Comparison of Unitary Coupled Cluster Ansatz Methods for the Variational Quantum Eigensolver

Ethan Hickman (ethanh@umd.edu), Aaron M. Roth, Yingyue Zhu

University of Maryland
CMSC 657
December 12, 2019

Abstract

The variational quantum eigensolver (VQE) is a hybrid quantum-classical algorithm that can be used to study the eigenvalues and corresponding eigenvectors of a physical system. In this work, we review and provide reference to key background in quantum chemistry for each step of the VQE algorithm, as well as demonstrate the effectiveness of the algorithm on a sparse matrix simulator. With the simulator, we compare several chemically motivated ansätze: generalized unitary coupled cluster singles and doubles (UCCGSD), singlet unitary coupled cluster singles and doubles (UCCSD), and a modified version of k unitary pair coupled cluster (k-UpCCSD) which employs singlet single excitations and sparse singlet pair double excitations.

1 Introduction

Current quantum devices are too small and noisy to reliably execute deep circuits or to employ error correction schemes that provide enough logical qubits for useful computations. As we approach the NISQ era in quantum computing, it will be possible to take advantage of limited quantum resources by strategically delegating classically intractable tasks in an algorithm to a quantum coprocessor. The variational quantum eigensolver (VQE), first proposed and demonstrated on a photonic quantum coprocessor in 2014 [1], is one such quantum-classical hybrid algorithm that was developed with this paradigm in mind.

Our goal in this project was to reach the level of familiarity with the VQE algorithm required to build a software pipeline that simulates VQE experiments in order to compare the effectiveness of multiple ansätze in computing the ground state energy of multiple small molecules. We succeeded in doing this using several open source software packages for quantum chemistry: `OpenFermion` [2], `PySCF` [3], and `OpenFermion-PySCF`. The code for our project is available at `https://github.com/eth-n/variational-quantum-eigensolver`.

2 Related Work

In early 2014, the VQE method was proposed and shown to work on a real-world photonic quantum coprocessor, where it was used to calculate the ground-state molecular energy for HeH^+ [1]. The main advantage of the algorithm is that it trades off the extensive coherence time requirements of quantum phase estimation (QPE) for many preparations of an ansatz state, each requiring and $O(1)$ coherence time in the size of the system.

Since then, there has been extensive interest in the VQE as a hybrid quantum-classical algorithm because it can take advantage of small and relatively noisy quantum resources to perform intractable computation. In the latter half of 2015, some of the original authors of the VQE paper wrote another paper to "extend the general theory of [the VQE] and suggest algorithmic improvements for practical implementations" [4]. This has been an indispensable resource both for understanding the VQE itself and as a source of guidance on how to implement the algorithm as a whole. In early 2019, McArdle et al wrote a review of quantum computational chemistry, covering a large amount of background material as well as the QPE and VQE algorithms [5]. These are two information-dense papers among a large body of research into advancing and improving the algorithm since it was proposed.

3 Background and Methodology

This section focuses on condensing relevant background for each of the steps of the algorithm and information that motivated our implementation decisions.

3.1 Fundamental background

There are several preliminary facts and conditions that predicate the VQE, and these are covered in greater detail in [4] and [5].

The system of interest should have a Hamiltonian that can be simulated with at most the number of qubits available. As the size and fidelity of quantum computers increase, we will have access to larger systems and obtain results faster and with higher accuracy.

The Hamiltonian must be decomposable into at most a polynomial number of simple operators so that the expectation value of the Hamiltonian can be measured efficiently. Many physical systems do meet this criteria.

The variational principle of quantum mechanics states that the average measured value of an observable with respect to a state is at least the observable operator's minimum eigenvalue.

$$\frac{\langle\psi|\hat{O}|\psi\rangle}{\langle\psi|\psi\rangle} \geq \lambda_1 \tag{1}$$

With $|\psi\rangle$ normalized, the equation simplifies to

$$\langle\psi|\hat{O}|\psi\rangle \geq \lambda_1 \tag{2}$$

In general, a parameterized ansatz wavefunction will be in a superposition of eigenstates of the Hamiltonian. Therefore the expected value of measurement of the Hamiltonian on the prepared ansatz will yield an average energy greater than or equal to the ground state energy of the system.

$$\langle\psi(\vec{\theta})|\hat{H}|\psi(\vec{\theta})\rangle \geq E_0 \tag{3}$$

This is the fundamental physical principle that gives the lower bound we are optimizing towards in the VQE.

3.2 The algorithm

The parameterized state preparation method is called an ansatz. The type of ansatz one chooses defines the family of states that the parameters can generate, as well as the number of experimental parameters that must be tuned. We would ideally like the ansatz to have as much flexibility in representation as possible, while using the fewest parameters possible an maximizing the likelihood of converging to a minimum in expected energy close to the global optimum.

At a high level, the VQE algorithm has three steps in its core loop that are repeated until convergence. As described in [4],

1. Prepare the ansatz state $|\psi(\vec{\theta})\rangle$ on the quantum computer where $\vec{\theta}$ are tunable parameters.
2. Measure the expectation value of H on the prepared ansatz.
3. Update $\vec{\theta}$ via classical optimizer to decrease the expectation value of H.

If the optimization converges to the global minimum, then the parameters $\vec{\theta}$ will prepare the ground state of the system of interest. The minimum expectation value of H with respect to the choice of sufficiently powerful ansatz will be the ground state energy of the system. Notably, the VQE can be modified to find excited states and energies of the system of interest as well [6].

3.3 Obtain the Coulomb Hamiltonian

These are the five terms of the molecular Hamiltonian. They are expressed in SI units followed by atomic units, which are used to reduce the number of constants floating around in the equations and calculations.

Kinetic energy of each nucleus:

$$\hat{T}_n = -\sum_i \frac{\hbar^2}{2M_i}\nabla^2_{\mathbf{R_i}} \implies -\sum_i \frac{\nabla^2_{\mathbf{R_i}}}{2M_i} \tag{4}$$

Kinetic energy of each electron:

$$\hat{T}_e = -\sum_i \frac{\hbar^2}{2m_e}\nabla^2_{\mathbf{r_i}} \implies -\sum_i \frac{\nabla^2_{\mathbf{r_i}}}{2} \tag{5}$$

Potential energy between electrons and nuclei:

$$\hat{U}_{en} = -\sum_i\sum_j \frac{Z_i e^2}{4\pi\epsilon_0|\mathbf{R_i}-\mathbf{r_j}|} \implies -\sum_i\sum_j \frac{Z_i}{|\mathbf{R_i}-\mathbf{r_j}|} \tag{6}$$

Potential energy due to electron-electron repulsion:

$$\hat{U}_{ee} = \frac{1}{2}\sum_i\sum_{j\neq i} \frac{e^2}{4\pi\epsilon_0|\mathbf{r_i}-\mathbf{r_j}|} \implies \frac{1}{2}\sum_i\sum_{j\neq i} \frac{1}{|\mathbf{r_i}-\mathbf{r_j}|} \tag{7}$$

Potential energy due to nuclei-nuclei repulsion:

$$\hat{U}_{nn} = \frac{1}{2}\sum_i\sum_{j\neq i} \frac{Z_i Z_j e^2}{4\pi\epsilon_0|\mathbf{R_i}-\mathbf{R_j}|} \implies \frac{1}{2}\sum_i\sum_{j\neq i} \frac{Z_i Z_j}{|\mathbf{R_i}-\mathbf{R_j}|} \tag{8}$$

The molecular Hamiltonian is the sum of these five terms:

$$\hat{H} = \hat{T}_n + \hat{T}_e + \hat{U}_{en} + \hat{U}_{ee} + \hat{U}_{nn} \tag{9}$$

3.4 Coulomb Hamiltonian under the Born-Oppenheimer approximation

Under the Born-Oppenheimer approximation, we elect to treat the motion of the electrons separately from that of the nuclei. This reduces the electronic Hamiltonian to the sum of only three of the five terms in the Coulomb Hamiltonian: the kinetic energy of each electron, the potential energy between each electrons and each nucleus, and the potential energy arising from electron-electron repulsion. The problem then is to solve for the eigenvectors and eigenvalues of the new Hamiltonian,

$$\hat{H} = -\sum_i \frac{\nabla^2_{\mathbf{r_i}}}{2} - \sum_i\sum_j \frac{Z_i}{|\mathbf{R_i}-\mathbf{r_j}|} + \frac{1}{2}\sum_i\sum_{j\neq i} \frac{1}{|\mathbf{r_i}-\mathbf{r_j}|} \tag{10}$$

where m_e is the mass of the electron, Z_i is the number of protons in nucleus i, e is the magnitude of the charge of a proton or electron, $\mathbf{R}_n$ is the displacement of nucleus n, and $\mathbf{r}_n$ is the displacement of electron n.

3.5 Chemical basis sets

Chemical basis sets are finite sets of functions used to represent the atomic orbitals of a molecule on a computer [7]. The basis functions are taken as linear combinations to represent molecular orbitals. The choice of basis set to use for a given molecule has important implications for the VQE algorithm.

There is a trade-off between the number of functions in a basis set and the accuracy of the representation of the atomic (and therefore molecular) orbitals. A minimal basis contains exactly enough functions to represent orbitals from 1s up to the highest energy orbitals likely relevant to the system of interest. Extended basis sets use multiple functions per atomic orbital, which allows for more flexibility in representation of the molecule, and in turn more accurate results.

