VILNIUS UNIVERSITY
FACULTY OF MATHEMATICS AND INFORMATICS
SOFTWARE ENGINEERING BACHELOR STUDY PROGRAM

Variational quantum eigensolver algorithm performance analysis on various molecules

Variacinio kvantinio tikrinių reikšmių algoritmo efektyvumo analizė įvairioms molekulėms

Bachelor's thesis

Author: Naglis Šuliokas

Supervisor: doc. dr. Linas Petkevičius

Reviewer: lect. Irus Grinis

Vilnius – 2025

Contents

Summary

Variational quantum eigensolver (VQE) is one of the most significant quantum algorithms, that can be implemented with near-term noisy quantum hardware, improving material science, medicine and other research fields. This bachelor's thesis analyzes the steps and performance of VQE described in the literature, then tests the findings with experiments on Qiskit Aer simulator, Tequila and Madness Python libraries, IBM Quantum hardware, resulting in more data available for research. Experimental data mostly supports the literature findings, showing Bravyi-Kitaev mapping slight advantage over Jordan-Wigner and Parity mappers, confirming significant accuracy improvement, when using k-UpCCGSD ansatz, instead of UCCSD. However, ADAPT-VQE algorithm accuracy improvement is contradicted with simulator process for small molecules. This study also has some limitations, like very narrow experimental data set and not thorough algorithm analysis, considering the conducted broad analysis. This study findings could help future research as an additional research data source, also providing insights about VQE, and it's implementation in code, so that those insights could be used for algorithm improvement.

Keywords: VQE, quantum computing, computational chemistry, IBM quantum hardware, benchmarks

Santrauka

Variacinis kvantinis tikrinių reikšmių algoritmas (VQE) yra vienas iš svarbiausių kvantinių algoritmų, kuris gali būti įgyvendintas naudojant artimos ateities triukšmingą kvantinę įrangą, patobulinant medžiagų mokslą, mediciną bei kitas mokslo sritis. Šis bakalauro baigiamasis darbas analizuoja VQE žingsnius bei efektyvumą, aprašytus literatūroje, tuomet testuoja rezultatus eksperimentuojant Qiskit Aer simuliatoriumi, Tequila ir Madness Python bibliotekomis, IBM Quantum įranga. To pasekoje yra paruošiama daugiau duomenų moksliniams tyrimams. Eksperimentiniai duomenys daugiausia patvirtina literatūros išvadas, parodydami, kad Bravyi–Kitaevo atvaizdavimas turi nežymų pranašumą, palyginus su Džordano-Vignerio ir pariteto atvaizdavimais, patvirtindami žymų tikslumo patobulinimą, kalbant apie k-UpCCGSD sakinio naudojimą, vietoj UCCSD. Tačiau, ADAPT-VQE algoritmo tikslumo patobulinimas yra paprieštaraujamas, naudojant simuliatorių su mažomis molekulėmis. Šis darbas taip pat turi tam tikrų trūkumų, tokių kaip labai mažas eksperimentinių duomenų kiekis ir nekruopšti algoritmo analizė, turint omeny plačią darbo apimtį. Šio darbo rezultatai galėtų padėti ateitiek moksliniams tyrimams kaip papildomas duomenų šaltinis, taip pat pateikiant įžvalgų apie VQE ir šio algoritmo įgyvendinimą kodu. Šiuo būdu įžvalgos apie algoritmą galėtų būti panaudoto algoritmo tobulinimui.

Raktiniai žodžiai: VQE, kvantiniai skaičiavimai, skaičiuojamoji chemija, IBM kvantinė įranga, lyginamieji testai

1. Introduction

1.1. Object of the study

In recent years, quantum computing has advanced significantly. Until 1988 it was only theorized to work - all the algorithms developed in theory. In 1988 a simple Deutsch's algorithm was tested on a first 2-qubit nuclear magnetic resonance quantum computer [JMH98, Jones et al. 1998]. Recent scientific advancements in this field lead to engineered quantum computers with more and more qubits. In 2023 IBM built, as of today, the largest gate-based system quantum computer with 1,121 physical qubits [Abu24]. Even the biggest problem in quantum computing, decoherence, is currently said to be solved by Microsoft, introducing quantum processor with topological qubits in 2025 [AAA+25]. This fast development of quantum hardware demonstrated the possibility of implementing quantum algorithms for solving currently classically unsolvable computation problems, such as large number prime factorization, NP-hard combinatorial problems, complex molecular system simulation. However, up to this date developed quantum computers have short coherence times and relatively small number of qubits and gates to show the quantum advantage. Developed devices can show practical use only after the hardware and algorithms are improved. For this reason it is important to continue testing and making this improvement. One way to facilitate the development in this field is to collect computation data and perform comparative analysis on the existing algorithms and their variations.

The problem we are analyzing is finding ground state energy for small molecules. In this bachelor's thesis we are only considering small structures, since with current computing capabilities it is not possible to fully run ground state energy algorithm for complex structures with neither of classical quantum computers. But the implementation for small molecules can be extended with the same logic, only using more hardware resources.

There are multiple ways to calculate the ground state energy of a molecule. Classically it is possible to get exact solution with an FCI (Full configuration interaction) [KH89], solving electronic Schrödinger equation. However, for the classical algorithm computational complexity is exponential against the complexity of a molecule (the basis set and number of electrons) - solution to this problem is the QPE (Quantum phase estimation) algorithm, which is run on a quantum computer, resulting in exponential speedup [SST+21]. The problem arising in the QPE implementation is that it can only run with small molecule data on NISQ (Noisy intermediate-scale quantum) devices, due to the generated deep circuits. Because of this problem with NISQ devices, a hybrid (classical and quantum) algorithm was developed, which can approximate ground state energy of molecules using variational method accurately while generating shallow quantum circuits. Variational method yields more errors (number of measurements scale as $\mathcal{O}(\frac{1}{\epsilon^2})$ vs $\mathcal{O}(\log(\frac{1}{\epsilon}))$ for precision ϵ). There are some algorithms developed that balance error complexity by combining both - QPE and variational method ([WHB19], [CAB+21]). However, in this bachelor's thesis we are not discussing all such specific cases, since the scope of this study is broad - only the most important algorithm frameworks whose research would contribute the most to the field.

The object of this study is Variational quantum eigensolver (VQE) ([PMS+14]) and it's

implementation for quantum chemistry problem of finding ground state energy for arbitrary molecule. VQE is an algorithm that approximates the smallest eigenvalue for a complex-valued matrix with classical and quantum subroutines using variational method (Fig. 1). Having molecule Hamiltonian, this is the way to approximate the ground state energy. There has been a number of successful VQE runs on quantum computers, such as one in 2017 by IBM, computing the ground state energy of H_2, LiH, and BeH_2 molecules [KMT+17]. However, computation for bigger, more complex molecules, as mentioned before, is still computationally not effective due to hardware limitations.

Methods to generate accurate Hamiltonian from molecule electronic and nuclear configurations are also a subject of scientific research, many studies have been conducted on this topic, but, since this is regarding quantum chemistry field and not related with the actual VQE algorithm implementation, it will not be analyzed thoroughly in this bachelor thesis.

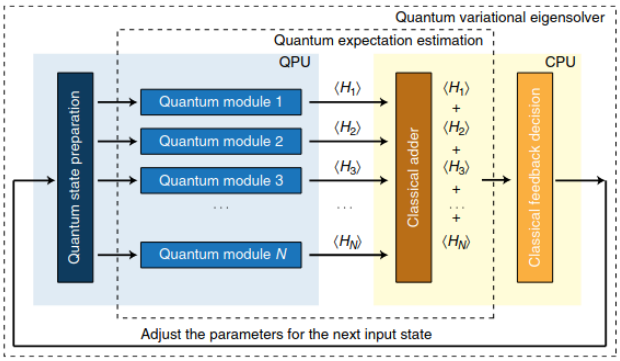

Figure 1. Architecture of the quantum-variational eigensolver [PMS+14]

1.2. Significance

Finding ground state energy of molecules holds great importance for research. It can help discover new materials, drugs, make new medicine [MMJ+22]. Having approximated ground state energy, VQE algorithm variations or other variational algorithms, like VQD (Variational quantum deflation), can be used to calculate the excited states of molecules, which has also a lot of implementations in research ([CAB+21], [HWB19]). Apart from these quantum chemistry use cases, in general case, VQE can also help solve combinatorial optimization problems (encoding them into Hamiltonians) which contribute to logistics and finance [MOS+21]. Since quantum hardware possibilities are still limited in the NISQ era, it is important to test and improve the algorithms to be able to achieve better results with less hardware resources. Analysis of the VQE algorithm and possibilities of it's variations would facilitate future research process by informing about the ways to test the algorithm on commertially accessible quantum and classical devices, providing computation data on recently developed algorithms, addressing areas for algorithm improvement.

