UNIVERSITÀ
DEGLI STUDI
DI PADOVA

University of Padova

Department of Chemical Sciences

Ph.D. Course in Molecular Sciences

Curriculum: **Chemical Sciences**

Cycle: **XXXVI**

Quantum algorithms design for molecular electronic structure theory

Supervisor: **Prof. Stefano Corni**
Ph.D. Student: **Davide Castaldo**

Academic Year: **2023/2024**

Contents

Introduction

Since I started working on my doctoral thesis, I have been fortunate enough to witness numerous milestones in scientific computing[1–3].

None of these results is inherently something we did not already know how to do, yet they have not left us any less surprised. The foundation of these discoveries lies in the recent understanding among the scientific community that to overcome certain barriers related to solving some problems, we need to adopt integrated techniques: hardware (the tool with which we solve the problem), algorithms (how we solve the problem), and software (how we apply the solving strategy) are closely interconnected. Although this approach is very reasonable, it can require significant time and effort. We are witnessing the culmination of these efforts in the field of artificial intelligence, where the initial algorithmic work dates back to the 1970s and 1980s, and today, with the development of specialized processors, these algorithms can be applied with extraordinary success[4].

This thesis is inspired by this approach. Indeed, it is unfair to ask machines designed for a particular purpose to serve as universal solving tools, even if limited to a specific application like that of this thesis: solving the Schrödinger equation for a system of electrons in a molecule. In the three central chapters of this thesis, three algorithms are proposed to solve specific issues in this domain. Each of these proposals is made with a particular quantum device in mind. Generally speaking, we refer as quantum device to any quantum system which is controllable in order to accomplish a given task. In Chapter 1 we expand this simple definition giving an introduction to quantum computers, their basic usage and applications to the simulation of molecular systems.

Moving onto Chapter 2, we discuss a proposal to use an optimal control procedure to find the temporal evolution capable of preparing the ground state of a molecule. This method is very similar to the more general variational approach proposed in the VQE[5], but it is primarily intended to be implemented on an analog quantum simulator of a molecule. These types of devices, still in a preliminary developmental stage[6], cannot solve any problem directly but can solely simulate a molecule through engineered quantum systems. The reason this is the chosen platform for this method is that, having a tool specifically designed to simulate the evolution of a particular Hamiltonian, it is reasonable to think that this task can be accomplished much faster than implementing the same routine on a universal quantum computer.

Another principle underlying my thesis work concerns the complexity of chemistry. While our goal is to develop algorithms that match the pace of hardware advancements, allowing for efficient solutions, there is also a growing interest in capturing more intricate phenomena. In other words, we aim not only to achieve efficiency higher than standard hardware and algorithms but also to address and match the complexity level they offer.

This perspective differs from much of the existing literature, which primarily emphasizes the efficiency of calculations representing single-point energies at *chemical accuracy*. In the subsequent sections of this thesis, I present algorithms designed to move beyond traditional approaches, aiming to catch up the intricacies of *chemical complexity*.

Thus, in the third chapter, I propose a strategy to unify a generic variational approach with polarizable continuum models[7] to extend the simulation of molecules from the gaseous phase to the condensed phase. In this chapter, we go through the first example of a quantum simulation of a molecule in a solution. This contribution is mainly geared towards Noisy Intermediate Scale Quantum (NISQ) devices, which are prototypes of digital quantum computers where noise still poses an obstacle that cannot be corrected (but can be mitigated) in information processing and reading.

Finally, in the fourth chapter, with the perspective of integrating different programming algorithms and paradigms, I propose an extension of Quantum Phase Estimation (QPE) that is integrable into software supporting automatic differentiation. The latter is a technique developed in the field of machine learning to efficiently compute arbitrary-order derivatives of complex functions, such as those represented in deep neural networks. In the chemical realm, the application of these methodologies has facilitated the calculation of spectroscopic observables and the development of geometry optimization procedures for different electronic structure methods[8]. As of today, QPE is the algorithm of choice for calculating energies of molecular Hamiltonians when a fault-tolerant quantum computer will be available. In this chapter, it is developed an algorithm designed for quantum architectures that are elaborated and expanded with qubit numbers orders of magnitude higher than those currently available.

Efficient quantum backpropagation techniques have not yet been developed, so scaling this algorithm using available quantum circuit differentiation methods is unfavorable. Nevertheless, I demonstrate the development of a pipeline for molecular geometry optimization and pinpoint the obstacles that are yet to be solved to efficiently scale up automatically differentiable interference methods[9].

Finally, in the last chapter I summarize the results presented and highlight possible avenues of these works.

References

(1) Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J. C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F. G.; Buell, D. A., et al. *Nature* **2019**, *574*, 505–510.

(2) Pederson, R.; Kozlowski, J.; Song, R.; Beall, J.; Ganahl, M.; Hauru, M.; Lewis, A. G.; Yao, Y.; Mallick, S. B.; Blum, V., et al. *Journal of Chemical Theory and Computation* **2022**, *19*, 25–32.

(3) Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A., et al. *Nature* **2021**, *596*, 583–589.

(4) Bubeck, S.; Chandrasekaran, V.; Eldan, R.; Gehrke, J.; Horvitz, E.; Kamar, E.; Lee, P.; Lee, Y. T.; Li, Y.; Lundberg, S., et al. *arXiv preprint arXiv:2303.12712* **2023**.

(5) Peruzzo, A.; McClean, J.; Shadbolt, P.; Yung, M.-H.; Zhou, X.-Q.; Love, P. J.; Aspuru-Guzik, A.; O'brien, J. L. *Nature communications* **2014**, *5*, 4213.

(6) Argüello-Luengo, J.; González-Tudela, A.; Shi, T.; Zoller, P.; Cirac, J. I. *Nature* **2019**, *574*, 215–218.

(7) Tomasi, J.; Mennucci, B.; Cammi, R. *Chemical reviews* **2005**, *105*, 2999–3094.

(8) Abbott, A. S.; Abbott, B. Z.; Turney, J. M.; Schaefer III, H. F. *The journal of physical chemistry letters* **2021**, *12*, 3232–3239.

(9) Lin, L. *arXiv preprint arXiv:2201.08309* **2022**.

Chapter 1

Methods

The purpose of this section is to establish the main tools and set up the definition for quantities and terms used throughout the rest of this thesis. To this end we will discuss the main concepts of quantum computation starting from the definition of a quantum gate to end up with the building blocks of quantum simulation.

1.1 Introduction to quantum computation

The initial proposals for a paradigm shift in computation, going beyond classical computer science, trace back to the seminal works of Feynman and Deutsch[1, 2]. They argued for leveraging quantum mechanics as a basis for a novel computing approach, suggesting substantial speedups in simulating quantum systems. Over the following decades, various algorithms demonstrating exponential speed enhancements compared to classical solutions have emerged[3, 4]. Currently, a fervent pursuit of achieving a quantum advantage is underway[5–7], aiming to realize a quantum computer capable of tackling computations deemed intractable for our most powerful classical supercomputers. At the same time, it is interesting to notice that as these first experiments were accomplished, a plethora of works have been published pushing the boundaries of classical simulation of quantum systems[8–11].

As most of the work on quantum computing requires the manipulation (i.e. programming) of the information at a deep level of the hardware, we start with an example of classical computation to demonstrate how quantum computation is a logical extension of the classical one and how the two differ for the physical regime at which we manipulate information.

In classical computers the fundamental unit of information is the bit which can assume two logical values, namely, $\{0,\ 1\}$. The physical implementation of this entity can be very different (as we will see, heading to the quantum realm, such a variety of choices will be mantained) but for sake of clarity we will consider only the classical **circuit model** of computation.

In the circuit model of computation the information, i.e. bit values, is stored as electrical voltage at the input/output of a circuit. In these kind of architectures circuits that perform standard logical operations are called **gates**. Classical gates implement all the fundamental constructions of logic, as an example, the *if-else* clause or the logical operations *and/or*. As long as all the fundamental arithmetical operations can be decomposed in terms of logical operations, classical circuits, i.e., gates, can be used as the foundation for classical computers. As an example, the numerical addition (modulo 2) of two bits is performed setting up a circuit where the eXclusive OR (XOR) gate and the AND gate are applied to the input qubits to output, respectively, a bit S (representing the sum) and a bit C (representing the carry-over).

The last example makes a clear connection between the bit as unit of information and its physical implementation; for this reason from now on we will not make any distinction between them.

Moving from the classical model to the quantum analogue we still preserve a circuit-like paradigm of computation (even though other possibilities are available[12–14]) although the nature of the unit of information has fundamentally changed. The main idea in quantum computing is to store the information in a **quantum bit** (from now on simply, the qubit); logical states are no more encoded in classical $\{0,\ 1\}$ states corresponding to electrical voltages but in $\{|0\rangle\ , |1\rangle\}$ states

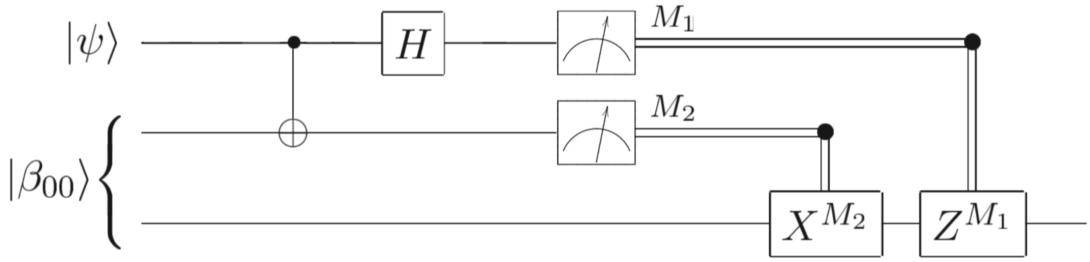


Figure 1.1: Example of quantum circuit diagram. Reproduced from [15].

of a two-level system. Like any quantum system, a qubit can be found in a coherent superposition of such states:

$$|\phi\rangle = a_0|0\rangle + a_1|1\rangle \tag{1.1}$$

Where the coefficients $\{a_0\ ,\ a_1\}$ satisfy the normalization condition $\sum_i |a_i|^2 = 1$. Such a basis to describe a two level system is commonly known in the field of quantum computation as **computational basis**.

All the differences between the qubit and the bit affect and determine distinctions between classical and quantum computation. As already mentioned, logical operations are performed through the action of classical gates; analogously we define a **quantum gate** as the operation performed by the computer at the qubit(s) level. Due to the nature of the qubit a quantum gate is formally described as an external perturbation which steers the system according to a particular task. Such a description encompasses the so called circuit model of quantum computation. As much as there are specific rules to read a classical circuit recognizing symbols for resistances, capacitors and other circuit elements, there are specific conventions to represent a quantum circuit. We refer to Fig. 1.1 to explain the basic rules to interpret these depictions.

The set of qubits on which we perform logical operations is called **quantum register**. In the figure above different labels of the input qubits stand to highlight the difference between the role of the different qubits in a given algorithm. This implies that in the same quantum device we can distinguish different sets of qubits (different quantum registers) having different roles. It is important to notice that, even though it is not the case of the circuit above, by convention the initial state of the quantum register is $|00...0\rangle$, i.e. the tensor product of each qubit in the state $|0\rangle$. In Fig. 1.1 we can see that any operation (i.e., any gate) performed on a qubit (or a set of qubits) is depicted as a box; in Fig. 1.2 we report a summary of the most important quantum gates and quantum circuital elements. The label in the box specifies the kind of action that will be performed.

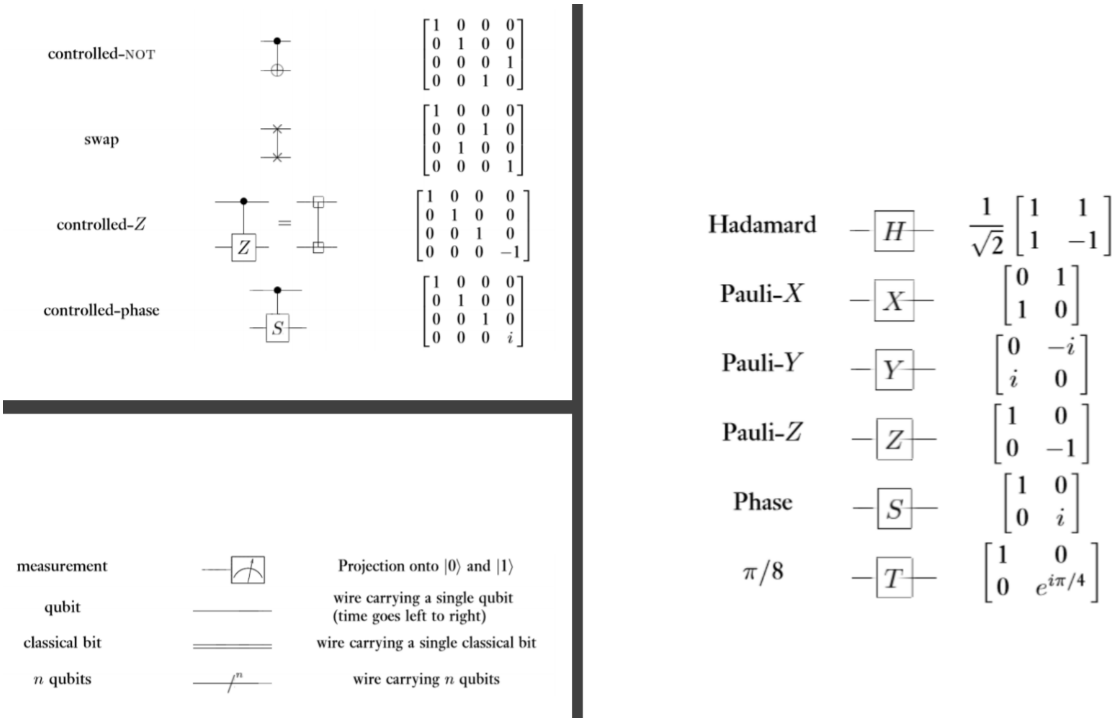


Figure 1.2: Summary of the most important quantum gates and circuital elements. Adapted from [15].

As we already know, in most cases quantum mechanical operator do not commute, for this reason is compulsory, in order to have a clear representation of the quantum dynamics underlying a given algorithm, to specify the order of application of each gate. This is accomplished by means of **quantum wires**.

Each wire is not to be meant as a physical component but nothing more than the arrow of time (from left to right) defining the order of application for each gate in a quantum circuit diagram. We may notice that the order given in the circuit diagram representation for the application of the gates is *chronological*, the direct transposition of a quantum circuit in terms of equations means that the first gate from the left in the circuit diagram is the rightmost matrix acting on the vector representing the quantum register.

Returning to Fig. 1.1, the box depicting a meter inside stands for the operation of measurement of the single qubit state. This operation results in a collapse of the wavefunction to a classical value (i.e. to a bit) represented as a double wire. As we will see in more detail afterwards, the black dots, or more generally, lines connecting gates and dots that are on different wires specify conditional operations. As shown above conditioned quantum operations can be performed also upon classical clauses.

1.1.1 On the physical implementation of a quantum computer

Even though it will not be a primary focus in this thesis, one of the main challenges in the field of quantum computing, perhaps the most significant, is the physical implementation of a quantum computer. Numerous efforts have been dedicated to finding the optimal physical implementation of a qubit, leading to a variety of options [16–21].

The requirements behind the physical implementation of a qubit have been formalized with the so called DiVincenzo criteria[22]:

1. Well defined structure of the energy spectrum having two distinguishable levels and any possible interference with other states characterized.
2. Coherent manipulation of the system allowing to prepare and measure the initial state.
3. Feasible way to build efficient quantum gates.
4. Decoherence time much longer than executional time of the gates.

5. Being able to perform the readout of the final state.

In particular the fourth statement is related to one of the major issues in quantum computation that still has to be properly addressed: quantum noise. Ideally speaking, the qubits in the computer are perfectly isolated from the environment and interacting only with each other. Such a feature is necessary to mantain the evolution of the system (our quantum program running) coherent (without loss of information). Indeed is widely known that interaction between a quantum system and the environment leads to loss of coherence[23] which is ultimately related to the amount of information stored. Even though we will not consider this aspect deeply in our numerical simulations, as we will see, it will directly affect the quality of the results and the strategy behind quantum programming.

1.2 Getting started with quantum programming

In this section we develop our toolkit for quantum computing; we start with the most basic operations to end up with the implementation of conditional statements and with the main schemes proposed in literature to decompose multiqubits gate.

1.2.1 Single qubit operations

As already pointed out in our introduction a qubit can be represented as a two level system with state vector $|\phi\rangle = a_0|0\rangle + a_1|1\rangle$. According to the standard textbooks notation the states of the computational basis have the following representation[15]:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad , \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{1.2}$$

In order to have a better understanding of the meaning of a single qubit operation we recall the Bloch picture of a two level system. The main idea is to give a geometrical representation of any possible state in a 2-dimensional Hilbert space.

Such an isomorphism is established considering the normalization condition of the wavefunction that allows to write:

$$|\chi\rangle = cos(\tfrac{\theta}{2})|0\rangle + e^{i\phi}sin(\tfrac{\theta}{2})|1\rangle = a_0|0\rangle + a_1|1\rangle$$

$$\begin{pmatrix} a_0 \\ a_1 \end{pmatrix} \Longleftrightarrow \begin{pmatrix} cos(\frac{\theta}{2}) \\ e^{i\phi}sin(\frac{\theta}{2}) \end{pmatrix} \tag{1.3}$$

In this manner we have set up a direct correspondence between the Hilbert space and a 3-dimensional unitary sphere:

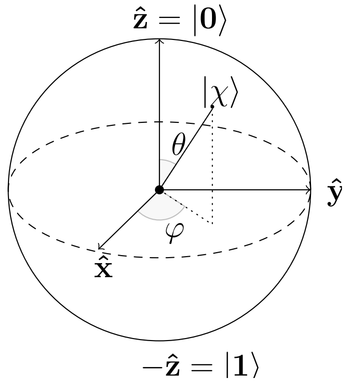


Figure 1.3: Bloch sphere representation of a qubit.

Thus all the operations on $|\chi\rangle$ can be represented either as rotations in the 2-dimensional Hilbert space or as rotations of the Bloch vector on the surface of the sphere. It is important to point out from the beginning that one of the main issues in quantum programming is to decompose the evolution we have in mind for our computer in terms of operations we can easily perform on one or two qubits at the time. The set of operations on which we decompose our "code" forms what is called the **gate library**.

For single qubit operations the Bloch sphere representation allows to derive an easy and general technique to decompose any single qubit operation. As far as we have a correspondence between rotations of a 3D vector and unitary evolutions we can exploit the formalism of the cartesian tensors[24] to obtain a general expression for a single qubit operation. This strategy relies on the use of the Pauli matrices, Eq. (2.4), and of the rotation operators, Eq. (2.5):

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \sigma_y \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{1.4}$$

We can notice the difference in the notation between Eq. 1.4 and Fig. 1.2 for what concerns the labeling of Pauli matrices; as they are ubiquitous in the field of quantum computing and both these notations are often exploited we will use them with no distinction.

$$\begin{aligned} R_z = e^{-\frac{i\theta\sigma_z}{2}} &= cos(\tfrac{\theta}{2})\mathbb{1} - isin(\tfrac{\theta}{2})\sigma_{\mathbb{z}} = \begin{pmatrix} e^{-\frac{i\theta}{2}} & 0 \\ 0 & e^{\frac{i\theta}{2}} \end{pmatrix} \\ R_y = e^{-\frac{i\theta\sigma_y}{2}} &= cos(\tfrac{\theta}{2})\mathbb{1} - isin(\tfrac{\theta}{2})\sigma_{\mathrm{y}} = \begin{pmatrix} cos(\frac{\theta}{2}) & -sin(\frac{\theta}{2}) \\ sin(\frac{\theta}{2}) & cos(\frac{\theta}{2}) \end{pmatrix} \\ R_x = e^{-\frac{i\theta\sigma_x}{2}} &= cos(\tfrac{\theta}{2})\mathbb{1} - isin(\tfrac{\theta}{2})\sigma_{\mathbb{x}} = \begin{pmatrix} cos(\frac{\theta}{2}) & -isin(\frac{\theta}{2}) \\ -isin(\frac{\theta}{2}) & cos(\frac{\theta}{2}) \end{pmatrix} \end{aligned} \tag{1.5}$$

Since it is known from angular momentum theory that any rotation of a point in a fixed frame of reference in a 3D space can be expressed as the composition of three rotations on different axis we obtain an expression for a general unitary U in terms of rotation operators:

$$U = R_z(\gamma)R_y(\beta)R_z(\alpha) = \begin{pmatrix} e^{-i(\gamma+\alpha)/2}cos(\frac{\beta}{2}) & -e^{-i(\gamma-\alpha)/2}sin(\frac{\beta}{2}) \\ e^{i(\gamma-\alpha)/2}sin(\frac{\beta}{2}) & e^{i(\gamma+\alpha)/2}cos(\frac{\beta}{2}) \end{pmatrix} \tag{1.6}$$

Where α, β and γ are commonly referred to as Euler angles[24].

1.2.2 Multiple qubits operations

Single qubit operations are fundamental to understand the most basic operations in quantum computation; nevertheless being able to manage more than one qubit at the time is mandatory to have a device powerful at least as a classical computer. From a theoretical point of view it has been demonstrated (Solovay-Kitaev theorem[25]) that a universal gate library (a gate library with sufficient elements to synthetize any kind of operation) requires at least one two-qubits operation, usually the gate library is completed adding the Controlled-NOT gate (CNOT)[26]. Before introducing all the elements to make clear this sentence we shall give a closer look at the tensor product formalism upon which the picture of a composite quantum system such as a quantum register is based.

The Hilbert space in which the quantum register wavefunction is defined is given by the tensor product of the Hilbert spaces of each qubit:

$$\mathcal{H} = \bigotimes_{i=1}^{n} \mathbb{C}^{2} = \mathbb{C}^{2^{n}}$$

As previously anticipated the basis vectors for this space are commonly chosen as the direct product between the basis vectors of each qubit in its Hilbert subspace, $\{|k\rangle\}$ with k = 0,..., N := 2^N-1. A useful convention is to order the product states such that the values of the direct product of the basis vectors of each qubit forms the binary representation of k. In other words, for instance, the state $|100\rangle$ is labeled as $|4\rangle$. Furthermore such a representation, in analogy with the classical version, is commonly known as **bitstring states** representation[27].

Now let us consider a quantum register with m qubits, a one qubit gate U operating on the n-th qubit is then given as the following tensor product:

$$U = \underbrace{\mathbb{1} \otimes ... \otimes \mathbb{1}}_{n-1\ times} \otimes \mathbb{U} \underbrace{\otimes \mathbb{1} ... \otimes \mathbb{1}}_{m-n\ times} \tag{1.7}$$

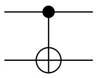
Analogously, concerning a quantum register with m qubits, a general multiqubit gate V operating on the subspace spanned by j qubits, for instance $\{n, n+1, ..., n+j\}$, is defined as:

$$V = \underbrace{\mathbb{1} \otimes ... \otimes \mathbb{1}}_{n-1\ times} \otimes \underbrace{\mathbb{V}}_{j-qubits\ operator} \otimes \underbrace{\mathbb{1} ... \otimes \mathbb{1}}_{m-(n+j)\ times} \tag{1.8}$$

The property of the universal gate library stated before can be rephrased as the fact that a general multiqubit gate can be decomposed as the tensor product of gates acting at most on a subspace spanned by two qubits. For this reason we introduce the most useful two qubits gate: CNOT. Its importance is due to the fact that all the conditional statements, very common and useful also in canonical programming (whose counterpart is given by the XOR gate mentioned at the beginning of the chapter), have a similar form and arise from this gate.

Its representation in the computational basis is:

$$U_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \equiv \tag{1.9}$$



Where the identity defines the circuit element convention for the symbol of the CNOT gate.

This operation involves two different kinds of qubits: the **control qubit** and the **target qubit**. More generally any controlled gate is specified by two sets of qubits: control and target. In the circuit representation black dots determine the control qubits while the actual operation is depicted as a labeled box on the target qubits. In the computational basis the action of the CNOT is given by the following relation: $U_{CNOT}|c,t\rangle = \mathbb{1}|c\rangle \otimes U_{CNOT}|t\rangle = |c, t \oplus c\rangle$, where $t \oplus c$ stands for the modulo-2 sum. In practice, if the control qubit is in the state $|1\rangle$ the target qubit is flipped, otherwise is left unchanged. For sake of completeness we also report, in the following picture, the truth table (i.e., the table that provides all possible outputs for all possible inputs which can be directly derived by multiplying the computational basis vector corresponding to each input to the U_{CNOT} matrix in Eq. 1.9) of this gate (Table 1.1).

At this stage we have seen that the action of the CNOT gate enables to perform a logical operation exactly identical to the classical XOR gate. Moreover the quantum analogoue of the XOR has also an additional feature that is creating

Table 1.1: Truth table of the CNOT gate.

Input		Output	
Control	Target	Control	Target
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

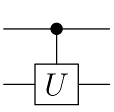
entanglement between different qubits. We can show this characteristic considering its action when applied to a target qubit in the state $|\psi\rangle = \frac{1}{\sqrt{(2)}}(|0\rangle + |1\rangle)$ that, incidentally, can be prepared by applying the Hadamard gate to the state $|0\rangle$, with a control qubit in the state $|0\rangle$:

$$U_{CNOT}|\phi\rangle = U_{CNOT}|\psi\rangle|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\begin{pmatrix}1\\0\\1\\0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix} = \frac{1}{\sqrt{2}}(|00\rangle+|11\rangle) \tag{1.10}$$

In this last equation we have shown that, starting from a state that can be written as the direct product of two states, the action of the CNOT gate results in a state that can not be separated anymore in a direct product of different states, i.e. is entangled.

It is important to notice that the CNOT gate is only a particular case of the more general controlled operation , **c-U**, which representation is a 4x4 diagonal block matrix:

$$U_{controlled} = \begin{pmatrix}\mathbb{1}&\mathbb{0}\\\mathbb{0}&\mathbb{U}\end{pmatrix} \quad \equiv \tag{1.11}$$



The notion of controlled gate can be extended further considering that would be useful implementing operators that have either more than one control qubit or more than one target qubit. In the most general case we could have, as previously mentioned, a set of control qubits and a set of target qubits. All this scenarios can be covered introducing the concept of quantum multiplexor due to Shende et. al[27].

The main idea behind the introduction of the quantum multiplexor is finding a tool which enables to directly transpose the if-then-else construction into the quantum realm. In this perspective determining if the predicate is whether true or not means that the clause to be performed depends on the state(s) of the control qubit(s). Thus if our control qubit is in a superposition of $|0\rangle$ and $|1\rangle$ we will have as outcome of our operation a linear combination of *if* and *else* clauses. To see that also the CNOT gate lies within the definition of a quantum multiplexor we transpose its truth table in a pseudocode:

```
if qubit(0) is |0>:
    U_cnot   ==  (1 0)
                 (0 1)
else:
    U_cnot   ==  (0 1)
                 (1 0)
```

In the quantum multiplexor perspective the control qubits are often referred as select qubits due to the fact that, while the bitstring state formed by their tensor product remains preserved by the action of the gate, it *selects* the actual behavior of the gate acting on the subspace spanned by the remaining qubits.

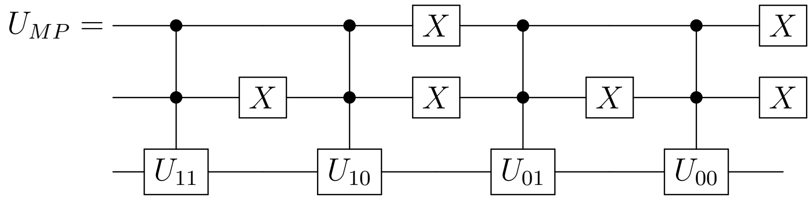
To give an example we consider the representation of a multiplexor with a single select qubit acting on a quantum register of k target qubits. The whole Hilbert space will be 2^{k+1}-dimensional and, with the select qubit being the leftmost in the bistring state representation, the matrix representation of this gate will be a 2x2 diagonal block matrix with each block being 2^k-dimensional:

$$U_{multiplexor} = \begin{pmatrix} \mathbb{U}_{\mathbb{0}} & \mathbb{0} \\ \mathbb{0} & \mathbb{U}_{\mathbb{1}} \end{pmatrix} \tag{1.12}$$

More generally, a quantum multiplexor with s select qubits and k-s target qubit, in this representation, is a diagonal block matrix with 2^s blocks of size 2^{k-s} x 2^{k-s}. The circuit diagram representation of these kind of gates is a series of multicontrolled gates. Control qubits are set up such that they form a particular bitstring state while the gate to be performed is the one selected by the state of the control qubits. For instance, if we consider a 4x4 block diagonal matrix (2 select qubits and one target qubit) we have:

$$U_{MP} = \begin{pmatrix} \mathbb{U}_{\mathbb{00}} & \mathbb{0} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & \mathbb{U}_{\mathbb{01}} & \mathbb{0} & \mathbb{0} \\ \mathbb{0} & \mathbb{0} & \mathbb{U}_{\mathbb{10}} & \mathbb{0} \\ \mathbb{0} & \mathbb{0} & \mathbb{0} & \mathbb{U}_{\mathbb{11}} \end{pmatrix} \tag{1.13}$$

as a matrix represented in the computational basis and



(1.14)

as a quantum circuit.

We notice that the X gate allows to use the usual black dot symbol, conventionally standing for a conditional clause set to true in the case of a control qubit in the state $|1\rangle$, also when the *if* statement is fulfilled for the control qubit in the state $|0\rangle$. Another way to understand this behaviour is looking at the Tab. 1.1 inverting the state of the control qubit in each entry.

As mentioned earlier, the basic elements for programming are defined by the gate library. Once this is established (according to the specifications of the available hardware), circuits such as Eq. 1.14 (or more complicated ones corresponding to dense matrices!) must, in turn, be decomposed.

Various techniques have been developed to accomplish this task; however, it is important to emphasize that, in general, the compilation of a quantum circuit—down to the instructions in terms of the computer's physical Hamiltonian—is a very active area of research in itself[28]. This is because it represents one of the ways to reduce the runtime of a program. In particular, optimal control techniques[29], reinforcement learning[30], and sophisticated mathematical decomposition techniques[31, 32] have been adopted to address the problem of optimal compilation.

In the following chapters our analysis will not consider the efficiency of the compilation routine given that rigorous statements shall require also a particular architecture in mind.