Perhaps the most important consideration when choosing a basis set is analyzing the requirements of your application. When representing the occupation of spatial orbitals on a quantum computer, because each orbital can contain a spin up and/or a spin down electron, the number of qubits required is twice the number of basis functions. This number is called the number of spin orbitals, because it accounts for both the spin up and spin down occupation of the spatial orbitals.

For demonstration purposes, and when constrained to a classical simulator, it is reasonable to use a minimal basis to represent the molecules we worked with. We use a minimal Slater type orbital, 3 Gaussian basis (STO-3G basis) in our experiments, the same basis used in the introductory paper.

There are several open source software packages that can generate and manipulate a plethora of basis sets and compute the relevant coefficients. Two that we encountered during this project are `PySCF` and `Psi4` [8].

3.6 Second quantization

Using [9] as a reference, here is a brief summary of the key points of this elegant representation of the wavefunction of a many-body system.

First, define a vacuum state $|\Omega\rangle$ to represent the unique state with no particles in it. Then define a set of annihilation operators a_λ and their adjoints, the creation operators $a_\lambda^\dagger$. An annihilation operator applied to a state removes a particle from the state, and a creation operator applied to a state adds a particle to that state.

Now we can define the general many-body or Fock space, $\mathcal{F}$, the direct sum of all N particle states for $N \in \mathbb{N}$.

$$\mathcal{F}_N = \frac{1}{\sqrt{\prod_\lambda n_\lambda}} a_N^\dagger \cdots a_1^\dagger \, |\Omega\rangle \tag{11}$$

$$\mathcal{F} = \bigoplus_{N=0}^{\infty} \mathcal{F}_N \tag{12}$$

To respect the symmetry of the wavefunction being represented, the creation and annihilation operators for bosons (fermions) obey (anti)commutation relations. When considering electronic Hamiltonians, the particles of interest are the electrons, which naturally obey the standard fermion anticommutation relations:

$$[a_\lambda, a_\mu^\dagger] = a_\lambda a_\mu^\dagger + a_\lambda^\dagger a_\mu = \delta_{\lambda,\mu} \tag{13}$$

$$[a_\lambda^\dagger, a_\mu^\dagger] = a_\lambda^\dagger a_\mu^\dagger + a_\lambda^\dagger a_\mu^\dagger = 0 \tag{14}$$

$$[a_\lambda, a_\mu] = a_\lambda a_\mu + a_\lambda a_\mu = 0 \tag{15}$$

These operators can be combined to represent one- and two-body operators, but for now we will focus on groupings that represent single and double excitations and deexcitations.

A single excitation or deexcitation operator will have one creation operator and one annihilation operator, like so:

$$a_\mu^\dagger a_\nu \tag{16}$$

Rather than thinking of this pair of operators as destroying and creating a particle, helpful intuition is that they effectively scatter a particle from state ν to state μ.

Similarly, a double excitation or deexcitation operator will have two creation operators and two annihilation operators, and can be thought of as scattering two particles to two different states:

$$a_\mu^\dagger a_{\mu'}^\dagger a_\nu a_{\nu'} \tag{17}$$

In general, to represent arbitrary operators, these single and double excitations can be taken in linear combinations, with coefficients defining the probabilities of excitation terms occurring. We defer to [9] for a thorough explanation of representing arbitrary operators in second quantized form.

3.7 Second quantization of the Hamiltonian

Now we are equipped to formulate the Hamiltonian for the system of interest in terms of creation and annihilation operations acting on a state representing the fermion occupation of the spin orbitals. Because the three terms in the Coulomb Hamiltonian under the Born-Oppenheimer approximation are all one- and two-body operators, the Hamiltonian can be expressed as a linear combination of single and double excitation operators acting on an occupation state of N fermions in the spin-orbital basis. The second quantization of the Hamiltonian for a fermionic system takes the form

$$\hat{H} = \sum_{mn} h_{mn} a_m^\dagger a_n + \sum_{mnpq} h_{mnpq} a_m^\dagger a_n^\dagger a_p a_q \tag{18}$$

with fermion creation operators $a_i^\dagger$ and annihilation operators a_i and with the one- and two-electron interaction coefficients determined by the integrals

$$h_{mn} = \int d\sigma \; \varphi_m^*(\sigma) \left(-\frac{\nabla_{\mathbf{r}}^2}{2} - \sum_i \frac{Z_i}{|\mathbf{R}_i - \mathbf{r}|} \right) \varphi_n(\sigma) \tag{19}$$

$$h_{mnpq} = \int d\sigma_1\ d\sigma_2\ \varphi^*_m(\sigma_1)\varphi^*_n(\sigma_2)\left(\frac{1}{2|\mathbf{r}_1 - \mathbf{r}_2|}\right)\varphi_p(\sigma_1)\varphi_q(\sigma_2) \tag{20}$$

In equations (18), (19), and (20), the indices run over an enumeration of the basis functions. The coefficients of the single excitations are the expectation of the one-body terms of the Hamiltonian and the coefficients of the double excitations are the expectation of the two-body term of the Hamiltonian, each with respect to the spatial orbitals the operators are acting on. Note that the 2 in the denominator of the two body integral can be pulled outside of the four-indexed sum in $\hat{H}$.

When using `PySCF`, after building a molecule, one can access the one- and two-electron integrals via the `mol.intor()` method. When using `OpenFermion-PySCF`, `OpenFermion`'s `MolecularData` class, one way to populate the one- and two-body integrals is by running a self-consistent field (SCF) calculation. The coefficients h_{mn} are the sum of the elements at the same index in the kinetic and potential energy tensors (the one-body integrals), which are each shape $(nbf,\ nbf)$, where nbf is the number of basis functions. The electron repulsion integral (ERI) tensor contains the coefficients h_{mnpq} (calculated by the two-body integrals), and is represented by a four dimensional array with shape $(nbf,\ nbf,\ nbf,\ nbf)$. These coefficients are calculated for the atomic orbitals and must be changed to the molecular orbital basis. This is also handled either by `PySCF` or internally in `OpenFermion`.

The same calculations can be performed in `Psi4`. When performing a Hartree-Fock self-consistent field theory energy calculation, the `MintsHelper` (used to compute many types of integral within `Psi4`) can be queried for single electron kinetic energy (`ao_kinetic`), single electron potential energy (`ao_potential`), and electron repulsion integral (ERI) (`ao_eri`) tensors. These coefficients must be transformed from the ao (atomic orbital) basis to the mo (molecular orbital basis, which is also possible in `Psi4`.

3.8 Transform the Hamiltonian to qubit space

There are two major transformations from the second quantized form of an operator to an operator that acts on qubits. These are the Jordan-Wigner and Bravyi-Kitaev [10] transforms. There are many modifications to these algorithms as well, but we did not focus on profiling them.

The Jordan-Wigner transformation from the second quantized Hamiltonian to an operator that acts on qubits representing the system is given by defining a transform for creation and annihilation operators. It is a no-frills transformation and is relatively simple to implement and understand. It is given in [4] as

$$a_p^\dagger = \left(\prod_{m<p} \sigma_m^z \right) \sigma_p^+ \tag{21}$$

$$a_p^\dagger = \left(\prod_{m<p} \sigma_m^z \right) \sigma_p^- \tag{22}$$

$$\sigma^\pm \equiv \frac{\sigma^x \mp i\sigma^y}{2} \tag{23}$$

where σ^x, σ^y, σ^z are the standard Pauli X, Y, and Z matrices respectively.

The Bravyi-Kitaev transform has improved asymptotic scaling in terms of the number of gates required when compared to the Jordan-Wigner transform. It requires $O(\log m)$ qubit operations, as opposed to Jordan-Wigner's $O(m)$ qubit operations, per fermionic operation.

The Jordan-Wigner and Bravyi-Kitaev transforms, as well as several of the modifications of them, are implemented in `OpenFermion`. We found, surprisingly, that the Jordan-Wigner transformation was faster to simulate (though it would require more gates on an actual quantum device). It's possible that's due to the more regular tensor product structure of the Jordan-Wigner transformation being advantageous in the state-vector simulator.

Beyond simulation, the savings that can be made in qubit operations, especially when investigating larger systems, make the Bravyi-Kitaev family of transforms preferable over Jordan-Wigner transforms [11].

3.9 Develop an ansatz

There are several types of ansatz that have been explored in the VQE literature. The three major families, hardware efficient ansätze, chemically inspired ansätze, and adiabatic state preparation, are reviewed in [5]. Chemically inspired methods and adiabatic preparation are discussed in [4].

We elected to investigate a class of chemically motivated ansätze known as unitary coupled cluster (UCC) methods. [12] summarizes the development of UCC from classical CC methods.