1.3. Aim of the study

In this study we aim to analyze and determine the best computation methods for the VQE algorithm, used for computational chemistry problems, simulating quantum (molecular) systems, determine the current computational limits for basic and more complex molecules.

1.4. Research objectives

To achieve the aim of this study, the following objectives are being set:

1. Conduct the literature analysis: find the variations of VQE and it's computation methods, determine and analyze computationally best methods for computational chemistry problems. This includes comparing
   (a) Hamiltonian mappers (from electronic to qubit),
   (b) ansätze,
   (c) classical optimizers,
   and deciding which is best in which case.
2. Run benchmarks on
   (a) Aer simulator,
   (b) IBM Quantum hardware,
   and compare with the findings of literature analysis. Perform analysis of selected VQE algorithm variations with H_2, LiH and BeH_2 molecule data. Measure speed of convergence, runtime, number of algorithm iterations. Evaluate circuit depth, gate fidelity, calculated ground state energy accuracy. Compare noisy quantum hardware results with noiseless simulator results.
3. Determine current limitations of computation. For VQE with different molecule data, consider qubit count, gate fidelity, coherence time, time-wise performance compared to classical implementation using FCI. Decide if for each analyzed case quantum advantage is reached.
4. Compare empirical findings with the existing research, evaluate alignment with published research results.

1.5. Theoretical background

The field of this study is quantum computing, but the analyzed algorithm also deals with computational quantum chemistry, quantum mechanics, classical computing and optimization. The basic knowledge in these fields is being assumed. This includes

1. quantum mechanical model of an atom, molecular orbital theory,
2. the concept of a Hamiltonian and molecule energy states,
3. the basics of the gate-based quantum computing,
4. basic function optimization methods.

1.6. Methodology and literature overview

To achieve the research objectives, these research methods are going to be conducted:

1. literature analysis on VQE algorithm variations and experimental data,
2. empirical experiment measuring speedups and accuracy on both classical and quantum devices,
3. comparative analysis of empirical and theoretical data.

The sources used for this study are mainly articles, recently published in scientific journals, or published some time ago, but are crucial to the subject. Experimental data, used for comparison with computed data for this study, is retrieved from either research articles, or from tested and approved public datasets in a form of a Python package or a data file. The Python scripts for experimental data are written with the help of online video tutorials, articles, official documentation and provided examples, publicly available code on GitHub platform.

1.7. Analysis structure and context

During the subject analysis, literature review on the VQE algorithm is conducted first. After that the basic VQE implementation on simulator and quantum hardware is analyzed, yielding basic numeric results. Having the basic implementation of VQE, the further analysis of implementation steps and frameworks (variations) is conducted, advantages and drawbacks of multiple Hamiltonian mappings, ansätze, optimizers are being outlined. After that, the information for conducting experiments is gathered, including the metrics used for evaluating and comparing VQE frameworks and research data from other studies that could be used as a comparison. And when the VQE frameworks and analysis metrics are selected, the experiments are conducted on both simulator and quantum hardware. Retrieved results are described, compared with literature, conclusions are being made.

The context of conducting this research is basic and minimalistic. No additional financing is used. Calculations are being made using personal computer and the free IBM Quantum hardware access, limited to 10 minutes of runtime per month for one account.

1.8. Tools and instruments

The experimental research part is done using IBM open source tool Qiskit. Algorithm implementation is done in Python scripts, generating graphs with mathplotlib Python package. PySCFDriver from Qiskit tools is used for retrieving molecule structure data. Majority of the VQE subroutines are implemented using Qiskit built-in functions. Also, Python package Tequila ([KAT+25]) is being used for some VQE subroutines. IBM Quantum hardware is being used through API, submitting jobs to a queue, after the run retrieving computed data. GitHub platform is used for experiment code and computed data storage, only summarized data is given in this bachelor's thesis. Code and computed data can be found on GitHub platform [Šul25].

2. Literature analysis

The aim of literature analysis is to familiarize with the VQE algorithm and it's computation steps -

1. qubit Hamiltonian generation (electronic to qubit Hamiltonian mappings),
2. ansatz generation (from qubit Hamiltonian and/or from electronic Hamiltonian),
3. classical optimisation subroutine (minimizing expectation value of a state generated by the parameterized ansatz on the Hamiltonian basis) -

comparing the methods used for each step, helping to define the experiment scope. For clarity, the diagram of the VQE algorithm process is provided (Fig. 2).

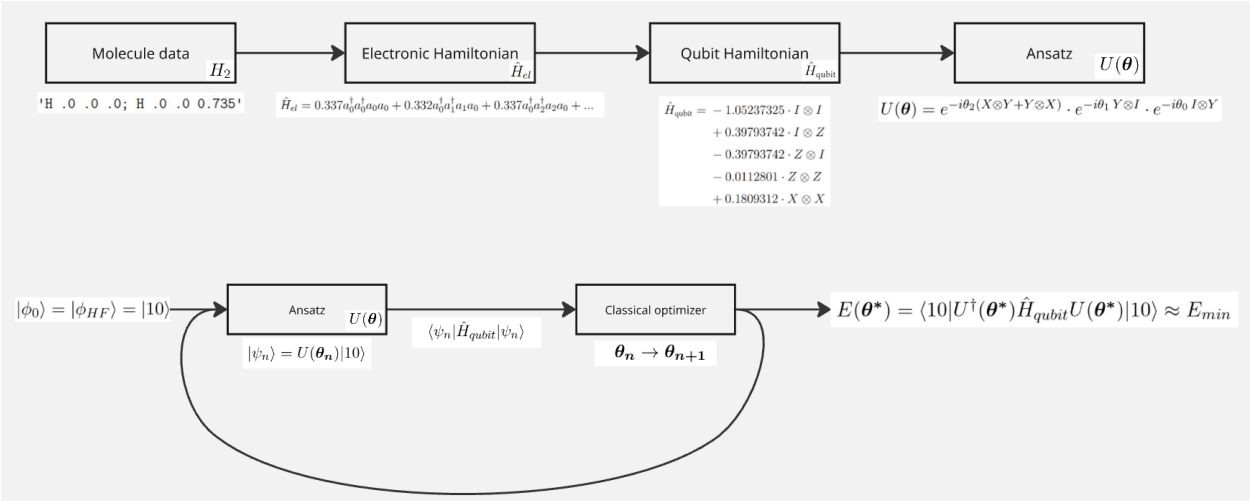

Figure 2. VQE algorithm process

VQE was first proposed by Peruzzo and colleagues in 2014 [PMS+14]. It is the first proposed VQA (Variational quantum algorithm) ([CAB+21]). VQA is an algorithm that consists of a cost function aimed to be minimized, and an ansatz - unitary operator (quantum circuit) with a set of continuous or discrete parameters $\boldsymbol{\theta}$ that can be optimized [CAB+21]. It is an iterative algorithm that has a subroutine running on a quantum device, evaluating the cost function, and a classical subroutine, optimizing the ansatz parameters. As the cost calculations converge, VQA produces an answer - minimized cost function value and a set of parameters that minimizes it.

In the case of VQE, the cost function is

$$C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|\, \hat{H}_{qubit}\, |\psi(\boldsymbol{\theta})\rangle\,.$$

Here $|\psi(\boldsymbol{\theta})\rangle$ is a trial state, produced by the ansatz from initial state, $\hat{H}_{qubit}$ is a qubit form of molecule Hamiltonian, used for computational chemistry problems. VQE can also solve optimization problems like Max-cut, but this is out of scope of this bachelor's thesis - we focus on a molecule ground state energy calculations for chemistry.

According to the Rayleigh-Ritz principle ([GOK88], [PMS+14]), function $H(\boldsymbol{\theta}) = \frac{C(\boldsymbol{\theta})}{\langle\psi(\boldsymbol{\theta})|\psi(\boldsymbol{\theta})\rangle}$ when minimized with a certain parameters $\boldsymbol{\theta}$, would approximate the smallest eigenvalue of $\hat{H}_{qubit}$,

i.e. would calculate the ground state energy. Since in VQE ansatz is built from unitary gates, states are always normalized, $\langle\psi(\boldsymbol{\theta})|\psi(\boldsymbol{\theta})\rangle = 1$, therefore $E(\boldsymbol{\theta}) = C(\boldsymbol{\theta})$.