1.3 Simulating molecular systems with quantum computers

After presenting the main tools needed to understand the language of a quantum computer we are now interested in finding all the ingredients necessary to use this model of computation to simulate physical systems of interest to chemistry. Thus the aim of this section is going through a general workflow to model chemistry using quantum computers. We start giving an account on a well-known mapping between second quantization formalism (a standard tool in quantum chemistry) and the algebra of Pauli matrices, that, as we have already seen, is the natural language of quantum computers. Afterwards we are going to look in detail how the general set up of a quantum chemistry calculation is transformed once it has to be set up on a quantum computer. First, we focus on how to upload a molecular wavefunction on a quantum computer and how to compute time evolution; finally, we will also discuss the measurement step needed to extract the final output.

1.3.1 The electronic structure problem

As stated in the introduction, the main goal of this thesis is to present the design of algorithms devoted to the solution of the Schroedinger equation which reads, for a sistem of N electrons and M nuclei in the Born-Oppenheimer approximation, as:

$$\hat{H}_{mol}|\Psi\rangle = (-\frac{1}{2}\sum_{i}^{N}\nabla_i^2 - \sum_{a}^{M}\sum_{i}^{N}\frac{Z_a e_i}{r_{ia}} + \sum_{i>j}^{N}\frac{e^2}{r_{ij}})|\Psi\rangle = E|\Psi\rangle \tag{1.15}$$

Where, as per the rest of this thesis, we have used atomic units to express all the terms in the Hamiltonian.

Since we will be working in second quantisation throughout the rest of this thesis, we also report the molecular Hamiltonian according to this formalism, which is the starting point for our translation of the electronic problem onto a quantum computer.

$$\hat{H}_{mol} = \sum_{p,q} h_{pq} a_p^{\dagger} a_q + \frac{1}{2}\sum_{p,q,r,s} g_{pqrs} a_p^{\dagger} a_r^{\dagger} a_s a_q \tag{1.16}$$

Where h_{pq} are the one-electron integrals containing the kinetic energy and the electron-nuclei repulsion terms and g_{pqrs} are the two-electron repulsion integrals.

In the next section we establish the tools needed to (i) upload a molecular wavefunction on a quantum computer and (ii) translate the molecular Hamiltonian into a qubit observable.

1.3.2 Encoding the molecular wavefunction

The standard description of molecular systems in quantum chemistry is based on Slater determinants that are N-electrons wavefunctions built as the antisymmetrized product of one-electron wavefunctions:

$$|\Theta\rangle = |\phi_1\phi_2\dots\phi_N| = \frac{1}{\sqrt{N!}}\begin{vmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & \phi_N(x_1) \\ \phi_1(x_2) & \phi_2(x_2) & \dots & \phi_N(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_N) & \phi_2(x_N) & \dots & \phi_N(x_N) \end{vmatrix} \tag{1.17}$$

Here ϕ_i represents the single particle basis function corresponding to the i-th spin orbital.

To identify a link with the formalism of quantum gates this formulation of a Slater determinant, based on the language of first quantization, is not as easy to handle as that of the second quantization where a Slater determinant is written as:

$$|\Theta\rangle = \prod_{j=1}^{N} a_j^\dagger |vac\rangle = |f_0 f_1 \dots f_N\rangle \tag{1.18}$$

Here $a_j^\dagger$ is the fermionic creation operator[33] and the state $|vac\rangle$ is a shorthand for $|vacuum\rangle$ being the standard notation in the number occupation representation for the empty state. f_i is the occupation number of the i-th spin-orbital which can take the values $\{0,\ 1\}$.

Generally speaking a mapping $\mathcal{M}$ between the fermionic and qubit algebras is a rule to establish a connection betwenn the Occupation Number (ON) vectors in the Fock space to the bitstring states of the quantum computer:

$$\mathcal{M} : |f_0 f_1 \dots f_N\rangle \to |q_0 q_1 \dots q_N\rangle \tag{1.19}$$

In this thesis our implementation always leverages the Jordan-Wigner mapping $\mathcal{M}_{JW}$ which implies $f_i = q_i$, i.e. the occupation number of a given spin orbital is locally encoded into the qubit state. This choice, combined with the fermionic commutation relations and the qubit commutation relations determines the following equations for the operator algebras:

$$a_j = -\prod_{k=1}^{j-1} \sigma_z^k \sigma_j^- \qquad a_j^\dagger = -\sigma_j^+ \prod_{k=1}^{j-1} \sigma_z^k \tag{1.20}$$

As we can see the antisymmetric nature of the fermionic wavefunction defines, in this mapping, that the (very efficient) locality of the occupation storage is acquired at the expense of inefficient parity storage on $\mathcal{O}(N)$ qubits (due to the string of σ_zs). Generally speaking, the maximum number of qubits on which each

Pauli string produced act is called Pauli weight (which we will see affects also the computational budget of measuring expectation values) and it is, along with the number of qubits needed for the mapping, an important metric for judging the performance of the mapping. Concerning the $\mathcal{M}_{JW}$ both the number of qubits and Pauli weight scale as $\mathcal{O}(N)$.

In this thesis we did not focus on the aspect of finding the optimal mapping w.r.t. the given algorithm proposed even though it can significantly impact the overall runtime if properly chosen. Nonetheless it is important to keep in mind that other choices are possible, as example, while keeping a linear scaling of qubits' requirement with the spin-orbitals, mapping based on classical data structure transformations such as Fenwick trees[34, 35] or ternary trees[36] can achieve Pauli-weights of, respectively, $\mathcal{O}(log_2 n)$ and $\mathcal{O}(log_3 2n)$. Moreover, very recently, strategies to build architecture-aware mappings[37] have been developed further increasing the specificity of this choice when looking for an optimal implementation.

Setting the input: preparation of the initial state.

In almost all cases of interest for quantum chemical simulations, we deal with states of N fermions that can be written as linear combination of K Slater determinants $|\nu_i\rangle$:

$$|\psi\rangle = \sum_{i=1}^{K} b_i |\nu_i\rangle \tag{1.21}$$

Thus, once we have chosen a mapping (i.e. correspondence between ON vectors and bitstring states), we could be interested in understanding how to prepare a given wavefunction on the quantum computer.

To accomplish this task several options are available; as a first example we could start from the amplitudes' vector $\mathbf{b}$ and employ a general embedding scheme like the ones presented in [27] as implemented in the Qiskit package[38].

Other strategies are possible if we look at the problem from a more physical standpoint: for example one could use adiabatic state preparation if interested in preparing the ground state of the Hamiltonian[39, 40]. Of course this strategy may not be computationally feasible depending on the Hamiltonian at hand but could be also used in conjunction to other strategies as proposed in Ref.[41] or explored in Chapter 2 to achieve efficient scalings.

Along this line, optimal control approaches in which we are interested in maximing the fidelity of a parametrically prepared quantum state with the state of interest are possible choices[42–44].

Finally also strategies based on uploading wavefunctions obtained with other electronic structure theory methods have been proposed[45]. Particularly, when considering Density Matrix Renormalization Group (DMRG) calculations[46] with Matrix Product States (MPS) ansatze an efficient upload is possible[47, 48].

1.3.3 Processing the information

Having described how the electronic structure problem can be mapped onto a quantum computer and outlined the main structures of quantum programming, we are now capable of performing transformations on the molecular wave function in a quantum computer. In this section, our primary focus is on the quantum dynamics problem, which serves as a subroutine in Chapters 2 and 4.

Quantum dynamics

The efficiency (in the sense of polynomial runtime scaling $\mathcal{O}(poly(N))$ with system size) of a quantum computer in simulating the dynamics of a molecule was demonstrated in the work of Ref. [49]. This work lays the ground for the theoretical motivation of possible quantum advantage in using quantum devices for the electronic structure calculations.

The time evolution operator acts on the wavefunction resulting in a translation along the time axis. We can have three different cases:

- Evolution under a time independent Hamiltonian:

$$U(t,0) = e^{-i\hat{H}t} \tag{1.22}$$

- Evolution under a time dependent Hamiltonian commuting at different instants:

$$U(t,0) = e^{-i\int_0^t \hat{H}(t')dt'} \tag{1.23}$$

- Evolution under a time dependent Hamiltonian non commuting at different instants:

$$U(t,0) = e_+^{-i\int_0^t \hat{H}(t')dt'} \tag{1.24}$$

  Where the plus subscript stands for the Dyson, *time ordered*, operator[50].

Independently on the kind of Hamiltonian defining the dynamics the time evolution operator always satisfies the following relationships:

$$\begin{aligned} &U(t,t) = \mathbb{1} \\ &U(-t,0) = U^\dagger(t,0) = U(0,t) \\ &U(t_2,t_1) = U(t_2,t_1)U(t_1,t_0) \quad with \quad t_2 > t_1 > t_0 \end{aligned} \tag{1.25}$$

Theoretically speaking if we knew the analytic representation of the time propagator at any time we should just use the most suitable decomposition technique over the available gate library and run the computation. Unfortunately, in most cases[51], since the time evolution is an exponential operator a closed analytic form is not available. For this reason several techniques to compute quantum dynamics have been developed so far[52–56].

As an example, we discuss briefly the Trotter-Suzuki approximation[57]. In this approach we consider the evolution of a system whose Hamiltonian $\hat{H}$ can be written as a sum of "k-locals" Hamiltonians, $\hat{H} = \sum_i \hat{h}_i$, meaning that each term in the sum is acting, at most, on k different particles (qubits). We notice that this is always the case for a molecular system in which we consider only the interaction between couple of particles (electrons or nuclei) mediated by the Coulombic interaction.

For these kind of Hamiltonians the time evolution, considering for instance the time independent case (similar asymptotic cost can be achieved for time-dependent Hamiltonians[58]), can be approximated at the first order as:

$$U(t,0) = e^{-i\sum_j \hat{h}_j t} = \Big(\prod_j e^{-i\hat{h}_j \frac{t}{N}}\Big)^N + O(t^2/N) \tag{1.26}$$

It is important to stress that the approximation of the last equation is due to the fact that the usual composition properties for exponentials is not valid for operators which do not commute. To this extent, not only the truncation order but also the ordering of the different terms in the Trotter expansion can affect the accuracy of the propagation[59].

More generally, considering for example a decomposition of $\hat{H} = \hat{A} + \hat{B}$ it is possible to show[56] that the error for a simulation up to time t divided into r steps is given by:

$$||e^{-i\frac{t}{r}\hat{B}} e^{-i\frac{t}{r}\hat{A}} - e^{-it(\hat{A}+\hat{B})}|| \leq \frac{t^2}{2r}||[\hat{A},\hat{B}]|| \tag{1.27}$$

To conclude, it is important to highlight that these techniques can be made more efficient when a problem-aware analysis is carried out. The importance of this further step is shown, for example, in Babbush et al.[60] where the authors have very recently demonstrated that exact quantum dynamics on quantum computer can scale more efficiently than classical mean-field quantum dynamics. Clearly these asymptotic estimates are useful for motivating the work in the field of quantum algorithms but still necessitate benchmarks on hardware capable of providing tight numerical scalings and benchmarks.

1.3.4 Measuring the result

The last step of a quantum algorithm is the evaluation of the result and transfer of the information into classical bits.

Tipically, standard experimental setups only allow to perform measurements on the computational basis which means performing a measurement on the z-axis for each qubit:

$$\begin{aligned} &\langle \sigma_j^z \rangle = Tr\{\sigma_j^z \rho_j\} = P_{|0\rangle} - P_{|1\rangle} \\ &P_{|0\rangle} + P_{|1\rangle} = 1 \end{aligned} \tag{1.28}$$

Where ρ_j is the reduced density matrix of the j-th qubit and $P_{|0,1\rangle}$ is the population of the state $|0\rangle$ or $|1\rangle$ of the j-th qubit. This construction can be easily generalized to more complicated operators by appending to the quantum circuit of interest gates that rotate the computational basis into the diagonal basis of the operator $\hat{O}_j$.

Hence, considering for instance the general task of measuring the expectation value of the operator $\hat{O}$ (now acting on N qubits) for a molecular wavefunction, we can have:

$$\langle \hat{O} \rangle = \sum_k m_k \langle \hat{P}_k \rangle \tag{1.29}$$

Where P_k are pauli strings obtained using one of the strategies presented in Sec. 4.2 on the operator $\hat{O}$ and m_k matrix elements depending on (i) the mapping strategy and (ii) the particular operator at hand.

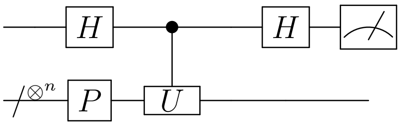
As Eq. 1.28 shows these kind of measurements require statistical sampling that scale, for an ϵ-good estimate, as $\mathcal{O}(\epsilon^{-2})$. This quadratic scaling is thought to hamper the scalability of variational quantum algorithms[61] for this reason many methods have been developed to minimize the computational cost of this step. Particularly, both strategies aimed to reduce the number of independent measured Pauli strings[62–64] and strategies to reduce the impact of error propagation due to the 1-norm of the operator $||\hat{O}||_1 = \sum_k |m_k|$[65] have been considered. It is important to highlight that this stage, even though one could think as independent from the previous steps described in this chapter, can be strongly influeced from the particular mapping (i.e. from the Pauli weight), problem and representation of the problem at hand.

Other proposals have recently been put forward to reconstruct expectation values from random (or informed[66]) sampling of the Pauli string space. Such an approach goes under the name of classical shadowing[67, 68]. These methods, which seem quite promising, have been used in Chapter 2 to estimate the runtime scaling of our variational analog simulation.

To conclude this section we want to highlight a last option to measure expectation values which scaling is still $\mathcal{O}(\epsilon^{-2})$ but is at the basis of almost all the algorithms considered promising in a fault-tolerant setting whose final scaling (as the one considered in Chapter 4) is $\mathcal{O}(\epsilon^{-1})$. In this option we do not measure directly the quantum register where the information resides but, instead, we introduce an ancillary register (or simply just a qubit) which we let interfere with the register of interest to get information with fewer measurements.

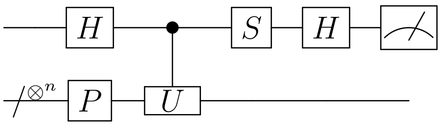
This protocol is known as Hadamard test (or scattering circuit as per the nomenclature of the work in which it is introduced[69]).

$$Re\{\langle\psi|U|\psi\rangle\} = \tag{1.30}$$



Where we used a slash to specify that a single wire encompasses a qubit register of n qubits. Notice that the gate P prepares the state $|\psi\rangle = \hat{P}|0\rangle^{\otimes n}$.

$$Im\{\langle\psi|U|\psi\rangle\} = \tag{1.31}$$



Where $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ is the phase gate.

Looking at Eqs. 1.30-1.31 we can understand how the burden of preparing multiple times the same state to sample expectation values of all the Pauli string of an operator $\hat{O}$ is replaced with the additional cost of an ancillary qubit and increased depth due to the controlled operation.

References

(1) Feynman, R. P. *Int. J. Theor. Phys* **1999**, *21.*

(2) Deutsch, D. *P. Roy. Soc. A-Math. Phy.* **1985**, *400*, 97–117.

(3) Shor, P. W. In *Proceedings 35th annual symposium on foundations of computer science*, 1994, pp 124–134.

(4) Weinstein, Y. S.; Pravia, M.; Fortunato, E.; Lloyd, S.; Cory, D. G. *Physical review letters* **2001**, *86*, 1889.

(5) Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J. C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F. G.; Buell, D. A., et al. *Nature* **2019**, *574*, 505–510.

(6) Zhong, H.-S.; Wang, H.; Deng, Y.-H.; Chen, M.-C.; Peng, L.-C.; Luo, Y.-H.; Qin, J.; Wu, D.; Ding, X.; Hu, Y., et al. *Science* **2020**, *370*, 1460–1463.

(7) Madsen, L. S.; Laudenbach, F.; Askarani, M. F.; Rortais, F.; Vincent, T.; Bulmer, J. F.; Miatto, F. M.; Neuhaus, L.; Helt, L. G.; Collins, M. J., et al. *Nature* **2022**, *606*, 75–81.

(8) Zhou, Y.; Stoudenmire, E. M.; Waintal, X. *Physical Review X* **2020**, *10*, 041038.

(9) Pan, F.; Zhang, P. *Physical Review Letters* **2022**, *128*, 030501.

(10) Wahl, T. B.; Strelchuk, S. *Physical Review Letters* **2023**, *131*, 180601.

(11) Zlokapa, A.; Villalonga, B.; Boixo, S.; Lidar, D. A. *npj Quantum Information* **2023**, *9*, 36.

(12) Albash, T.; Lidar, D. A. *Reviews of Modern Physics* **2018**, *90*, 015002.

(13) Briegel, H. J.; Browne, D. E.; Dür, W.; Raussendorf, R.; Van den Nest, M. *Nature Physics* **2009**, *5*, 19–26.

(14) Bartolucci, S.; Birchall, P.; Bombin, H.; Cable, H.; Dawson, C.; Gimeno-Segovia, M.; Johnston, E.; Kieling, K.; Nickerson, N.; Pant, M., et al. *Nature Communications* **2023**, *14*, 912.

(15) Nielsen, M. A.; Chuang, I. L., *Quantum computation and quantum information*; Cambridge university press: 2010.

(16) Kjaergaard, M.; Schwartz, M. E.; Braumüller, J.; Krantz, P.; Wang, J. I.-J.; Gustavsson, S.; Oliver, W. D. *Annual Review of Condensed Matter Physics* **2020**, *11*, 369–395.

(17) Ferrando-Soria, J.; Moreno Pineda, E.; Chiesa, A.; Fernandez, A.; Magee, S. A.; Carretta, S.; Santini, P.; Vitorica-Yrezabal, I. J.; Tuna, F.; Timco, G. A., et al. *Nature communications* **2016**, *7*, 11377.

(18) Kok, P.; Munro, W. J.; Nemoto, K.; Ralph, T. C.; Dowling, J. P.; Milburn, G. J. *Reviews of modern physics* **2007**, *79*, 135.

(19) Menicucci, N. C.; Van Loock, P.; Gu, M.; Weedbrook, C.; Ralph, T. C.; Nielsen, M. A. *Physical review letters* **2006**, *97*, 110501.

(20) Jones, J. A. *Progress in nuclear magnetic resonance spectroscopy* **2011**, *59*, 91–120.

(21) Liu, G.-Q.; Pan, X.-Y. *Chinese Physics B* **2018**, *27*, 020304.

(22) DiVincenzo, D. P. *Fortschritte der Physik: Progress of Physics* **2000**, *48*, 771–783.

(23) Joos, E.; Zeh, H. D.; Kiefer, C.; Giulini, D. J.; Kupsch, J.; Stamatescu, I.-O., *Decoherence and the appearance of a classical world in quantum theory*; Springer Science & Business Media: 2013.

(24) Rose, M. E., *Elementary theory of angular momentum*; Courier Corporation: 1995.

(25) Kitaev, A. Y. *Russian Mathematical Surveys* **1997**, *52*, 1191.

(26) Barenco, A.; Bennett, C. H.; Cleve, R.; DiVincenzo, D. P.; Margolus, N.; Shor, P.; Sleator, T.; Smolin, J. A.; Weinfurter, H. *Physical review A* **1995**, *52*, 3457.

(27) Shende, V. V.; Bullock, S. S.; Markov, I. L. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, 2005, pp 272–275.

(28) Maronese, M.; Moro, L.; Rocutto, L.; Prati, E. In *Quantum Computing Environments*; Springer: 2022, pp 39–74.

(29) Koch, C. P.; Boscain, U.; Calarco, T.; Dirr, G.; Filipp, S.; Glaser, S. J.; Kosloff, R.; Montangero, S.; Schulte-Herbrüggen, T.; Sugny, D., et al. *EPJ Quantum Technology* **2022**, *9*, 19.

(30) Moro, L.; Paris, M. G.; Restelli, M.; Prati, E. *Communications Physics* **2021**, *4*, 178.

(31) Vatan, F.; Williams, C. *Physical Review A* **2004**, *69*, 032315.

(32) Zulehner, A.; Wille, R. In *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, 2019, pp 185–190.

(33) Helgaker, T.; Jorgensen, P.; Olsen, J., *Molecular electronic-structure theory*; John Wiley & Sons: 2014.

(34) Seeley, J. T.; Richard, M. J.; Love, P. J. *J. Chem. Phys.* **2012**, *137*, 224109.

(35) Bravyi, S. B.; Kitaev, A. Y. *Annals of Physics* **2002**, *298*, 210–226.

(36) Jiang, Z.; Kalev, A.; Mruczkiewicz, W.; Neven, H. *Quantum* **2020**, *4*, 276.

(37) Miller, A.; Zimborás, Z.; Knecht, S.; Maniscalco, S.; García-Pérez, G. *PRX Quantum* **2023**, *4*, 030314.

(38) Aleksandrowicz, G.; Alexander, T.; Barkoutsos, P.; Bello, L.; Ben-Haim, Y.; Bucher, D.; Cabrera-Hernández, F.; Carballo-Franquis, J.; Chen, A.; Chen, C., et al. Qiskit: An open-source framework for quantum computing, 2019.

(39) Aspuru-Guzik, A.; Dutoi, A. D.; Love, P. J.; Head-Gordon, M. *Science* **2005**, *309*, 1704–1707.

(40) Veis, L.; Pittner, J. *The Journal of Chemical Physics* **2014**, *140*.

(41) Matsuura, S.; Yamazaki, T.; Senicourt, V.; Huntington, L.; Zaribafiyan, A. *New Journal of Physics* **2020**, *22*, 053023.

(42) Castaldo, D.; Rosa, M.; Corni, S. *Physical Review A* **2021**, *103*, 022613.

(43) Li, J.; Yang, X.; Peng, X.; Sun, C.-P. *Physical review letters* **2017**, *118*, 150503.

(44) Magann, A. B.; Arenz, C.; Grace, M. D.; Ho, T.-S.; Kosut, R. L.; McClean, J. R.; Rabitz, H. A.; Sarovar, M. *PRX Quantum* **2021**, *2*, 010101.

(45) Tubman, N. M.; Mejuto-Zaera, C.; Epstein, J. M.; Hait, D.; Levine, D. S.; Huggins, W.; Jiang, Z.; McClean, J. R.; Babbush, R.; Head-Gordon, M., et al. *arXiv preprint arXiv:1809.05523* **2018**.

(46) Chan, G. K.-L.; Sharma, S. *Annual review of physical chemistry* **2011**, *62*, 465–481.

(47) Fomichev, S.; Hejazi, K.; Zini, M. S.; Kiser, M.; Morales, J. F.; Casares, P. A. M.; Delgado, A.; Huh, J.; Voigt, A.-C.; Mueller, J. E., et al. *arXiv preprint arXiv:2310.18410* **2023**.

(48) Rudolph, M. S.; Chen, J.; Miller, J.; Acharya, A.; Perdomo-Ortiz, A. *Quantum Science and Technology* **2023**, *9*, 015012.

(49) Lloyd, S. *Science* **1996**, *273*, 1073–1078.

(50) Tannor, D. J., *Introduction to quantum mechanics: a time-dependent perspective*, 2007.

(51) Atia, Y.; Aharonov, D. *Nature communications* **2017**, *8*, 1572.

(52) Low, G. H.; Chuang, I. L. *Physical Review Letters* **2017**, *118*, 010501.

(53) Low, G. H.; Wiebe, N. *arXiv preprint arXiv:1805.00675* **2018**.

(54) Low, G. H.; Chuang, I. L. *Quantum* **2019**, *3*, 163.

(55) Low, G. H.; Su, Y.; Tong, Y.; Tran, M. C. *PRX Quantum* **2023**, *4*, 020323.

(56) Childs, A. M.; Su, Y.; Tran, M. C.; Wiebe, N.; Zhu, S. *Physical Review X* **2021**, *11*, 011020.

(57) Suzuki, M. *Physics Letters A* **1993**, *180*, 232–234.

(58) Poulin, D.; Qarry, A.; Somma, R.; Verstraete, F. *Physical review letters* **2011**, *106*, 170501.

(59) Tranter, A.; Love, P. J.; Mintert, F.; Wiebe, N.; Coveney, P. V. *Entropy* **2019**, *21*, 1218.

(60) Babbush, R.; Huggins, W. J.; Berry, D. W.; Ung, S. F.; Zhao, A.; Reichman, D. R.; Neven, H.; Baczewski, A. D.; Lee, J. *Nat. Comm.* **2023**.

(61) Gonthier, J. F.; Radin, M. D.; Buda, C.; Doskocil, E. J.; Abuan, C. M.; Romero, J. *Physical Review Research* **2022**, *4*, 033154.

(62) Hamamura, I.; Imamichi, T. *npj Quantum Information* **2020**, *6*, 56.

(63) Choi, S.; Loaiza, I.; Izmaylov, A. F. *Quantum* **2023**, *7*, 889.

(64) Yen, T.-C.; Izmaylov, A. F. *PRX Quantum* **2021**, *2*, 040320.

(65) Koridon, E.; Yalouz, S.; Senjean, B.; Buda, F.; O'Brien, T. E.; Visscher, L. *Physical Review Research* **2021**, *3*, 033127.

(66) García-Pérez, G.; Rossi, M. A.; Sokolov, B.; Tacchino, F.; Barkoutsos, P. K.; Mazzola, G.; Tavernelli, I.; Maniscalco, S. *Prx quantum* **2021**, *2*, 040342.

(67) Huang, H.-Y.; Kueng, R.; Preskill, J. *Nature Physics* **2020**, *16*, 1050–1057.

(68) Nakaji, K.; Endo, S.; Matsuzaki, Y.; Hakoshima, H. *Quantum* **2023**, *7*, 995.

(69) Miquel, C.; Paz, J. P.; Saraceno, M.; Knill, E.; Laflamme, R.; Negrevergne, C. *Nature* **2002**, *418*, 59–62.

Chapter 2

Optimal control allows ground state preparation

Background and personal contribution

The work presented in the following chapter represents the starting point of my Ph.D. project. My experience with quantum computing began by developing an optimal control procedure executed on a quantum processor, with the goal of maximizing population transfer between molecular states[1].

Building on this initial achievement, additional efforts were required to adapt this approach to the realm of electronic structure. At the outset of the project, determining the type of perturbation (with the constraint of maintaining a physical interpretation conducive to easy application on an analogue simulator) and the appropriate parameterization for this perturbation were not clear.

From a technical standpoint, I grappled with the challenge of devising an efficient implementation that would enable the exploration of the simulation protocol by varying the number of optimization iterations and, as much as possible, increasing the size of the studied system.

This chapter is presented in the form of an article. Following the introduction, where I delineate the protocol in relation to the literature on variational algorithms, the results demonstrate the capability to find the ground state of molecular systems in times shorter than estimated with theoretical bounds for quantum speed limit.

I developed the algorithm (along with exploring different parameterizations and control shapes) and its software implementation with the help of Dr. Rosa (post-doc in the group of Prof. Corni) to produce the numerical results. Finally, I drafted the article in its original version.

Abstract

We show that optimal control of the electron dynamics is able to prepare molecular ground states, within chemical accuracy, with evolution times approaching the bounds imposed by quantum mechanics. We propose a specific parameterization of the molecular evolution only in terms of interaction already present in the molecular Hamiltonian. Thus, the proposed method solely utilizes quantum simulation routines, retaining their favourable scalings. Due to the intimate relationships between variational quantum algorithms and optimal control we compare, when possible, our results with state-of-the-art methods in literature. We found that the number of parameters needed to reach chemical accuracy and algorithmic scaling are in line with compact adaptive strategies to build variational ansatze. The algorithm, which is also suitable for quantum simulators, is implemented emulating a digital quantum processor (up to 16 qubits) and tested on different molecules and geometries spanning different degrees of electron correlation.

2.1 Introduction

At the heart of the second quantum revolution are two main characters: those working to counteract the harmful effects of quantum noise and those seeking the most efficient strategies to gain practical advantage from new quantum devices as soon as possible. Both face this challenge because properly harnessing physics at the nanoscale would enable the leap forward envisaged by the use of quantum computers[2–4]. First evidences of a quantum advantage[5–7], albeit with some *caveat*[8], are now coming to light.

Within this framework, the precise manipulation of quantum matter is central to the emergence of new ideas leveraging non-classical properties. Coherent control, before being the basis of quantum information processing techniques [9, 10], has been proven a pivotal tool for the exploration of exotic states of matter enabling the preparation of ultracold atoms [11, 12] or the investigation of ultrafast electron dynamics [13, 14]. Accomplishing these tasks means to push the boundaries of engineering into the realm of quantum physics, which has led to the rapid development, both in terms of techniques [15, 16] and applications [17], of Quantum Optimal Control Theory (QOCT) [18]. In this context the physical knowledge of the system and the ability of simulating its evolution in presence of an external control are exploited to enhance a desired response by specifically tailoring a tunable perturbation.

Here we propose an optimal control approach to find the ground state of a molecular Hamiltonian: the real time evolution in presence of an external perturbation is handled by a quantum device, while the optimisation of the perturbation is carried out on a classical hardware to minimize the energy of the system. In this manner the computational burden of the simulation is addressed by the quantum device for which the solution of this task is expected to be one of the first applications with a significant advantage over classical hardwares [19, 20]. The implementation of this routine does not require the universality of the quantum platform and is applicable to both an analog simulator and digital quantum computers. Our work will be focused on the problem of determining the ground state energy of a molecular system and the implementation that we will show directly relates to digital quantum simulators whose technology, to date, is more mature than the one of analog quantum simulators for chemistry [21, 22]. Nevertheless, given the potential of this latter alternative methodology, we will comment the possible development of our algorithm on an analog platform.

Similar optimal protocols have already been applied to the case of laser cooling in a fully classical implementation [23]. Differently from the standard laser cooling procedure (which is mostly related to the vibrational and rotational degrees of freedom [24, 25]), here we do not drive the evolution towards states that are prone to relax towards the target state but, instead, point directly to the target state minimizing the energy of a closed quantum system. For this reason we will refer to the proposed method as quantum simulated cooling since the ultimate goal of the algorithm is to find the optimal perturbation that realizes a trajectory (among those that can be realised by the adopted time-dependent Hamiltonian) driving the system from an initial guess state (higher in energy) to its ground state.

The algorithm we are proposing is based on the idea of solving an optimal control problem using a quantum system as a co-processor; is therefore inscribed in a research line where other problems have been addressed similarly. In particular, Li *et al.* have considered a state preparation problem on a NMR quantum processor [26]; Judson and Rabitz [27] focused on the issue of optimal population transfer in ultrafast spectroscopy exploiting a closed-loop feedback control strategy. They proposed to shape the impinging light pulse on the basis of the molecular response until the evolution reaches optimally the desired state. Recently this idea has been extended, using a quantum computer, to those cases for which it is not possible to build such an experimental apparatus [1].