First, define a reference state that represents an initial state of the system with the required number of electrons in it. Remember we are adding particles to a state with respect to the spin-orbitals of a basis set which we will say has M functions.

$$|\psi_{ref}\rangle = a_N \cdots a_1 |\Omega\rangle \tag{24}$$

Classical coupled cluster (CC) methods traditionally consider single and double excitations from occupied to virtual spin orbitals in $|\psi_{ref}\rangle$, where parameters of the ansatz are coefficients of the excitations.

$$T = \sum_{i=1}^{k} T_i \tag{25}$$

$$T_1 = \sum_{\substack{\alpha \in virtual \\ i \in occupied}} t_i^{\alpha} a_{\alpha}^{\dagger} a_i \tag{26}$$

$$T_2 = \sum_{\substack{\alpha,\beta \in virtual \\ i,j \in occupied}} t_{ij}^{\alpha\beta} a_{\alpha}^{\dagger} a_{\beta}^{\dagger} a_i a_j \tag{27}$$

where single and double excitations would limit k to 2, and thus

$$\theta = \left\{ t_i^{\alpha}, t_{ij}^{\alpha\beta} \right\}_{i,j,\alpha,\beta} \tag{28}$$

$$|\psi(\vec{\theta})\rangle = e^{T} |\psi_{ref}\rangle \tag{29}$$

Unfortunately, CC approaches become non-variational when breaking bonds or when there is significant correlation in the system and, classically, avoiding this non-variational breakdown requires cost exponential in system size [12]. The non-variational breakdown can be avoided by using a unitary cluster operator,

$$|\psi(\vec{\theta})\rangle = e^{T-T^{\dagger}} |\psi_{ref}\rangle \tag{30}$$

Now factors determining the power and structure of the ansatz are defined by the structure of T.

If T is developed from a single reference state considering only single and double excitations, it is known as singlet UCCSD.

$$T = \sum_{\substack{\alpha \in virtual \\ i \in occupied}} t_i^{\alpha} a_{\alpha}^{\dagger} a_i + \sum_{\substack{\alpha,\beta \in virtual \\ i,j \in occupied}} t_{ij}^{\alpha\beta} a_{\alpha}^{\dagger} a_{\beta}^{\dagger} a_i a_j \tag{31}$$

If T is instead developed with independent parameterization of every possible single and double excitation (respecting the spin of the starting and ending spin-orbital), the cluster operator is known as generalized UCCSD.

$$T = \sum_{m,n} t_n^m a_m^{\dagger} a_n + \sum_{m,n,p,q} t_{pq}^{mn} a_m^{\dagger} a_n^{\dagger} a_p a_q \tag{32}$$

The third method we investigated was one similar to the k unitary pair coupled cluster singles and doubles (k-UpCCSD) ansatz introduced by [12]. In particular, the ansatz we implemented considers generalized singles excitations and only the singlet doubles excitations that move a pair of spin up and spin down electrons from an occupied spatial orbital to a virtual spatial orbital.

This is the T which generates singlet pair coupled cluster:

$$T = \sum_{\substack{\alpha \in virtual \\ i \in occupied}} t_i^\alpha a_\alpha^\dagger a_i + \sum_{\substack{\alpha \in virtual \\ i \in occupied}} t_i^\alpha a_{\alpha\uparrow}^\dagger a_{\alpha\downarrow}^\dagger a_{i\uparrow} a_{i\downarrow} \tag{33}$$

The extension of this operator to k products of the operator yielded more accurate results in [12], and is as follows,

$$|\psi(\vec{\theta})\rangle = \left(\prod_{i=1}^{k} e^{T^{(i)} - T^{(i)\dagger}} \right) |\psi_{ref}\rangle \tag{34}$$

where each $T^{(i)}$ has an independent set of parameters

$$\vec{\theta} = \bigcup_i \vec{\theta_i} \tag{35}$$

Another change that [12] found beneficial was generalizing the excitations in k-UpCCSD, calling the new cluster operator k-UpCCGSD.

In general, UCC operators are composed of a sum of a polynomial number of parameterized excitations that can be broken up into some number of non-commuting subsets of terms. There is a large body of research into variations on the UCC method for electronic structure problems [5].

3.10 The quantum circuit

This step is where there are significant differences between simulating the algorithm and running it on a real quantum device. As shown above, once the cluster operator T is transformed to a qubit operator, the qubit operator ansatz must be exponentiated. While this can be done numerically to be used in a matrix simulator, in order to apply the ansatz to qubits, the operation must be Trotterized. [4] emphasizes that the Trotterization is more of a redefinition of the ansatz rather than an approximation of it.

The bigger difference is in how we take the expectation value. In the state-vector simulator, the expectation value can be taken exactly given the Hamiltonian matrix and ansatz state.

On a real quantum device, the process is not so simple. For each term in the qubit Hamiltonian, one must prepare the ansatz state from $|0 \cdots 0\rangle$, apply the Pauli matrices of the term's excitation operators and measure. The preparation and measurement has to be repeated until an acceptable variance is achieved, and then that term's expectation can weighted by its coefficient in the Hamiltonian and added to a sum over terms. An analysis of the variance can be found in [4]. This must be repeated every time the expectation value of energy is needed in the optimization loop.

This is an argument for adiabatic state preparation or hardware ansätze. If the circuit is not fixed, the overhead of recompiling and running may simply take too long (practically speaking). In a hardware ansatz, the order of gates of the circuit are fixed, but would be parameterized by rotation angle, for example.

3.11 Classical optimization of ansatz parameters

The selection of classical optimization method has enormous impact on the feasibility of optimizing the ansatz parameters. It approximately determines (due to the stochasticity of measurement) the number of function evaluations required for convergence. If the classical optimization has difficulty converging to a value within chemical accuracy of the global minimum, then the algorithm is essentially defeated.

Nelder-Mead (NM) was used in the introductory paper for the VQE, and rejected noise from their photonic quantum processor well [1]. Several methods in TOMLAB (a MATLAB library for optimization) were shown to require orders of magnitude fewer iterations to converge [4]. A short review of several other experiments focused on evaluating optimization methods is included in [5].

The Nelder-Mead method [13] is a numerical optimization method that uses heuristics and simplices. An alternative to Nelder-Mead is the Broyden-Fletcher-Goldfarb-Shanno method (BFGS) [14], which estimates the gradient. The L-BFGS-B method is a variant of the BFGS method that limits the memory used by the optimizer. L-BFGS-B was shown to be the best of several methods tested numerically in [15]. It's also available in Scipy, so we were able to easily integrate the method into a pipeline with the other open source packages.

4 Results

Unless you have direct access to a real quantum device, simulation is the most reasonable way to demonstrate VQE experiments. This is because of the overhead state preparation, and each term, or at least each commuting subset, of the Hamiltonian has to be averaged separately in each iteration. In the time frame of our project, we were able to construct a software pipeline that, given a molecule specification, chemical basis set, choice of ansatz, and optimization parameters, runs the VQE for multiple steps in bond length to produce bond dissociation curves. In particular, we demonstrated the effectiveness of the algorithm on a noiseless sparse-matrix simulator for the hydrogen molecule (H_2), helium hydride (HeH^+), lithium hydride (LiH), and for stretching one of the OH bonds in a water molecule (H_2O) without changing the bond angle.

When we began this project, we were unsure exactly what resources we would require to simulate the VQE for different molecules and basis sets. We had initially hoped to show the VQE applied to Li^2, but found that because the 2s and 2p orbitals are close in energy, each lithium would contribute five spatial orbitals even in a minimal basis. That would require twenty qubits, and to represent the (moderately sparse) Hamiltonian requires hundreds of GB of memory. Table 1 shows resource requirements for the STO-3G basis and projected requirements correlation-consistent polarized valence only double zeta (cc-pVDZ) basis, a more exact representation of the orbitals than a minimal basis.

	#e$^-$	# qubits (STO-3G)	# qubits (cc-pVDZ)
H_2	2	4	20
HeH^+	2	4	20
LiH	4	12	38
H_2O	10	14	48

Table 1: System size vs quantum resource requirements

One of the critical metrics to track is the number of parameters in the system, which will correlate with the difficulty of the optimization. Table 2 contains the number of parameters for the three UCC ansatz methods we tried for the minimal STO-3G basis, and table 3 contains a projection of the number of parameters for the ansätze applied to the cc-pVDZ basis. It becomes obvious that the fully generalized UCCSD has more parameters ($nbf^2 + nbf^4$, where nbf is number of basis functions) than is reasonable to expect to optimize for, even for relatively small molecules.

	# params UCCGSD	# params UCCSD	# params per k, k-UpCCSD
H_2	272	2	2
HeH^+	272	2	2
LiH	20880	44	16
H_2O	38612	65	20

Table 2: Number of parameters for each ansatz in the STO-3G basis

	# params UCCGSD	# params UCCSD	# params per k, k-UpCCSD
H_2	160400	54	18
HeH^+	160400	54	18
LiH	2086580	629	68
H_2O	5310720	4655	190

Table 3: Number of parameters for each ansatz in the cc-pVDZ basis

In Appendix A, we compiled pairs of graphs for each ansatz that we could apply to the four molecules. All calculations interpreted below were carried out with a minimal STO-3G basis set. The Hartree-Fock (HF) energy (orange curve with circle markers) and full configuration interaction (FCI) energy (solid green curve) are calculated with `PySCF`. Our goal is to match the VQE (blue dotted-dashed curve with 'x' markers)to the FCI energy, or to at least be closer to the FCI than the HF energy is.

For both H^2 and HeH^+, it was feasible to apply UCCGSD, singlet UCCSD, and k-UpCCSD with $k = 1$ and $k = 3$. In all four cases, the VQE matched the FCI calculation within $5E - 11ha$; essentially 0.

For LiH, the UCCGSD and UCCSD optimizations were running too slowly to obtain results. The k-UpCCSD did converge however, and the data show that with $k = 1$ the VQE is more accurate than the HF calculation, and even more so with $k = 2$. Neither is within $1E - 4ha$, the general threshold for chemical accuracy. A longer-running simulation with higher k would likely be increasingly accurate.