There are multiple ways to map electronic Hamiltonian in the second quantized form (which is used as an approximation of full molecule Hamiltonian) to qubit Hamiltonian representation ([JW93], [BK02], [BGM+17]). There are also many different possible ansätze that can evaluate the cost function, since, regarding to the Rayleigh-Ritz principle, it does not matter how the state of which the expectation value in $\hat{H}_{qubit}$ basis is calculated changes with the iterations - any state generated by any ansatz unitary would yield exact or greater value than the ground state energy (smallest eigenvalue of $\hat{H}_{qubit}$). So the algorithm is not dictated to what unitary it should use to generate the series of states, as long as their expectation value converges, i.e. yields smallest possible expectation value (the approximated ground state energy). The initial trial state is important to be chosen close enough to the precise ground state, so that the optimisation would be successful and would not yield to other possible local minima or Baren plateaus. From the initial state, other trial states are generated. It is common to choose Hartree-Fock state as the initial state ([PMS+14], [KZU25], [CAB+21], [TCC+22], [QX23]), which is close enough to the ground state (because it is calculated with mean field approximation) ([Bac93]), therefore in this bachelor's thesis we will simplify our analysis by performing the VQE using Hartree-Fock initial trial state:

$$|\phi_0\rangle = |\phi_{HF}\rangle, \;\; |\psi(\boldsymbol{\theta}_n)\rangle = U(\boldsymbol{\theta}_n)|\phi_0\rangle.$$

However, we acknowledge other ways to generate initial trial state, mentioned in literature ([FHZ+24]). Finally, there are many ways to perform the classical optimization algorithm, that would determine what parameters should be chosen for each iteration. These optimisation algorithms should also be analyzed and compared.

2.1. Hamiltonian mappings

The first step of VQE, for which we are conducting analysis, is the construction of qubit Hamiltonian $\hat{H}_{qubit}$. In the context of quantum chemistry, it is constructed from the second quantization electronic Hamiltonian $\hat{H}_{el}$, using one of many possible Hamiltonian mappings. After the operators are mapped, we get the qubit Hamiltonian, which is the one that is used in VQE or other quantum chemistry quantum simulation algorithms ([SW18]). There are other ways to approximate molecule Hamiltonian ([AF11]), but second quantization electronic Hamiltonian is the most commonly used in literature and is accurate enough, therefore, since analysis of those methods' falls out of this study scope, for simplicity we are only using the electronic Hamiltonian. For any molecule electronic (fermionic) Hamiltonian can be constructed using second quantization fermionic creation and annihilation operators. Constructed it is of this form:

$$\hat{H}_{el} = \sum_{i,j} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{i,j,k,l} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l,$$

where h_{ij} and h_{ijkl} are the Coulombic overlap and exchange integrals for electrons, $a_i^\dagger$ and a_i are fermionic creation and annihilation operators. The first part, $\sum_{i,j} h_{ij} a_i^\dagger a_j$, represents a single electron contribution to the molecule energy, the second part, $\frac{1}{2}\sum_{i,j,k,l} h_{ijkl} a_i^\dagger a_j^\dagger a_k a_l$, represents electron-electron interaction contribution to the molecule energy ([TLM+18], [FPG+22], [QX23]). Integrals, which can be efficiently computed classically, are determined by the chosen basis set (of functions), of which linear combinations are approximations of molecular orbitals. Many sources ([QX23]), [TLM+18], [STH+22], [KVO+18]) use STO-3G basis, which, for simplicity, will be also used in this bachelor's thesis. Here is an example electronic Hamiltonian for the H_2 molecule with 0.735 Å inter-atomic distance case, calculated using Qiskit second quantization driver PySCFDriver, STO-3G basis:

$$\hat{H}_{el} = 0.337 a_0^\dagger a_0^\dagger a_0 a_0 + 0.332 a_0^\dagger a_1^\dagger a_1 a_0 + 0.337 a_0^\dagger a_2^\dagger a_2 a_0 + ...$$

Here indexes 0, 1, 2 respectively mean the first atom 1s-up electron, the first atom 1s-down electron and the second atom 1s-up electron. The exact full Hamiltonian can be found together with Python code that produces it in the GitHub repository [Šul25].

There are many possible ways to perform the mapping to qubit Hamiltonian that can be used in VQE, like Checksum, Binary addressing codes ([SW18]), Derby-Klassen, Super-compact ([OS24]), but the most commonly used and efficient ones include Jordan-Wigner, Bravyi-Kitaev and Parity mappings ([FPG+22], [TLM+18], [GAN14], [TCC+22], [YLS+25]). After the transformation, qubit Hamiltonian should be of this form:

$$\hat{H}_{qubit} = \sum_i \alpha_i P_i$$

where α_i is a complex coefficient, P_i is a Pauli string (tensor product of Pauli matrices X, Y, Z, I). For example, this is hydrogen qubit Hamiltonian made with Parity mapper with qubit reduction (code implementation can be found in GitHub repository [Šul25]):

$$\begin{aligned}\hat{H}_{\text{qubit}} = &-1.05237325 \cdot I \otimes I \\ &+0.39793742 \cdot I \otimes Z \\ &-0.39793742 \cdot Z \otimes I \\ &-0.0112801 \cdot Z \otimes Z \\ &+0.1809312 \cdot X \otimes X\end{aligned}$$

The key metric for mapping method comparison is the Pauli weight ([TCC+22]) - the length of the Pauli string. For example, Pauli weight of Pauli string $ZXYIZ$ is 4, as there are 4 non-identity Pauli matrices. In a generated qubit Hamiltonian there are usually multiple terms of with Pauli strings (e.g. $\hat{H}_{qubit} = -0.81I + 0.17Z_0 + 0.17Z_1 + 0.17Z_0Z_1 + 0.12X_0X_1 + 0.12Y_0Y_1$). Therefore, when considering then scaling of Pauli weight, only the maximum possible Pauli weight is compared to the number of qubits (in the previously analyzed mappings it is the same as the

number of fermionic modes). Also, the operator scaling is taken into account (the number of Pauli string terms in qubit Hamiltonian).

2.1.1. Jordan-Wigner mapping

Jordan-Wigner mapping ([JW93]) is the most commonly used, simplest ([TLM+18]) mapping, arguably the oldest one ([OS24]). In this mapping, occupation of spin orbitals is encoded with $|1\rangle$ for occupied and $|0\rangle$ for unoccupied. These are the transformation rules ([TCC+22], [TLM+18]):

$$a_i^\dagger \to \frac{1}{2}(X_i - iY_i) \otimes Z_0 \otimes Z_1 \otimes \cdots \otimes Z_{i-1},$$

$$a_i \to \frac{1}{2}(X_i + iY_i) \otimes Z_0 \otimes Z_1 \otimes \cdots \otimes Z_{i-1}.$$

Maximum Pauli weight for this mapping scales as $\mathcal{O}(N)$. Operator scaling is $\mathcal{O}(N^4)$.

2.1.2. Parity mapping

Parity mapping ([BGM+17]), has better qubit tapering - removing qubits because of symmetries, that are better exposed, without losing physical information. It also produces more sparse Hamiltonians (fewer $\alpha_i P_i$ terms). This mapping, instead of encoding occupation of each spin orbital separately, encodes the parity of orbital occupation. $|1\rangle_i$ means that the number of occupied orbitals up to the i-th (including) is odd, and $|0\rangle_i$ means it is even ([TCC+22]). These are the transformation rules:

$$a_i^\dagger \to \frac{1}{2}(Z_{i-1} \otimes Z_i - iY_i) \otimes X_{i+1} \otimes \cdots \otimes X_{n-1},$$

$$a_i^\dagger \to \frac{1}{2}(Z_{i-1} \otimes Z_i + iY_i) \otimes X_{i+1} \otimes \cdots \otimes X_{n-1}.$$

Maximum Pauli weight for this mapping scales as $\mathcal{O}(N)$. Operator scaling is $\mathcal{O}(N^4)$.

2.1.3. Bravyi-Kitaev mapping

Another mapping introduced to tackle the linear scaling of Pauli weights is Bravyi-Kitaev mapping ([BK02]). This mapping stores both occupation and parity numbers in qubits. These are the transformation rules ([TCC+22], [TLM+18]):

$$a_i^\dagger \to \frac{1}{2}(Z_{P(i)} \otimes X_i \otimes X_{U(i)} - iZ_{R(i)} \otimes Y_i \otimes X_{U(i)}),$$

$$a_i \to \frac{1}{2}(Z_{P(i)} \otimes X_i \otimes X_{U(i)} + iZ_{R(i)} \otimes Y_i \otimes X_{U(i)}),$$

where $P(i)$, $U(i)$, $R(i)$ are the qubit sets, respectively, called "parity", "update" and "remainder". For brevity, exact construction of these sets is not discussed in this bachelor's thesis, since the main sources ([TCC+22], [TLM+18]) also exclude exact explanation.