In addition to these works, the relation between optimal control and variational hybrid algorithms is deep and has been discussed in Ref. [28]. Among the plethora of ansatze proposed in literature to solve quantum chemistry problems, the works of Wecker *et al.* [29] and Choquette *et al.* [30] are closely related to this work. The former has proposed, as ansatz for the Variational Quantum Eigensolver (VQE), a parametrized quantum circuit of the same structure of the system Hamiltonian allowing to restrict the variational search within a symmetry-preserving subspace. This eases the optimization that is challenging when occuring in the entire qubits register Hilbert space. The work of Ref. [30] moves from this point to include an additional term in the variational circuit accounting for temporary drifts in subspaces where the symmetries of the system are not conserved. Even though the variational circuit is not meant to realize a real time evolution of the system, the additional term that is included is thought as an external control highlighting, once again, the close link between optimal control and variational hybrid algorithms.

Finally, we also mention the work of Meitei *et al.*[31] that, very recently, has explored the possibility of rephrasing a VQE approach to optimize not a unitary generated by a parametrized quantum circuit (i.e., the typical approach) but rather to shape the state preparation unitary applying an optimal control protocol directly on the hardware Hamiltonian. Following up this work, many efforts[32] have been done to reduce the duration of the pulses during the control solution either including the pulse length into the variational optimization[33] or allowing leakages of the computer wavefunction outside the standard computational space[34]. This work aims to contribute to these research lines considering an analog quantum simulator specifically devised for the molecular Hamiltonian.

This paper is organized as follows: in Sec. 2.2 we present the general structure of the method, in particular, the main steps that must be followed when applying this procedure to any system are identified. In this regard, in section Sec. 2.2.1 we describe our choice for the energy optimization task on the classical hardware. Section 2.3 presents applications to molecular systems describing in detail the control problem and the parameterization of the control operators (Sec.2.3). Results and technical details about the implementation are provided in Sec. 4.5. We first show how the optimal control procedure is able to find the fundamental state while maintaining chemical accuracy. Then we focus on the effects that the length of the dynamics can have on optimization. We find that the dynamics has an optimal length in terms of convergence and result found, we compare these times of the dynamics to quantum speed limit estimates (i.e. minimum times to accomplish evolution according to quantum mechanics) for processes driven by time-dependent Hamiltonians, and, as reported in Ref.[35], we find that theoretical bounds for time-dependent processes provide quite loose estimates compared to numerical results. Moreover, we study the convergence of the problem by keeping the evolution duration fixed and increasing the number of control parameters. Finally, section 2.4.3 is devoted to a semi-empirical estimate of the computational cost of this method obtained applying this method to hydrogen chains of different lengths. To conclude, we summarize the results obtained and discuss potential future extensions of this work.

2.2 Quantum simulated cooling

In this work we assume that a reference wavefunction $|\Psi_0\rangle$, approximating the target ground state $|\Psi_{GS}\rangle$ of the problem Hamiltonian $\hat{H}$, can be computed efficiently with a preliminar classical computation (e.g. such as the solution of a mean field effective Hamiltonian). We propose the use of a quantum processor to simulate the dynamics of the system Hamiltonian in presence of a time-dependent external control $\hat{V}(t)$. If the unitarity of the dynamics is preserved, $\hat{V}(t)$ can be adapted differently depending on the problem at hand, with the purpose of finding the ground state of the system. The perturbation can either represent a real physical process (such as the presence of an external field coupled to some system's degree of freedom) or a process without experimental counterpart, not affecting the viability of the proposed computational method. Once the evolution is computed, the quantum processor is used to evaluate the Hamiltonian expectation value for the system at the final time $t = T$, then the perturbation is iteratively shaped by a classical optimizer which aims to minimize the energy of the system (see Fig. 2.1).

More formally, the quantum computer provides the evolution of the system due to a parametrized Hamiltonian $\hat{H}_{\mathbf{a}}(t)$:

$$\hat{H}_{\mathbf{a}}(t) = \hat{H} + \hat{V}_{\mathbf{a}}(t) \tag{2.1}$$

Where $\mathbf{a}$ is the set of control parameters shaping the external perturbation.

In order to compute the evolution, the simulated Hamiltonian is mapped onto a N-qubits quantum register:

$$\hat{H}_{\mathbf{a}}^{QC}(t) = \sum_j \gamma_j^{\mathbf{a}}(t)\hat{P}_j \tag{2.2}$$

Here j is an index running on different Pauli strings $\hat{P}_j \in \{\sigma_x, \sigma_y, \sigma_z, \mathbb{I}\}^{\otimes N}$ that are operators acting non trivially on k different qubits (in order to ensure the simulation routine efficiency [19]). The coefficients $\gamma_j^{\mathbf{a}}(t)$ include both the system

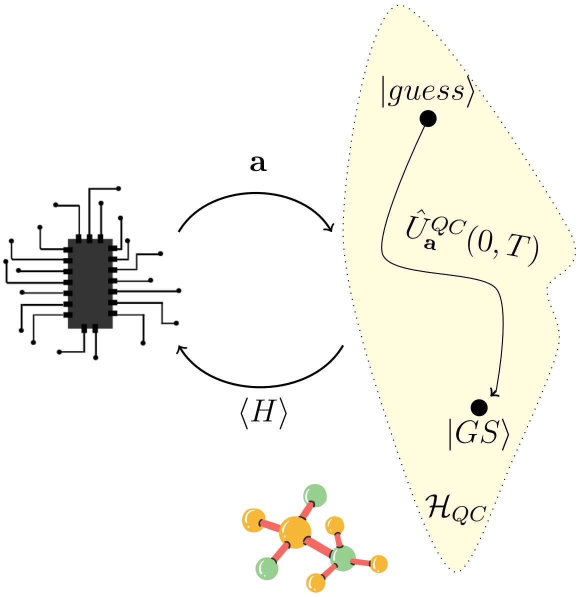


Figure 2.1: Schematic diagram for the hybrid algorithm. The evolution of the system wavefunction is performed on a quantum simulator, the Hamiltonian expection value $\langle H \rangle$ is measured feeding a classical optimization routine which outputs a new set of control parameters $\mathbf{a}$. The control parameters shape the evolution specified by the time evolution operator $\hat{U}_{\mathbf{a}}^{QC}(0,T)$ which starts from an higher energy initial guess state, $|guess\rangle$, and drives the evolution towards the exact ground state of the system, $|GS\rangle$. The loop ends when the energy is below a user-specified threshold.

Hamiltonian and time dependent perturbation matrix elements, whose explicit form (as for the Pauli strings) depends on the adopted mapping.

Notably, we have not made any observation regarding the quantum simulation paradigm to adopt, as the application of this methodology is not limited to universal quantum computers but can be applied with any suitable quantum simulator. In the following we will make explicit reference to the implementation on a digital quantum computer, bearing in mind that other routes are available which may be more or less convenient depending on the system under consideration.

Several methods have been developed to implement the evolution of a time dependent Hamiltonian [36–38] here we compute the approximate time evolution operator $\hat{U}(0, T)$, discretizing the time axis with K slots of width $\Delta t = \frac{T}{K}$. Within each time slot we consider the Hamiltonian as time independent. The accuracy of this procedure depends on the precision with which the Hamiltonian is simulated within each time slot, the time step used and the complexity of the perturbation [39].

So far we have presented the general framework of the method and discussed the role played by the quantum device. In the following section we discuss the complementary step of the procedure: the classical update of the control parameters.

2.2.1 Classical optimization of the energy functional

In the framework of variational hybrid algorithms the choice of the classical optimization routine is a crucial step [40] not only on its own but also with respect to the quantum resource requirements. Indeed, as a first approximation we can estimate the cost of running the optimal control problem to find the ground state as $C = \mathcal{O}(\frac{KMG}{\epsilon^2})$, where K is the number of iterations needed to achieve convergence of the result, M is the number of circuits one needs to execute to do one step of the optimization and G is the gate count of each circuit. The term ϵ^{-2} comes from the finite-shot sampling noise. We will discuss the computational scaling of the control protocol proposed in this work in Sec.2.4.3, here we want to discuss only the possible choices of the optimizer that impact primarily the quantity K.

For our purposes the cost function that we want to minimize is given by:

$$J[\mathbf{a}] = \langle \Psi_{\mathbf{a}}(T)|\hat{H}|\Psi_{\mathbf{a}}(T)\rangle \tag{2.3}$$

Where $|\Psi_{\mathbf{a}}(T)\rangle$ is the wavefunction of our system of interest at time T after the application of the parameterized evolution $\hat{U}_{\mathbf{a}}(0,T)$.

If we consider current quantum processors, it is reasonable to prefer gradient-free optimizers as the calculation of functional gradients numerically or exactly[41–43] would imply adding further noise sources. On the other hand, trying to ensure scalability of the algorithm requires that the optimization task to be accomplished into a reasonable number of iterations to avoid the circuit number executions to grow too rapidly. To this extent the most natural option would be using a gradient-based optimization. Indeed, it is known that the convergence rate for many optimization problems is higher for these latter kind of algorithms than for gradient-free optimizers[44]. As described in Sec.2.4 here we adopted the L-BFGS[45] optimizer.

Before concluding this section it is important to remark that the close relationship between optimal control, variational algorithms and supervised learning may lead to improvements of our implementation that in turn can reduce the overall scaling. Particularly, very recently natural gradient based methods[46, 47] have been developed in the context of VQAs showing promising results in terms convergence rate and avoidance of barren plateaux. Further, other strategies implementing reinforcement learning techniques have shown great improvements w.r.t. gradient based methods that could possibly lead to further speedups to the proposed algorithm[48–50]. We will seek to explore these aspects in future contributions.

2.3 Molecular ground state energies

In this section we provide a detailed description of the algorithm sketched above applied to the case of molecular systems. Therefore, analogue to the previous section, we identify with the problem hamiltonian H the molecular hamiltonian H_{mol}:

$$\hat{H}_{mol} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} g_{pqrs} a_p^\dagger a_r^\dagger a_s a_q \tag{2.4}$$

Where h_{pq} are the one-electron integrals containing the kinetic energy and the electron-nuclei repulsion terms and g_{pqrs} are the two-electron repulsion integrals.

To compose the parametrized Hamiltonian $\hat{H}_{\mathbf{a}}(t)$ we have to specify a form for the perturbation operator and a proper parametrization to implement the optimization routine. As already mentioned in Sec. 2.2 different options are viable, here we considered a time-dependent modification of the Hamiltonian expressed in terms of five ingredients: (i) an effective electron mass $m_e(t)$, (ii) effective nuclear charges $\tilde{Z}_i(t)$, (iii) screened electron-electron interactions $\tilde{\epsilon}(t)$, (iv) a time-dependent effective scalar mean field term $b_0(t)$ and, finally, (v) an overall scalar prefactor $a_0(t)$. These lead to the following expression for $\hat{V}_{\mathbf{a}}(t)$:

$$\hat{V}_{\mathbf{a}}(t) = a_0(t)\Big[\sum_{p,q} \tilde{h}^{\mathbf{a}}_{pq}(t) a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} \tilde{g}^{\mathbf{a}}_{pqrs}(t) a_p^\dagger a_r^\dagger a_s a_q\Big] \tag{2.5}$$

where $\tilde{h}^{\mathbf{a}}_{pq}(t)$ and $\tilde{g}^{\mathbf{a}}_{pqrs}(t)$ are given by:

$$\tilde{h}^{\mathbf{a}}_{pq}(t) = \begin{cases} \frac{1}{2m_e(t)} \int \phi_p^*(\mathbf{x}) \nabla^2 \phi_q(\mathbf{x}) d\mathbf{x} - \int \phi_p^*(\mathbf{x}) \sum_i \frac{\tilde{Z}_i(t)}{r_i} \phi_q(\mathbf{x}) d\mathbf{x} & p \neq q \\ (b_0(t) + \frac{1}{2m_e(t)}) \int \phi_p^*(\mathbf{x}) \nabla^2 \phi_q(\mathbf{x}) d\mathbf{x} - b_0(t) \sum_i (Z_i + \frac{\tilde{Z}_i(t)}{b_0(t)}) \int \phi_p^*(\mathbf{x}) \frac{1}{r_i} \phi_q(\mathbf{x}) d\mathbf{x} & p = q \end{cases} \tag{2.6}$$

and

$$\tilde{g}^{\mathbf{a}}_{pqrs}(t) = \begin{cases} (\frac{1}{\tilde{\epsilon}(t)} + b_0(t)) g_{pqrs} & p = r, q = s \\ (\frac{1}{\tilde{\epsilon}(t)} - b_0(t)) g_{pqrs} & p = s, q = r \\ \frac{1}{\tilde{\epsilon}(t)} g_{pqrs} & \text{else} \end{cases} \tag{2.7}$$

The choice of this parameterization is guided by three factors: (i) linking the perturbation explicitly to physical quantities that are accessible in the presence of an analog simulator; (ii) minimizing the number of parameters to be optimized to reduce the computational cost of optimization; and (iii) ensuring the necessary expressivity for the perturbation to effectively generate dynamics that lead to the target state. Particularly, the last principle motivated the choice of a differential treatment for each atom in the molecule: in fact, it may allow a more subtle discrimination between spatial orbitals (that already experience different nuclear charge due to a_0 and b_0 factors) enhancing effects due to the distinct atoms electronegativity. The extent to which this feature impacts the optimization of the wavefunction is an intriguing question in itself, as it may provide additional physical insight into the solution of the control problem. We aim to delve deeper into investigating this particular feature in a follow-up study.

Concerning the trainability of the proposed parameterization, it is important to mention the problem of barren plateaus which is of specific relevance to hybrid variational algorithms [51, 52]. It has been shown that the landscape parameters' shape is strongly affected by the exponentially big dimension of the quantum processor's Hilbert space. More precisely, as we consider larger systems (i.e. a greater number of qubits), the average value of the gradient objective function tends to zero and more and more states embody this typical value. Thus, if we do not leverage the physical intuition coming from the model Hamiltonian of interest (e.g. exploit symmetry constraint), the control parameters landscape becomes flat over a larger portion of the Hilbert space that we explore during our optimization procedure. We will discuss this aspect in more detail in Sec. 2.4.2.

Parametrization of the control Hamiltonian

Now we turn our attention to the parametrization of the external control. Within the context of quantum optimal control various shapes for the control fields have been proposed ranging from superposition of Gaussian pulses [44] and Fourier-based parametrizations (such as the CRAB and DCRAB methods [53, 54]) to a point-wise definition of the temporal profile as in the case of the GRAPE algorithm

[55]. In this work, we have opted to represent the control functions using the latter option. Previous studies that applied similar parameterizations to optimal gate synthesis have demonstrated that this approach is not always preferable compared to the analytic parameterization[56]. Indeed, analytic controls may allow a faster computation of the gradients and have been shown to be less prone to introducing unwanted high-frequency components leading to leakage errors. However, when transitioning to experimental setups where discretization of control pulses becomes inevitable, these disadvantages fall short[49]. Since we wanted to focus on the development of an algorithm as more oriented to analog simulators as possible we focused on this approach.

With these choice of the parameterization we get an explicit scaling of the number of parameters w.r.t. the time steps of the evolution and system size which is $\mathcal{O}(MT)$. Where M is the number of nuclei and T is the number of controlled steps of the discretized evolution. In Sec.2.4.3 we will provide semi-empirical estimates of the scaling w.r.t. the number of spin-orbitals showing results on hydrogen chains of different length.

It is worth noticing that the energy expectation value measured at the end of the perturbation, as mentioned in Sec. 2.2, is related to the system Hamiltonian $\hat{H}_{mol}$. Hence, no boundary conditions on the perturbation that impose the driving Hamiltonian to coincide with the system Hamiltonian at the end of the evolution are needed. Finally, in contrast with usual optimal control protocols applied to laboratory experiments, unless it is useful for the optimization, the control parameters are allowed to take arbitrary values without maximum (or minimum) thresholds or penalty.

2.4 Results

Computational details

In this section we provide the computational details for the implementation of all our numerical results shown in Sec. 2.4.1-2.4.3. All the calculations were performed with a Python code using a locally modified version of the PennyLane library[57] to construct a representation of all the operators in the computational basis. The evolution and the optimization of the wavefunction were carried out using JAX[58] and the JaxOpt[59] library in order to exploit automatic differentiation and fast evaluation of the quantum dynamics with just-in-time compilation of the code. To ease the computational burden of exactly simulating the quantum dynamics into the qubit space we adopted symmetry reduction of the operator representation to taper-off redundant qubits as shown in Ref.[60] and implemented in [57]. The code is available open-source at [61]. Regarding the specifics of the optimization we used the L-BFGS algorithm as implemented in JaxOpt with default settings. All the calculations were performed either with a maximum number of iterations (N_{iter}) or an energy error of 1 mHa as termination condition.

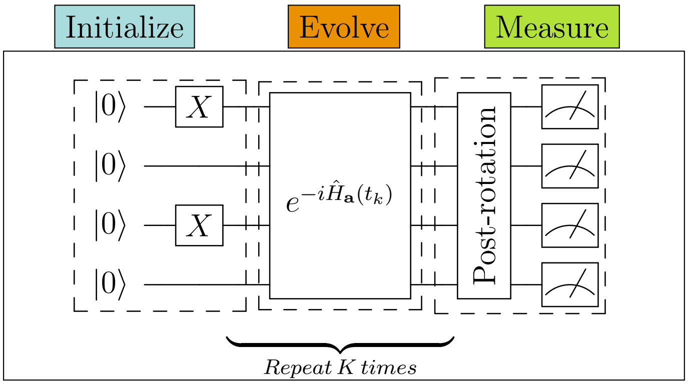


Figure 2.2: Example circuit needed for the implementation on a digital quantum processor. Quantum computer's initial state is given by all the qubits being in the state $|0\rangle$; as an example we reported the initialization circuit for the H_2 molecule in the minimal basis. The second step provides the evolution of the molecular wavefunction, due to the time-dependent external control the exponentiation of the Hamiltonian is repeated for K different steps of the propagation. Here we mantained the circuit as more general as possible (i) to resemble the numerical exponentiation that we used in the computational protocol and (ii) to highlight that one can choose the more suitable simulation routine at hand. Finally the circuit is repeated several times to evaluate the hamiltonian expectation value.

The initial state for all our simulations is the Hartree-Fock wavefunction using the STO-3G basis set in all cases. The initial guess parameters were drawn from a uniform distribution between 0 and 1. The parameters a_0 and b_0 were initialized according to a linear schedule (see Appendix A) as is done by adiabatic state preparation protocols of varying lengths (depending on the time step of the simulation), ranging from 0.2 to 7.5 atomic units (a.u.).

In Fig. 2.2 we report the general structure of the quantum circuits adopted in all the calculations. The qubits are initialized in the $|0\rangle$ state, the mapping between the qubits and the molecular spin-orbitals is accomplished according to the Jordan-Wigner method [62] (e.g. each occupied spin-orbital is represented by a qubit in the state $|1\rangle$). As previously mentioned in Sec. 2.2, the digital quantum simulation is performed numerically exponentiating the time-dependent hamiltonian at each time step in the computational basis spanned by the qubit register. The time step used varies between $\Delta t = 0.00125$ a.u. and $\Delta t = 0.05$ a.u. for all the simulation reported in this work. The stability of the numerical integration is assessed on the basis of previous works[1].

2.4.1 Molecular ground state energies

In this section we report numerical examples of the control protocol applied to three different molecular systems: the H_4 molecule with atoms arranged in a square lattice, H_6 in a linear configuration and lithium hydride (LiH).

The selection of these systems was made to specifically assess the algorithm's performance on systems that, despite their small size, are well-established benchmarks for quantum chemistry methods. Notably, hydrogen chains, although experimentally unstable[63], have been extensively characterized being the simplest systems revealing strong electronic correlation phenomena. We opted for two distances, r=1 Å and r=2 Å, since the former is both in proximity of the observed metal-to-insulator phase transition point expressed in longer analogues of the same series[64] and results from other quantum variational methods are available for comparison. The geometry associated with r=2 Å, which is farther from the equilibrium bond distance, allows us to put our method to the test in a regime approaching dissociation. On the other hand, the H_4 molecule in squared configuration is another prototypical system used to study multireference effects. At this geometry HOMO and LUMO orbitals become degenerate giving a diradical character to the electronic system [65]. Farther from equilibrium, the second geometry we have considered, on top of these effects we add a fourfold bond dissociation process that requires multiply excited configurations to be described. Finally, we have also considered lithium hydride as to test our parameterization (explicitly involving nuclear charges) with an heteroatomic system.

In Fig.2.3 we show the result of the control problem solution for the systems described above. These calculations demonstrate that the control problem can be effectively resolved, achieving error energies below 1 mHa ($< 0.67\,\mathrm{kcal/mol}$), even in regimes characterized by strong correlation. From a dynamical perspective, this implies our capability to identify a pathway leading from the HF state to the exact ground state, even when these states are significantly separated within the Hilbert space. However, it is worth noting that commencing the optimization process from a state further away in the Hilbert space tends to prolong the optimization, despite the protocol's capacity to reach a chemically accurate state. In this regard, in Section 2.4.3, we have examined the convergence rate's scaling with respect to the system size to assess the computational scalability of this approach.

Before discussing these aspects we focus on the effect of varying the duration of the controlled dynamics.

2.4.2 Ground state preparation at the quantum speed limit

Quantum speed limits define the minimum time needed for a quantum system to transition between states. In quantum technologies, they have been extensively studied[66] as being able to engineer transformations achieving this boundaries directly impacts the efficiency and capabilities of quantum devices.

Several theoretical bounds have been developed to quantify this times for different kind of processes[67–69]; here, following Ref. [35] we estimated the quantum

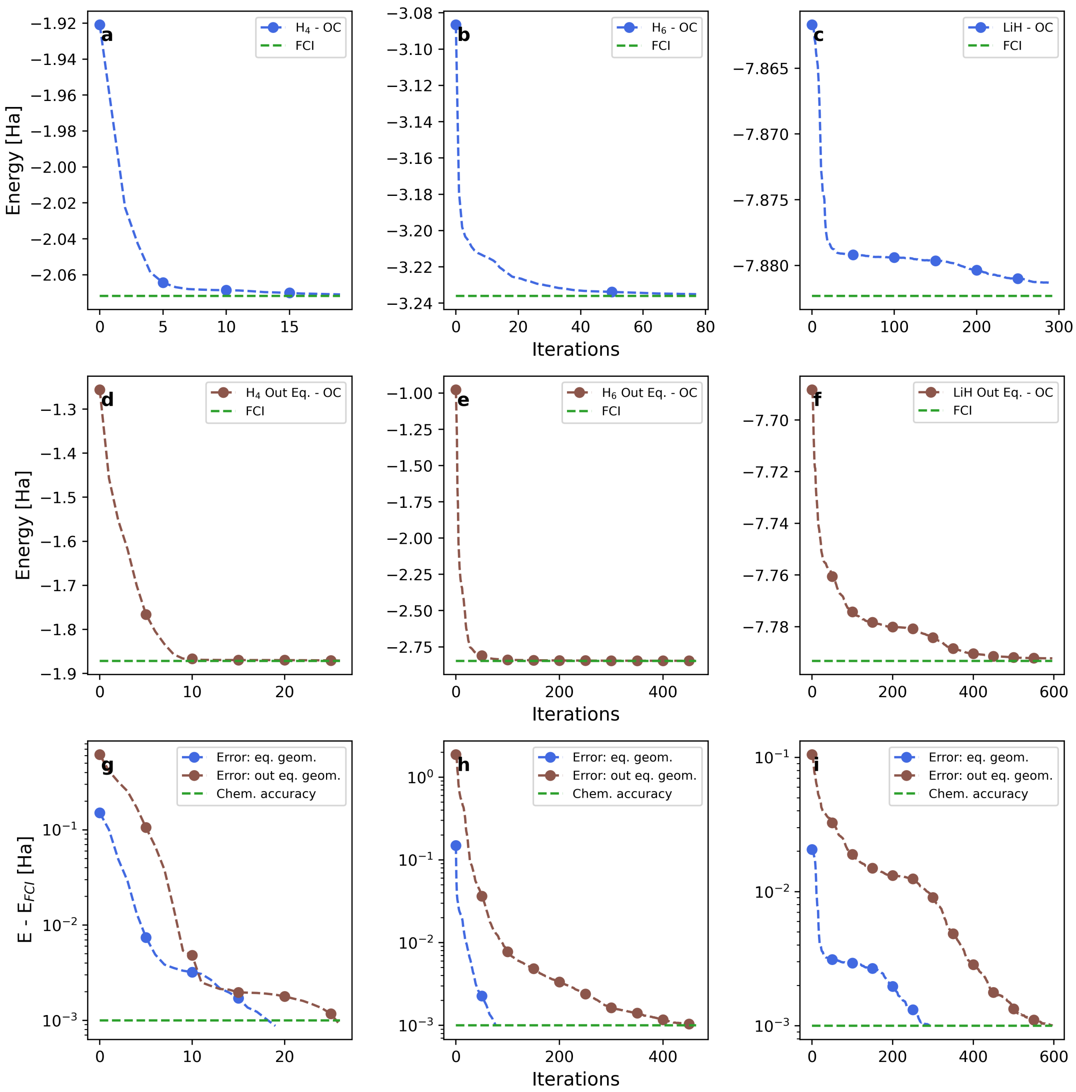


Figure 2.3: Optimal control for molecular ground state preparation. a-c) Energy (blue dashed-dotted line) vs. iterations for squared H_4 at r = 1.2, Å; linear H_6 chain at r = 1 Å and LiH at r = 1.6 Å. d-f) Energy (brown dashed-dotted line) vs. iterations for squared H_4 at r = 2.4 Å; linear H_6 chain at r = 2 Å and LiH at r = 3.2 Å. Error (dashed-dotted line) w.r.t. FCI Energy vs. iterations. Same color code as in panels above; green dashed lines represent either the FCI energy value or a threshold for chemical accuracy posed at 1mHa. Dashed lines are guides to the eyes.

speed limit T_{QSL} (see Tab. 2.1) for the optimally controlled trajectory as:

$$T_{QSL} \leq \frac{\pi}{2} \frac{T}{\int_0^T \sqrt{\langle\psi(t)|[\hat{H}(t) - E(t)]^2|\psi(t)\rangle} dt} \tag{2.8}$$

Where $E(t) = \langle\psi(t)|H|\psi(t)\rangle$.

This quantity estimates the quantum speed limit as the mean energy spread along the computed trajectory and represent an extension of the Battcharayya bound[70] to time-dependent hamiltonians. Please notice that T_{QSL} depends on the control parameters $\mathbf{a}$ as $|\psi(t)\rangle = U_{\mathbf{a}}(0,t)|\psi(0)\rangle$.

As reported in Ref. [35], a rigorous definition of a quantum speed limit for time-dependent processes is elusive and discrepancies with numerical experiments reflect this aspect. The authors find for the problem of optimal population transfer along a spin chain that Eq. 2.8 overestimates (on average) by a factor of 3 the numerical results. These discrepancies have been shown with even tighter bounds as reported in Ref. [71] for the case of a time-optimal SWAP gate. Here we found that the numerical estimate is sensibly lower than the theoretical one in all cases with edge cases such as H_4 (r = 1.2 Å) where Eq. 2.8 overestimates the numerical result by two order of magnitudes.

We can compare these findings with other similar works in literature. Particularly, the work of Matsuura et al.[72] proposes the coupling of an optimal control of the annealing schedule and a VQE with a UCCSD ansatz to improve molecular ground state preparation which they dub VanQver. We find similar times for the optimized evolution as they report. Particularly, for the rectangular H_4 molecule (notice that here we qualitatively compare the results as we have instead a perfectly squared geometry) they find $\mathrm{T}_{VanQver} = 0.088\,a.u.$, for the LiH molecule $\mathrm{T}_{VanQver} = 0.14\,a.u.$. For comparison their reported times for Annealing State Preparation (ASP) are respectively $T_{ASP} = 9.5\,a.u.$ and $T_{ASP} = 11.5\,a.u.$ for LiH and rectangular H_4 both close to the equilibrium bond length (i.e. to compare with the first and third column of Tab. 2.1).

We can also compare the number of parameters needed by our optimal con-

Table 2.1: Estimated time according to the Bhattacharyya bounds[35, 70] (Eq. 2.8) on time-dependent quantum evolutions (T_{QSL}), time required by the optimally controlled evolution to reach the ground state (T_{OC}) and number of parameters optimized in each trajectory ($\# \ \theta_{QSL}$) for the molecules considered in this study. Time is expressed in atomic units.

	LiH @ 1.6 Å	H_4 @ 1.2 Å	H_6 @ 1 Å	LiH @ 3.2 Å	H_4 @ 2.4 Å	H_6 @ 2 Å
T_{QSL}	7.91	2.39	6.27	10.84	7.43	39.11
T_{OC}	0.25	0.01	0.5	0.75	0.5	0.75
$\#\theta_{QSL}$	30	32	50	90	80	150

trol procedure with the number of parameters generated by the UCCSD ansatz and compact adaptive strategies[73, 74]. Particularly, as discussed in [75] for the LiH molecule, spin-adapted UCCSD requires 64 parameters in comparison adaptive ansatze built with a fermionic operator pool require around 10 parameters to reach chemical accuracy. In the same work the authors report, concerning the H_6 molecule, that the UCCSD circuit requires almost 70 parameters without being able to reach chemically accurate results as the bond distance increases (at about distances greater than 1.3 Å) while the adaptive procedure requires at most the same number of parameters reaching chemical accuracy even in the dissociation limit. As we can see, the optimal control procedure requires a similar set of parameters as the adaptive approach when considering bond lengths close to the equilibrium geometry. However, in cases involving more stretched bonds, it becomes evident that the adaptive procedure shows greater efficiency.

In addition to estimating the quantum speed limit through equation 2.8, we analyzed the relationship between optimal control procedures for the systems presented in the previous section and the duration of the dynamics (see Fig. 2.4). What is observed is a dual behavior: concerning dynamics shorter than a certain length (which we identify as the actual quantum speed limit), the optimal control procedure is unable to achieve chemical accuracy within 500 iterations. Furthermore, the optimizer amplifies the perturbation strength by increasing the average energy injected into the system, which we tried to estimate with the quantity $\langle \frac{||H(t)||}{||H_{mol}||} \rangle$ (see Fig.2.4-2.5). As we can see, consistently with the time-energy uncertainty relationship, the average energy injected into the system decreases as the evolution length increases.