For H_2O, each optimization step of $k - UpCCSD$ takes hundreds of times more wall time to execute because of the exponential bump in the number of elements in each matrix that is generated each step. For context, each data point for the water molecule was taking roughly 9000 seconds to generate. Clearly, the VQE is still outperforming the HF calculation, but for $k = 1$ is again not within chemical accuracy of the FCI calculation.

5 Open problems

Many chemical basis sets are approximations that were designed for classical quantum chemistry. There is potential for new basis sets to be developed that may do better on a quantum computer [5].

Random parameter initialization for ansätze can lead to gradient plateaus making classical optimization difficult. Studying methods to generate better initial guesses for parameters is another way to decrease the number of optimization steps to convergence. [16].

There is also ongoing study and comparison and development of classical optimization algorithms. Both stochastic gradient descent and the Nelder-Mead method empirically tend to get stuck in local minima as system size grows, so numerical studies [5].

Reducing the number of measurements required for the VQE algorithm is also of great importance. As system size grows, the number of measurements required grows rapidly. Even with polynomial scaling, the number of measurements can balloon from hundreds of millions up to millions of billions. The latter cases, even with optimistic gate speed, would take billions of years for the algorithm to run to completion. [5].

6 Conclusion

By creating a small framework that allows for the relatively easy comparison of ansätze, optimization methods, initial guesses for parameters, and other aspects of the VQE, we have shown a proof of concept of a tool that could help researchers iterate on their ideas. We hope that either future extension of the capabilities and interface of our tool, or other projects with similar aspiration, can make future experimentation easier for researchers studying any of the components of the VQE. Also, by collecting information on the steps of the VQE, we hope that this work can help others who are unfamiliar with the VQE algorithm begin to grasp its core concepts.

References

[1] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature communications*, vol. 5, p. 4213, 2014.

[2] J. R. McClean, I. D. Kivlichan, D. S. Steiger, Y. Cao, E. S. Fried, C. Gidney, T. Häner, V. Havlíček, Z. Jiang, M. Neeley, J. Romero, N. Rubin, N. P. D. Sawaya, K. Setia, S. Sim, W. Sun, K. Sung, and R. Babbush, "Openfermion: The electronic structure package for quantum computers," 2017, cite arxiv:1710.07629. [Online]. Available: http://arxiv.org/abs/1710.07629

[3] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters, and G. K. Chan, "Pyscf: the python-based simulations of chemistry framework," p. e1340, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/wcms.1340

[4] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," 2015.

[5] S. McArdle, S. Endo, A. Aspuru-Guzik, S. Benjamin, and X. Yuan, "Quantum computational chemistry," 2018.

[6] O. Higgott, D. Wang, and S. Brierley, "Variational Quantum Computation of Excited States," *Quantum*, vol. 3, p. 156, July 2019. [Online]. Available: https://doi.org/10.22331/q-2019-07-01-156

[7] V. Hänninen, "Introduction to computational chemistry," http://www.helsinki.fi/kemia/fysikaalinen/opetus/jlk/luennot/Lecture5.pdf, 2012.

[8] R. M. Parrish, L. A. Burns, D. G. A. Smith, A. C. Simmonett, A. E. DePrince, E. G. Hohenstein, U. Bozkaya, A. Y. Sokolov, R. Di Remigio, R. M. Richard, J. F. Gonthier, A. M. James, H. R. McAlexander, A. Kumar, M. Saitow, X. Wang, B. P. Pritchard, P. Verma, H. F. Schaefer, K. Patkowski, R. A. King, E. F. Valeev, F. A. Evangelista, J. M. Turney, T. D. Crawford, and C. D. Sherrill, "Psi4 1.1: An open-source electronic structure program emphasizing automation, advanced libraries, and interoperability," *Journal of Chemical Theory and Computation*, vol. 13, no. 7, pp. 3185–3197, 2017, pMID: 28489372. [Online]. Available: https://doi.org/10.1021/acs.jctc.7b00174

[9] B. Simons, “Second quantisation,” Quantum Condensed Matter Field Theory (Part III Physics) Lecture Notes Chapter 2. [Online]. Available: http://www.tcm.phy.cam.ac.uk/~bds10/tp3/secqu.pdf

[10] S. B. Bravyi and A. Y. Kitaev, “Fermionic quantum computation,” *Annals of Physics*, vol. 298, no. 1, pp. 210 – 226, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0003491602962548

[11] J. T. Seeley, M. J. Richard, and P. J. Love, “The bravyi-kitaev transformation for quantum computation of electronic structure,” *The Journal of Chemical Physics*, vol. 137, no. 22, p. 224109, 2012. [Online]. Available: https://doi.org/10.1063/1.4768229

[12] J. Lee, W. J. Huggins, M. Head-Gordon, and K. B. Whaley, “Generalized unitary coupled cluster wave functions for quantum computation.” *Journal of chemical theory and computation*, vol. 15 1, pp. 311–324, 2019.

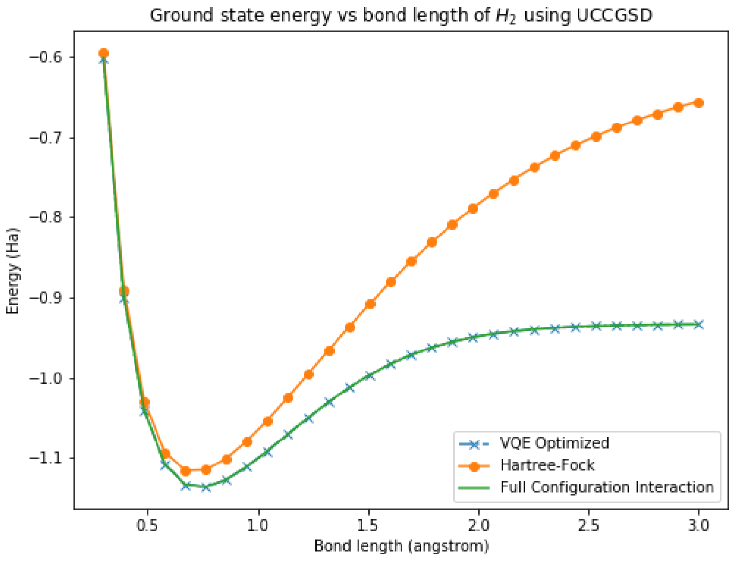
[13] D. M. Olsson and L. S. Nelson, “The nelder-mead simplex procedure for function minimization,” *Technometrics*, vol. 17, no. 1, pp. 45–51, 1975.

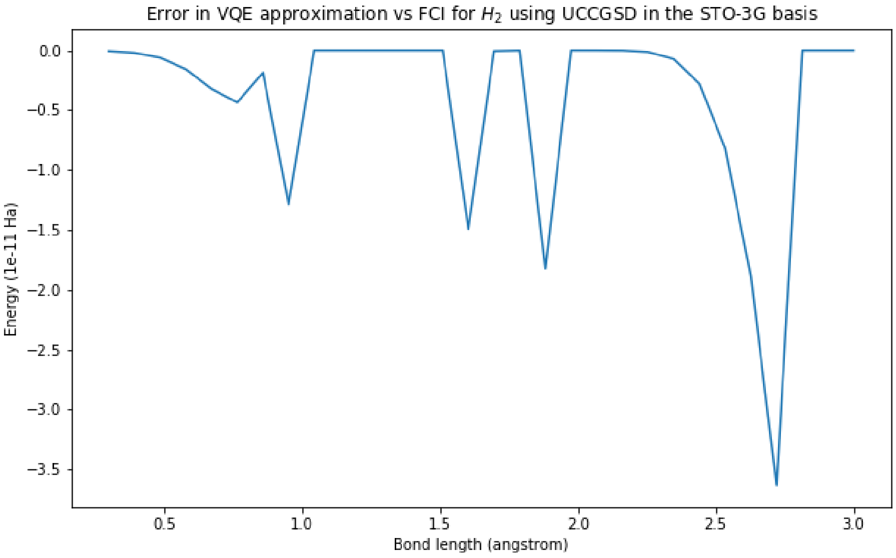
[14] R. Fletcher, *Practical methods of optimization.* John Wiley & Sons, 2013.

