Boutin, S., Andersen, C. K., Venkatraman, J., Ferris, A. J. & Blais, A. Resonator reset in circuit qed by optimal control for large open quantum systems. *Phys. Rev. A* **96**, 042315 (2017).
45. Lu, D. et al. Enhancing quantum control by bootstrapping a quantum processor of 12 qubits. *npj Quant. Inf.* **3**, 45 (2017).
46. Gradl, T., Spörl, A., Huckle, T., Glaser, S. J. & Schulte-Herbrüggen, T. *Euro-Par 2006 Parallel Processing* (eds Nagel, W. E., Walter, W. V. & Lehner, W.) p. 751–762 (Springer Berlin Heidelberg, 2006).
47. Cheng, J., Deng, H. & Qia, X. Accqoc: Accelerating quantum optimal control based pulse generation. In *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*, p. 543–555 (IEEE, 2020).
48. Williams, C. P. *Explorations in Quantum Computing* (Springer, 1998).
49. Economou, S. E. & Barnes, E. Analytical approach to swift nonleaky entangling gates in superconducting qubits. *Phys. Rev. B* **91**, 161405 (2015).
50. Barron, G. S., Calderon-Vargas, F. A., Long, J., Pappas, D. P. & Economou, S. E. Microwave-based arbitrary cphase gates for transmon qubits. *Phys. Rev. B* **101**, 054508 (2020).
51. Ball, H. et al. Software tools for quantum control: Improving quantum computer performance through noise and error suppression. *Quantum Sci. Technol.* **6**, 044011 (2021).
52. Bravyi, S. B. & Kitaev, A. Y. Fermionic quantum computation. *Ann. Phys.* **298**, 210 (2002).
53. Seeley, J. T., Richard, M. J. & Love, P. J. The bravyi-kitaev transformation for quantum computation of electronic structure. *J. Chem. Phys.* **137**, 224109 (2012).
54. Bartlett, R. J. & Musial, M. Coupled-cluster theory in quantum chemistry. *Rev. Mod. Phys.* **79**, 291–352 (2007).
55. Yung, M.-H. et al. From transistor to trapped-ion computers for quantum chemistry. *Sci. Rep.* **4**, 3589 (2014).
56. Ostaszewski, M., Grant, E. & Benedetti, M. Quantum circuit structure learning. *arXiv:1905.09692 [quant-ph]* (2019).
57. Chivilkhin, D. et al. MoG-VQE: multiobjective genetic variational quantum eigensolver. *arXiv:2007.04424 [cond-mat, physics:quant-ph]* (2020).
58. Matsuzawa, Y. & Kurashige, Y. Jastrow-type decomposition in quantum chemistry for low-depth quantum circuits. *J. Chem. Theory Comput.* **16**, 944–952 (2020).
59. Lee, J., Huggins, W. J., Head-Gordon, M. & Whaley, K. B. Generalized unitary coupled cluster wavefunctions for quantum computation. *J. Chem. Theory Comput.* **15**, 311 (2019).
60. Babbush, R. et al. Low-depth quantum simulation of materials. *Phys. Rev. X* **8**, 011044 (2018).
61. Verteletskyi, V., Yen, T.-C. & Izmaylov, A. F. Measurement optimization in the variational quantum eigensolver using a minimum clique cover. *J. Chem. Phys.* **152**, 124114 (2020).
62. Huggins, W. J. et al. Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers. *arXiv:1907.13117 [physics, physics:quant-ph]* (2019).
63. Zhao, A. et al. Measurement reduction in variational quantum algorithms. *Phys. Rev. A* **101**, 062322 (2020).
64. Ku, H. S. et al. Single qubit operations using microwave hyperbolic secant pulses. *Phys. Rev. A* **96**, 042339 (2017).
65. Choquette, A. et al. Quantum-optimal-control-inspired ansatz for variational quantum algorithms. *arXiv:2008.01098 [quant-ph]* (2020).
66. Yang, Z.-C., Rahmani, A., Shabani, A., Neven, H. & Chamon, C. Optimizing variational quantum algorithms using ponyragin’s minimum principle. *Phys. Rev. X* **7**, 021027 (2017).
67. Magann, A. B. et al. From pulses to circuits and back again: a quantum optimal control perspective on variational quantum algorithms. *PRX Quant.* **2**, 010101 (2021).
68. Alexander, T. et al. Qiskit pulse: programming quantum computers through the cloud with pulses. *Quant. Sci. Technol.* **5**, 044006 (2020).
69. Johansson, J. R., Nation, P. D. & Nori, F. Qutip 2: A python framework for the dynamics of open quantum systems. *Comput. Phys. Commun.* **184**, 1234–1240 (2013).
70. Sun, Q. et al. Pyscf: the python-based simulations of chemistry framework. *WIREs Comput. Mol. Sci.* **8**, e1340 (2018).

ACKNOWLEDGEMENTS

N.J.M., S.E.E., E.B. and D.P.P. are grateful for financial support provided by the U.S. Department of Energy (Award No. DE-SC0019199). S.E.E. acknowledges support provided by the U.S. Department of Energy (Award No. DE-SC0019318). This work was done in part while N.J.M. was visiting the Simons Institute for the Theory of Computing. The authors thank the Advanced Research Computing at Virginia Tech for the computational infrastructure.

AUTHOR CONTRIBUTIONS

N.J.M., S.E.E., E.B. and D.P.P. conceived the project. O.R.M. wrote the python/C++ code and helped design the adaptive pulse parameterization algorithm. B.T.G. wrote mathematica code. B.T.G. and O.R.M performed numerical experiments. All authors contributed to the conceptual analysis of the results and to ideas that refined the algorithm. All authors wrote the paper. O.R.M. and B.T.G. contributed equally.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41534-021-00493-0>.

Correspondence and requests for materials should be addressed to Nicholas J. Mayhall.

Reprints and permission information is available at <http://www.nature.com/reprints>

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



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2021