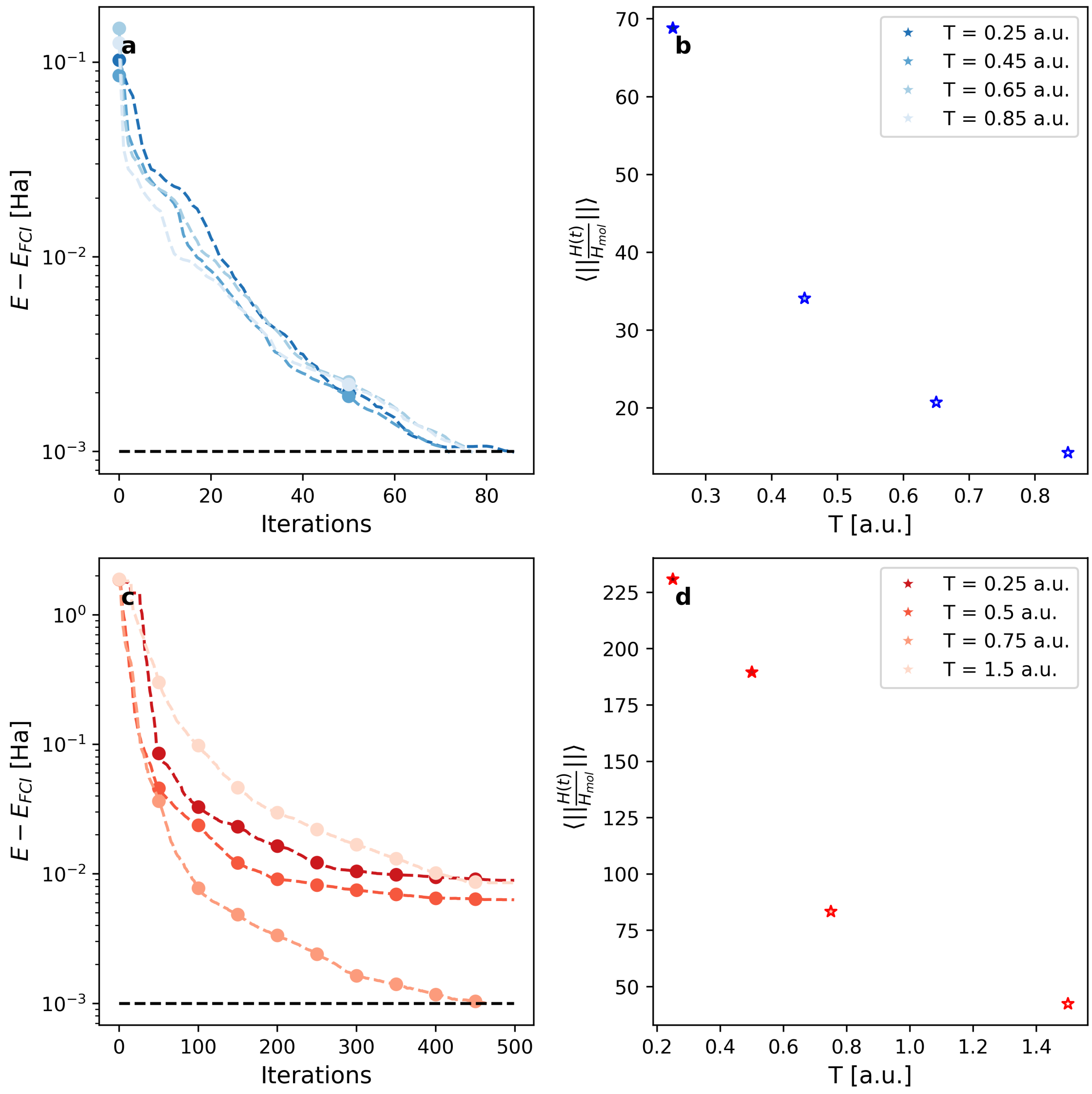


Figure 2.4: Ground state preparation at the quantum speed limit for the H_6 molecule. a, c) Optimal control for various duration of the dynamics. Blues refer to H_6 at r=1 Å, reds refer to H_6 at r=2 Å. b,d) Mean driving Hamiltonian norm vs. duration length for the same systems and color codes.

On the other hand, the optimized dynamics with a duration exceeding the quantum speed limit converges more slowly to the optimal result. This effect is consistent with other findings in literature, where has been shown that quantum dynamics in presence of an external perturbation tends to converge towards unitary q-designs (i.e. unitaries that can uniformly cover the Hilbert space) as time increases[76]. This justifies a slower convergence since these types of unitaries are highly expressive and, as previously shown, are much more prone to encountering barren plateaus during optimization[51, 77]. Interestingly, as reported in Fig.2.4c, an initial plateaux in the optimization is present both for very short dynamics and for the ones beyond our estimate T_{QSL}. Even though they look similar the former may arise from a lack of controllability (i.e. too few control parameters available), while the latter are a direct manifestation of the barren plateaux. Having identified this sweet spot in terms of the length of the dynamics suggests that, to avoid encountering optimization problems with larger systems, it might be beneficial to progressively optimize the dynamics starting from shorter evolutions and initializing the control parameters to achieve idle evolution. We plan to assess the effect of the initialization and adaptive optimization of the dynamics in a future work. We refer the reader to appendix B where similar results are shown for the LiH molecule.

Finally, to get an additional insight of this multiple interplay among controllability and time-energy uncertainty relationships, we report in Fig.2.5 for the H_4 molecule the solution of the control problem varying (i) the length of the dynamics (Fig.2.5a-b) and (ii) the number of controllable steps keeping the duration fixed at T=0.01 a.u. (Fig.2.5c-d). Again, we can notice that the amount of energy that the perturbation inputs into the system decreases as the evolution length increases as already shown in Fig.2.4. Nevertheless we can notice that moving from T=0.01 to T=0.05 a.u. there is an abrupt decrease as compared to all the values reported both in Fig.2.4 and Fig.2.5. This motivated us to study the effect of increasing the number of controllable steps at this shorter duration length of the dynamics. As we can see in Fig.2.5c increasing the number of controllable steps immediately leads the control problem to find a state within chemical accuracy. Moreover, the mean energy input into the system decreases of almost two order of magnitudes reaching values more in line with the trends observed fort the other systems. From this calculation arises a picture where the quantity $\langle \frac{||H(t)||}{||H_{mol}||} \rangle$ allows to diagnose possible controllability issues in the definition of the optimization problem. Indeed, its unexpected increase indicates that the optimizer is striving to reach the optimal solution, still attainable within the given time frame. In doing so, it injects more energy into the system but lacks the flexibility to effectively address the control problem. By relaxing this constraint and adding more controllable steps, both the excess injected energy and convergence issues vanish.

2.4.3 Computational cost analysis

In this section we provide an empirical estimate of the algorithmic scaling of the proposed method. Particularly, as mentioned in Sec.2.2.1 we can approximate the cost (or runtime of the algorithm) as $C = \mathcal{O}(\frac{KMG}{\epsilon^2})$. We recall that K is the number of iterations needed to achieve 1 mHa of error, M is the number of circuits per iteration and G is the gate count of the circuit.

In Fig. 2.6, we provide an estimate of C as a function of the spin-orbitals N studying the hydrogen chain series from H_2 to H_8 at $r = 1\,\text{Å}$. As we can see Fig.2.6a reports an almost linear scaling of the number of iterations as a function of the spin-orbitals and Fig.2.6b a quadratic scaling concerning the number of control parameters. In order to estimate the overall number of circuits executed during the optimization we considered the relation:

$$\#Circuits \approx \mathcal{O}(\#Iterations \text{ x } \#params \text{ x } m) \tag{2.9}$$

Where m is the number of independent measurements needed to compute the 2-RDM of the molecular Hamiltonian according to the shadow procedure developed in Ref.[78]:

$$m = \mathcal{O}(\frac{4\eta^2}{\epsilon^2}) \tag{2.10}$$

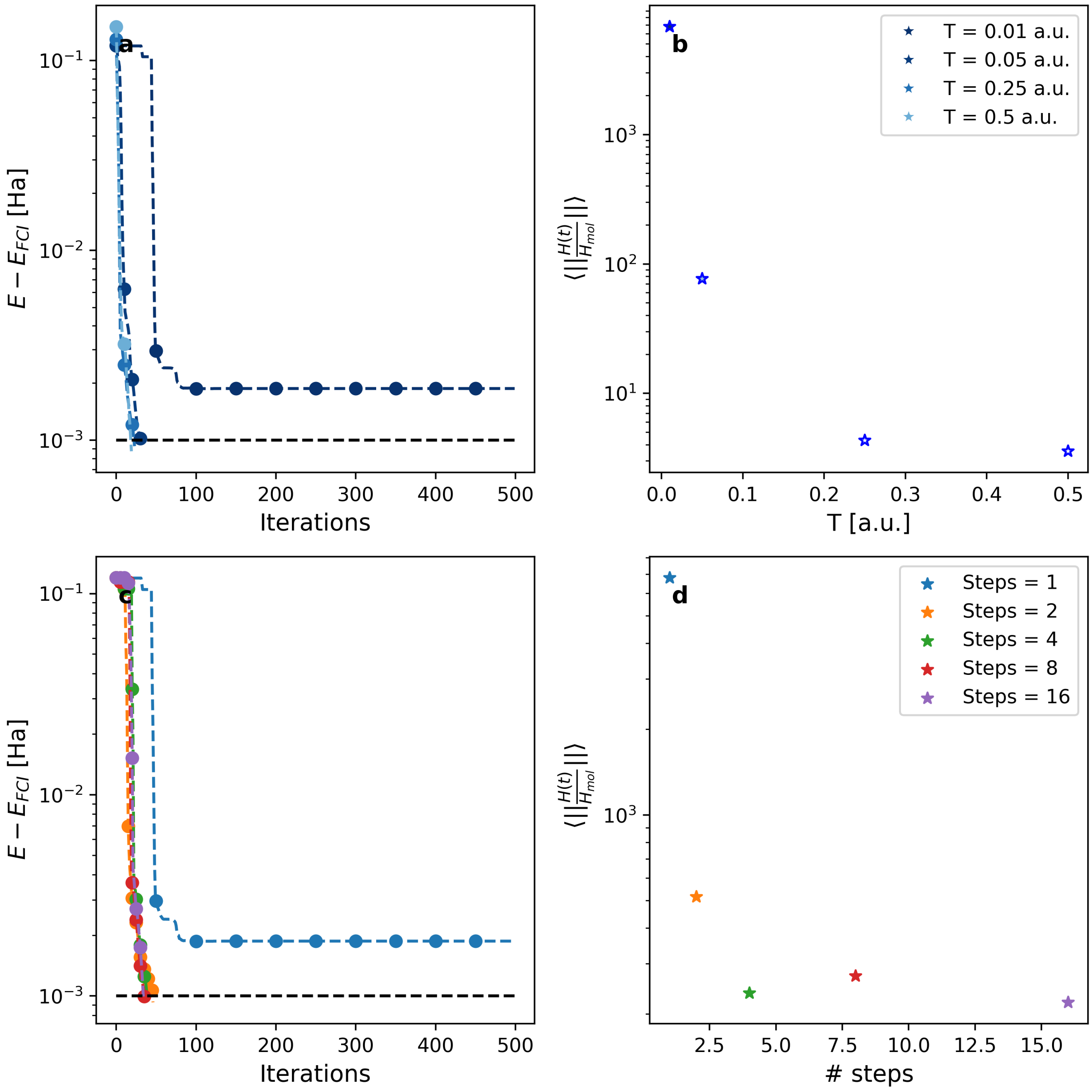


Figure 2.5: Effect of controllability on the time-energy uncertainty relationship. a) Optimal control for the H_4 molecule at r=1.2 Å for different quantum dynamics' lengths (darker blue shorter length). b) Mean driving Hamiltonian norm vs. duration length. c, d) Control problem at fixed duration length (T=0.01 a.u.) with finer time stepping (more controllable dynamics).

Where η is the number of electrons in the system.

It is important to notice that the estimate of m could be further improved adopting system-dependent methods to reduce the overall number of measurements based on Pauli strings partitioning[79, 80] in combination with matrix completion techniques[81].

Plugging Eq. 2.10 into Eq. 2.9 we got an overall scaling of $\#Circuits \approx \mathcal{O}(N^4)$. Finally, the total runtime reads:

$$C = \mathcal{O}(\frac{GN^4}{\epsilon^2}) \tag{2.11}$$

In the last equation we decided to keep the runtime of the quantum simulation routine unexpressed as the designated platform for the execution of this algorithm is an analog simulator for the molecular Hamiltonian. Currently, only prototypes of this simulator have been developed[21, 22]. Nevertheless, if we were to consider implementing the algorithm on digital quantum computers, it would be reasonable (actually conservative) to assume a linear increase in computational overhead as the number of spin orbitals grows[82, 83].

To conclude this section we would like to comment on our findings. First of all, we are aware that the results of Fig. 2.6 give rise to a crude estimate for at least two reasons: (i) the modest range of the active space explored and (ii) an additional uncertainty due to the random guess initialization. Moreover, in Sec. 2.4.1 we showed that depending on the degree of correlation the number of iterations needed may vary; to this extent, expanding this benchmark to other systems will surely increase its reliability.

Nevertheless, these results already represent a good starting point to understand if this method is worth further refinements. Given that very promising methods, such as adaptive strategies[75], have $\mathcal{O}(N^8)$ scaling if implemented naively and can achieve $\mathcal{O}(N^5)$ scaling only if clever strategies for evaluating gradients are adopted[84], we think that the method proposed in this paper can be of interest regardless of the future development of an analog simulator. To this extent incremental optimization strategies, i.e. optimizing the evolution step-by-step, could provide significant speedups and provide tighter bounds on the overall scaling. We plan to explore these aspects in a future work.

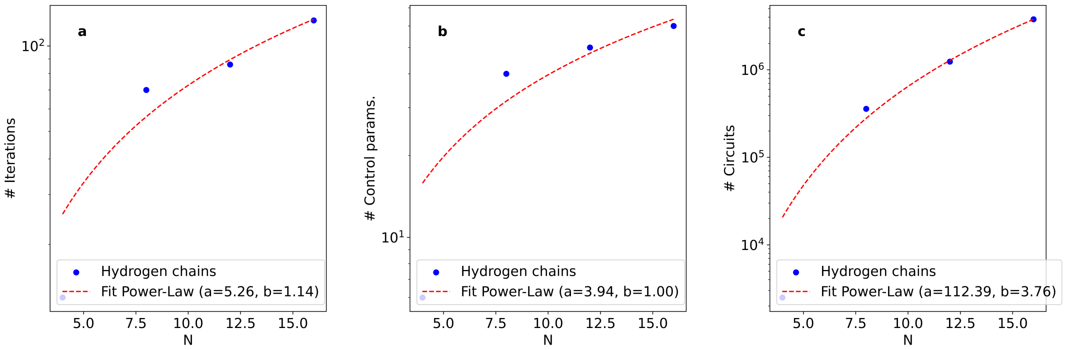


Figure 2.6: Empirical scaling of the optimal control algorithm. a) Number of iterations needed to reach chemical accuracy as a function of the number of spin-orbitals (N). b) Number of control parameters as a function of the number of spin-orbitals. c) Number of circuits evaluated during the all control problem solution. Power-law fitting function: $y = aN^b$.

2.5 Conclusions

We have proposed an optimal control approach utilizing a quantum device to steer the evolution of a quantum system towards its ground state. Demonstrating its application, we have computed ground state energies for molecular systems up to 16 spin-orbitals (qubits). Our results indicate the potential to discover pathways reaching states within the chemical accuracy energy threshold more rapidly than the adiabatic path, edging closer to the quantum speed limit. Moreover, this study underscores the intertwined nature of controllability (defined as the minimum number of controls needed to achieve desired precision), duration of the dynamics and convergence of the optimization protocol.

Additionally, we have offered an empirical estimate of the computational cost conducting various calculations on hydrogen chains of different lengths. Our findings reveal that the overall algorithm runtime execution scales as $\mathcal{O}(N^5)$, aligning with results from adaptive ansatze. Future avenues of this work aim to mitigate the computational scaling by refining the procedure via a step-by-step adaptive optimization of the quantum dynamics.

To conclude, we highlight how this method could serve to speed up phase estimation-like algorithms. Particularly the optimally controlled dynamics could impact the quantum phase estimation procedure in two ways: (i) on one hand as initial state preparation routine; on the other hand (ii), we can imagine of replacing the evolution of the circuit within the QPE to directly sample a correlation function (from a simple initial state such as HF) corresponding to an initial state that includes contributions from multiple electronic configurations. We plan to explore this latter aspect in future works to understand whether this option could bring fault-tolerant algorithms' implementation one step closer.

Appendix A: a simpler example on $\mathbf{H}_2$

The aim of this section is to provide a simpler example of the general protocol, for the case of the hydrogen molecule H_2, focusing on the comparison of the results with a quantum simulated annealing protocol for different values of annealing time T. We show that the optimal solution of our procedure represents a shortcut to adiabaticity[85, 86], i.e. an alternative fast route that allows to obtain the same final state given by a slow, adiabatic evolution.

As to test the optimal control framework in different settings we modified both the choice of the perturbation and the optimization routine (here we used differential evolution as implemented in Scipy[87]). Particularly, concerning the control operators here we only needed a time-dependent modulation of the electron-nuclei interaction. Thus, the perturbation operator $V_{\mathbf{a}}(t)$ reads:

$$\hat{V}_{\mathbf{a}}(t) = \sum_{p,q} \tilde{h}^{\mathbf{a}}_{pq}(t) a^{\dagger}_{p} a_{q} \tag{2.12}$$

with $\tilde{h}^{\mathbf{a}}_{pq}(t)$ given by:

$$\tilde{h}^{\mathbf{a}}_{pq}(t) = f_{\mathbf{a}}(t) \int \phi^*_p(\mathbf{x}) \sum_i \frac{Z_i}{r_i} \phi_q(\mathbf{x}) d\mathbf{x} \tag{2.13}$$

with $f_{\mathbf{a}}$ defined locally in time such that each different value of the function at different time steps of the propagation is a unique control parameter.

The theoretical foundation of quantum simulated annealing is the adiabatic theorem [88]. It states that the evolution of a quantum state, being in the ground state of an initial Hamiltonian $\hat{H}_0$, will occur transitionless (i.e. adiabatically, without excitations) under a time dependent Hamiltonian $\hat{H}(t)$ if the variation rate of the Hamiltonian is small enough. As a consequence, a slow evolution under a perturbation that modifies the Hamiltonian until it becomes the one of interest allows to compute its ground state. Here we chose the initial state to be the solution of a classical Hartree-Fock calculation: the initial Hamiltonian is the Hartree-Fock Hamiltonian $\hat{H}_{HF}$. The adiabatic evolution is meant to reach the exact ground state for the complete molecular Hamiltonian $\hat{H}_{mol}$, therefore the overall evolution takes place under the Hamiltonian:

$$\hat{H}(t) = A(t)\hat{H}_{HF} + B(t)\hat{H}_{mol} \tag{2.14}$$

The functions $A(t)$ and $B(t)$ define the annealing schedule, i.e. the switching factors between the two Hamiltonians. They must be defined such that $\hat{H}(0) = \hat{H}_{HF}$ and $\hat{H}(T) = \hat{H}_{mol}$.

The choice of the annealing schedule influences the perfomance of the quantum algorithm[89–91]. In particular, different strategies have been adopted such as optimal control protocols devised to shorten the annealing time or to shape the annealing schedule profile enhancing the success probability [92, 93]. Bearing this in mind, here we have considered a linear schedule (Eq. 2.15) as it is still used as typical benchmark in the field [94].

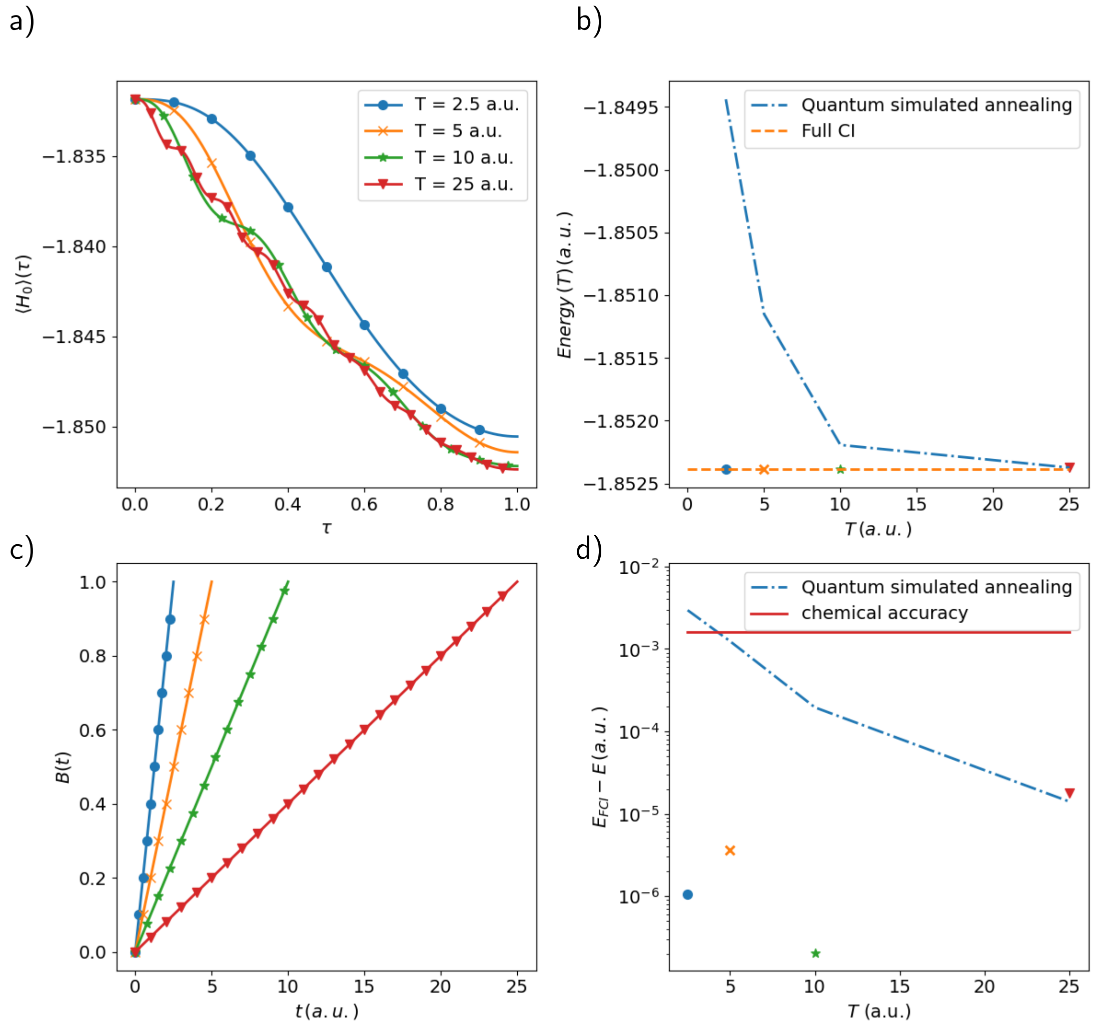


Figure 2.7: Comparison between quantum simulated cooling and quantum simulated annealing results for the H_2 molecule at equilibrium geometry. a) Expectation value of the driving Hamiltonian (Eq. 2.14) as a function of the dimensionless instantaneous time, $\tau = \frac{t}{T}$. Results are reported for different values of annealing time T: $T = 2.5$ a.u. (blue dots), $T = 5$ a.u. (orange crosses), $T = 10$ a.u. (green stars) and $T = 25$ a.u. (red downward triangles). b) Electronic energies after quantum simulated annealing (dashed-dotted blue line) and quantum simulated cooling (scatter plot) as a function of the evolution time. Scatter plot symbols refer to the same T values of panel a; orange dashed line is the reference FCI energy. c) Linear schedules for the quantum simulated annealing. d) Absolute energy difference from FCI for the quantum simulated cooling (scattered) and quantum simulated annealing (dashed-dotted blue line). The red solid line poses a threshold for chemical accuracy at 0.0016 Ha (i.e., 1 kcal/mol).

$$
\begin{aligned}
A(t) &= 1-\frac{t}{T} \\
B(t) &= \frac{t}{T}
\end{aligned}
\tag{2.15}
$$

In Fig. 2.7 we show the results of our comparison between quantum simulated cooling and quantum simulated annealing, for this analysis we focused on the H_2 molecule at the equilibrium distance. We considered four increasing evolution times ranging from 2.5 a.u. to 25 a.u.. In the first panel (Fig. 2.7a), the driving Hamiltonian expectation value is plotted as a function of the instantaneous time, $\tau = \frac{t}{T}$. As expected, increasing the annealing time the energy at the final instant decreases as the evolution occurs without transitions to any excited electronic configuration. Moving to panel 2.7b we compare the results of the adiabatic evolution (dashed-dotted blue line) with the exact Full CI energy (dashed orange line) and the energy outcome of the quantum simulated cooling (scatter plot symbols differ according to the evolution length). We observe that the non-adiabatic evolution provided by the quantum simulated cooling is always performing better than the quantum simulated annealing protocol. The only exception is for the last point at 25 a.u., which, as better highlighted in Fig. 2.7d, equals the result obtained with the Hamiltonian of Eq. 2.14. To explain this behavior we point out that the differential evolution is a non-deterministic algorithm, as such small fluctuations on the final results are expected between different runs.

Appendix B: additional calculations on the LiH molecule

Here we report additional results on the LiH molecule (see Fig.2.8) concerning the analysis on the dynamics' duration length as discussed in Sec.2.4.2. These results confirm the considerations drawn in the main text: short evolution lengths imply higher energy injection from the perturbation which results in higher values of $\langle \frac{||H(t)||}{||H_{mol}||} \rangle$. Concerning the convergence behavior of the control problem w.r.t. the

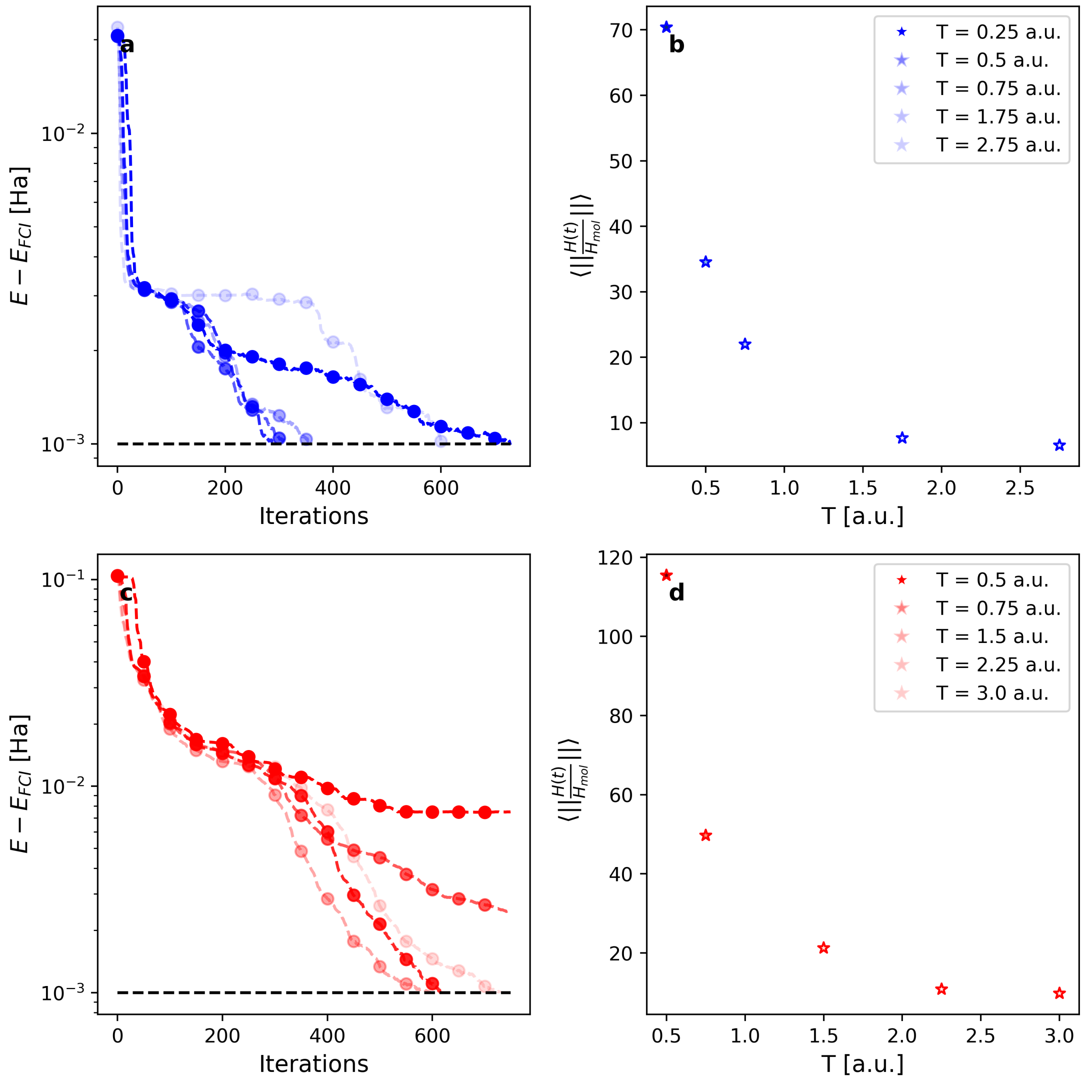


Figure 2.8: Ground state preparation at the quantum speed limit for the LiH molecule. a, c) Optimal control for various duration of the dynamics. Blues refer to LiH at r=1.6 Å, reds refer to LiH at r=3.2 Å. b,d) Mean driving Hamiltonian norm vs. duration length for the same systems and color codes.

length of the evolution at the equilibrium geometry (Fig.2.8a) we can clearly see the same pattern previously discussed, i.e. that very short (or very long) evolutions can hamper the optimization to reach a chemically accurate result. On the other hand, this trend is not so evident looking at Fig.2.8c (r = 3.2 Å). The dynamics' length and convergence patterns appear less interdependent in this system. Further studies collecting statistics on various randomly initialized optimizations could resolve this inconsistency.

References

(1) Castaldo, D.; Rosa, M.; Corni, S. *Physical Review A* **2021**, *103*, 022613.