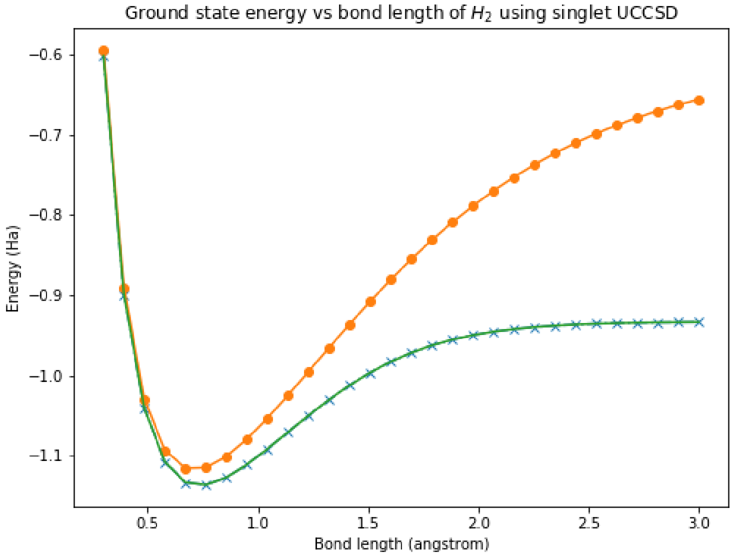
[15] J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik, “Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz,” *Quantum Science and Technology*, vol. 4, no. 1, p. 014008, oct 2018. [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Faad3e4

[16] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, “Barren plateaus in quantum neural network training landscapes,” *Nature Communications*, vol. 9, no. 1, p. 4812, 2018. [Online]. Available: https://doi.org/10.1038/s41467-018-07090-4

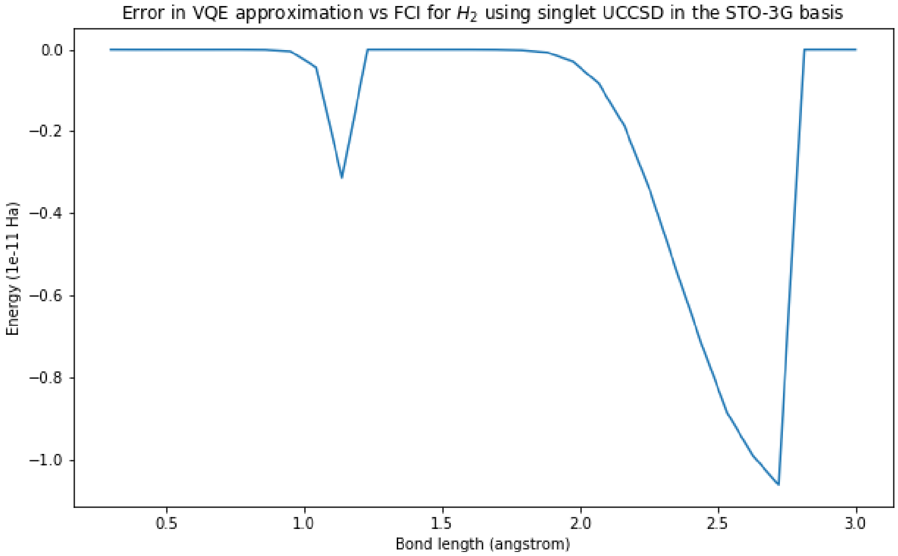
A Experimental data



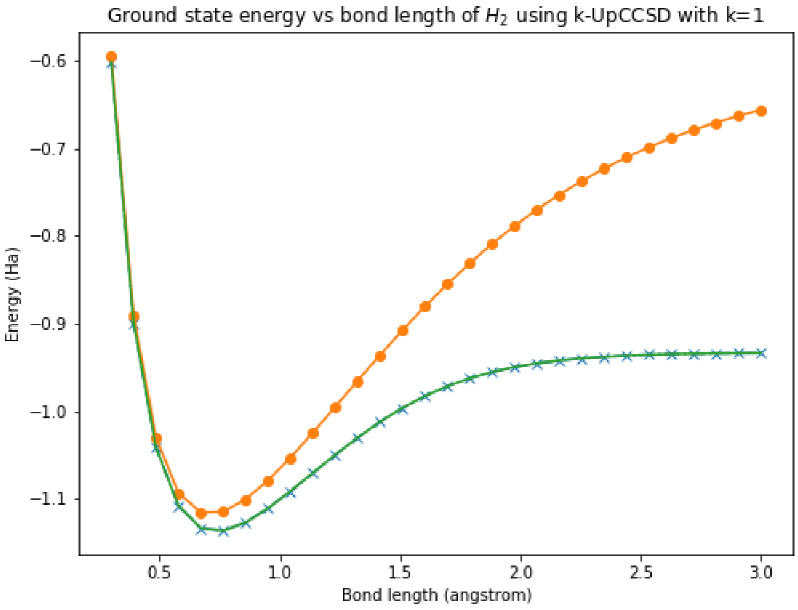


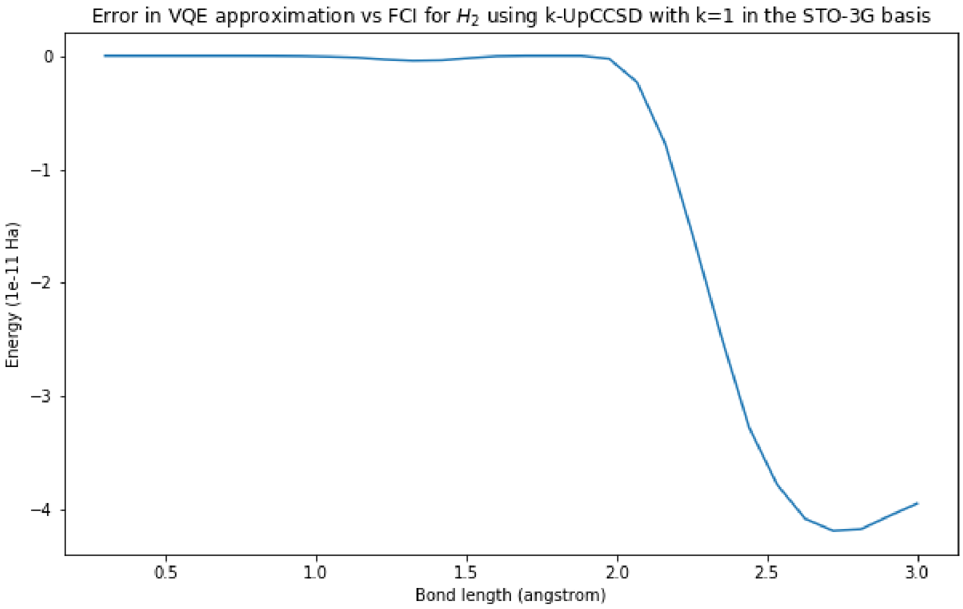


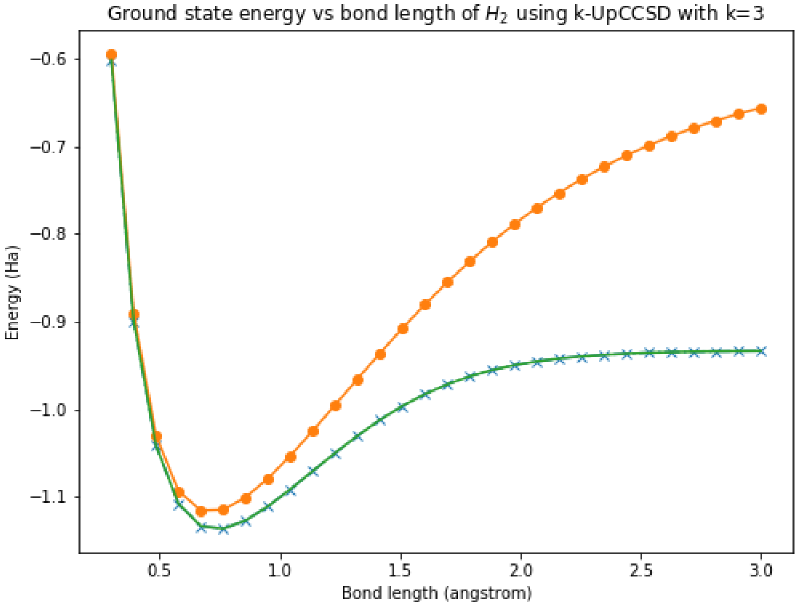
Ground state energy vs bond length of H_2 using singlet UCCSD