Maximum Pauli weight for this mapping scales as $\mathcal{O}(log_2(N))$. Operator scaling is $\mathcal{O}(N^4)$. In theory, because of the lower Pauli weights, Bravyi-Kitaev mapping should have less read-out errors. However, with grouping of the Pauli operators for joint measurements, Jordan-Wigner and Parity mappers gain advantage ([TCC+22]). Bravyi-Kitaev method shows advantage when ansatz is being constructed from the generated Hamiltonian (e.g. UCCSD, see section 2.2.1), or when it is used with QPE (Quantum phase estimation) algorithm, because it significantly reduces the number of CNOT or other entanglement gates, without impacting other metrics ([TSS+15], [TCC+22]).

Regarding the Pauli weight, the mentioned mappings can be formalized using ternary trees. When this is done, ternary trees can be used to find the optimal Hamiltonian encoding, which scales as $\mathcal{O}(log_3(2N))$ ([YLS+25]). However, the implementation of this mapping is more complex and in comparison shows little performance improvement, therefore in this study we will not analyze it in detail.

2.2. Ansätze

The main component of VQE is the ansatz - parameterized quantum circuit that iteratively prepares a quantum state and estimates it's expectation value. The equation for state preparation is this:

$$|\psi(\boldsymbol{\theta}_n)\rangle = U(\boldsymbol{\theta}_n)|\phi_0\rangle,$$

where $U(\boldsymbol{\theta})$ is a unitary (the ansatz) made of gates depending on parameters $\boldsymbol{\theta}$, and $|\phi_0\rangle$ is the reference (initial) state - we use Hartree-Fock state. The expectation value is also calculated, evaluating the cost function $C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|\,\hat{H}_{qubit}\,|\psi(\boldsymbol{\theta})\rangle$. After that the work is given to the classical subroutine to choose the parameters for the next algorithm iteration.

Since VQE can solve various different problems - mainly combinatorial optimization and computational chemistry - ansätze are also built differently, depending on the use case. QAOA (Quantum Approximate Optimization Algorithm [FGG14]) is the most widely used ansatz, built and referenced in many papers ([GM19], [LJJ+22]) specifically to solve combinatorial optimization problems. Since combinatorial optimization problems are out of this work scope, ansätze related to them will not be discussed. Then there is HEAs (Hardware Efficient Ansatz) and chemistry inspired ansätze. We will focus on the chemistry inspired ansätze and briefly discuss HEA for the use on IBM quantum hardware.

The standard chemistry inspired ansatz used for VQE is UCCSD (Unitary coupled cluster for singles and doubles) ([FPG+22]). We will analyze this ansatz usage in the algorithm, together with some algorithm improvements: k-UpCCGSD, ADAPT-VQE. We acknowledge there are more methods ([MMN+20], [MBK+20]), but these are the most common, recent and most performant ones ([TCC+22], [FPG+22]).

2.2.1. UCCSD

UCCSD was first used in VQE together with the introduction of VQE itself by Peruzzo et al. (2014) [PMS+14]. However, the coupled-cluster (CC) theory has been a subject to many studies since it's first introduction in 1966 ([Číž66]). The general case of coupled-cluster state (approximation to the ground state) is described with cluster operator, using Hartree-Fock reference state:

$$|\psi_{CC}\rangle = e^{\hat{T}}|\phi_{HF}\rangle,$$

here $\hat{T} = \hat{T}_1 + \hat{T}_2 + \hat{T}_3 + \dots$ is the cluster operator, that can be expanded into excitation operators $\hat{T}_1$ (single excitations), $\hat{T}_2$ (double excitations) etc. ([BM07]). UCCSD case approximates ground state, using only two terms of $\hat{T}$, them being:

$$\hat{T}_1 = \sum_{ia} t_i^a \, \hat{a}_a^\dagger \hat{a}_i,$$

$$\hat{T}_2 = \frac{1}{4} \sum_{ijab} t_{ij}^{ab} \, \hat{a}_a^\dagger \hat{a}_b^\dagger \hat{a}_j \hat{a}_i,$$

here t_{ij}^{ab} and t_i^a are excitation amplitudes (unknowns that can be found by solving Schrödinger's equation), $\hat{a}_i$ and $\hat{a}_i^\dagger$ are previously mentioned fermionic creation and annihilation operators. To implement CC ansatz on quantum computer, we need to make the exponential operator unitary. This way it becomes UCC (Unitary coupled-cluster) ansatz ([ASA+22]):

$$|\psi_{UCC}\rangle = e^{T - T^\dagger}|\phi_{HF}\rangle.$$

Again, it accurately approximates ground state with only single and double excitations, becoming UCCSD:

$$|\psi_{UCCSD}\rangle = e^{(T_1+T_2)-(T_1^\dagger+T_2^\dagger)}|\phi_{HF}\rangle.$$

For fermionic creation and annihilation operators we would use the same mapping as for Hamiltonian (see Section 2.1). To create a circuit on quantum computer we need to split the exponential operator into multiple operators (steps):

$$U(\boldsymbol{\theta}) = U_1(\boldsymbol{\theta})U_2(\boldsymbol{\theta})U_3(\boldsymbol{\theta})...,$$

however, as some operators in $\hat{T}$ expansion might be non-commuting, in general case (if at least one operator is non-commuting) we cannot expand it by separating terms:

$$e^{\hat{A}+\hat{B}+\hat{C}+\dots} \neq e^{\hat{A}} e^{\hat{B}} e^{\hat{C}} \dots$$

For that we use Trotter decomposition ([ASA+22]):

$$e^{\hat{T}-\hat{T}^\dagger} = e^{\sum_i \theta_i(\hat{T}_i-\hat{T}_i^\dagger)} \approx \left(\prod_i e^{\frac{\theta_i}{t}(\hat{T}_i-\hat{T}_i^\dagger)}\right)^t + \mathcal{O}\left(\frac{1}{t}\right),$$

here t is the order of decomposition (commonly chosen as $t = 1$), θ_i are the free parameters. Calculated gate count scaling is $\mathcal{O}((N-\eta)^2\eta^2)$, circuit depth scales as $\mathcal{O}((N-\eta)^2\eta)$ ([LHH+18]), where N is the number of spin-orbitals in the active space, η is the number of electrons. Since the UCCSD method is very common, many other papers have analyzed the ground state energy computation of small molecules using simulators ([ABD+20], [QX23]). The results from other studies are used as a comparison for the experiment of this study.

2.2.2. k-UpCCGSD

k-UpCCGSD (k-Unitary product coupled-cluster with generalized singles and doubles) ([LHH+18]) is the UCCSD improvement, increasing accuracy, however, at computational cost ([FPG+22]). Differently from UCCSD, it has generalized singles and sparse generalized doubles. Increasing k value yields more accurate results, while the computational cost also scales (linearly). This is the formula for generating the state:

$$|\psi\rangle = \prod_{\alpha=1}^{k}\left(e^{\hat{T}^{(\alpha)}-\hat{T}^{(\alpha)\dagger}}\right)|\phi_{HF}\rangle,$$

here k is the number of unitary clusters. Calculated gate count scaling is $\mathcal{O}(kN^2)$, circuit depth scales as $\mathcal{O}(kN)$ ([LHH+18]), where N is the number of spin-orbitals in the active space, k is the number of products in k-UpCCGSD function.

2.2.3. ADAPT-VQE

One of the most performative variations of VQE is ADAPT-VQE (Adaptive derivative-assembled pseudo-Trotter ansatz variational quantum eigensolver) ([GEB+19]) - an adaptive algorithm, that iteratively grows the ansatz. This way the depth of ansatz is significantly reduced, chemical accuracy is increased. The algorithm works by defining the "Operator Pool", selecting from one-, two body- and higher body operators by choosing (recommended) the highest gradient of the energy functional with respect to the operator parameter ([TCC+22]). This way the generated ansatz contains only the most contributing to the correlation energy operators, excluding many near-zero contribution terms. Also, accuracy of the ansatz can be controlled by changing the convergence criteria.

We acknowledge the recent improvements of this adaptive algorithm [TSB+21], [FHT+23], [ACM+24], however, since the analysis of the VQE algorithm in this bachelor thesis is broad, we do not conduct experimental analysis on the more specific variations of the algorithm.