(2) Deutsch, I. H. *PRX Quantum* **2020**, *1*, 020101.

(3) Feynman, R. P. *Int. J. Theor. Phys* **1999**, *21*.

(4) Deutsch, D. *P. Roy. Soc. A-Math. Phy.* **1985**, *400*, 97–117.

(5) Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J. C.; Barends, R.; Biswas, R.; Boixo, S.; Brandao, F. G.; Buell, D. A., et al. *Nature* **2019**, *574*, 505–510.

(6) Zhong, H.-S.; Wang, H.; Deng, Y.-H.; Chen, M.-C.; Peng, L.-C.; Luo, Y.-H.; Qin, J.; Wu, D.; Ding, X.; Hu, Y., et al. *Science* **2020**, *370*, 1460–1463.

(7) Madsen, L. S.; Laudenbach, F.; Askarani, M. F.; Rortais, F.; Vincent, T.; Bulmer, J. F.; Miatto, F. M.; Neuhaus, L.; Helt, L. G.; Collins, M. J., et al. *Nature* **2022**, *606*, 75–81.

(8) Zhou, Y.; Stoudenmire, E. M.; Waintal, X. *Physical Review X* **2020**, *10*, 041038.

(9) Werninghaus, M.; Egger, D. J.; Roy, F.; Machnes, S.; Wilhelm, F. K.; Filipp, S. *npj Quantum Information* **2021**, *7*, 1–6.

(10) Montangero, S.; Calarco, T.; Fazio, R. *Physical Review Letters* **2007**, *99*, 170501.

(11) Sklarz, S. E.; Tannor, D. J. *Physical Review A* **2002**, *66*, 053619.

(12) Koch, C. P.; Palao, J. P.; Kosloff, R.; Masnou-Seeuws, F. *Physical Review A* **2004**, *70*, 013402.

(13) Klamroth, T. *The Journal of Chemical Physics* **2006**, *124*, 144310.

(14) Rosa, M.; Gil, G.; Corni, S.; Cammi, R. *The Journal of Chemical Physics* **2019**, *151*, 194109.

(15) Doria, P.; Calarco, T.; Montangero, S. *Physical Review Letters* **2011**, *106*, 190501.

(16) Zhu, W.; Botina, J.; Rabitz, H. *The Journal of Chemical Physics* **1998**, *108*, 1953–1963.

(17) Brif, C.; Chakrabarti, R.; Rabitz, H. *New J. Phys.* **2010**, *12*, 075008.

(18) Werschnik, J.; Gross, E. *Journal of Physics B: Atomic, Molecular and Optical Physics* **2007**, *40*, R175.

(19) Lloyd, S. *Science* **1996**, *273*, 1073–1078.

(20) Georgescu, I. M.; Ashhab, S.; Nori, F. *Reviews of Modern Physics* **2014**, *86*, 153.

(21) Argüello-Luengo, J.; González-Tudela, A.; Shi, T.; Zoller, P.; Cirac, J. I. *Nature* **2019**, *574*, 215–218.

(22) Argüello-Luengo, J.; González-Tudela, A.; Shi, T.; Zoller, P.; Cirac, J. I. *Physical Review Research* **2020**, *2*, 042013.

(23) Rahmani, A.; Kitagawa, T.; Demler, E.; Chamon, C. *Physical Review A* **2013**, *87*, 043607.

(24) Bartana, A.; Kosloff, R.; Tannor, D. J. *Chemical Physics* **2001**, *267*, 195–207.

(25) Bartana, A.; Kosloff, R.; Tannor, D. J. *The Journal of Chemical Physics* **1997**, *106*, 1435–1448.

(26) Li, J.; Yang, X.; Peng, X.; Sun, C.-P. *Physical review letters* **2017**, *118*, 150503.

(27) Judson, R. S.; Rabitz, H. *Physical Review Letters* **1992**, *68*, 1500.

(28) Magann, A. B.; Arenz, C.; Grace, M. D.; Ho, T.-S.; Kosut, R. L.; McClean, J. R.; Rabitz, H. A.; Sarovar, M. *PRX Quantum* **2021**, *2*, 010101.

(29) Wecker, D.; Hastings, M. B.; Troyer, M. *Physical Review A* **2015**, *92*, 042303.

(30) Choquette, A.; Di Paolo, A.; Barkoutsos, P. K.; Sénéchal, D.; Tavernelli, I.; Blais, A. *Physical Review Research* **2021**, *3*, 023092.

(31) Meitei, O. R.; Gard, B. T.; Barron, G. S.; Pappas, D. P.; Economou, S. E.; Barnes, E.; Mayhall, N. J. *arXiv preprint arXiv:2008.04302* **2020**.

(32) Meirom, D.; Frankel, S. H. *arXiv preprint arXiv:2212.12911* **2022**.

(33) Egger, D. J.; Capecci, C.; Pokharel, B.; Barkoutsos, P. K.; Fischer, L. E.; Guidoni, L.; Tavernelli, I. *Physical Review Research* **2023**, *5*, 033159.

(34) Asthana, A.; Liu, C.; Meitei, O. R.; Economou, S. E.; Barnes, E.; Mayhall, N. J. *arXiv preprint arXiv:2203.06818* **2022**.

(35) Caneva, T.; Murphy, M.; Calarco, T.; Fazio, R.; Montangero, S.; Giovannetti, V.; Santoro, G. E. *Physical Review Letters* **2009**, *103*, 240501.

(36) Wiebe, N.; Berry, D.; Høyer, P.; Sanders, B. C. *Journal of Physics A: Mathematical and Theoretical* **2010**, *43*, 065203.

(37) Berry, D. W.; Childs, A. M.; Cleve, R.; Kothari, R.; Somma, R. D. *Physical Review Letters* **2015**, *114*, 090502.

(38) Low, G. H.; Chuang, I. L. *Physical Review Letters* **2017**, *118*, 010501.

(39) Wiebe, N.; Berry, D. W.; Høyer, P.; Sanders, B. C. *Journal of Physics A: Mathematical and Theoretical* **2011**, *44*, 445308.

(40) McArdle, S.; Endo, S.; Aspuru-Guzik, A.; Benjamin, S. C.; Yuan, X. *Reviews of Modern Physics* **2020**, *92*, 015003.

(41) Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; Killoran, N. *Physical Review A* **2019**, *99*, 032331.

(42) Mari, A.; Bromley, T. R.; Killoran, N. *Physical Review A* **2021**, *103*, 012405.

(43) Mitarai, K.; Negoro, M.; Kitagawa, M.; Fujii, K. *Physical Review A* **2018**, *98*, 032309.

(44) Machnes, S.; Assémat, E.; Tannor, D.; Wilhelm, F. K. *Physical Review Letters* **2018**, *120*, 150401.

(45) Wright, S. J., *Numerical optimization*, 2006.

(46) Stokes, J.; Izaac, J.; Killoran, N.; Carleo, G. *Quantum* **2020**, *4*, 269.

(47) Fitzek, D.; Jonsson, R. S.; Dobrautz, W.; Schäfer, C. *arXiv preprint arXiv:2304.13882* **2023**.

(48) Dalgaard, M.; Motzoi, F.; Sørensen, J. J.; Sherson, J. *npj Quantum Information* **2020**, *6*, 1–9.

(49) Niu, M. Y.; Boixo, S.; Smelyanskiy, V. N.; Neven, H. *npj Quantum Information* **2019**, *5*, 33.

(50) Wright, E.; de Sousa, R. *arXiv preprint arXiv:2305.01169* **2023**.

(51) McClean, J. R.; Boixo, S.; Smelyanskiy, V. N.; Babbush, R.; Neven, H. *Nature Communications* **2018**, *9*, 1–6.

(52) Cerezo, M.; Sone, A.; Volkoff, T.; Cincio, L.; Coles, P. J. *Nature Communications* **2021**, *12*, 1–12.

(53) Caneva, T.; Calarco, T.; Montangero, S. *Physical Review A* **2011**, *84*, 022326.

(54) Rach, N.; Müller, M. M.; Calarco, T.; Montangero, S. *Physical Review A* **2015**, *92*, 062343.

(55) Khaneja, N.; Reiss, T.; Kehlet, C.; Schulte-Herbrüggen, T.; Glaser, S. J. *Journal of Magnetic Resonance* **2005**, *172*, 296–305.

(56) Willsch, D.; Nocon, M.; Jin, F.; De Raedt, H.; Michielsen, K. *Physical Review A* **2017**, *96*, 062302.

(57) Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Ahmed, S.; Ajith, V.; Alam, M. S.; Alonso-Linaje, G.; AkashNarayanan, B.; Asadi, A., et al. *arXiv preprint arXiv:1811.04968* **2018**.

(58) Bradbury, J.; Frostig, R.; Hawkins, P.; Johnson, M. J.; Leary, C.; Maclaurin, D.; Necula, G.; Paszke, A.; VanderPlas, J.; Wanderman-Milne, S.; Zhang, Q. JAX: composable transformations of Python+NumPy programs, version 0.3.13, 2018.

(59) Blondel, M.; Berthet, Q.; Cuturi, M.; Frostig, R.; Hoyer, S.; Llinares-López, F.; Pedregosa, F.; Vert, J.-P. *arXiv preprint arXiv:2105.15183* **2021**.

(60) Setia, K.; Chen, R.; Rice, J. E.; Mezzacapo, A.; Pistoia, M.; Whitfield, J. D. *Journal of Chemical Theory and Computation* **2020**, *16*, 6091–6097.

(61) https://github.com/davidecast/oc_quantum_simulation Accessed by date: 02/01/2023.

(62) Jordan, P.; Wigner, E. P. *Z. Phys.* **1928**, *47*, 631–651.

(63) Stella, L.; Attaccalite, C.; Sorella, S.; Rubio, A. *Physical Review B* **2011**, *84*, 245117.

(64) Motta, M.; Genovese, C.; Ma, F.; Cui, Z.-H.; Sawaya, R.; Chan, G. K.-L.; Chepiga, N.; Helms, P.; Jiménez-Hoyos, C.; Millis, A. J., et al. *Physical Review X* **2020**, *10*, 031058.

(65) Sand, A. M.; Mazziotti, D. A. *Computational and Theoretical Chemistry* **2013**, *1003*, 44–49.

(66) Deffner, S.; Campbell, S. *Journal of Physics A: Mathematical and Theoretical* **2017**, *50*, 453001.

(67) Margolus, N.; Levitin, L. B. *Physica D: Nonlinear Phenomena* **1998**, *120*, 188–195.

(68) Chau, H. *Physical Review A* **2010**, *81*, 062133.

(69) Ness, G.; Alberti, A.; Sagi, Y. *Physical Review Letters* **2022**, *129*, 140403.

(70) Bhattacharyya, K. *Journal of Physics A: Mathematical and General* **1983**, *16*, 2993.

(71) Lee, J.; Arenz, C.; Rabitz, H.; Russell, B. *New Journal of Physics* **2018**, *20*, 063002.

(72) Matsuura, S.; Yamazaki, T.; Senicourt, V.; Huntington, L.; Zaribafiyan, A. *New Journal of Physics* **2020**, *22*, 053023.

(73) Tang, H. L.; Shkolnikov, V.; Barron, G. S.; Grimsley, H. R.; Mayhall, N. J.; Barnes, E.; Economou, S. E. *PRX Quantum* **2021**, *2*, 020310.

(74) Shkolnikov, V. O.; Mayhall, N. J.; Economou, S. E.; Barnes, E. *Quantum* **2023**, *7*, 1040.

(75) Grimsley, H. R.; Economou, S. E.; Barnes, E.; Mayhall, N. J. *Nature Communications* **2019**, *10*, 3007.

(76) Banchi, L.; Burgarth, D.; Kastoryano, M. J. *Physical Review X* **2017**, *7*, 041015.

(77) Holmes, Z.; Sharma, K.; Cerezo, M.; Coles, P. J. *PRX Quantum* **2022**, *3*, 010313.

(78) Babbush, R.; Huggins, W. J.; Berry, D. W.; Ung, S. F.; Zhao, A.; Reichman, D. R.; Neven, H.; Baczewski, A. D.; Lee, J. *Nat. Comm.* **2023**.

(79) Martínez-Martínez, L. A.; Yen, T.-C.; Izmaylov, A. F. *Quantum* **2023**, *7*, 1086.

(80) Choi, S.; Loaiza, I.; Izmaylov, A. F. *Quantum* **2023**, *7*, 889.

(81) Peng, L.; Zhang, X.; Chan, G. K. *arXiv preprint arXiv:2306.05640* **2023**.

(82) Low, G. H.; Su, Y.; Tong, Y.; Tran, M. C. *PRX Quantum* **2023**, *4*, 020323.

(83) Babbush, R.; Berry, D. W.; McClean, J. R.; Neven, H. *npj Quantum Information* **2019**, *5*, 92.

(84) Anastasiou, P. G.; Mayhall, N. J.; Barnes, E.; Economou, S. E. *arXiv preprint arXiv:2306.03227* **2023**.

(85) Guéry-Odelin, D.; Ruschhaupt, A.; Kiely, A.; Torrontegui, E.; Martínez-Garaot, S.; Muga, J. G. *Reviews of Modern Physics* **2019**, *91*, 045001.

(86) Martinis, J. M.; Geller, M. R. *Physical Review A* **2014**, *90*, 022307.

(87) Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J., et al. *Nature Methods* **2020**, *17*, 261–272.

(88) Albash, T.; Lidar, D. A. *Reviews of Modern Physics* **2018**, *90*, 015002.

(89) Roland, J.; Cerf, N. J. *Physical Review A* **2002**, *65*, 042308.

(90) Albash, T.; Lidar, D. A. *Physical Review A* **2015**, *91*, 062320.

(91) Chen, H.; Lidar, D. A. *Physical Review Applied* **2020**, *14*, 014100.

(92) Brady, L. T.; Baldwin, C. L.; Bapat, A.; Kharkov, Y.; Gorshkov, A. V. *Physical Review Letters* **2021**, *126*, 070505.

(93) Chen, Y.-Q.; Chen, Y.; Lee, C.-K.; Zhang, S.; Hsieh, C.-Y. *arXiv preprint arXiv:2004.02836* **2020**.

(94) Callison, A.; Festenstein, M.; Chen, J.; Nita, L.; Kendon, V.; Chancellor, N. *PRX Quantum* **2021**, *2*, 010338.

Chapter 3

Quantum simulation of molecules in solution

Background and personal contribution

As stated in the Introduction, the ambition of this Ph.D. thesis is to contribute not only to the development of methods for calculating the energies of molecular states but also to the development of quantum computing protocols capable of handling the complexity of molecular systems.

In the following chapter, already published as a journal article in *Journal of Chemical Theory and Computation*[1], we developed a methodology to include in the Variational Quantum Eigensolver the possibility of describing an environment surrounding the molecule implicitly with a polarizable dielectric.

In this paper, we demonstrate not only the extension of quantum computing methods from gas-phase to condensed-phase molecules but also that the variational nature of the VQE allows the inherently non-linear problem to be solved without additional costs in terms of circuits to be performed.

I developed the algorithm and the code to obtain the numerical results. This has been accomplished developing an interface betweenPennyLane (circuit execution), Psi4 (solving the electronic structure), and PCMSolver (solving the BEM problem). The adaptive circuits were built by Dr. Alain Delgado (Xanadu), and

Dr. Soran Jahangiri (Xanadu) assisted me in developing the PennyLane part. I wrote the initial draft of this paper.

Abstract

Quantum chemical calculations on quantum computers have been focused mostly on simulating molecules in gas-phase. Molecules in liquid solution are however most relevant for Chemistry. Continuum solvation models represent a good compromise between computational affordability and accuracy in describing solvation effects within a quantum chemical description of solute molecules. In this work we extend the Variational Quantum Eigensolver to simulate solvated systems using the Polarizable Continuum Model. To account for the state dependent solute-solvent interaction we generalize the Variational Quantum Eigensolver algorithm to treat non-linear molecular Hamiltonians. We show that including solvation effects does not impact the algorithmic efficiency. Numerical results of noiseless simulations for molecular systems with up to twelve spin-orbitals (qubits) are presented. Furthermore, calculations performed on a simulated noisy quantum hardware (IBM Q Mumbai) yield computed solvation free energies in fair agreement with the classical calculations.

3.1 Introduction

Nowadays, multiscale modelling is a workhorse of computational chemistry and physics [2–5]. Its recent development has been fueled by the constant quest to understand and harness more complex phenomena which necessarily call for the inclusion of details arising from the composite nature of the studied systems [3]. Such a demand for greater detail in the simulation of experiments is accompanied by an increase of the computational resources required. In particular, the need for more accurate molecular simulations using wave function based methods has motivated significant research efforts at the intersection of quantum chemistry and quantum computing. [6–9].

In this work, we aim to contribute to both fields by reporting the first example of a hybrid quantum-classical algorithm (i.e., with the meaning given in the quantum computing literature to hybrid[10], a protocol whose implementation relies both on classical and quantum computation) in which the quantum simulation of a molecular system takes also into account the presence of a solvating environment. The importance of these effects is glaring given their ubiquity in Nature at all levels: plants, animals and microorganisms base their existence on molecular mechanisms in which the presence of a solvent is essential [11–15]. It is also needless to remark the importance of accounting for solvation effects in almost all branches of Chemistry[16–19].

Concerning the strategies adopted so far to include solute-solvent interactions we can distinguish two broad classes pertaining to an explicit or implicit treatment of the solvent in the system description. The former is typically represented by Molecular Dynamics simulations or Monte Carlo simulations in which average properties are obtained from sampling the phase space of the system where the solvent degrees of freedom are explicitly accounted for [20, 21]. This option, due to the high number of molecules needed, is limited to a classical description of the solvent (possibly coupled to a quantum chemical description of the solute in a QM/MM approach[22, 23]) or to a quantum description for a relatively small systems and limited statistical sampling [24, 25]. In the long term, one may expect that the development of quantum computers will be sufficiently advanced to allow explicit, full-fledged quantum simulations of solutions. However, to date, this possibility is still far from being realised, and extending implicit solvation models to quantum computing approaches is a suitable option to address solvation effects.

In particular, implicit methods of solvation are the most commonly adopted providing a methodology to describe the surrounding solvent as a continuum medium [26]. Within this framework, the Polarizable Continuum Model (PCM)[27] represents *de facto* the standard approach due to its flexibility and accuracy. In particular, the Integral Equation Formalism version of the PCM (IEF-PCM) allows to describe with very little modifications to the working equations the presence of both an isotropic or anisotropic polarizable medium as well as ionic solutions [28–30]. Beyond the environment complexity available, this theoretical framework has been developed in many directions giving the possibility to describe the solute at different levels of theory[31–34], picturing the overall process with the use of the open quantum system formalism [35] and also accounting for the presence of nanometallic structures [36] and optimal control procedures [37]. Finally, we also mention a recent development which exploits the emerging tool of machine learning to improve the estimates of solvation free-energy obtained from PCM[38].

In this contribution we leverage the standard formulation of the IEF-PCM to include solvation effects in the flagship algorithm of quantum simulation for Noisy Intermediate Scale Quantum (NISQ) devices: the Variational Quantum Eigensolver (VQE) [39–41]. The choice of this method has been dictated by the recent literature that has showed its successful applications on near-term quantum processors to simulate molecules, condensed matter physics and other phenomena of physico-chemical interest[42–46]. In particular, we exploit a specific flavor of VQE[47] where the trial wavefunction is built exploiting an adaptive concept[48]. In the following, we will refer to the new algorithm as PCM-VQE.

This work is organized as follows: first we review the basics of PCMs with particular attention to the IEF-PCM formulation, subsequently we discuss the changes incorporated into the VQE to include solvent effects. In the Results sections we report various numerical tests on three different molecules, namely H_2O, BeH_2 and H_3^+ in dimethyl sulfoxide (DMSO) as a test bed for the algorithm implementation. Further, we assess the estimate of the solvation free energy with a noisy quantum simulation adopting a noise model based on the IBM Q Mumbai quantum processor. We conclude discussing our results and future perspectives of this work.

3.2 Theory

3.2.1 The Polarizable Continuum Model

The purpose of this section is to recall the PCM concepts and quantities that enter the modifications we have made to the VQE algorithm. For comprehensive summaries we refer to Refs. [27, 49].

The physical picture encompassed by PCMs is of a solute embedded in a molecular shaped cavity interacting with the solvent, located outside, which is described as a structureless polarizable dielectric. In this approach, the charge density of the solute molecule polarizes the external environment which generates an electric field (the reaction field) that acts back on the solute. Such reaction field is obtained as the field produced by a set of polarization charges, the so called apparent surface charges (ASC), spread on the cavity boundary, whose values depend, in turn, on the solute molecular electrostatic potential.

To organize the discussion, we first present how such ASC can be calculated within IEF-PCM. Then, we describe how the solute-solvent interaction is accounted for in the quantum mechanical description of the molecule.

The electrostatic problem

We start by solving an electrostatic problem in which we look for the electrostatic potential $\varphi(\mathbf{r})$ generated by the molecular (nuclear and electronic) charge density $\rho(\mathbf{r})$ embedded in a polarizable surrounding solvent characterized by the dielectric constant ϵ. This is accomplished solving the appropriate Poisson's equation [50]:

$$\nabla \cdot [\epsilon(\mathbf{r}) \nabla \varphi(\mathbf{r})] = -4\pi\rho(\mathbf{r}) \tag{3.1}$$

where $\epsilon(\mathbf{r})$ is defined as:

$$\epsilon(\mathbf{r}) = \begin{cases} 1 & \text{inside the cavity} \\ \epsilon & \text{outside the cavity} \end{cases} \tag{3.2}$$

Equation 3.2 implies the use of a set of additional boundary conditions to solve the Poisson equation ensuring the continuity of the potential and the electric field at the interface of the cavity [51].

In the framework of IEF-PCM[28, 30] the electrostatic problem is recasted in an integral equation that directly provides the ASC density:

$$\big(\frac{\epsilon+1}{\epsilon-1}\hat{I}-\frac{1}{2\pi}\hat{D}\big)\hat{S}\sigma(\mathbf{s})=-\big(\hat{I}-\frac{1}{2\pi}\hat{D}\big)\Phi(\mathbf{s})\,. \tag{3.3}$$

Here $\Phi(\mathbf{s})$ is the Molecular Electrostatic Potential (MEP) at the surface Γ of the cavity, $\mathbf{s}$ is a point on the cavity surface, $\hat{I}$ is the identity operator, $\hat{D}$ and $\hat{S}$ are the components of the Calderòn projector[30] that are related, respectively, to the normal component w.r.t. Γ of the field generated by $\sigma(\mathbf{s})$ and the related electrostatic potential at the surface. Their explicit expression depends only on the cavity shape and the dielectric properties of the solvent.[30]

The numerical solution of Eq. 3.3 involves a discretization of the cavity surface into N_{tess} tesserae and a corresponding discrete representation of the operators $\hat{D}$, $\hat{S}$ and of the ASC density. The formal details of the cavity discretization procedure is described in Refs.[52, 53]. Here, we will focus on reporting the working equations of the IEF-PCM method after this step is completed.

The discretization of $\sigma(\mathbf{s})$ results in the introduction of a set of charges $\mathbf{q}$ positioned at the centre of each tessera:

$$\sigma(\mathbf{r})=\sum_{i=1}^{N_{tess}}\frac{q_i}{a_i}\delta(\mathbf{r}-\mathbf{s}_i)\,, \tag{3.4}$$

where a_i and q_i are, respectively, the area and the point charge located at the ith tessera, $\delta(\mathbf{r}-\mathbf{s}_i)$ is a Dirac delta function peaked on the tessera representative point $\mathbf{s}_i$.

Once the discretization procedure has been accomplished we obtain an expression for the polarization charges on all the tesserae, which model the response of the solvent to the presence of the solute:

$$\mathbf{q}=-\left(2\pi\frac{\epsilon+1}{\epsilon-1}\mathbf{S}-\mathbf{DAS}\right)^{-1}(2\pi\mathbf{1}-\mathbf{DA})\,\mathbf{V}=\mathbf{Q}^{\mathrm{PCM}}\mathbf{V} \tag{3.5}$$

The equation above is the discretized version of Eq. 3.3, which gives explicitly $\mathbf{q}$ as a function of $\mathbf{V}$, which is the vector collecting all the values of the MEP $\Phi(\mathbf{s}_i)$ on the tesserae representative points $\mathbf{s}_i$ ($V_i=\Phi(\mathbf{s}_i)$).

The quantities in bold indicate vectors and matrices that represent the quantities and operators in Eq. 3.3. Particularly, $\mathbf{q}$ and $\mathbf{V}$ are column vectors of dimension N_{tess}, and $\mathbf{A}$ is a diagonal matrix collecting the areas of all the surface elements. $\mathbf{Q}^{\text{PCM}}$ is implicitly defined in Eq. 3.5, and it is called the solvent response matrix.

So far, we have seen how to obtain both formally (Eq. 3.3) and practically (Eq. 3.5) an expression for calculating the polarization of the solvent (polarization charges, $\mathbf{q}$) due to the presence of the solute (MEP, $\mathbf{V}$). Let us now see how this impacts the quantum-mechanical description of the molecular system.

The quantum mechanical problem

The standard approach for including solvation effects in the quantum description of the molecule is to define a new quantity with respect to which optimize the quantum state of the molecule, such a quantity is known as free energy in solution $\mathcal{G}$:

$$\mathcal{G}[|\Psi\rangle] = \langle\Psi|\hat{H}_0 + \frac{1}{2}\hat{V}_\sigma|\Psi\rangle \tag{3.6}$$

As we can see $\mathcal{G}$ is a functional of the electronic state only, since we are implicitly adopting the Born-Oppenheimer approximation. In the previous equation, $\hat{H}_0$ is the electronic Hamiltonian of the molecule in gas phase and $\hat{V}_\sigma$ is the operator accounting for the Coulomb interaction between the ASCs $\sigma(\mathbf{s})$ representing the solvent polarization generated by the molecule's charge density $\rho(\mathbf{s})$ and the electrons of the molecule.

If we apply the variational principle to the free energy functional $\mathcal{G}[|\Psi\rangle]$ (Eq. 3.6), under the constraint of a normalized wavefunction, it is possible to derive a non-linear Schrödinger equation with an effective Hamiltonian $\hat{H}^{\text{eff}}$ that includes the solute-solvent interaction[54]:

$$\hat{H}^{\text{eff}}_{|\Psi\rangle}|\Psi\rangle = [\hat{H}_0 + \hat{V}_\sigma(|\Psi\rangle)]|\Psi\rangle = E|\Psi\rangle\,, \tag{3.7}$$

where we have highlighted the non-linearity of the equation by explicitly reporting the dependence of the interaction operator on the electronic wavefunction.

For the sake of our purposes, is convenient to define the interaction operator in second quantization. This allows us to get a deeper insight into the meaning of this operator and also to illustrate better how to calculate the solvation free energy within the VQE algorithm.

Therefore, by separating the contributions of electrons and nuclei, in second quantization we can write $\hat{V}_\sigma$ as:

$$\hat{V}_\sigma = W_{NN} + \mathbf{v}_N^T \cdot \langle\Psi|\hat{\mathbf{Q}}|\Psi\rangle + \sum_{p,q}[\frac{1}{2}(j_{pq} + y_{pq}) + x_{pq}]\hat{E}_{pq}\,. \tag{3.8}$$

Here W_{NN} is the interaction between nuclei and their polarization charges; the indices p, q run over the basis of molecular orbitals, j_{pq} is the interaction term between the electrostatic potential produced by the electronic charge distribution $-\chi_p(\mathbf{r})^*\chi_q(\mathbf{r})$, evaluated at each tessera, with the ASC generated by the nuclear charge distribution:

$$j_{pq} = \mathbf{v}_{pq}^T \cdot \mathbf{q}_N \qquad (\mathbf{v}_{pq})_i = -\langle\chi_p|\frac{1}{|\mathbf{r} - \mathbf{s}_i|}|\chi_q\rangle\,, \tag{3.9}$$

Similarly, y_{pq} represents the interaction between the nuclear potential and the ASC generated by the elementary electronic charge distribution $-\chi_p(\mathbf{r})^*\chi_q(\mathbf{r})$, called $\mathbf{q}_{pq}$

$$y_{pq} = \mathbf{v}_N^T \cdot \mathbf{q}_{pq} \qquad (\mathbf{v}_N)_i = \sum_m \frac{Z_m}{|\mathbf{R}_m - \mathbf{s}_i|} \qquad (\mathbf{q}_{pq})_i = \sum_j \mathbf{Q}_{ij}^{\mathrm{PCM}}(\mathbf{v}_{pq})_j\,, \tag{3.10}$$

where $\mathbf{v}_N$ is the nuclear potential and Z_m, $\mathbf{R}_m$ are the nuclear charge and position of the mth nucleus.

Finally, we have the interaction term between the electrons and the ASC generated by themselves:

$$x_{pq} = \mathbf{v}_{pq}^T \cdot \langle \Psi | \hat{\mathbf{Q}} | \Psi \rangle \,, \tag{3.11}$$

where we have introduced the apparent charge operator $\hat{\mathbf{Q}}$, also appearing in Eq. 3.8, given by:

$$\hat{\mathbf{Q}} = \sum_{pq} \mathbf{q}_{pq} \hat{E}_{pq} \,. \tag{3.12}$$

The operator $\hat{V}_\sigma$ reported in Eq. 3.8 is a one-body operator since it represents the interaction between a charge distribution (the ASC) and the electrons of the molecule, formally analogous to the interaction term between nuclei and electrons in the standard molecular Hamiltonian. Since it is a spin-free operator, we have written it directly in terms of singlet excitation operators $\hat{E}_{pq}$:

$$\hat{E}_{pq} = a_{p\alpha}^\dagger a_{q\alpha} + a_{p\beta}^\dagger a_{q\beta} \tag{3.13}$$

To conclude this section we summarize the standard procedure to find the solution of the coupled equations for the solvent (Eq. 3.5) and the solute (Eq. 3.7) response.

The idea is to find the minimum of the $\mathcal{G}$ functional with a self-consistent procedure. For a given initial approximation of the many-electron wave function of the molecule, the electrostatic potential $\mathbf{V}$ is calculated on each tessera. Subsequently, the polarization charges are obtained using (Eq. 3.5). In turn, such charges enter directly the definition of the effective Hamiltonian (see Eq. 3.8) that allows us to compute the an improved wavefunction and the corresponding $\mathcal{G}$, (Eq. 3.6). Then, one iterates these steps to converge the value of the free energy.