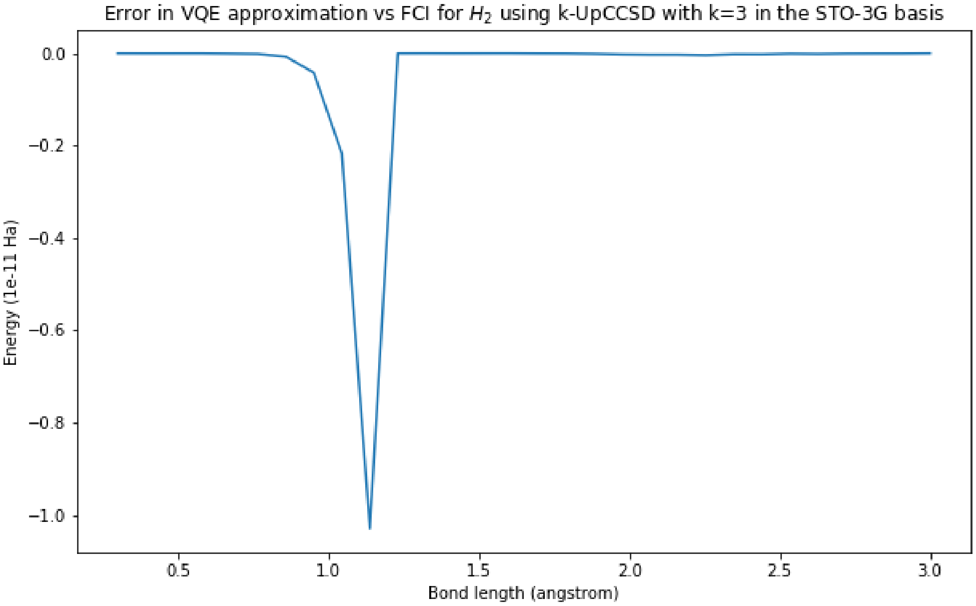


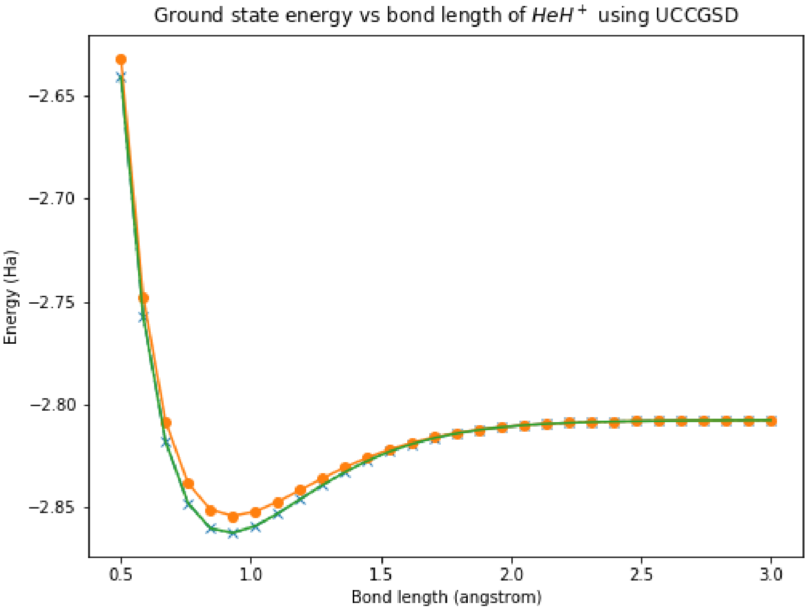
Error in VQE approximation vs FCI for H_2 using singlet UCCSD in the STO-3G basis

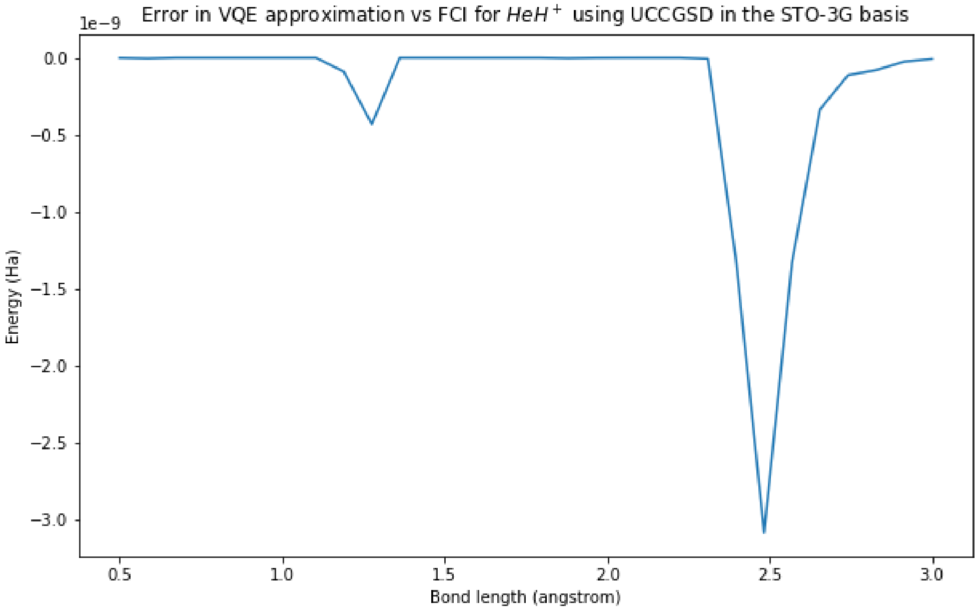

Ground state energy vs bond length of H_2 using k-UpCCSD with k=1
Energy (Ha)
−0.6
−0.7
−0.8
−0.9
−1.0
−1.1
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

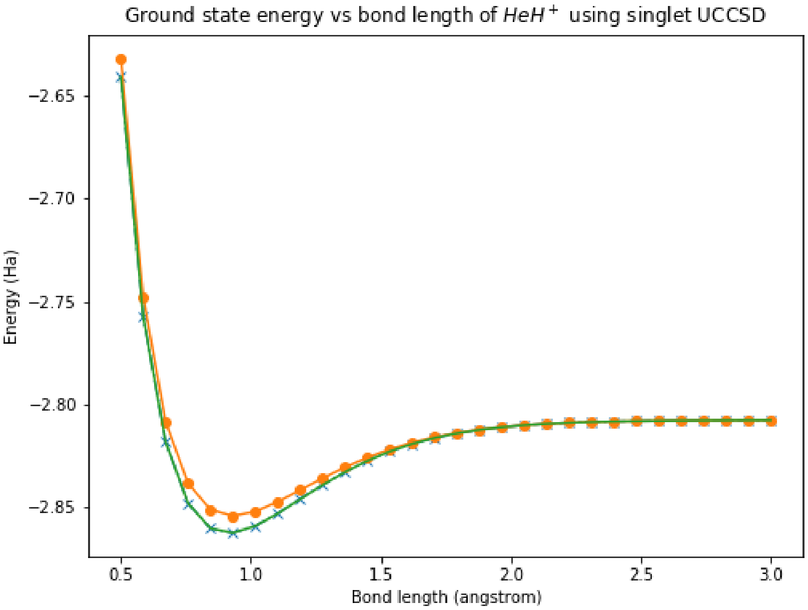

Error in VQE approximation vs FCI for H_2 using k-UpCCSD with k=1 in the STO-3G basis
Energy (1e-11 Ha)
0
−1
−2
−3
−4
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)


Ground state energy vs bond length of H_2 using k-UpCCSD with k=3
Energy (Ha)
−0.6
−0.7
−0.8
−0.9
−1.0
−1.1
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

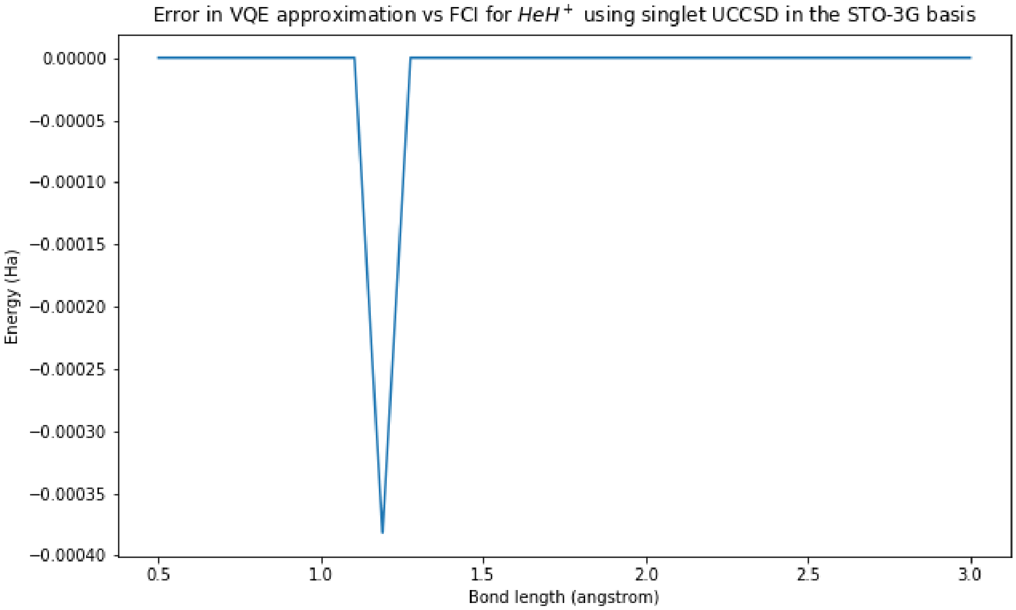

Error in VQE approximation vs FCI for H_2 using k-UpCCSD with k=3 in the STO-3G basis
Energy (1e-11 Ha)
0.0
−0.2
−0.4
−0.6
−0.8
−1.0
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)


Ground state energy vs bond length of HeH^+ using UCCGSD
Energy (Ha)
−2.65
−2.70
−2.75
−2.80
−2.85
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

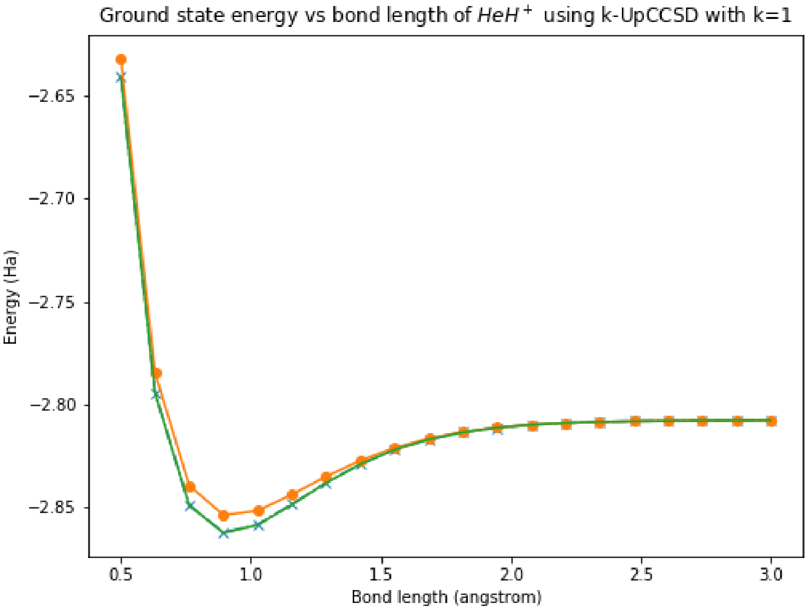

Error in VQE approximation vs FCI for HeH^+ using UCCGSD in the STO-3G basis
1e−9
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
−3.0
Energy (Ha)
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