2.3. Classical optimizers

The last part of VQE is the classical optimization subroutine that chooses parameters to pass for ansatz to generate a trial state. The most common choice is L-BFGS-B (Limited-memory Broyden–Fletcher–Goldfarb–Shanno bound constrained) algorithm ([FPG+22], [TCC+22], [QX23]). However, there are also other optimization algorithms that are used in VQE. Them being COBYLA (Constrained optimization by linear approximation) ([LAS+22]), SPSA (simultaneous perturbation stochastic approximation) ([MPP+22]) and others. Often the optimizer is chosen without much focus, since the performance of mentioned optimizers is similar and their choosing does not directly improve the quantum algorithm. However, in 2022 study ([LTM+20]) we can see BFGS performing better in noiseless environments, COBYLA performing better in low-noise, but also underperforming with higher noise. In this study we will briefly compare L-BFGS-B and COBYLA optimizers.

3. Methodology

From the literature analysis we found a few methods used for each step of VQE. Those methods have different performance evaluations, are used in different cases of VQE implementation. After the literature analysis, the experiments with comparative analysis are conducted, their results compared with the literature findings, VQE performance is benchmarked.

3.1. Hardware and tools

Classical calculations, including Aer simulator, are done on a computer with specifications, listed in Table 1. Calculations are done using Python version 3.13.2 with Qiskit and other supplementary Qiskit or Qiskit community libraries (such as Qiskit Nature). k-UpCCGSD ansatz preparation is done using Python library Tequila ([KAT+25]) with Madness backend. Virtual environments created with built-in Venv module, some parts use Conda virtual environment management system. Calculations are done in Jupyter notebook format.

Component	**Specification**
CPU	AMD Ryzen 5 7530U with Radeon Graphics
RAM	14 GiB
GPU	AMD/ATI Barcelo (rev c5)
Operating System	Ubuntu 24.04.2 LTS
Storage	477 GB NVMe SSD

Table 1. Specifications of the local machine used for simulation and analysis.

For quantum hardware calculations the IBM Quantum hardware API, ibm_brisbane version 1.1.115 device is used. Specifications can be seen in Table 2.

Property	**Value**
Qubits	127
Processor type	Eagle r3
Region	us-east
Basis gates	ECR, ID, RZ, SX, X
Median T_1 time	222.77 μs
Median T_2 time	131.59 μs
Median SX error	2.56×10^{-4}
Median readout error	1.733×10^{-2}
Median ECR error	6.942×10^{-3}
2Q error (best)	3.33×10^{-3}
2Q error (layered)	1.68×10^{-2}
CLOPS	180K

Table 2. Specifications of the used IBM backend (ibm_brisbane).

3.2. Hamiltonian mappings

For Hamiltonian mappings, we conduct experimental comparison analysis and consider these cases:

1. Jordan-Wigner mapping, yielding a certain qubit Hamiltonian,
2. Bravyi-Kitaev mapping, yielding a certain qubit Hamiltonian,
3. Parity mapping with qubit reduction for number of particles, yielding a certain qubit Hamiltonian,

These graphs are displayed:

1. maximum Pauli weight, depending on qubit count (number of fermionic modes, i.e. molecule complexity),
2. if generated Hamiltonians are different enough, for H_2, LiH, BeH_2 data full VQE benchmark graphs.

3.3. Ansätze

For the ansätze, we collect experimental data for these cases:

1. UCCSD ansatz, L-BFGS-B optimizer, Jordan-Wigner mapping, Aer simulator,
2. k-UpCCGSD ansatz with $k \in \{1; 2\}$, L-BFGS-B optimizer, Jordan-Wigner mapping, Aer simulator,
3. ADAPT-VQE method, L-BFGS-B optimizer, Jordan-Wigner mapping, Aer simulator,
4. (only for H_2, due to the IBM Quantum hardware limitations) HEA, L-BFGS-B optimizer, Parity mapping, 2 qubit reduction, quantum hardware,
5. (for accuracy comparison) FCI, classical, precise calculation.

HEA for H_2, being the most basic parameterized quantum circuit of 2 qubits, is taken from the literature [CW23]. To run performance benchmarks for mentioned ansätze cases, we implement full VQE instances and depict these graphs:

1. converged ground state energy graph, depending on the inter-atomic distance,
2. accuracy, with reference of FCI calculations, absolute error (chemical accuracy being absolute error **1 kcal/mol ≈ 0.043 eV ≈ 1.593 × 10^{-3} Hartree (Eh)**, not percent error, [SH06]), depending on the inter-atomic distance,
3. number of iterations needed to achieve convergence (gradient tolerance is 10^{-5}), depending on the inter-atomic distance,
4. graph of the iterations for computation convergence (if implementation is not with built-in tools),
5. runtime, depending on the inter-atomic distance.

Mentioned full VQE benchmark graphs are computed for H_2, LiH, BeH_2 molecules. We acknowledge that some of the mentioned calculations can be not possible to implement due to the simulator and quantum hardware limitations.

3.4. Classical optimizers

To test classical optimizer performance, we choose the UCCSD ansatz for H_2 molecule and test these algorithms:

1. L-BFGS-B,
2. COBYLA.

3.5. Code and results

Exact code implementation with setup instructions, short explanations and exact results data are accessible on GitHub platform [Šul25].

4. Results and discussion

The conducted experiment yielded the following results, achieving the aim of the study:

1. Hamiltonian mapping choice doesn't make difference for small molecules, for larger one's Bravyi-Kitaev mapping scales slightly better,
2. the best computation method for VQE (for small molecules, running on a simulator) is k-UpCCGSD ($k = 1$) ansatz,
3. choosing classical optimizer for small molecules doesn't make difference.

4.1. Hamiltonian mappings

In figures 3 and 4 Hamiltonian mappings are compared by their Pauli weights and the runtime it takes to generate the qubit Hamiltonian. As it can be seen from the graphs, for small molecules choosing of the mapping doesn't make a difference. However, when scaling to higher fermionic modes, it can be seen that Bravyi-Kitaev mapping yields a little bit of advantage over Jordan-Wigner, as it is also mentioned in the literature [TCC+22]. On the other hand, Parity mapping yields very similar results to Bravyi-Kitaev, which is not expected. And since small molecule data yields very similar results in all mappings, the full VQE algorithm run is not executed for comparison.

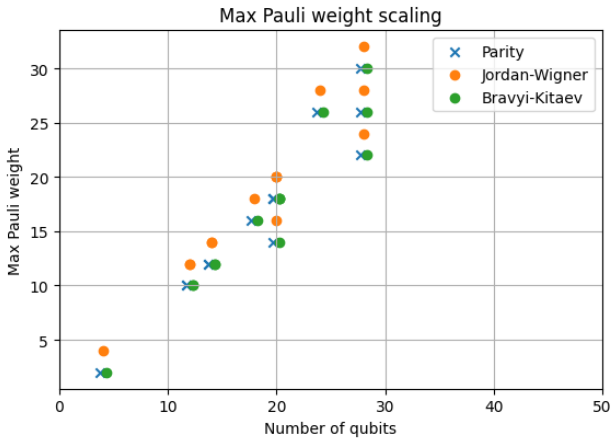

Figure 3. Max Pauli weight scaling comparison with different mappings

4.2. Ansätze

Ansätze performance comparison is done by grouping results by molecule data.

4.2.1. H_2

In figure 5 results of the full VQE runs are depicted, showing the comparison between ansätze generation method in the case of H_2 molecule. It can be seen that for bigger bond lengths accuracy is much better in all computation methods (comparing to FCI). However, when small bond lengths are considered, accuracy significantly falls behind for ADAPT-VQE, UCCSD and

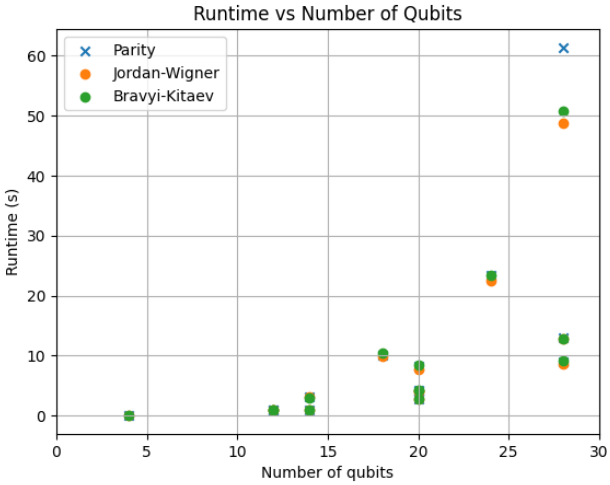

Figure 4. Runtime comparison with different mappings

HEA methods. The most accurate is the 1-UpCCGSD ansatz case, also requiring more runtime than other cases. Even with a very small error calculations, 1-UpCCGSD still doesn't reach the chemical accuracy (expected $< 1.6 \times 10^{-3}$ Hartree ([MEA+20])