In the next section we will describe the PCM-VQE algorithm. At the heart of this new hybrid quantum-classical algorithm there is the idea of translating the just mentioned self-consistent procedure to a procedure where the minimization of the free energy functional and the solution of the electrostatic problem are performed classically while the quantum computer is used to generate the trial wavefunction and evaluate the expectation values needed to calculate the corresponding solvation free energy.

3.3 PCM-VQE

Here we describe the extension of the VQE algorithm to include the solvation effects using the PCM model described in the previous section.

We start considering the standard workflow of the VQE. We use a quantum computer to prepare a trial state of the N-electrons molecular wave function and to measure the expectation value of the corresponding Hamiltonian. Subsequently, the prepared state is variationally optimized to find the ground state energy. A classical optimizer is used to adjust the variational parameters $\bar{\theta}$ that define the quantum circuit preparing the many-electron wave function.

In order to account for solvation effects within the VQE algorithm, we generalize the objective function to be the free energy in solution as defined in Eq. 3.6:

$$\mathcal{G}[\bar{\theta}] = \langle \hat{H}_0 \rangle_{\bar{\theta}} + \frac{1}{2} \langle \hat{V}_\sigma(\bar{\theta}) \rangle_{\bar{\theta}} \tag{3.14}$$

Where we recall $\hat{H}_0$ is the molecular Hamiltonian of the molecule *in vacuo* and V_σ is the solute-solvent interaction operator defined in Eq. 3.8. By taking the expectation values of these observables in the prepared state $|\Psi(\bar{\theta})\rangle$ we re-write the cost function as,

$$\mathcal{G}[\bar{\theta}] = \sum_{p,q} h_{pq} d_{pq}(\bar{\theta}) + \frac{1}{2} \sum_{p,q,r,s} g_{pqrs} D_{pqrs}(\bar{\theta}) + \frac{1}{2} \sum_{p,q} [(j_{pq} + y_{pq}) + x_{pq}(\bar{\theta})] d_{pq}(\bar{\theta}) + \frac{1}{2} W_{NN} \tag{3.15}$$

Here p, q, r, s are indices running over the orbitals (note we are considering a spin free Hamiltonian, proper for the usual condition when no magnetic fields or spin-orbit coupling is considered), $d_{pq}(\bar{\theta})$ and $D_{pqrs}(\bar{\theta})$ are the one- and two-electrons orbital Reduced Density Matrices (1-, 2-RDMs) defined as follows[55]:

$$\begin{aligned} d_{pq}(\bar{\theta}) &= \langle E_{pq} \rangle_{\bar{\theta}} \\ D_{pqrs}(\bar{\theta}) &= \langle E_{pq}E_{rs} - \delta_{rq}E_{ps} \rangle_{\bar{\theta}} \,. \end{aligned} \tag{3.16}$$

The possibility of retrieving $d_{pq}(\bar{\theta})$ and $D_{pqrs}(\bar{\theta})$ as expectation values is guaranteed by the inherent variational procedure of the method, which is also the case for the UCCSD ansatz [56]. This is non-trivial in general, and for non variational approaches the expression should be replaced by strategies such as the introduction of an auxiliary variational Lagrangian. In those cases, the use of Eq. 3.16 would represent just an approximation[55] to the proper density matrices.

From what we have seen in the previous section it is easy to see that the solvation free energy contribution to the total free energy depends both implicitly and explicitly on the circuit parameters. The implicit dependency stems from the definition of the new interaction operator, the explicit dependency is a result of the relaxation of the wavefunction in presence of the reacting field. Concerning the dependence of the interaction operator matrix elements on the variational parameters, it is instructive to get a better intuition on the modified procedure of the PCM-VQE to make explicit the presence of the variational parameters in Eq. 3.11:

$$x_{pq} = \mathbf{v}_{pq}^T \cdot \langle \Psi | \hat{\mathbf{Q}} | \Psi \rangle \xrightarrow{\text{PCM-VQE}} x_{pq}(\bar{\theta}) = \mathbf{v}_{pq}^T \cdot \langle \Psi(\bar{\theta}) | \hat{\mathbf{Q}} | \Psi(\bar{\theta}) \rangle \tag{3.17}$$

This last feature gives rise to an hybrid algorithm in which the classical optimization routine is tasked with the optimization of a cost functional with a dependence on the parameters that is different to that of standard VQEs: as a consequence of the non-linearity here we jointly optimize the quantum state and the observable w.r.t. which we compute the expectation value.Whether this feature has an impact on the convergence properties of the algorithm is a topic that deserves further study in terms of the theory of hybrid variational algorithms per se. In this paper, we will address this problem numerically only for a few examples.

Another important point to comment is that with the definition of this new cost functional we are including solvent effects in our description without any additional cost of quantum computational resources (see the SI Sec. "Algorithmic complexity: PCM overhead" for a more in-depth analysis of the computational cost) as the same quantities needed to measure the Hamiltonian expectation value in gas-phase are needed to update the interaction operator matrix elements (as shown in Eq. 3.5 and Eq. 3.11 they only depend on the 1-RDM) and to compute the solute-solvent contribution to the free energy (Eq. 3.14).

Finally, we notice that once the density matrices are extracted from the QC, the solution of the electrostatic problem using Eq. 3.5 is straightforward as the solvent response matrix $\mathbf{Q}^{\text{PCM}}$ remains unchanged through all the calculation.

The PCM-VQE algorithm consists of five steps, as it is sketched in Fig. 3.1:

1. The molecular wavefunction is encoded into the state of the quantum computer. Several mappings have been developed in the literature, see Ref.[57] for an extended review on the topic.

2. Transform the quantum computer initial state according to a unitary operation (often referred as the VQE ansatz) $U(\bar{\theta})$ which depends parametrically on the set parameters $\bar{\theta}$.

3. Evaluate the one- and two-electron orbital RDMs using the trial state prepared by the quantum computer.

4. Update the polarization charges using Eq. 3.5 and the matrix elements j_{pq}, y_{pq} and $x_{pq}(\bar{\theta})$ accordingly.

5. Evaluate the free energy functional $\mathcal{G}[\bar{\theta}]$ and compute, if needed, its gradient with respect to the circuit parameters.

Finally, steps 2-5 are repeated until the value of the molecule's free energy in solution is converged.

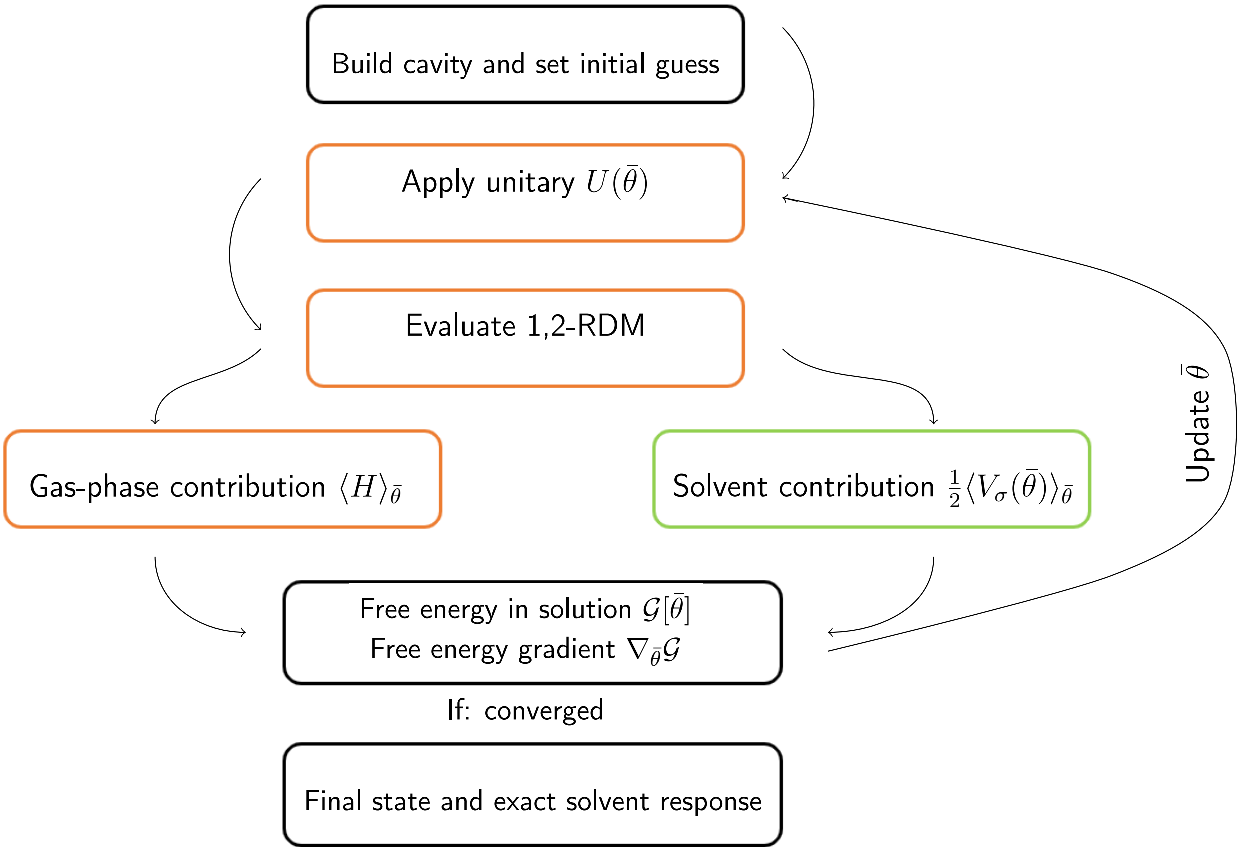


Figure 3.1: Schematic representation of the PCM-VQE algorithm. Black boxes represent operations that involve uniquely classical computation, orange boxes refer to operations that are performed by the quantum computer. Computing the solvent response (green box) is an hybrid computing operation as the classical solver of the IEF-PCM equation is fed by the quantum processor with the 1-RDM. The PCM-VQE loop is iterated until a convergence criterion is satisfied providing the final state of the solute molecule and the corresponding reaction field of the solvent.

For the sake of clarity, here we stress that the non-linearity of the effective Hamiltonian in Eq. 3.7 prevents the inclusion of the solvation effects in the VQE by simply substituting the molecular integrals computed in gas-phase with the molecular integrals computed in solution.

Indeed, the result obtained in this fashion would only provide the optimal variational parameters which enable to prepare the ground state of a molecule *in* *vacuo* whose MOs result from a HF calculation in solution. Such a wavefunction would differ (providing inaccurate results) both from the solution given by the IEF-PCM model coupled to a standard method of quantum chemistry and from the PCM-VQE. A scheme of the algorithm highlighting the interplay of classical and quantum libraries is given as Fig. S3.1; the code is available on GitHub[58].

This concludes the description of the PCM-VQE algorithm and the Theory section. In the following we will discuss the technical details of the implementation and the results obtained both with a noiseless simulation and in presence of a simulated quantum noise.

3.3.1 Computational details

The PCM-VQE algorithm has been implemented in a Python code[58] realizing the interface between the Psi4[59] quantum chemistry package and the PennyLane quantum library[60]. Psi4 was used to compute the molecular integrals, build the solute cavity and solve the electrostatic problem (through its interface with PCMSolver[61]), and PennyLane functionalities were used to implement the quantum algorithm. That is, defining the quantum circuit preparing the molecular trial state, computing the expectation value of the many-body observables and optimizing the quantum circuit parameters.

We have performed numerical simulations to compute the free energy in solution of the trihydrogen cation (H_3^+), beryllium hydride (BeH_2) and water (H_2O) molecules at their equilibrium geometry, shown in Fig. 3.2, computed in gas-phase using the STO-3G basis set. Two examples using a larger basis set (6-31G) are provided in the Supporting Information. Here we remark that the use of STO-3G as basis set should be avoided when the goal of the numerical simulation is to quantitavely predict a property and/or compare the result with an experimental measure. This was not the case for the present study where the purpose of the numerical experiments is to showcase the newly developed algorithm on a set of different molecules. The choice of a minimum basis set is therefore motivated to avoid overflowing the computational resources required by the used quantum computer simulators as previously done in other works[40, 62, 63]. The molecular cavities are built in PCMSolver according to the GePol algorithm [53] using the atomic radii reported in Ref. [64]. The choice of the investigated systems has been made to span a set of molecules with different dipole moment, charge state (quantities that are deeply involved when solvation effects are taken into account) and spatial symmetry, so as to test the implementation and algorithmic robustness over different situations. The classical reference calculations have been performed with the Psi4 code at the CCSD/IEF-PCM level of theory[31] using the same solute cavities.

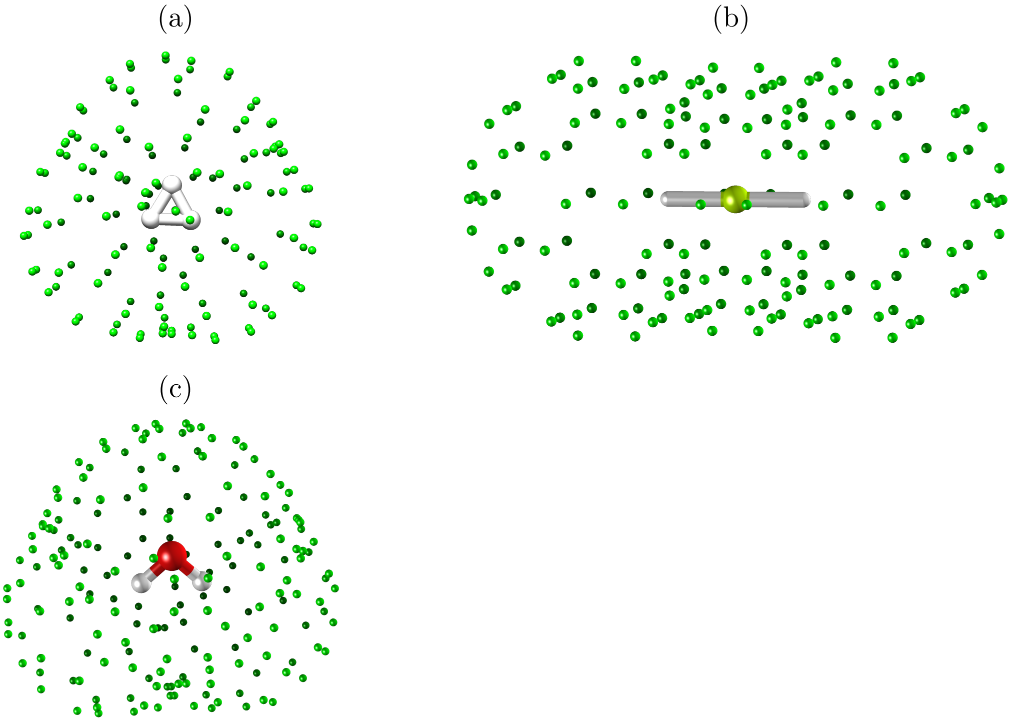


Figure 3.2: Structures of studied systems: (a) trihydrogen cation, (b) beryllium hydride and (c) water. The small green dots are the representative points of each tessera, where ASCs are located, and are spread on the molecularly shaped cavity boundary. Please note that different cavity sizes are not to scale. These figures have been produced with the software Chimera[65].

The variational quantum circuits to prepare the trial states $|\Psi(\bar{\theta})\rangle$ of the simulated molecules are built following the methodology reported by the authors in

Table 3.1: Variational parameters and maximum number of iterations for the PCM-VQE calculations showed in Fig.3.4.

	H_3^+	BeH_2	H_2O
Variational parameters	2	18	30
Max. iterations	12	15	12

Ref. [66]. For the sake of completeness, we outline here the main steps for building the quantum circuit. First, the N-qubit register encoding the molecular spin-orbitals is initialized to the Hartree-Fock (HF) state of the molecule. That is, the first N_e qubits, with N_e being the number of electrons, are set in the state $|1\rangle$ and the other $N - N_e$ qubits in the state $|0\rangle$. Thus, to prepare a many-electron state beyond the meanfield approximation, we apply particle-conserving single- and double-excitation gates on the initial state (see Fig. 3.3 and SI for the circuits used in this work). These excitation operations are implemented in PennyLane as Givens rotations which act on the subspace of two and four qubits [47]. As a result, the final state is a superposition of the HF state and other states encoding multiply-excited configurations [66]. In Tab.4.2 we report the number of variational parameters (i.e., number of Givens rotations) for the systems studied in this work and the corresponding number of maximum iterations needed to achieve the results shown in Fig.3.4 .

Furthermore, we have used the adaptive method proposed in Ref. [66] to select the excitation operations that are important to compute the ground state of the solvated molecules. In addition, to check the reliability of the method to different implementations of VQE, we also explored a more system-agnostic ansatz, the unitary coupled-cluster ansatz truncated at the level of single and double excitations (UCCSD) [39] (see SI).

On the other hand, evaluating the cost function defined in Eq. (3.15) for given set of the variational parameters $\bar{\theta}$ requires to compute the one- and two-electron orbital reduce density matrices $d_{pq}(\bar{\theta})$ and $D_{pqrs}(\bar{\theta})$, respectively. To that aim, we use the Jordan-Wigner transformation [67] to map the fermionic operator $\hat{E}_{pq}$ into the Pauli basis, and compute their expectation values in the trial state prepared by the quantum circuit. Thus, we proceed to minimize the cost function to obtain the free energy of the solvated molecules. The optimization of the circuit parameters in the absence of noise was performed using an adaptive gradient descent algorithm while a gradient-free optimizer [68] was used in the case of noisy simulations.

In addition, we investigated the capability of the present PCM-VQE implementation to estimate solvent effects in a system in the presence of a high degree of static correlation such as the double dissociation bonding profile of water. For the latter calculations we have used the same variational ansatz exploited for the single point calculations at the equilibrium geometry.

The results concerning the implementation on a simulated noisy quantum hardware are obtained by using a noise model for the IBM Q Mumbai quantum processor as implemented in the Qiskit library[69]. It includes one- and two-qubits gate error probabilities, pulse duration for the basis gates, readout errors and thermal relaxation effects tuned upon the experimental parameters. Each circuit has been repeated 8192 times to build relevant statistics, we set the number of shots per circuit to match the maximum number allowed on the actual quantum device. In the SI (Sec. "Measurement budget allocation and statistical errors") we provide a theoretical discussion on the error due to a finite sampling of the expectation value. Here we summarize from a practical point of view. The error bars reported in the results' section have been obtained assuming the standard deviation on the expectation value of each Pauli string equal to 1, which is an upper bound for this quantity. Subsequently the error on each Pauli string is obtained dividing by the square root of the number of shots executed (8192 in our case), since their expectation value is obtained by independent measures. The final error bars on (free) energies are then given by standard error propagation applied to the Hamiltonian mapped on the Pauli strings.

3.4 Results

3.4.1 Numerical simulations in noise free conditions

Single point calculations

Fig. 3.4 shows the values of the free energy in solution $\mathcal{G}[\bar{\theta}]$ as a function of the iterations for the molecules depicted in Fig. 3.2. All the results of this section are reported following the same color code in the plots. The solid blue line corresponds to the free energy evaluated with the PCM-VQE algorithm, the orange dashed line refers to the free energy evaluated classically at the PCM-CCSD level of theory, and the green dashed line gives the free energy value computed classically with the PCM-HF method. The left panel (Fig. 3.4a) refers to the trihydrogen cation; a two electron system whose wavefunction in the STO-3G basis can be encoded by using six qubits.

This circuit is able to prepare the parameterized state $|\Psi(\theta_1, \theta_2)\rangle$ defined as:

$$|\Psi(\theta_1, \theta_2)\rangle = \cos(\theta_1)\cos(\theta_2)|\mathrm{HF}\rangle - \cos(\theta_1)\sin(\theta_2)a_5^\dagger a_4^\dagger a_1 a_0|\mathrm{HF}\rangle - \sin(\theta_1)a_3^\dagger a_2^\dagger a_1 a_0|\mathrm{HF}\rangle\,. \tag{3.18}$$

Particularly, the last expression shows the efficiency of the adaptive procedure which enables to generate a variational ansatz that spans selectively the subspace corresponding to the set of Slater determinants that contribute to the FCI wavefunction without allowing to reach states having components along different electronic configurations.

As a first comment, we can notice how the quantum simulation algorithm is able to recover all the correlation energy and solvent effects contribution w.r.t. the value given by the CCSD reference which is exact in this case. Moreover, we can see that the optimization convergence is reached within only ten iterations. This is due to the optimization settings that comprise an educated guess encoding the $|\mathrm{HF}\rangle$ state. Furthermore, we have adopted a variational ansatz that prepares the FCI ground-state for this molecule. Such a strategy allows to further reduce the cost of including solvation effects: in the PCM-VQE theory section we have

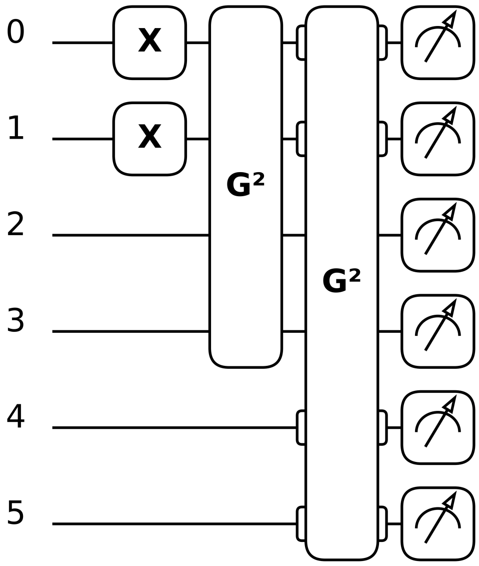


Figure 3.3: Quantum circuit used for the PCM-VQE simulations on the H_3^+ molecule. As mentioned in the main text, the circuits used are composed by a set of initial state preparation gates (here the X gates acting on the first two qubits) and by particle-conserving unitary operators here implemented as Givens rotations (whence the label G). Figure obtained using the quantum circuit drawer function as implemented in PennyLane[60].

seen that no other additional costs are present concerning the quantum part of the algorithm. Here we point out that only a few more iterations are needed to account for effects of the solvent. In this regard, we highlight that constructing the variational ansatz with an adaptive procedure on the wavefunction in gas-phase is an approximation that applies best if the electronic structure of the solute is not severely modified by the inclusion of the solvent. When this occurs, it may happen that contributions from a few excited configurations, not relevant in the electronic structure *in vacuo*, are lost. In such cases, the resulting wavefunction remains a good guess for the effective Hamiltonian in solution, and the adaptive procedure may be restarted with such Hamiltonian to include the relevant excitations. In the SI we show additional results for a different system (HeH^+) in which a more system agnostic ansatz is considered (UCCSD) with both STO-3G and 6-31G basis sets to further assess the performance of the modified algorithm. Moreover, we include an additional calculation with an adaptive circuit for the H_3^+ molecule using the 6-31G basis set (SI Sec. "H_3^+ calculations with PCM-VQE/6-31G").

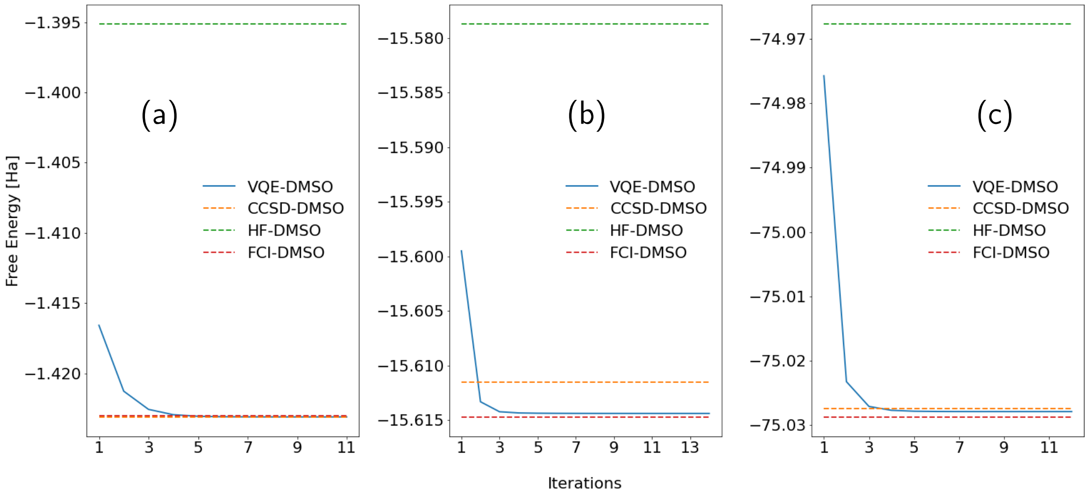


Figure 3.4: PCM-VQE results. Ground state free energies in dimethyl sulfoxide (DMSO) for the (a) Trihydrogen ion H_3^+, (b) Beryllium hydride BeH_2 and (c) Water H_2O. The solid blue line represents the free energy in solution obtained with the PCM-VQE as a function of the iterations using a STO-3G basis set. As a reference we show the free energy obtained with an HF-PCM calculation (green dashed line) and with a CCSD-PCM calculation (orange dashed line) using the same basis set. The red dashed line shows the reference value obtained running a FCI calculation *in vacuo* and adding to the total energy contribution the polarization energy.

Figs. 3.4b-3.4c plot the numerical results for the BeH_2 and the H_2O molecules. For these larger systems the core electrons localized in the s-type orbitals of the beryllium and the oxygen atoms are excluded from the active space. That means that we have four and eight active electrons in the BeH_2 and the H_2O molecules, respectively, whose wavefunctions are represented using twelve qubits. We chose to keep the core electrons frozen for practical purposes as their optimization does not impact effectively neither the electronic structure nor the description of the solute-solvent interaction.

Table 3.2: Free energies in solution for the studied systems (Fig. 3.2). Comparison between FCI/CCSD/VQE (gas-phase) and between IEF-FCI/IEF-CCSD/PCM-VQE (DMSO). Molecular geometries for the calculations as well as values obtained for the VQE in gas phase with the adaptive ansatz have been taken from Ref. [66]. Energy values are reported in Hartree.

	Gas-phase			Solvent		
	FCI	CCSD	VQE	FCI-PCM^a	CCSD-PCM	PCM-VQE
$\mathcal{G}_{H_3^+}$	-1.2744	-1.2744	-1.2744	-1.4230	-1.4231	-1.4231
$\mathcal{G}_{BeH_2}$	-15.5952	-15.5945	-15.5945	-15.6147	-15.6115	-15.6144
$\mathcal{G}_{H_2O}$	-75.0233	-75.0231	-75.0230	-75.0287	-75.0273	-75.0279

(a) The results reported under the label FCI-PCM are obtained by adding to the FCI energy in gas phase the polarization energy computed with the 1-RDM of the corresponding wavefunction.

As we can see looking at the figures all the consideration made above in the case of H_3^+ still hold. This implies that, at least for the simple systems that we can tackle with current NISQ devices, the procedure involving first the simulation in absence of the solvent to detect the more relevant excited configurations retains its effectiveness as the size of the system increases. In addition, we can note that in this case, where more electrons are involved, the PCM-VQE algorithm predicts lower energy states as compared with the classical simulations at the level IEF-CCSD. This is not surprising as the variational ansatz spans a larger space than CCSD. Further, this is in accordance with the comparison, *in vacuo*, between the FCI and CCSD wavefunctions, Tab. 3.2a.

We observe from Tab. 3.2a that the variational ansatz is able to recover, in gas-phase, almost all the correlation energy in all cases with maximum differences of $\approx$ 0.4 kcal/mol (i.e., within chemical accuracy). Further, moving to the calculations in DMSO, with the PCM-VQE we are able to improve the results of the IEF-CCSD wavefunctions up to 1.9 kcal/mol as in the case of BeH_2. For the sake of completeness we reported in Tab. 3.3 the solvation free energies for the simulated molecules in DMSO. We recall that the solvation free energy is computed by taking the difference of the free energy in solution and the energy of the solute *in vacuo*:

Table 3.3: Solvation free energies for the studied systems (Fig. 3.2). Comparison between IEF-CCSD and PCM-VQE in DMSO. Energy values are reported in eV.

	H_3^+	BeH_2	H_2O
$\Delta\mathcal{G}_{\text{PCM-VQE}}$	-4.046	-0.539	-0.133
$\Delta\mathcal{G}_{\text{IEF-CCSD}}$	-4.046	-0.460	-0.114

$$\Delta\mathcal{G}_{sol} = \mathcal{G}[\bar{\theta}_{sol}] - \langle\Psi(\bar{\theta}_{vac})|H|\Psi(\bar{\theta}_{vac})\rangle \tag{3.19}$$

While in the next section we will look at this quantity evaluated in presence of quantum noise, here we focus on the accuracy obtained with the noiseless calculations. In all cases we are able to recover quantitatively the description obtained with a IEF-CCSD calculation with the larger deviation observed for the BeH_2 molecule.

In the SI (Tab. S1) we have also reported analogous results obtained for the same molecules in water.

Solvation effects along the water molecule double bond dissociation profile

In this section we apply the PCM-VQE to estimate solvation effects along the water molecule double bond dissociation profile. Such a system has been largely investigated in the context of electronic structure theory [70][71][72] and very recently it has been considered in Ref.[73] to benchmark the effects of adding an orbital optimization procedure to the VQE method.

The reason behind the interest in this problem is that such a system, due to its symmetry, exhibits strong static correlation effects due to two equivalent electronic configurations arising in the bond dissociation limit. In particular, as thoroughly discussed by Olsen *et al.*[74], around the equilibrium geometry the HF determinant contribution to the FCI wavefunction largely outclasses all the other electronic configurations. On the other hand, as the bond length increases, its contribution becomes smaller and smaller until it vanishes at the dissociation limit.

Simultaneously, the occupation of the orbital pair (3a$_1$, 1b$_2$), almost completely populated in the HF determinant, loses occupation to the orbital pair (4a$_1$, 2b$_2$) until they are equally populated in the dissociation limit. Here we show that, despite the inherent problem complexity, we are able to obtain a fair estimate of the solvation effects even using a variational network tailored specifically on the electronic structure of the equilibrium geometry.