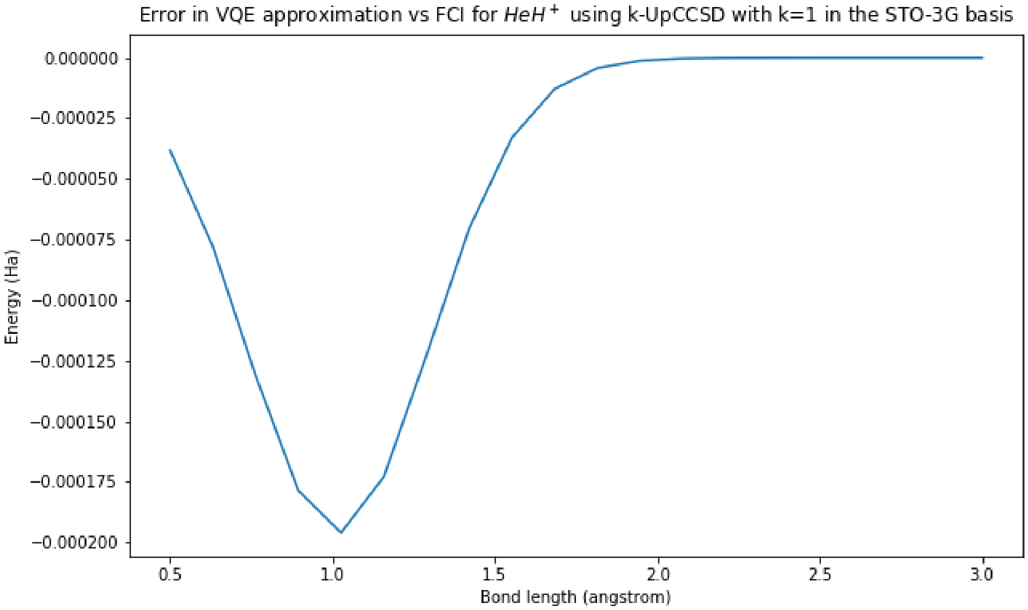


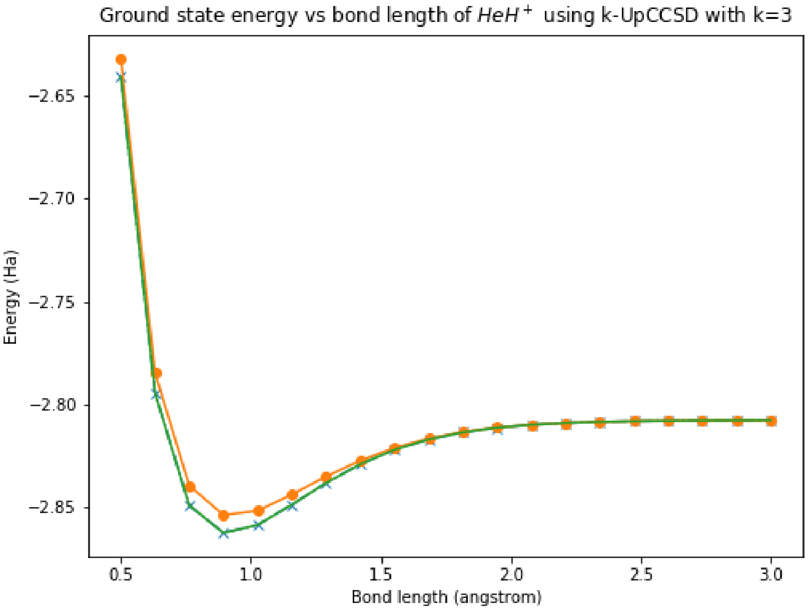
Ground state energy vs bond length of HeH^+ using singlet UCCSD



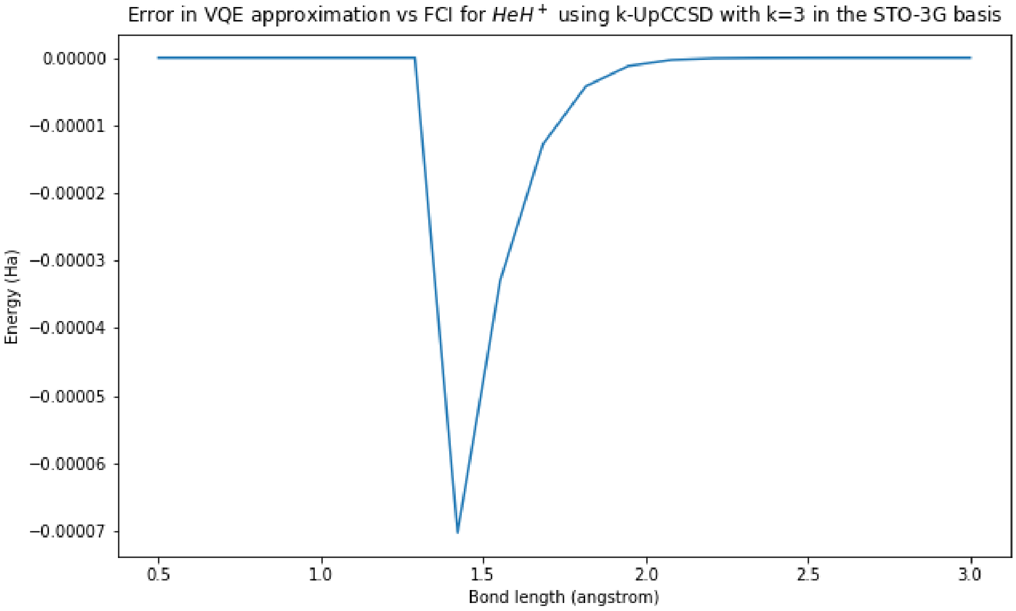
Error in VQE approximation vs FCI for HeH^+ using singlet UCCSD in the STO-3G basis


Ground state energy vs bond length of HeH^+ using k-UpCCSD with k=1
Energy (Ha)
−2.65
−2.70
−2.75
−2.80
−2.85
Bond length (angstrom)
0.5
1.0
1.5
2.0
2.5
3.0

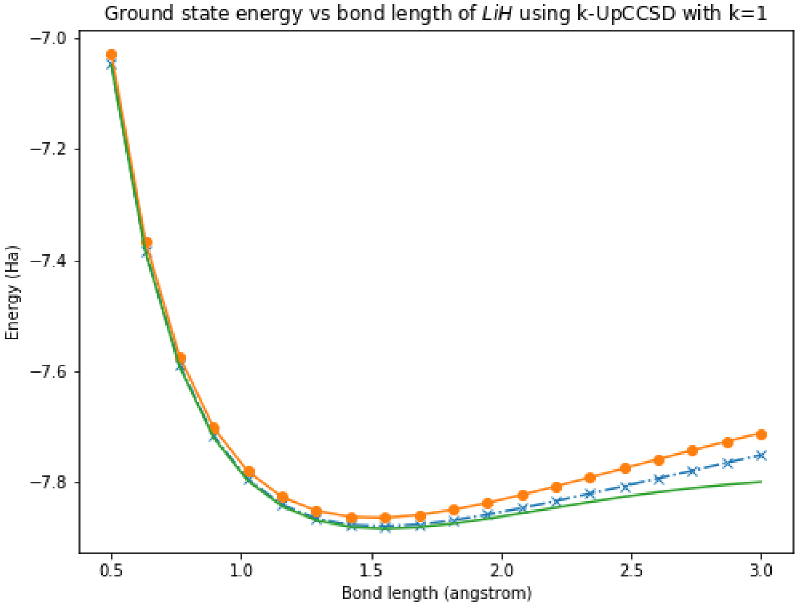

Error in VQE approximation vs FCI for HeH^+ using k-UpCCSD with k=1 in the STO-3G basis
Energy (Ha)
0.000000
−0.000025
−0.000050
−0.000075
−0.000100
−0.000125
−0.000150
−0.000175
−0.000200
Bond length (angstrom)
0.5
1.0
1.5
2.0
2.5
3.0