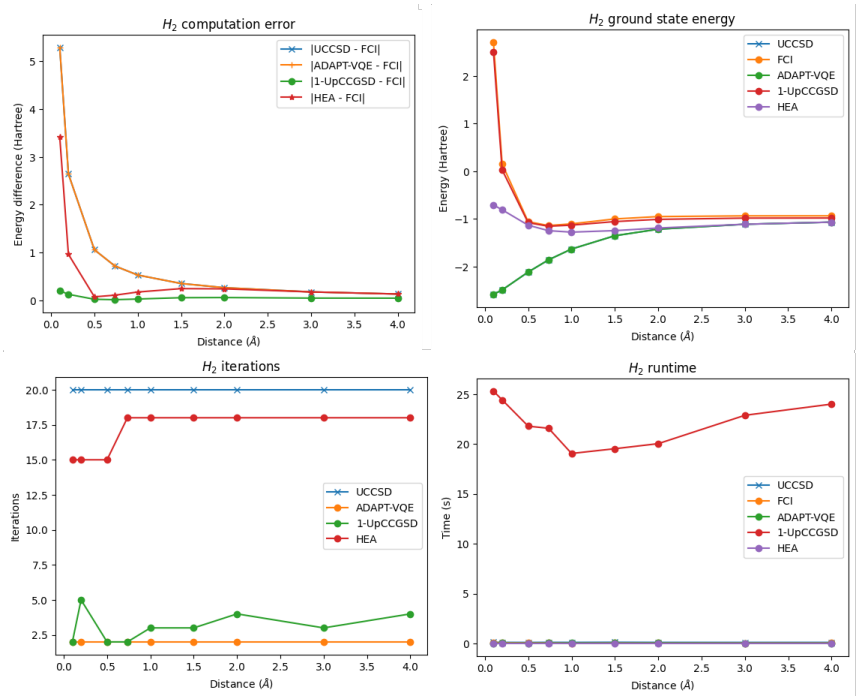

Figure 5. H_2 ground state energy computation comparison with different ansätze and methods

In figure 6 results of single bond length hydrogen molecule ground state energy calculation are depicted, comparing the convergence on simulator and on quantum hardware. Each cost function evaluation on quantum computer took approximately the same amount - 13 seconds. It can be seen that the hardware calculated energies are converging more noisy. Because of this, the converging criteria of 10^{-5} gradient tolerance wasn't achieved and the free limits of 10 minutes of IBM Quantum hardware were exceeded. To effectively conduct experiments on IBM Quantum hardware, it is recommended to set lower gradient tolerance.

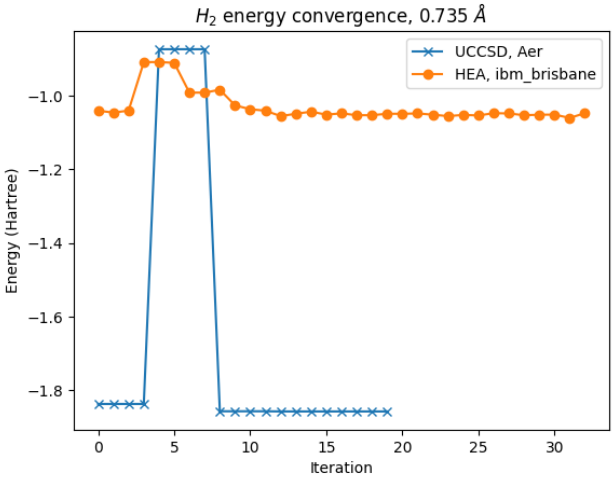

Figure 6. H_2 energy convergence on Aer simulator vs IBM Quantum hardware

4.2.2. LiH

Figure 7 shows the comparison of VQE computation methods for LiH molecule case. The results are pretty similar to the ones of hydrogen - ADAPT-VQE was the least precise, UCCSD also showed significant error deviations for small bond lengths. The reason why the UCCSD ansatz case wasn't implemented for all bond lengths is because the runtime for UCCSD on a simulator is significantly larger than of any other method.

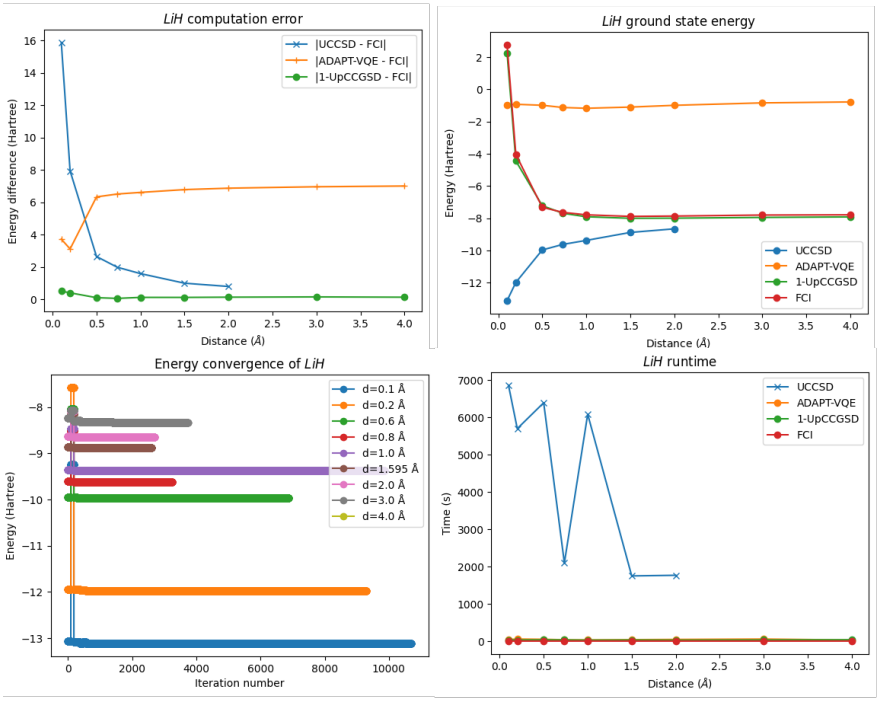

Figure 7. LiH ground state energy computation comparison with different ansätze and methods

4.2.3. BeH_2

Figure 8 displays BeH_2 molecule ground state energy computation data for comparing different methods. The results are pretty similar to the previously mentioned cases. ADAPT-VQE performed the worst, 1-UpCCGSD - the best, but still not accurately enough.

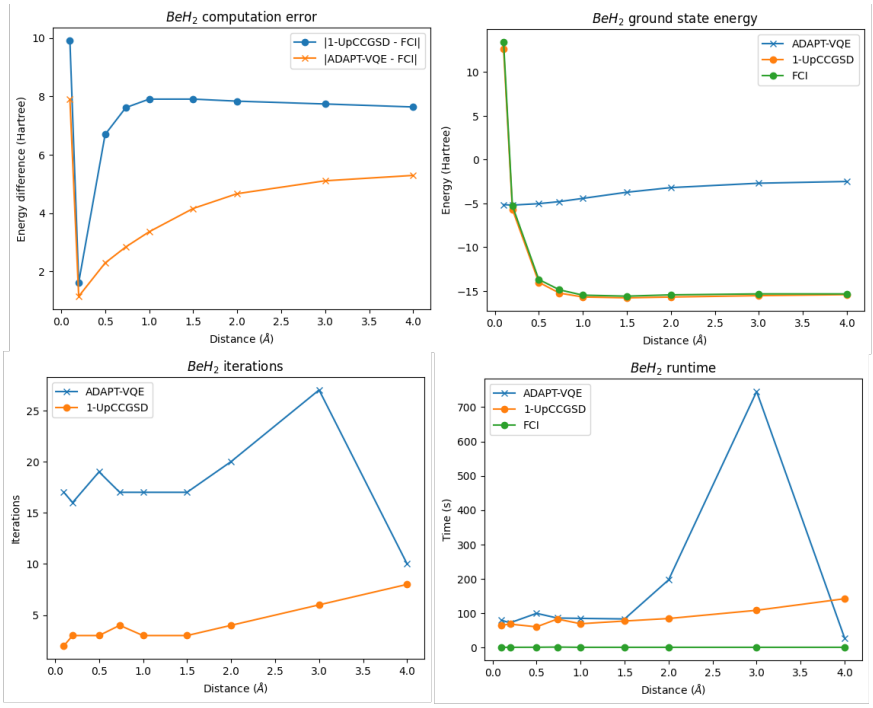

Figure 8. BeH_2 ground state energy computation comparison with different ansätze and methods

Figure 9 displays unfinished calculation of 0.1 bond distance BeH_2 ground state energy. It can be seen that it converges poorly, often "spiking" up.