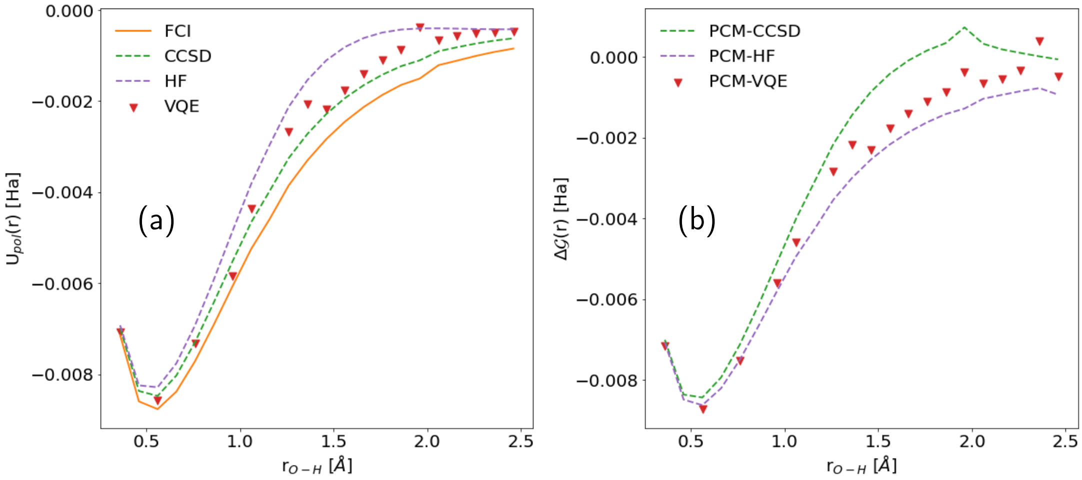


Figure 3.5: Solvation effects for the water molecule (in water) along the double bond dissociation profile. (a) U_{pol}, polarization energy; (b) $\Delta\mathcal{G}$, solvation free energy. As a reference we report the results obtained at the FCI (orange solid line), CCSD (green dashed line) and HF (purple dashed line). Red triangles are obtained with the PCM-VQE algorithm where the number of optimization steps has been increased (up to 500) with the bond distance. For more details see SI. All calculations were performed using a STO-3G basis set.

In Fig. 3.5a we show the results for the polarization energy, defined as $U_{pol} = \frac{1}{2}\langle V_\sigma \rangle$, computed as a function of the bond distance. This quantity accounts for the interaction between the solute charge density in gas phase and the corresponding polarization charges. Due to the lack of relaxation of the electronic degrees of freedom in presence of the solvent it may deviate from the solvation free energy defined in Eq. 3.19.

As a first comment, we can notice that the double bond dissociation profile for the polarization energy curve predicts the greatest energetical stabilization due to the hydration process around ≈ 0.6 Å which, interestingly, differs from the equilibrium distance $r_{OH} = 0.96$ Å . For short bond lengths the PCM-VQE (red triangles) predicts values of polarization energy slightly closer to the FCI result (solid orange line) than the CCSD (green dashed line). On the other hand, as the bond length increases, we observe larger deviations from the FCI result. In spite of this, we obtain better estimates w.r.t. the HF level of theory, with a maximum variation from the CCSD estimate < 1 kcal/mol. It is important to emphasise that this degree of agreement is not readily apparent from the results shown in the SI regarding the absolute energy values obtained with the different methods.

Now we consider the results for the hydration free energy reported in Fig. 3.5b. Here the observed qualitative behavior is reversed: the PCM-VQE predicts a slightly favourable energy contribution to the hydration process compared to CCSD for almost all bond length values considered. As noted previously, for short bond lengths, the discrepancies among the values obtained with the different methods are small; moving to the region where the strong static correlation arises larger deviations between the different approximations become more apparent with the PCM-VQE outcomes producing values in between the ones of CCSD and HF.

Finally, we want to discuss the slightly noisy trend observed in both graphs for the polarisation energy and the solvation energy estimated with the (PCM-)VQE. This could result from an interplay of different effects. First of all we highlight that the overall quality of these results could be easily further improved adopting a different ansatz such as those used in Ref. [73]. Here we did not focus on the technical refinement of the implementation as was beyond the scope of the present work. Further, as shown in more detail in the SI and mentioned in the computational details, the optimisation procedure does not take place with the same number of iterations for each bond length. This choice has been dictated by the fact that, to preserve consistency with the computational procedure, in all cases the initial state for the VQE optimization was considered to be the $|\mathrm{HF}\rangle$ state which is an increasingly worse ansatz as electronic correlation increases and thus requires a greater number of iterations before achieving convergence. Although the convergence rate of the optimisation procedure does not seem to be particularly affected by the presence of solvent effects in the cost functional (see SI), this may have influenced the estimation of much smaller numerical values, such as those of $\Delta\mathcal{G}$, compared to the absolute energies (or free energies). This last point is particularly relevant for r_{OH} values greater than 2 Å in which the overall convergence rate is lower for both the case in vacuum and in solution w.r.t. shorter bond lengths.

In this section we have analyzed the results obtained with a PCM-VQE procedure on a noiseless quantum processor. This allows to prove the efficacy of the method but does not give us an idea of its viability on the NISQ devices that are currently available. In the following section we will look at the results obtained simulating a noisy quantum device to understand if the effects of a microscopic environment on a small molecule can be reliably caught by an actual NISQ device.

3.4.2 Effect of quantum noise on solvation free energy

The aim of this section is to understand if we are able to reliably compute with present quantum processors an estimate of the solvation free energy for a molecular system. We will focus on the six qubit system H_3^+ whose charged state determines a strong stabilization of the system due to the solvation process. Here it is important to notice, as explicitly expressed in Eq. 3.19 that the definition of the solvation free energy involves two different sets of variational circuit parameters corresponding to the optimal result found with a standard VQE and to the optimal result after the PCM-VQE procedure.

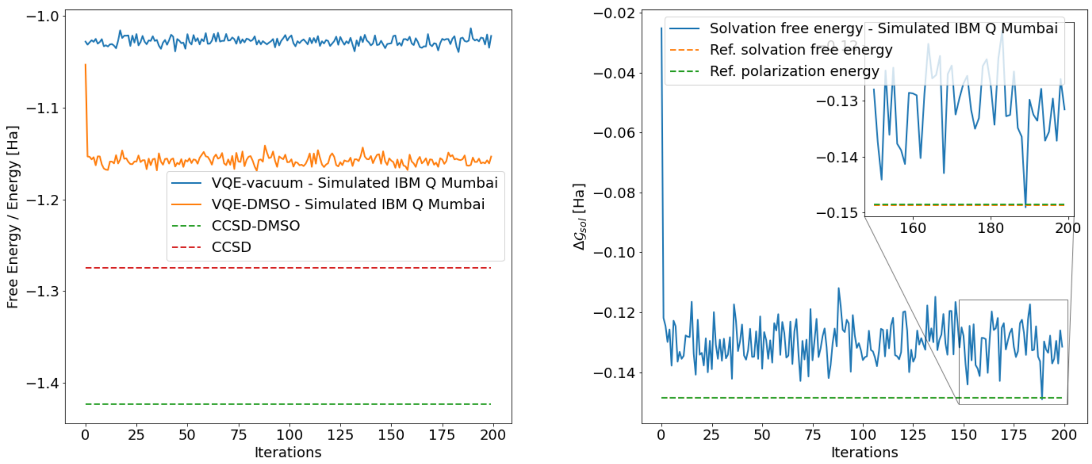
In Fig. 3.6a we report the free energy in solution (solid orange line) and the gas-phase energy (solid blue line) computed by running the PCM-VQE and VQE problem on a classical quantum emulator with a noise model built to mimic the IBM Q Mumbai quantum computer. The variational ansatz adopted is the same used for the calculations in noise free conditions reported in the previous section. In this case we initialized the variational parameters to random values.

As we can see the presence of quantum noise affects the coherent state of the processor and hampers the classical optimization procedure; the result is that the in vacuo energy and free energy estimates are significantly higher w.r.t. the theoretical values reported with dashed lines in the same plot, and both continue to fluctuate with the iteration number. This is in agreement with what has been reported in other works[75, 76]. It represents the current limitations of this technology and calls for the pressing need of error mitigation strategies. Here we notice that the deviation (quantum vs. theoretical benchmark) on the estimated energy in vacuo and on the free energy in solution is very similar, $\Delta \approx 0.275\,\mathrm{Ha}$. This is reasonable since both the quantities to be measured (1,2-RDMs) and the gas-phase vs. solvated wavefunctions are very similar.

Indeed, as shown in Fig. 3.6b, even without any error mitigation procedure, we were able to give a reasonable estimate of the solvation free energy ($\Delta\mathcal{G}_{sol}[\bar{\theta}] = -3.564 \pm 0.163$ eV vs. $\Delta\mathcal{G}_{sol} = -4.054$ eV) reported with the solid blue line as a function of the optimization steps. This value has been obtained performing two independent runs of the VQE algorithm *in vacuo* and in solution and taking the energy difference. Further, to mitigate the effect of stochastic fluctuations, the value reported results from an average over the last 50 points of the iteration plot (see inset of Fig. 3.6) and the error is estimated as $\pm\sigma$ (standard deviation of the last 50 points).

We have also analyzed the behavior of the polarization energy. As we have seen in the previous section, in general it is different from $\Delta\mathcal{G}_{sol}$ since it lacks the contribution of the wavefunction change from gas-phase to solution. However for the present system these two quantities are very similar w.r.t. the classical benchmark calculations. This is also evident in Fig. 3.6b where both reference lines are completely overlapped.

We have also investigated the impact of quantum noise on the computed polarization energy. The goal of this analysis is to understand if estimating the solvation free energy approximating it with U_{pol} gives better results on a NISQ device. In this case we obtained a value of $U_{pol} = -4.054 \pm 0.217$ eV that matches the quantity obtained with a noise free calculation. Error bars are obtained according to the procedure explained in the SI and computational details' section.



(a) (b)

Figure 3.6: PCM-VQE simulation of the H_3^+ molecule in DMSO with a noise model built upon the IBM Q Mumbai quantum processor. (a) Energy obtained with a VQE simulation in presence of the simulated quantum noise (solid blue line); Free energy in solution calculated with the PCM-VQE in presence of the simulated quantum noise (solid orange line). As a reference we show the energy obtained with a classical calculation at the CCSD level of theory (*in vacuo* - red dashed line, DMSO - green dashed line). (b) Solvation free energy for the H_3^+ molecule in DMSO with a noise model built upon the IBM Q Mumbai quantum processor (solid blue line). As a reference we have reported the solvation free energy (orange dashed line) and polarization energy (green dashed line) computed with a CCSD calculation, these latter are superimposed and not distinguishable. The estimated value reported in the main text is obtained taking the average over the last 50 points of the iteration plot (see inset) $\pm\sigma$ (standard deviation of the last 50 points).

We can read these results in light of multiple effects. First of all, as evidenced in the theory section, solvent effects are reproduced optimizing a perturbation operator whose contribution is captured by measuring only the 1-RDM. As such, the number of Pauli strings to measure is dramatically reduced. To better discuss this point we have reported the exact 1-RDM and the estimated 1-RDMs with the calculation in vacuum and in solution (see Fig. 3.7). Furthermore, we have computed the trace distance, $D(\rho, \rho_{exp}) = \frac{1}{2N_e}||\rho - \rho_{exp}||_1$ where N_e is the number of electrons, between the theoretical and experimental 1-RDMs obtaining the same value of $D = 0.075$. Note that the trace distance ranges from 0 to 1 where the former value stands for two identical RDMs and 1 is obtained for two matrices spanning two orthogonal supports. The same value is obtained comparing noisy vs. noiseless PCM-VQE calculations. As we can see a quantitative agreement between the density matrices is apparent. To get the same values for the trace distance is not surprising, since the procedure to extract the 1-RDM is the same (in terms of measurement needed) both for the PCM-VQE and VQE algorithm. This is also shown by the reference values of the polarization energy and of the solvation free energy.

On the other hand, moving to the estimated solvent effects, the outcome is soundly improved when only the polarization energy is considered. Such a result could be rationalized considering that when taking the difference between two values coming from two independent runs the errors due to the sampling of the 2-RDMs (that require a much larger number of Pauli strings to be measured) and the errors for each computed 1-RDM accumulate by propagating. Instead, when the solvation free energy is approximated with the polarization energy the errors arising from the sampling of the 2-RDM are not present. In addition we also avoid the combination of the errors coming from the subtraction of the two quantities. Since the latter are not correlated we do not expect cumulative effects but simply a propagation of more errors on the calculated quantity that contribute to worsen the accuracy of the estimated solvation free energy. We point the reader to the SI for a more thorough analysis. We emphasise that all these results come from a combination of smaller numerical values of the matrix elements and a smaller number of Pauli words to be measured.

Finally, it is important to stress that the accuracy w.r.t. the exact value could be further reduced applying the protocol shown in Ref.[77] (both for $\Delta\mathcal{G}_{sol}$ and U_{pol}) that means post-selecting the measured values upon a total occupation number criterion and applying the McWeeny purification[78]. Here, since a technical a)

$$\begin{pmatrix} 1.96 & 0.00 & 0.00 \\ 0.00 & 1.85 * 10^{-2} & 0.00 \\ 0.00 & 0.00 & 1.85 * 10^{-2} \end{pmatrix}$$

b)

$$\begin{pmatrix} 1.67 & 2.13 * 10^{-3} & -3.19 * 10^{-3} \\ -2.01 * 10^{-3} & 1.98 * 10^{-1} & -2.60 * 10^{-3} \\ 1.01 * 10^{-2} & -4.73 * 10^{-4} & 1.35 * 10^{-1} \end{pmatrix}$$

c)

$$\begin{pmatrix} 1.96 & 0.00 & 0.00 \\ 0.00 & 1.85 * 10^{-2} & 0.00 \\ 0.00 & 0.00 & 1.85 * 10^{-2} \end{pmatrix}$$

d)

$$\begin{pmatrix} 1.67 & 4.60 * 10^{-3} & 1.42 * 10^{-3} \\ -1.53 * 10^{-3} & 1.96 * 10^{-1} & 5.48 * 10^{-3} \\ -1.22 * 10^{-2} & 7.26 * 10^{-3} & 1.36 * 10^{-1} \end{pmatrix}$$

Figure 3.7: H_3^+ noisy simulation of the PCM-VQE algorithm. (a) Comparison between exact orbital 1-RDM, (b) orbital 1-RDM computed with a noisy simulation of the PCM-VQE algorithm and used to compute the solvation free energy, (c) orbital 1-RDM computed with a noise-less simulation of the PCM-VQE algorithm and (d) orbital 1-RDM computed with a noisy simulation of the VQE in gas-phase and used to compute the polarization energy.

optimization of the implementation goes beyond the purposes of this study, we only applied a normalization factor to the matrix elements of the 1-RDM to obtain the correct number of electrons when taking the trace.

In conclusion, although the use of variational quantum algorithms for the simulation of chemical systems of interest is still hampered by the present NISQ computer accuracy, these results suggest that the technical gap to be overcome to accurately evaluate solvation effects may be lower than that to obtain accurate values of the total electronic energy.

3.5 Discussion

In this work we have proposed a direct method, without the need of additional quantum resources, to extend the Variational Quantum Eigensolver algorithm to simulate solvated systems. The analysis of the numerical results obtained from quantum simulations of simple molecules suggests that computing the solvation free energy is a computational task that can be reasonably tackled with current quantum computers.

The inclusion of solvation effects through the IEF-PCM allows to describe *a* *plethora* of microscopic environments thus extending the computational reach of current quantum computers even more than what we have showed in this work. Indeed, here we focused on the simple (yet most common) situation of an homogeneous isotropic solvent but the inclusion of more complex environments such as metal nanoparticles or liquid-liquid phase separations is straightforward and does not require any modification of the proposed quantum algorithm.

Future investigations that can benefit from the present work regard the integration of the proposed algorithm into more elaborated computational pipelines. The first step in this direction is the application of the method presented here to quantum algorithms that perform molecular geometry optimization being another feature strongly affected by the presence of the solvent. Other examples are the problems of calculating optical responses and excited state properties in solution that give rise to a variety of photochemical and photophysical phenomena otherwise unexplorable. In this regard, we mention the very recent contribution of Lee *et al.*[79] where the authors couple classical molecular dynamics and variational quantum simulation[80] to compute the optical response of a multi-chromophoric excitonic system.

Along with the exploration of more exotic systems with NISQ devices, this work opens up to a more theoretical question related to the theory of variational hybrid algorithms: how does the variational landscape topology change when a non-linear Hamiltonian is considered? The importance of this question has been remarked very recently by the work of Bittel and Kliesch[81] where the authors show (not considering observables analogue to the non-linear effective Hamiltonian used in this work) that the classical task of training a variational quantum circuit is NP-Hard. Moreover, the authors find that the complexity class of the classical optimization is not inherited by the complexity of finding the ground state of a local Hamiltonian, which is known to be QMA-Hard[82], but it is an intrinsic feature of the underlying optimization landscape. Given the importance of the question for the possible developments and applications of this type of algorithms, it would be surely worth to show if their finding applies also to classes of cost functionals, such as the free energy in solution considered in this work, where the variational parameters shape not only the quantum computer state but also the measured observable.

In conclusion, to the extent that large-scale simulation of reactivity is among the long-term goals of the computational chemistry community, and that this cannot ignore the insights provided by including environmental effects, this work paves the way for quantum simulation of molecular systems in the next generation quantum processors. We believe that the development of quantum multiscale methods must be part of the second quantum revolution agenda if we want to go beyond the current computational capabilities.

Supporting Information

Pseudocode for the PCM-VQE implementation

Here we show a pseudocode scheme which summarizes the implementation of the PCM-VQE algorithm as it is implemented in our code[58]. Particularly this realizes a dynamical interface between Psi4/PCMSolver/PennyLane.

Our implementation allows to build the `Observable` and `Hamiltonian` objects of PennyLane with the data encoded in the Psi4 object `Wavefunction`. From the latter we have also access the PCMSolver routine that we leverage to update the polarization charges and to compute the corresponding value of polarization energy $\frac{1}{2}\langle\hat{V}_\sigma\rangle$. This last step summarized in the pseudocode as `Compute_sol_solv_int(d(`$\bar{\theta}_k$`), Q`$^{\texttt{PCM}}$`)` is realized linking the call to the Psi4/PCMSolver function to the output of the `QNode` built with the PennyLane library. Finally PennyLane's optimizers update the variational parameters until the convergence criteria are met.

UCCSD Ansatz

Here we report additional results with a noiseless simulation concerning the calculation of the ground state energy of the HeH^+ molecule at the equilibrium distance

Algorithm 1 PCM-VQE - Pseudocode of the implementation.

```
Require: Molecular_integrals, $\mathbf{Q}^{\mathrm{PCM}}$                          ▷ Psi4, PCMSolver
Ensure: $\mathcal{G}[\bar{\theta}]$, $\bar{\theta}$
  Initialization:                                                        ▷ PennyLane
  Fermion_to_qubit_mapping(Molecular_integrals)
  Set_Variational_Form()
  Set_Classical_Optimizer()
  k=0
  $\bar{\theta}_k \leftarrow \bar{\theta}_{guess}$
  while $k \leq max\_iter \;\; or \;\; \|\mathcal{G}[\bar{\theta}_k] - \mathcal{G}[\bar{\theta}_{k-1}]\| \leq \epsilon$ do
      Execute_variational_circuit($\bar{\theta}_k$) return $d(\bar{\theta}_k), \langle \hat{H}_0 \rangle$          ▷ PennyLane
      Compute_sol_solv_int(d($\bar{\theta}_k$), $\mathbf{Q}^{\mathrm{PCM}}$) return $\langle \hat{V}_\sigma(\bar{\theta}_k) \rangle_{\bar{\theta}_k}$          ▷ Psi4,
  PCMSolver
      $\mathcal{G}[\bar{\theta}_k] \leftarrow \langle \hat{H}_0 \rangle_{\bar{\theta}_k} + \frac{1}{2} \langle \hat{V}_\sigma(\bar{\theta}_k) \rangle_{\bar{\theta}_k}$
      k+=1
      $\bar{\theta}_k \leftarrow \bar{\theta}_{k+1}$                                                  ▷ PennyLane
      $\mathcal{G}[\bar{\theta}_k] \leftarrow \mathcal{G}[\bar{\theta}_{k+1}]$
  end while
  $\bar{\theta} \leftarrow \bar{\theta}_k$
  $\mathcal{G}[\bar{\theta}] \leftarrow \mathcal{G}[\bar{\theta}_k]$
  return $\mathcal{G}[\bar{\theta}]$, $\bar{\theta}$
```

in dimethyl sulfoxide (DMSO) (Fig. S1) using the PCM-VQE.

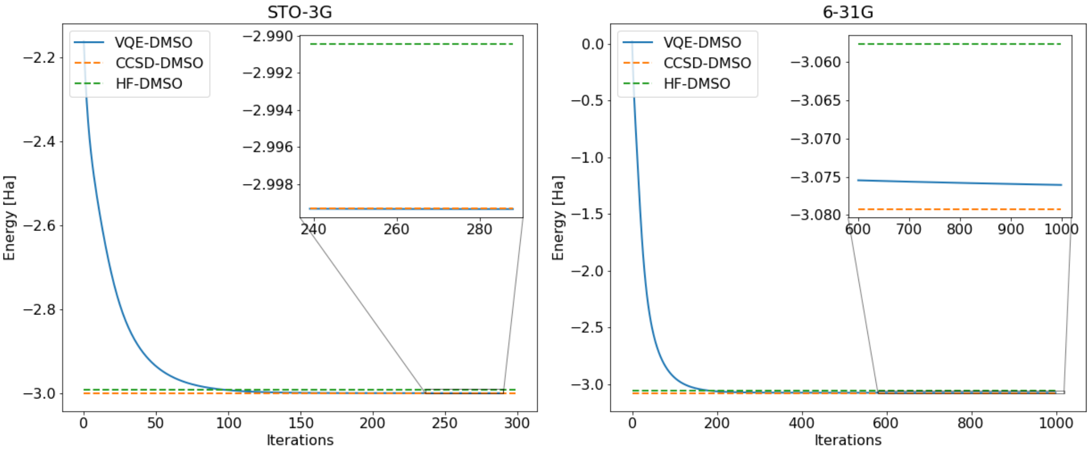


Figure S1: Ground state energies of the HeH^+ molecule in DMSO. (Left panel) The solid blue line represents the energy obtained with the PCM-VQE as a function of the iterations using a STO-3G basis set. As a reference we show the energy obtained with an HF-PCM calculation (green dashed line) and with a CCSD-PCM calculation (orange dashed line) using the same basis set. (Right panel) The same plot is reported adopting the 6-31G basis set.

We have run these calculations to test the hybrid algorithm also on a well established ansatz such as the Unitary Coupled Cluster with Singles and Doubles (UCCSD). We recall that in this approach the optimization parameters coincide with the cluster amplitudes of the single and double excitation operators:

$$\hat{U}(\bar{\theta}) = e^{\hat{T}_1(\bar{\theta})+\hat{T}_2(\bar{\theta})} \quad , \quad \begin{cases} \hat{T}_2(\bar{\theta}) = \sum_{i>j>k>l} \theta_{ijkl}(a_k^\dagger a_l^\dagger a_j a_i - a_j^\dagger a_i^\dagger a_k a_l) \\ \hat{T}_1(\bar{\theta}) = \sum_{i>j} \theta_{ij}(a_j^\dagger a_i - a_i^\dagger a_j) \end{cases} \tag{S1}$$

The implementation of the previous unitary is done performing a Trotterization since the two excitation operators do not commute. The order of the Trotterization determines the number of control parameters and the depth of the circuit, here we employed a first order approximation[83]. The circuit template is provided by default in the PennyLane library.

Moreover, here we wanted to explore not only a different ansatz but also the effect of a larger basis set. For this reason we chose to investigate HeH^+, as the smallest system exhibiting substantial impact by the presence of a solvating environment, with the STO-3G basis and with the 6-31G basis set. As for the results shown in the main text, the initial state is the Hartree-Fock wavefunction, the molecular wavefunction is mapped onto the quantum computer state with the Jordan-Wigner method and the variational network parameters have been optimized with the same adaptive gradient descent algorithm. The variational parameters are initialized at random.

With regard to the quantum resources required for this calculation the helium hydride wavefunction with a STO-3G basis is mapped in a four qubits' quantum register, on the other hand for the calculation in the 6-31G basis eight qubits are needed.

As we can see we are able to obtain good results recovering for the STO-3G basis set calculation (left panel) $> 99\%$ of the free energy in solution with respect to the exact calculation, while, for the 6-31G basis set (right panel), we reach the 85.4% . It is important to notice that, in this case, the classical optimization algorithm did not meet the convergence criteria and the result could be further improved by increasing the number of iterations.

These results further contribute to show that the performance of the PCM-VQE is similar to what has already been reported in literature for the UCCSD *in vacuo*[39, 84] and that the inclusion of solvation effects does not impact the overall algorithmic efficiency.

H_3^+ calculations with PCM-VQE/6-31G

Here we report additional results on the H_3^+ molecule using the 6-31G basis set in DMSO (see Fig. S2). To perform the calculations we have adopted an adaptive ansatz including only the relevant excitations that contribute to the ground state wavefunction as described in the computational details' section of the main text. We have also reported the circuit structure in Fig. ??. As we can see in Fig. S2 the adaptive ansatz is able to reproduce exactly the energy in vacuum. Indeed we recall that for a two electron system the CCSD calculation is equivalent to Full Configuration Interaction.

Moving to the results for the solvated molecule (right panel) again we can see a very good agreement between the PCM-CCSD and the PCM-VQE calculations but in contrast with the gas phase result the numerical values are not exactly matching. Particularly, the PCM-VQE free energy is 8 meV lower. This is due to the fact that for the solvated system the PCM-CCSD calculation is performed approximating the reaction field at the HF level (PTE approximation in the original reference [31]) as implemented in Psi4 [59]. Interestingly this effect is not noticeable in the STO-3G results of the main text because adopting a smaller basis set the error due to the PTE approximation is negligible.

Finally, it is worth to comment the convergence rate of this calculations w.r.t. the results obtained with a 6-31G basis set in the previous subsection (Fig. S1). As expected, the number of iterations and the convergence rate of the optimization procedure is dramatically improved using the adaptive ansatz instead of a less chemically-aware UCCSD circuit. This is in agreement with what already discussed in literature: the expressibility[85] of a parameterized quantum circuit affects the convergence properties of the classical optimizer due to wider Hilbert space sectors explored[86] w.r.t. adaptive circuits and to the emergence of barren plateaus[87].

Additional results in aqueous solution

In this section we show additional calculations performed to test the PCM-VQE on the same systems studied in the main text but with a different solvent, i.e. water. These results, in accordance with what already discussed for the case of a DMSO solution, allow us to discuss the performance of the algorithm on another very common solvent in the chemical practice.

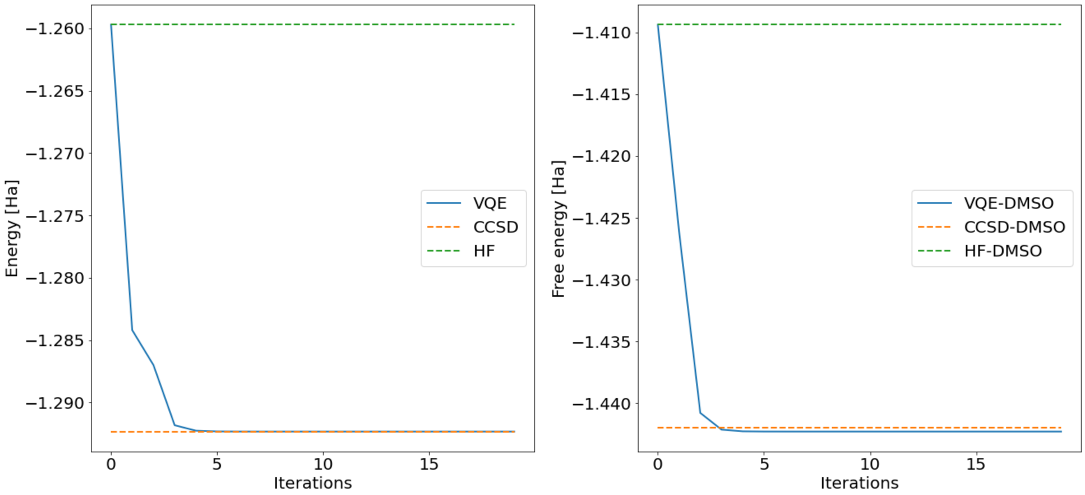


Figure S2: Ground state energies of the H_3^+ molecule with a 6-31G basis set. (Left panel) VQE results *in vacuo*; (Right panel) PCM-VQE results in DMSO. Solid blue line represents the energy obtained from the trial wavefunction as a function of the iterations. As a reference we show the energy obtained with an HF-PCM calculation (green dashed line) and with a CCSD-PCM calculation (orange dashed line) using the same basis set.

Single point calculations

In Tab. S1 we report the results obtained running a PCM-VQE calculation in water. Analogously to what discussed in the main text for the case of DMSO, with the PCM-VQE we are able to reproduce the exact results reported for the CCSD-PCM calculation of the H_3^+ molecule and to recover a value of free energy in solution for the larger molecules (BeH_2 and H_2O) close to the total FCI energy comprehensive of the polarization energy contribution. All the calculations have been carried out according to the computational protocol described in the computational details' section referring to the results shown in the main text.

Energy and free energy plot for the H_2O double dissociation profile

Here we report additional results regarding our analysis on the polarization energy and hydration free energy discussed in the main text for the double bond dissocia-

Table S1: Solvation free energies in water solution for the studied systems. Comparison between IEF-FCI/IEF-CCSD/PCM-VQE (H_2O). Molecular geometries for the calculations have been taken from Ref. [66]. Energy values are reported in eV.

	FCI-PCM^a	CCSD-PCM	PCM-VQE
$\Delta\mathcal{G}_{\mathrm{H}_3^+}$	-4.082	-4.082	-4.082
$\Delta\mathcal{G}_{\mathrm{BeH}_2}$	-0.539	-0.468	-0.545
$\Delta\mathcal{G}_{\mathrm{H_2O}}$	-0.142	-0.117	-0.133

(a) For FCI-PCM, polarization energies are reported rather than solvation free energy.

tion profile of the water molecule. In Fig. S3a-b we report, respectively, the total energy (Fig. S3a) and the free energy in solution (Fig. S3b) computed with the PCM-VQE (red triangles). The energies reported correspond to an optimization convergence rate $\epsilon < 10e^{-8}$ or to a maximum number of 500 iterations. In this case, while maintaining the same trend as observed for U_{pol} and $\Delta\mathcal{G}$, we can see that the differences are considerably more pronounced for the total energy E and the free energy in solution $\mathcal{G}$. In particular, while up to 1.4 Å the (PCM-)VQE result is quantitatively in agreement with that of the FCI (orange solid line) or CCSD (green dashed line), for higher bond length values we observe deviations from the value calculated by the FCI/CCSD method of about 0.2 Ha.