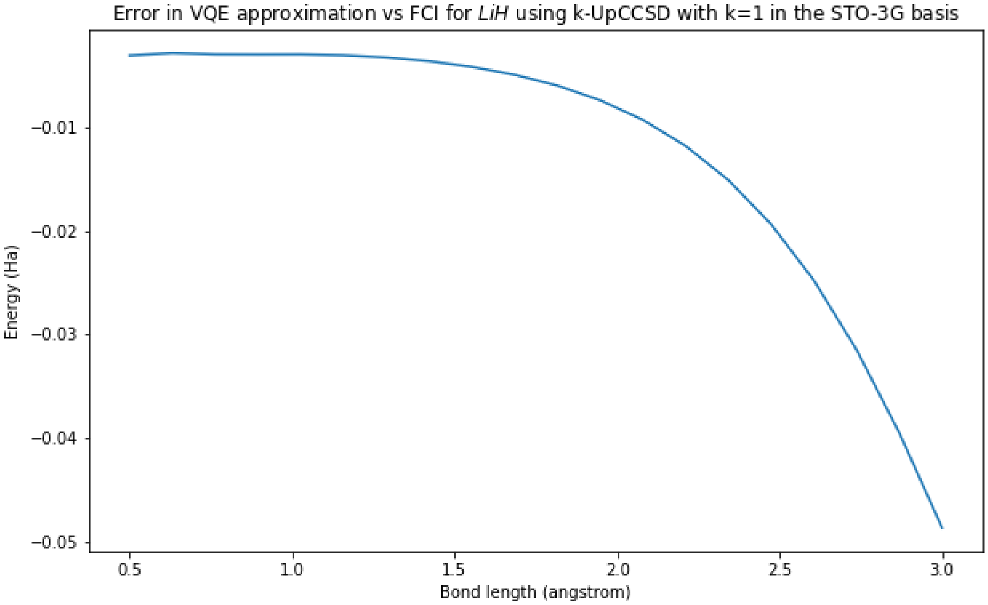


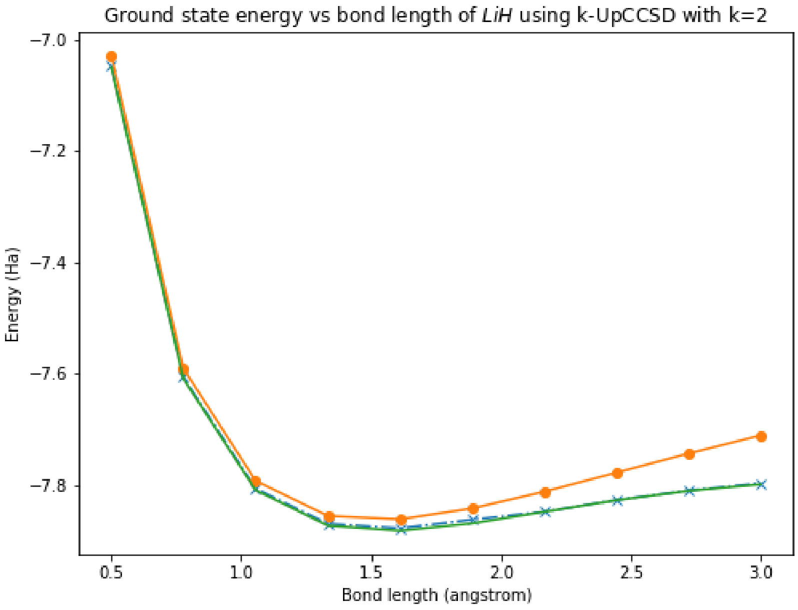
Ground state energy vs bond length of HeH^+ using k-UpCCSD with k=3



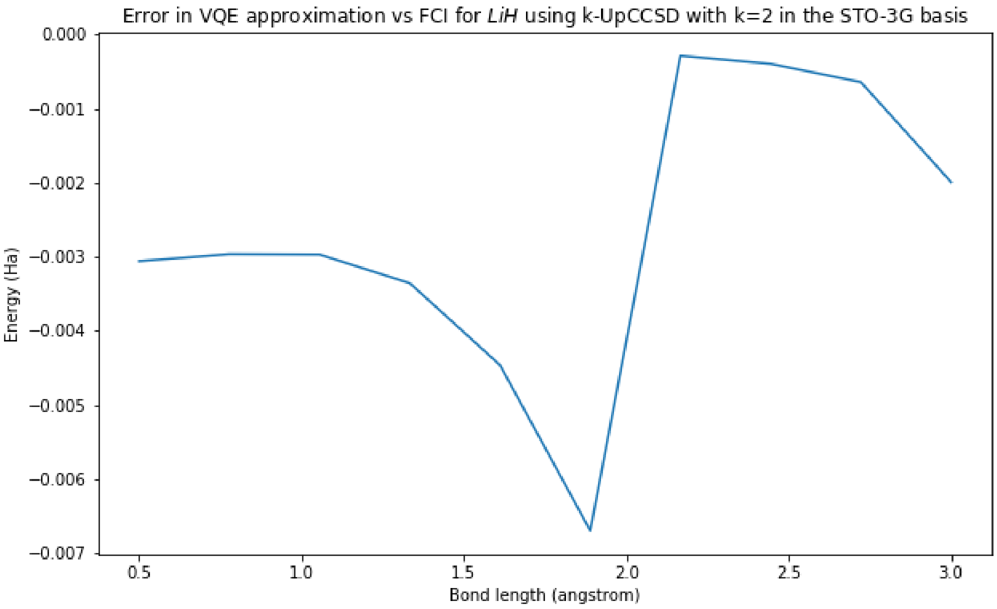
Error in VQE approximation vs FCI for HeH^+ using k-UpCCSD with k=3 in the STO-3G basis


Ground state energy vs bond length of *LiH* using k-UpCCSD with k=1
−7.0
−7.2
−7.4
−7.6
−7.8
Energy (Ha)
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

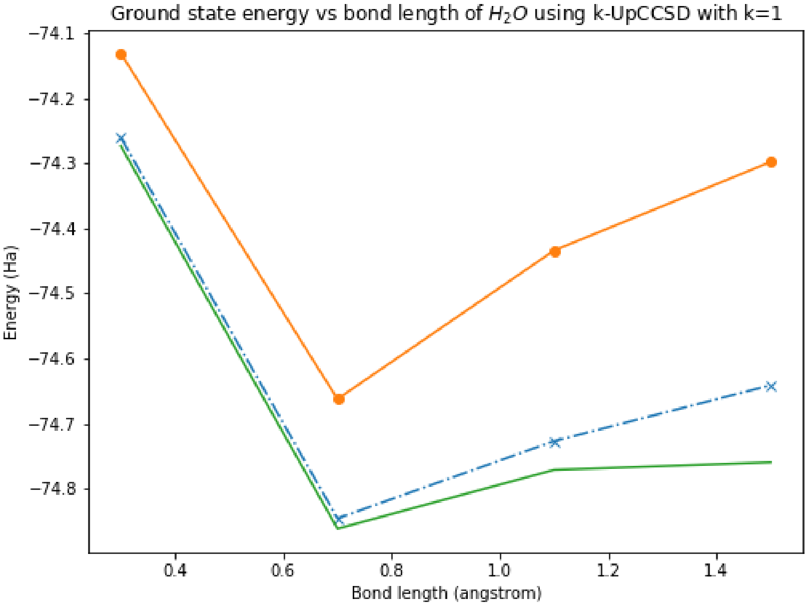

Error in VQE approximation vs FCI for *LiH* using k-UpCCSD with k=1 in the STO-3G basis
−0.01
−0.02
−0.03
−0.04
−0.05
Energy (Ha)
0.5
1.0
1.5
2.0
2.5
3.0
Bond length (angstrom)

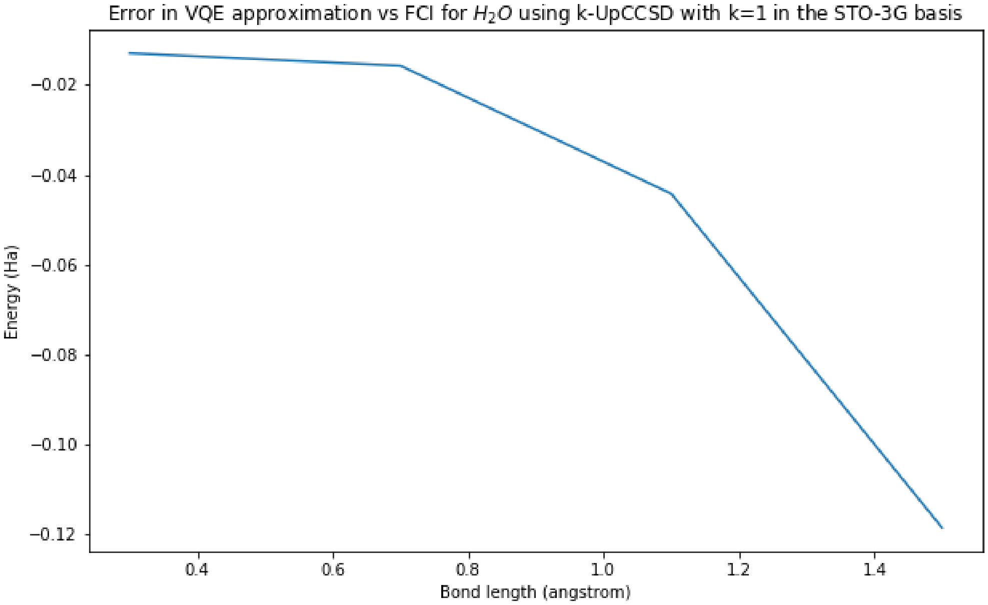


Ground state energy vs bond length of *LiH* using k-UpCCSD with k=2



Error in VQE approximation vs FCI for *LiH* using k-UpCCSD with k=2 in the STO-3G basis


Ground state energy vs bond length of H_2O using k-UpCCSD with k=1
−74.1
−74.2
−74.3
−74.4
−74.5
−74.6
−74.7
−74.8
Energy (Ha)
0.4
0.6
0.8
1.0
1.2
1.4
Bond length (angstrom)


Error in VQE approximation vs FCI for H_2O using k-UpCCSD with k=1 in the STO-3G basis
−0.02
−0.04
−0.06
−0.08
−0.10
−0.12
Energy (Ha)
0.4
0.6
0.8
1.0
1.2
1.4
Bond length (angstrom)