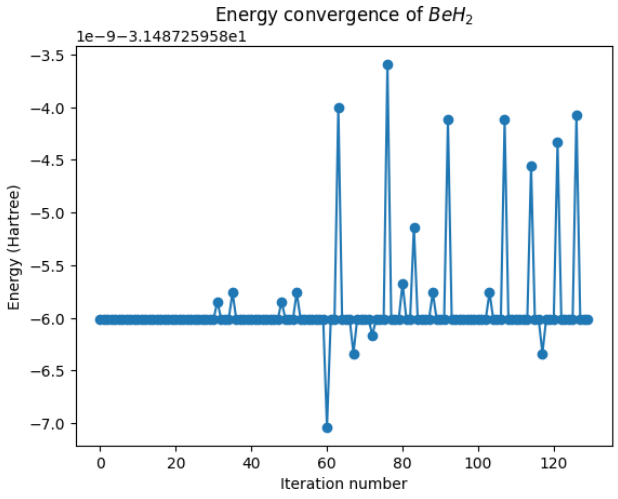

Figure 9. BeH_2 computation iterations for 0.1 Å bond distance

4.3. Molecule complexity benchmark

Unfortunately, running all the ansätze cases was not successful for all - H_2, LiH and BeH_2 - molecule data due to the computational limits. This greatly depicts the limitations of quantum simulators and NISQ quantum hardware. None of the cases achieved quantum advantage, some of them were unreasonable to even compute: 2-UpCCGSD ansatz didn't yield any results for even one inter-atomic distance after running the most basic hydrogen molecule case for more than an hour. Thorough details of the failing cases are not being discussed in this study, however, investigating problems and evaluating the scaling is suggested for the future research.

4.4. Optimizers

Finally, figure 10 depicts UCCSD ansatz VQE computation of hydrogen ground state energy, when implementing COBYLA and L-BFGS-B optimizers in comparison. Results show that for such simple molecule like hydrogen, the choice of optimizer does not matter.

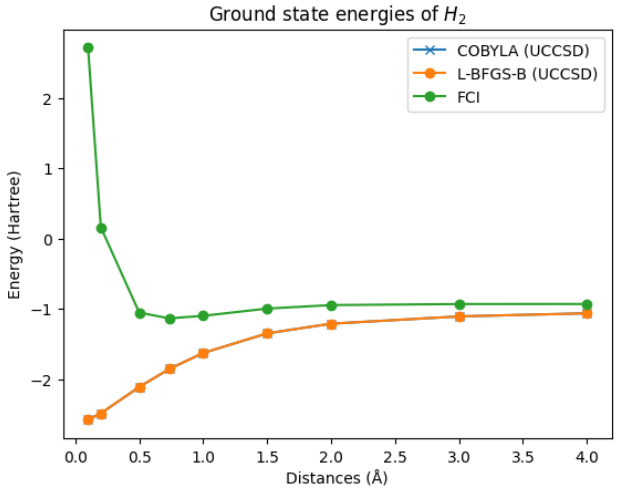

Figure 10. BeH_2 ground state energy computation comparison with different ansätze and methods

5. Conclusion

This thesis investigated the VQE algorithm implementation for quantum chemistry problems, analyzing the methods and their improvements described in the literature. An experiment with simulators and quantum hardware was done, testing the literature findings about the most common and performing Hamiltonian mappings, ansätze and their generation methods, classical optimizers.

The results of this thesis show that using currently commercially accessible quantum hardware does not yield quantum advantage for VQE algorithm on small molecules. When considering the simulators, the best VQE variation is the one that uses k-UpCCGSD ($k = 1$) ansatz for computing. However, FCI still holds the best performance for classical algorithms. The choice of Hamiltonian mapping and classical optimizer algorithm for small molecules doesn't show difference.

The goal of this study was to analyze the performance of VQE with various molecules, comparing the existing methods and improvements in research, testing the findings with an experiment, collecting more research data. Therefore, the goal is successfully reached. More detailed information on the computation methods used in this thesis together with the numerical data can be found on GitHub platform [Šul25].

However, this study has some limitations. Firstly, the experiment data set is relatively small - data is collected only for 3 molecule cases with two types of ansätze and another generative VQE variation. Secondly, analysis is very broad, therefore it does not dive deep into the algorithm specifics and very recent improvements on certain subroutines. Because of these limitations, this study findings should be interpreted with a caution.

Quantum computing field stays very relevant in today's research, and possibly will be more relevant in the future. Therefore, it is important to keep improving the quantum algorithms, VQE being one of the most important one's. Following this study, future research could expand more on the chemistry part of the algorithm, combining electronic Hamiltonian (or other approximation) construction with the quantum algorithms, since currently in literature the chemistry part of the VQE is often being ignored, focusing on the quantum computer implementation. Future research could also include more thorough analysis of other optimization algorithms, testing their capabilities. Analysis could be done also in more noisy simulations, since the quantum hardware is not yet in fault tolerant era.

References

[AAA+25] D. Aasen, M. Aghaee, Z. Alam, M. Andrzejczuk, et al. *Roadmap to fault tolerant quantum computation using topological qubit arrays* [arXiv preprint arXiv:2502.12252]. 2025. Available also from: https://arxiv.org/abs/2502.12252.

[ABD+20] V. Armaos, D. A. Badounas, P. Deligiannis, K. Lianos. Computational chemistry on quantum computers: Ground state estimation. *Applied Physics A*. 2020, volume 126, number 8, p. 625.

[Abu24] M. AbuGhanem. *IBM quantum computers: Evolution, performance, and future directions* [arXiv preprint arXiv:2410.00916]. 2024. Available also from: https://arxiv.org/abs/2410.00916.

[ACM+24] P. G. Anastasiou, Y. Chen, N. J. Mayhall, E. Barnes, S. E. Economou. TETRIS-ADAPT-VQE: an adaptive algorithm that yields shallower, denser circuit ansätze. *Physical Review Research*. 2024, volume 6, number 1, p. 013254.

[AF11] P. W. Atkins, R. S. Friedman. *Molecular quantum mechanics*. Oxford university press, 2011.

[ASA+22] A. Anand, P. Schleich, S. Alperin-Lea, P. W. Jensen, et al. A quantum computing view on unitary coupled cluster theory. *Chemical Society Reviews*. 2022, volume 51, number 5, pp. 1659–1684.

[Bac93] V. Bach. Accuracy of mean field approximations for atoms and molecules. *Communications in mathematical physics*. 1993, volume 155, pp. 295–310.

[BGM+17] S. Bravyi, J. M. Gambetta, A. Mezzacapo, K. Temme. Tapering off qubits to simulate fermionic Hamiltonians. *arXiv preprint arXiv:1701.08213*. 2017.

[BK02] S. B. Bravyi, A. Y. Kitaev. Fermionic quantum computation. *Annals of Physics*. 2002, volume 298, number 1, pp. 210–226.

[BM07] R. J. Bartlett, M. Musiał. Coupled-cluster theory in quantum chemistry. *Reviews of Modern Physics*. 2007, volume 79, number 1, pp. 291–352.

[CAB+21] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, et al. Variational quantum algorithms. *Nature Reviews Physics*. 2021, volume 3, number 9, pp. 625–644. Available from: https://doi.org/10.1038/s42254-021-00348-9.

[Číž66] J. Čížek. On the correlation problem in atomic and molecular systems. Calculation of wavefunction components in Ursell-type expansion using quantum-field theoretical methods. *The Journal of Chemical Physics*. 1966, volume 45, number 11, pp. 4256–4266.

[CW23] B. Choy, D. J. Wales. Molecular energy landscapes of hardware-efficient ansatze in quantum computing. *Journal of chemical theory and computation*. 2023, volume 19, number 4, pp. 1197–1206.

[FGG14] E. Farhi, J. Goldstone, S. Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*. 2014.

[FHT+23] C. Feniou, M. Hassan, D. Traoré, E. Giner, Y. Maday, J.-P. Piquemal. Overlap-ADAPT-VQE: practical quantum chemistry on quantum computers via overlap-guided compact Ansätze. *Communications Physics*. 2023, volume 6, number 1, p. 192.

[FHZ+24] S. Fomichev, K. Hejazi, M. S. Zini, M. Kiser, et al. Initial state preparation for quantum chemistry on quantum computers. *PRX Quantum*. 2024, volume 5, number 4, p. 040339.

[FPG+22] D. A. Fedorov, B. Peng, N. Govind, Y. Alexeev. VQE method: a short survey and recent developments. *Materials Theory*. 2022, volume 6, number 1, p. 2.

[GAN14] I. M. Georgescu, S. Ashhab, F. Nori. Quantum simulation. *Reviews of Modern Physics*. 2014, volume 86, number 1, pp. 153–185.

[GEB+19] H. R. Grimsley, S. E. Economou, E. Barnes, N. J. Mayhall. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nature communications*. 2019, volume 10, number 1, p. 3007.

[GM19] G. G. Guerreschi, A. Y. Matsuura. QAOA for Max-Cut requires hundreds of qubits for quantum speed-up. *Scientific reports*. 2019, volume 9, number 1, p. 6903.

[GOK88] E. K. Gross, L. N. Oliveira, W. Kohn. Rayleigh-Ritz variational principle for ensembles of fractionally occupied states. *Physical Review A*. 1988, volume 37, number 8, pp. 2805–2808. Available from: https://doi.org/10.1103/PhysRevA.37.2805.

[HWB19] O. Higgott, D. Wang, S. Brierley. Variational Quantum Computation of Excited States. *Quantum*. 2019, volume 3, p. 156. ISSN 2521-327X. Available from: https://doi.org/10.22331/q-2019-07-01-156.

[JMH98] J. A. Jones, M. Mosca, R. H. Hansen. Implementation of a quantum search algorithm on a quantum computer. *Nature*. 1998, volume 393, number 6683, pp. 344–346. Available from: https://doi.org/10.1038/30694.

[JW93] P. Jordan, E. P. Wigner. *Über das paulische äquivalenzverbot*. Springer, 1993.

[KAT+25] J. S. Kottmann, S. Alperin-Lea, T. Tamayo-Mendoza, A. Cervera-Lierta, et al. *Tequila: A platform for rapid development of quantum algorithms*. Zenodo, 2025. Version v1.9.9. Available from: https://doi.org/10.5281/zenodo.15114692.

[KH89] P. J. Knowles, N. C. Handy. Unlimited full configuration interaction calculations. *Journal of Chemical Physics*. 1989, volume 91, number 4, pp. 2396–2398.

[KMT+17] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, J. M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*. 2017, volume 549, number 7671, pp. 242–246. Available from: https://doi.org/10.1038/nature23879.

[KVO+18] K. Kapusta, E. Voronkov, S. Okovytyy, V. Korobov, J. Leszczynski. Reconstruction of STO-3G family basis set for the accurate calculation of magnetic properties. *Russian Journal of Physical Chemistry A*. 2018, volume 92, pp. 2827–2834.

[KZU25] A. Karim, S. Zhang, M. Usman. Fast and Noise-aware Machine Learning Variational Quantum Eigensolver Optimiser. *arXiv preprint arXiv:2503.20210*. 2025.

[LAS+22] X. Liu, A. Angone, R. Shaydulin, I. Safro, Y. Alexeev, L. Cincio. Layer VQE: A variational approach for combinatorial optimization on noisy quantum computers. *IEEE Transactions on Quantum Engineering*. 2022, volume 3, pp. 1–20.

[LHH+18] J. Lee, W. J. Huggins, M. Head-Gordon, K. B. Whaley. Generalized unitary coupled cluster wave functions for quantum computation. *Journal of chemical theory and computation*. 2018, volume 15, number 1, pp. 311–324.

[LJJ+22] J. Larkin, M. Jonsson, D. Justice, G. G. Guerreschi. Evaluation of QAOA based on the approximation ratio of individual samples. *Quantum Science and Technology*. 2022, volume 7, number 4, p. 045014. Available from: https://doi.org/10.1088/2058-9565/ac6973.

[LTM+20] W. Lavrijsen, A. Tudor, J. Müller, C. Iancu, W. De Jong. Classical optimizers for noisy intermediate-scale quantum devices. In: *2020 IEEE international conference on quantum computing and engineering (QCE)*. IEEE, 2020, pp. 267–277.

[MBK+20] M. Metcalf, N. P. Bauman, K. Kowalski, W. A. De Jong. Resource-efficient chemistry on quantum computers with the variational quantum eigensolver and the double unitary coupled-cluster approach. *Journal of chemical theory and computation*. 2020, volume 16, number 10, pp. 6165–6175.

[MEA+20] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, X. Yuan. Quantum computational chemistry. *Reviews of Modern Physics*. 2020, volume 92, number 1, p. 015003.

[MMJ+22] H. Mustafa, S. N. Morapakula, P. Jain, S. Ganguly. Variational Quantum Algorithms for Chemical Simulation and Drug Discovery. In: *2022 International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT)*. IEEE, 2022, pp. 1–8. Available from: https://doi.org/10.1109/TQCEBT56225.2022.10054360.

[MMN+20] W. Mizukami, K. Mitarai, Y. O. Nakagawa, T. Yamamoto, T. Yan, Y.-y. Ohnishi. Orbital optimized unitary coupled cluster theory for quantum computer. *Physical Review Research*. 2020, volume 2, number 3, p. 033421.

[MOS+21] T. Miki, R. Okita, M. Shimada, J. I. Shirakashi. Combinatorial Optimization with Variational Approaches on Noisy Quantum Devices. In: *2021 IEEE International Conference on Manipulation, Manufacturing and Measurement on the Nanoscale (3M-NANO)*. IEEE, 2021, pp. 214–217. Available from: https://doi.org/10.1109/3M-NANO52173.2021.9599740.

[MPP+22] I. Miháliková, M. Pivoluska, M. Plesch, M. Friák, D. Nagaj, M. Šob. The cost of improving the precision of the variational quantum eigensolver for quantum chemistry. *Nanomaterials*. 2022, volume 12, number 2, p. 243.

[OS24] O. O'Brien, S. Strelchuk. Ultrafast hybrid fermion-to-qubit mapping. *Physical Review B*. 2024, volume 109, number 11, p. 115149.

[PMS+14] A. Peruzzo, J. McClean, P. Shadbolt, M. H. Yung, X. Q. Zhou, P. J. Love, A. Aspuru-Guzik, J. L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*. 2014, volume 5, number 1, p. 4213. Available from: `https://doi.org/10.1038/ncomms5213`.

[QX23] M. Qing, W. Xie. Use VQE to calculate the ground energy of hydrogen molecules on IBM Quantum. *arXiv preprint arXiv:2305.06538*. 2023.

[SH06] J. S. Sims, S. A. Hagstrom. High precision variational calculations for the Born-Oppenheimer energies of the ground state of the hydrogen molecule. *The Journal of chemical physics*. 2006, volume 124, number 9.

[SST+21] K. Sugisaki, C. Sakai, K. Toyota, K. Sato, D. Shiomi, T. Takui. Quantum algorithm for full configuration interaction calculations without controlled time evolutions. *The Journal of Physical Chemistry Letters*. 2021, volume 12, number 45, pp. 11085–11089.

[STH+22] Y. Shee, P.-K. Tsai, C.-L. Hong, H.-C. Cheng, H.-S. Goan. Qubit-efficient encoding scheme for quantum simulations of electronic structure. *Physical Review Research*. 2022, volume 4, number 2, p. 023154.

[Šul25] N. Šuliokas. *Repository on GitHub platform: Code implementation and exact data* [`https://github.com/naglissul/vqe-analysis-paper`]. 2025. Accessed: 2025-05-23.

[SW18] M. Steudtner, S. Wehner. Fermion-to-qubit mappings with varying resource requirements for quantum simulation. *New Journal of Physics*. 2018, volume 20, number 6, p. 063010. Available from: `https://doi.org/10.1088/1367-2630/aac54f`.

[TCC+22] J. Tilly, H. Chen, S. Cao, D. Picozzi, et al. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*. 2022, volume 986, pp. 1–128.

[TLM+18] A. Tranter, P. J. Love, F. Mintert, P. V. Coveney. A comparison of the Bravyi–Kitaev and Jordan–Wigner transformations for the quantum simulation of quantum chemistry. *Journal of chemical theory and computation*. 2018, volume 14, number 11, pp. 5617–5630.

[TSB+21] H. L. Tang, V. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes, S. E. Economou. qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor. *PRX Quantum*. 2021, volume 2, number 2, p. 020310.

[TSS+15] A. Tranter, S. Sofia, J. Seeley, M. Kaicher, et al. The B ravyi–K itaev transformation: Properties and applications. *International Journal of Quantum Chemistry*. 2015, volume 115, number 19, pp. 1431–1441.

[WHB19] D. Wang, O. Higgott, S. Brierley. Accelerated variational quantum eigensolver. *Physical review letters*. 2019, volume 122, number 14, p. 140504.

[YLS+25] J. Yu, Y. Liu, S. Sugiura, T. Van Voorhis, S. Zeytinoğlu. Clifford circuit based heuristic optimization of fermion-to-qubit mappings. *arXiv preprint arXiv:2502.11933*. 2025.