Finally, we comment the data reported in Fig. S4 which show the convergence behavior of the classical optimization procedure as a function of the iterations (different colors) and as function of the bond length $\mathrm{r_{OH}}$. As a first comment we can notice that the converge rate of the optimization procedure for the PCM-VQE (crosses) is very similar to the one observed *in vacuo* (triangles) in accordance with the plots shown in the main text for the single point calculations and with the results shown in Fig. S1 suggesting that the inclusion of a non-linear term in the cost functional should not further affect the classical optimisation algorithm.

Finally, we also notice that the rate of convergence of the optimization slows down considerably as the bond distance increases. This behavior is not surprising and can be explained considering (i) that as the dissociation regime approaches the initial guess state $|\mathrm{HF}\rangle$ provides a worse approximation of the optimal result

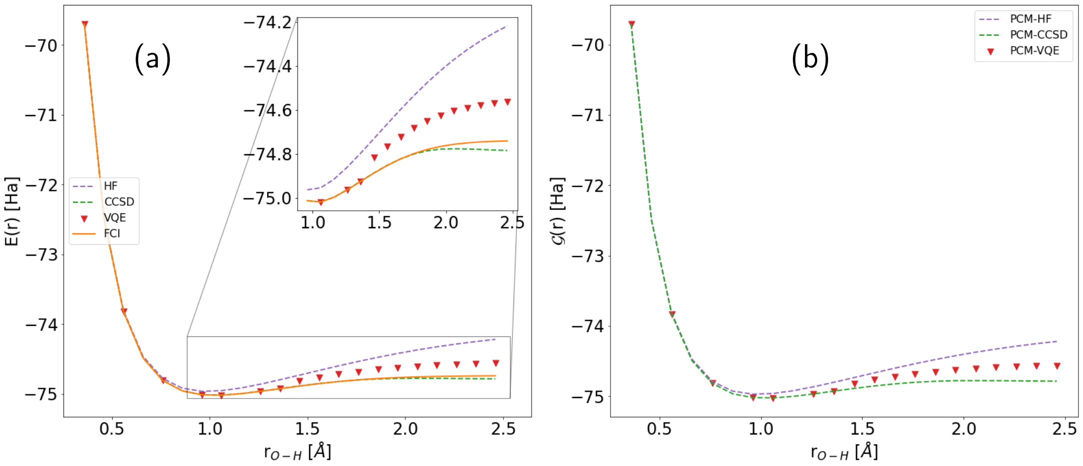


Figure S3: Double bond dissociation profile for the water molecule in gas phase (a) and in water (b). Reference values with classical methods are reported: FCI (solid orange line), CCSD (green dashed line) and HF (purple dashed line). Red triangles show the results obtained with the (PCM-)VQE method. All the calculations were performed with a STO-3G basis set.

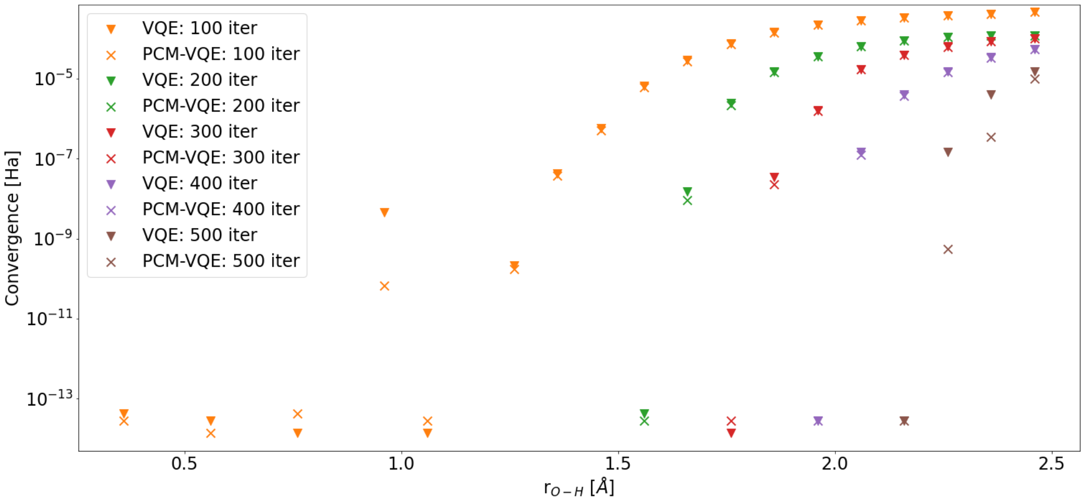


Figure S4: Convergence plot of the classical optimization procedure for the VQE (triangles) and PCM-VQE (crosses) as a function of the bond length. Color code: yellow (≤ 100 iter.); green (≤ 200 iter); red (≤ 300 iter.); brown (≤ 400 iter.); purple (≤ 500 iter.).

and therefore a greater number of iterations is required to achieve convergence. (ii) The adaptive ansatz here used includes excited configurations among the most relevant to capture electronic correlation at the equilibrium geometry whose electronic structure is increasingly different from the one at the bond dissociation limit.

Measurement budget allocation and statistical errors

In this section we give a brief account of the effects of finite sample size on the accuracy of the expectation value estimates performed during the VQE and PCM-VQE procedure. The goal of this section is two-fold: on one hand we want to provide a practical recipe to estimate the number of shots N needed to achieve a desired accuracy on solvation quantities (and, in general, on any expectation value calculated from the VQE wavefunction); on the other hand we want to rationalize the magnitude of the errors at fixed number of shots for different systems and solvation quantities.

The analysis we report explicitly refers to an error due to finite sampling for which it is exact to consider the variances of the expectation values. We point out that in presence of quantum computer noise it would be more appropriate to work explicitly with the mean squared error $\mathrm{MSE}(X)$ defined as

$$\mathrm{MSE}(X) = \mathrm{Var}(X) + \mathrm{Bias}(X)\,, \tag{S2}$$

where X is a random variable representing the expectation value, $\mathrm{Var}(X)$ is its variance and $\mathrm{Bias}(X)$ comprises the error due to the presence of the quantum computer noise.

We first consider the problem of estimating the number of optimal shots N needed to estimate the expectation value of an observable $\hat{O}$ within a desired accuracy ϵ. We follow Rubin *et al.*[88] who derived an optimal number of measurement for $\langle \hat{O} \rangle = \sum_\gamma h_\gamma P_\gamma$ given by

$$N = \frac{K}{\epsilon^2}\,, \tag{S3}$$

where the factor K is obtained taking into account the fluctuation of each independently measured Pauli word P_α needed to compute the desired expectation value:

$$K = \Big(\sum_B \sqrt{\sum_{\alpha,\beta \in B} h_\alpha h_\beta \mathrm{Covar}(P_\alpha, P_\beta)}\Big)^2 \tag{S4}$$

Here B is a label that runs over the different sets of non-commuting Pauli words and h_α, h_β are the corresponding matrix elements of the mapped operator $\hat{O}$.

We highlight that Eq. S4 is a direct consequence of error propagation theory and formalizes two important statements: (i) the number of shots increases as the number of independently measured Pauli words increases, (ii) the number of shots increases as the magnitude of the matrix elements of $\hat{O}$ increases.

In order to use Eq. S4 in practice we need to replace $\mathrm{Covar}(P_\alpha, P_\beta)$ with a reasonable estimate. A common choice[89, 90], since $\mathrm{Var}(P_\alpha) = 1 - \langle P_\alpha \rangle^2$, is to set all the variances to 1 and all the covariances to zero. It is possible to show that this corresponds to assuming that the Pauli strings are sampled from a uniform distribution [91].

With this assumption, K (and in turn the number of shots N) will depend only on the Frobenius norm of the operator $||O||$:

$$K \approx ||O||^2 = \sum_B \sum_{\alpha \in B} |h_\alpha|^2 \tag{S5}$$

Gonthier and coworkers[90] found numerically (for quantum chemistry ground state calculations on organic molecules at CCSD(T) level of accuracy) that putting covariances at zero overestimates K (and thus the required number of shots N) by a factor of 2, while assuming the variances to be all equal to 1 lead to a 20%-30% overestimation of K. Overall, their results show that using Eq. S3 together with the K estimate in Eq. S5 yields quite conservative (i.e., on the safe side) values of

N.

To use what reported here in practice, if a given precision ϵ is sought, one has first to estimate the factor K summing all the squared $\hat{O}$ matrix elements corresponding to non-commuting Pauli strings and then to divide by ϵ^2 to obtain the corresponding number of samples N to collect.

Now we shall consider the error estimated on the total energy and total free energy. This is also relevant for the error on solvation free energies, which are obtained as differences of total free energy in solution and total energy in gas-phase. Suppose for example that an error of $\epsilon = 0.1$ Ha on the total energy (or free energy) for water at the STO-3G level is desired. The first step is to calculate $K = ||H||^2$ for water from eq. S5, that turns out to be $\approx 6 \cdot 10^3$. Then, eq. S3 gives a number of samples $N \approx 6 \cdot 10^5$.

The presented discussion can also be used to numerically explore how error estimates depend on the specific calculation, via the Hamiltonian norm and the number of non-commuting sets of Pauli words at fixed number of shots, see Tab. S2.

Table S2: Frobenius norm of the Hamiltonian $\hat{H}$, number of non-commuting sets of Pauli words B and statical errors ϵ for the molecules studied in this work at the STO-3G basis set level. Estimates computed according to Eq. S5. Values obtained for a number of samples $N = 8192$. All quantities are given in atomic units.

	$\|\|H\|\|$	B	ϵ
H_3^+	1.12	10	0.0124
BeH_2	17.7	17	0.195
H_2O	78.5	23	0.867

As we can see from such Table, the error increases both as the number of non commuting Pauli terms and the Hamiltonian norm increases. This is in accordance with Eq. S3 and Eq. S5.

Turning now to the error on the polarization energy, that we identify in the main text as a robust solvation quantity also for noisy quantum computers, we note that the same procedure just described to estimate N to obtain a desired accuracy

ϵ can be applied directly to the operator V_σ. Moreover, by using Eq. S4 we can also rationalize the different accuracy observed in estimating the solvation free energy and the polarization energy. Ultimately this should be ascribed to the fact that the number of independently measured Pauli strings is much smaller for the polarization energy alone than for the overall solvation free energy since in one case we deal with a one-electron operator while in the latter case we have to include all the measurements coming from the two-electrons part. More specifically, in Tab. S3 we show estimated sampling errors for $\frac{1}{2}V_\sigma$ (the operator needed to compute the polarization energy). The comparison of the magnitude of the errors reported in this Table with those in Tab. S2 confirms our qualitative analysis. Interestingly, the error for the smallest molecule H_3^+ here is the largest in the Table. This is not surprising, since H_3^+ is the only charged molecule in the set, and as a consequence the V_σ norm is the largest.

Table S3: Frobenius norm of the operator $\frac{1}{2}V_\sigma$, number of non-commuting set of Pauli words B and statistical errors ϵ for the molecules studied in this work at the STO-3G basis set level. Estimates computed according to Eq. S5. Values obtained for a number of samples $N = 8192$. All quantities are given in atomic units.

	$\frac{1}{2}\|\|V_\sigma\|\|$	B	ϵ
H_3^+	0.712	3	0.00786
BeH_2	0.0735	5	0.00081
H_2O	0.104	6	0.00115

Algorithmic complexity: PCM classical overhead

The proposed PCM-VQE algorithm is an hybrid one, combining a classical portion with a quantum one. In view of realistic simulations on future quantum computers, it is important to assess that the extension to PCM does not hamper potential quantum advantage from the VQE approach. The goal of this section is not to provide a thorough analysis of the algorithmic cost, which depends on the particular flavour of VQE considered, but to highlight the computational overhead due to the inclusion of solvation effects within the PCM framework.

Since we are considering an hybrid algorithm we estimate the overall cost $C = C_Q + C_C$ where C_Q is the cost associated with operations run on the quantum computer and C_C the classical cost to compute the molecular integrals .

Concerning C_Q, we note that gas-phase VQE requires evaluation of 1- and 2-RDM to obtain the expectation value of the molecular Hamitonian $\langle \hat{H}_0 \rangle$ (see Figure 1 in the main text). The 1-RDM is the only information from the quantum computer that is also needed to calculate $\langle \hat{V}_\sigma \rangle$, i.e., the additional part with respect to $\langle \hat{H}_0 \rangle$ to calculate the total free energy. Since the 1-RDM is calculated for the standard VQE as well, there is no quantum overhead for PCM-VQE with respect to gas-phase VQE (i.e., $C_Q^{\text{gas}} = C_Q^{\text{PCM}}$).

Now we move our analysis to the classical cost of the algorithm. Here the inclusion of solvation effects may potentially affect the algorithmic cost. Particularly, the standard molecular Hamiltonian is usually built computing the molecular one- and two-electron integrals [55]. On the other hand, inclusion of solvation effects calls for (i) computing the solvent response at each iteration $\mathbf{q}$ and (ii) calculating the solute-solvent interaction term $\langle \hat{V}_\sigma \rangle$ at each iteration. (i) Formally, computing the response charges implies the inversion of the PCM response matrix whose computational cost is $\mathcal{O}(N_{tess}^3)$. In practice it is possible to achieve a linear scaling with N_{tess} for this step adopting techniques such as the Fast Multipole Method (FMM) and parallelization [92]. In turn, N_{tess} is scaling with the size of the molecular surface, that is linear with the number of atoms N_{atoms} in the worst case of linear molecules. This is a negligible scaling w.r.t. the following contribution. (ii) Concerning the calculation of $\langle \hat{V}_\sigma \rangle$, at each step we have to contract the 1-RDM with the $\mathbf{v}_{rs}$ array a step costing $\mathcal{O}(N_a^2 N_{tess}) = \mathcal{O}(N_a^2 N_{\text{atoms}})$ additional operations. Reasonably assuming that N_a is scaling linearly with N_{atoms}, this step has a $\mathcal{O}(N_a^3)$ scaling. This is still a better scaling than the cost of the contraction of the 2-RDM with the gas-phase bielectronic integrals (that dominates the gas-phase C_C), where a sum over four index is needed ($\mathcal{O}(N_a^4)$). As such the scaling of C_C^{PCM} will be at most equal to that of C_C^{gas}. Of course, any strategy to improve such $\mathcal{O}(\mathrm{N}_a^4)$ scaling (e.g., use of molecular symmetry, prescreening of the molecular integrals) can also be used to improve the scaling of the classical PCM term.

In conclusion, the overhead of the PCM extension to VQE is null for the quantum part, and does not worsen the scaling of the classical part with respect to gas-phase calculations; a potential quantum advantage of gas-phase VQE is therefore unhampered by the inclusion of the PCM solvent. The bottleneck remains to be the solution of the electronic structure problem.

Quantum circuits

References

(1) Castaldo, D.; Jahangiri, S.; Delgado, A.; Corni, S. *Journal of Chemical Theory and Computation* **2022**, *18*, 7457–7469.

(2) Weinan, E., *Principles of multiscale modeling*; Cambridge University Press: 2011.

(3) Mennucci, B.; Corni, S. *Nat. Rev. Chem.* **2019**, *3*, 315–330.

(4) Segatta, F.; Cupellini, L.; Garavelli, M.; Mennucci, B. *Chemi. Rev.* **2019**, *119*, 9361–9380.

(5) Franco, A. A.; Rucci, A.; Brandell, D.; Frayret, C.; Gaberscek, M.; Jankowski, P.; Johansson, P. *Chem. Rev.* **2019**, *119*, 4569–4627.

(6) Argüello-Luengo, J.; González-Tudela, A.; Shi, T.; Zoller, P.; Cirac, J. I. *Nature* **2019**, *574*, 215–218.

(7) Nielsen, M. A.; Chuang, I. Quantum computation and quantum information, 2010.

(8) Deutsch, I. H. *PRX Quantum* **2020**, *1*, 020101.

(9) Georgescu, I. M.; Ashhab, S.; Nori, F. *Reviews of Modern Physics* **2014**, *86*, 153.

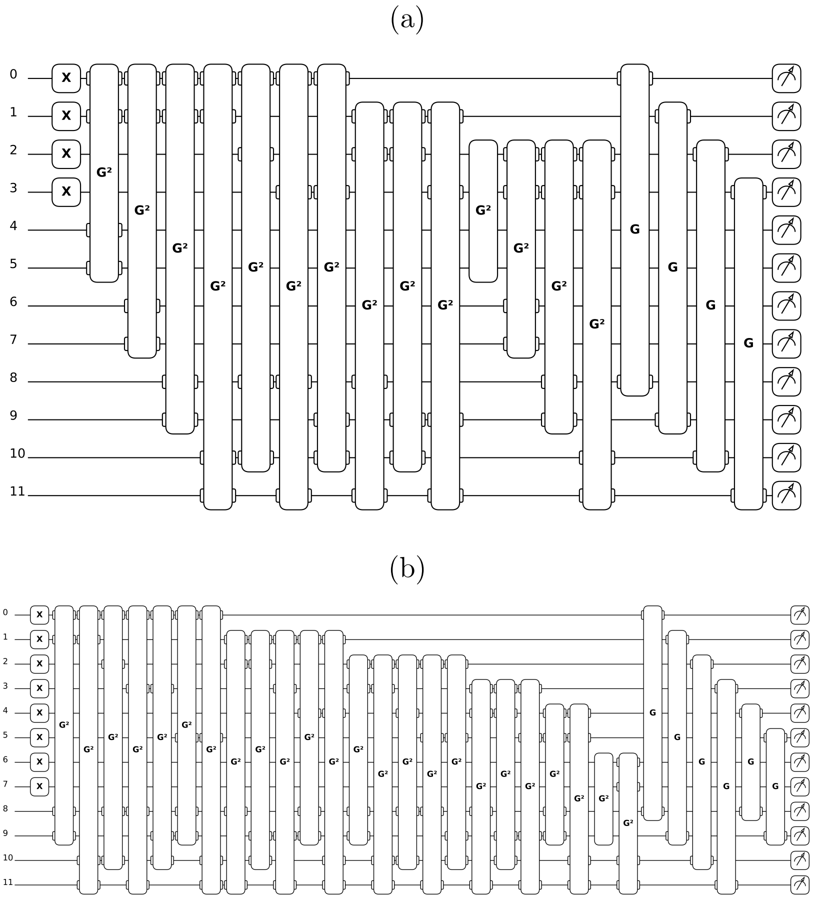


Figure S5: Quantum circuits used for the PCM-VQE simulations for the BeH_2 (a) and H_2O (b) molecules. Please note the difference w.r.t. the circuit reported in the main text for H_3^+. The space of multiple excitations is realised by applying more Givens rotations (18 and 30 for BeH_2 and H_2O respectively). We also note the presence of single excitation operators missing in the H_3^+ circuit as a result of the adaptive procedure that discards irrelevant excitations. Labels G and G^2 refer to single- and double-excitation unitaries implemented as rotations in the subspace spanned by two or four qubits. Figure obtained using the quantum circuit drawer function as implemented in PennyLane[60].

(10) Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; Patrick, C. *Nat. Rev. Phys.* **2021**, 1–20.

(11) Jaramillo, P.; Coutinho, K.; Cabral, B. J.; Canuto, S. *Chem. Phys. Lett.* **2011**, *516*, 250–253.

(12) Lovell, T.; Li, J.; Case, D. A.; Noodleman, L. *J. Biol. Inorg. Chem.* **2002**, *7*, 735–749.

(13) Röhrig, U. F.; Guidoni, L.; Rothlisberger, U. *ChemPhysChem* **2005**, *6*, 1836–1847.

(14) Prabhu, N.; Sharp, K. *Chem. Rev.* **2006**, *106*, 1616–1623.

(15) Timasheff, S. N. *COSB* **1992**, *2*, 35–39.

(16) Chen, X.; Regan, C. K.; Craig, S. L.; Krenske, E. H.; Houk, K.; Jorgensen, W. L.; Brauman, J. I. *J. Am. Chem. Soc.* **2009**, *131*, 16162–16170.

(17) Anslyn, E. V.; Dougherty, D. A., *Modern physical organic chemistry*; University science books: 2006.

(18) Sherwood, J.; Clark, J. H.; Fairlamb, I. J.; Slattery, J. M. *Green Chem.* **2019**, *21*, 2164–2213.

(19) Lucent, D.; Vishal, V.; Pande, V. S. *Proc. Natl. Acad. Sci.* **2007**, *104*, 10430–10434.

(20) Florová, P.; Sklenovsky, P.; Banaš, P.; Otyepka, M. *J. Chem. Theory Comput.* **2010**, *6*, 3569–3579.

(21) Christen, M.; Van Gunsteren, W. F. *J. Comp. Chem.* **2008**, *29*, 157–166.

(22) Caricato, M.; Lipparini, F.; Scalmani, G.; Cappelli, C.; Barone, V. *J. Chem. Theory Comput.* **2013**, *9*, 3035–3042.

(23) Chibani, S.; Jacquemin, D.; Laurent, A. D. *Comput. Theor. Chem.* **2014**, *1040*, 321–327.

(24) Cao, D.; Lu, G.-Q.; Wieckowski, A.; Wasileski, S. A.; Neurock, M. *J. Phys. Chem. B* **2005**, *109*, 11622–11633.

(25) Bryantsev, V. S.; Diallo, M. S.; Goddard Iii, W. A. *J. Phys. Chem. B* **2008**, *112*, 9709–9719.

(26) Roux, B.; Simonson, T. *Biophys. Chem.* **1999**, *78*, 1–20.

(27) Tomasi, J.; Mennucci, B.; Cammi, R. *Chemical reviews* **2005**, *105*, 2999–3094.

(28) Cances, E.; Mennucci, B.; Tomasi, J. *J. Chem. Phys.* **1997**, *107*, 3032–3041.

(29) Mennucci, B.; Cances, E.; Tomasi, J. *J. Phys. Chem. B* **1997**, *101*, 10506–10517.

(30) Cancès, E.; Mennucci, B. *J. Math. Chem.* **1998**, *23*, 309–326.

(31) Cammi, R. *J. Chem. Phys.* **2009**, *131*, 164104.

(32) Cammi, R.; Frediani, L.; Mennucci, B.; Tomasi, J.; Ruud, K.; Mikkelsen, K. V. *J. Chem. Phys.* **2002**, *117*, 13–26.

(33) Cammi, R.; Mennucci, B. *J. Chem. Phys.* **1999**, *110*, 9877–9886.

(34) Caricato, M. *J. Chem. Phys.* **2011**, *135*, 074113.

(35) Guido, C. A.; Rosa, M.; Cammi, R.; Corni, S. *J. Chem. Phys.* **2020**, *152*, 174114.

(36) Corni, S.; Tomasi, J. *J. Chem. Phys.* **2001**, *114*, 3739–3751.

(37) Rosa, M.; Gil, G.; Corni, S.; Cammi, R. *The Journal of Chemical Physics* **2019**, *151*, 194109.

(38) Alibakhshi, A.; Hartke, B. *Nat. Comm.* **2021**, *12*, 1–7.

(39) Peruzzo, A.; McClean, J.; Shadbolt, P.; Yung, M.-H.; Zhou, X.-Q.; Love, P. J.; Aspuru-Guzik, A.; O'brien, J. L. *Nature communications* **2014**, *5*, 4213.

(40) Kandala, A.; Mezzacapo, A.; Temme, K.; Takita, M.; Brink, M.; Chow, J. M.; Gambetta, J. M. *Nature* **2017**, *549*, 242–246.

(41) Tilly, J.; Chen, H.; Cao, S.; Picozzi, D.; Setia, K.; Li, Y.; Grant, E.; Wossnig, L.; Rungger, I.; Booth, G. H.; Tennyson, J. *Phys. Rep.* **2022**, *986*, 1–128.

(42) Di Paolo, A.; Barkoutsos, P. K.; Tavernelli, I.; Blais, A. *Phys. Rev. Res.* **2020**, *2*, 033364.

(43) Yoshioka, N.; Nakagawa, Y. O.; Ohnishi, Y.-y.; Mizukami, W. *arXiv preprint arXiv:2008.09492* **2020**.

(44) Pravatto, P.; Castaldo, D.; Gallina, F.; Fresch, B.; Corni, S.; Moro, G. J. *New J. Phys.* **2021**, *23*, 123045.

(45) Barkoutsos, P. K.; Gonthier, J. F.; Sokolov, I.; Moll, N.; Salis, G.; Fuhrer, A.; Ganzhorn, M.; Egger, D. J.; Troyer, M.; Mezzacapo, A. *Phys. Rev. A* **2018**, *98*, 022322.

(46) Liu, J.; Wan, L.; Li, Z.; Yang, J. *arXiv preprint arXiv:2008.02946* **2020**.

(47) Arrazola, J. M.; Di Matteo, O.; Quesada, N.; Jahangiri, S.; Delgado, A.; Killoran, N. *arXiv preprint arXiv:2106.13839* **2021**.

(48) Grimsley, H. R.; Economou, S. E.; Barnes, E.; Mayhall, N. J. *Nature Communications* **2019**, *10*, 3007.

(49) Tomasi, J.; Persico, M. *Chem. Rev.* **1994**, *94*, 2027–2094.

(50) Jackson, J. D. Classical electrodynamics, 1999.

(51) Mennucci, B.; Cammi, R., *Continuum solvation models in chemical physics: from theory to applications*; John Wiley & Sons: 2008.

(52) York, D. M.; Karplus, M. *J. Phys. Chem. A* **1999**, *103*, 11060–11079.

(53) Pascual-Ahuir, J. L.; Silla, E. *J. Comp. Chem.* **1990**, *11*, 1047–1060.

(54) Cammi, R.; Tomasi, J. *J. Comp. Chem.* **1995**, *16*, 1449–1458.

(55) Helgaker, T.; Jorgensen, P.; Olsen, J., *Molecular electronic-structure theory*; John Wiley & Sons: 2014.

(56) Taube, A. G.; Bartlett, R. J. *Int. J. Quantum Chem.* **2006**, *106*, 3393–3401.

(57) Cao, Y.; Romero, J.; Olson, J. P.; Degroote, M.; Johnson, P. D.; Kieferová, M.; Kivlichan, I. D.; Menke, T.; Peropadre, B.; Sawaya, N. P. *Chem. Rev.* **2019**, *119*, 10856–10915.

(58) https://github.com/davidecast/PCM-VQE Accessed by date: 10/11/2022.

(59) Smith, D. G.; Burns, L. A.; Simmonett, A. C.; Parrish, R. M.; Schieber, M. C.; Galvelis, R.; Kraus, P.; Kruse, H.; Di Remigio, R.; Alenaizan, A., et al. *The Journal of Chemical Physics* **2020**, *152*, 184108.

(60) Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Ahmed, S.; Ajith, V.; Alam, M. S.; Alonso-Linaje, G.; AkashNarayanan, B.; Asadi, A., et al. *arXiv preprint arXiv:1811.04968* **2018**.

(61) Di Remigio, R.; Steindal, A. H.; Mozgawa, K.; Weijo, V.; Cao, H.; Frediani, L. *Int. J. Quantum Chem.* **2019**, *119*, e25685.

(62) Nam, Y.; Chen, J.-S.; Pisenti, N. C.; Wright, K.; Delaney, C.; Maslov, D.; Brown, K. R.; Allen, S.; Amini, J. M.; Apisdorf, J.; Beck, K.; Blinov, A.; Chaplin, V.; Chmielewski, M.; Kim, J. *Npj Quantum Inf.* **2020**, *6*, 1–6.

(63) Ratini, L.; Capecci, C.; Benfenati, F.; Guidoni, L. *J. Chem. Theory Comput.* **2022**, *18*, 899–909.

(64) Bondi, A. v. *J. Phys. Chem.* **1964**, *68*, 441–451.

(65) Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Couch, G. S.; Greenblatt, D. M.; Meng, E. C.; Ferrin, T. E. *J. Comp. Chem.* **2004**, *25*, 1605–1612.

(66) Delgado, A.; Arrazola, J. M.; Jahangiri, S.; Niu, Z.; Izaac, J.; Roberts, C.; Killoran, N. *Phys. Rev. A* **2021**, *104*, 052402.

(67) Seeley, J. T.; Richard, M. J.; Love, P. J. *J. Chem. Phys.* **2012**, *137*, 224109.

(68) Ostaszewski, M.; Grant, E.; Benedetti, M. *Quantum* **2021**, *5*, 391.

(69) Aleksandrowicz, G.; Alexander, T.; Barkoutsos, P.; Bello, L.; Ben-Haim, Y.; Winston, E.; Wood, C.; Wood, S.; Wörner, S.; Akhalwaya, I. Y.; Zoufal, C. Qiskit: An Open-source Framework for Quantum Computing, version 0.7.2, 2019.

(70) Yanai, T.; Chan, G. K.-L. *J. Chem. Phys.* **2006**, *124*, 194106.

(71) Van Voorhis, T.; Head-Gordon, M. *J. Chem. Phys.* **2000**, *113*, 8873–8879.

(72) Evangelista, F. A.; Gauss, J. *J. Chem. Phys.* **2011**, *134*, 114102.

(73) Mizukami, W.; Mitarai, K.; Nakagawa, Y. O.; Yamamoto, T.; Yan, T.; Ohnishi, Y.-y. *Phys. Rev. Res.* **2020**, *2*, 033421.

(74) Olsen, J.; Jorgensen, P.; Koch, H.; Balkova, A.; Bartlett, R. J. *J. Chem. Phys.* **1996**, *104*, 8007–8015.

(75) O'Malley, P. J.; Babbush, R.; Kivlichan, I. D.; Romero, J.; McClean, J. R.; Barends, R.; Kelly, J.; Roushan, P.; Tranter, A.; Ding, N. *Phys. Rev. X* **2016**, *6*, 031007.

(76) Kandala, A.; Temme, K.; Córcoles, A. D.; Mezzacapo, A.; Chow, J. M.; Gambetta, J. M. *Nature* **2019**, *567*, 491–495.

(77) Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J. C.; Barends, R.; Boixo, S.; Broughton, M.; Buckley, B. B.; Buell, D. A. *Science* **2020**, *369*, 1084–1089.

(78) McWeeny, R. *Rev. Mod. Phys.* **1960**, *32*, 335.

(79) Lee, C.-K.; Hsieh, C.-Y.; Zhang, S.; Shi, L. *arXiv preprint arXiv:2106.10767* **2021**.

(80) Li, Y.; Benjamin, S. C. *Phys. Rev. X* **2017**, *7*, 021050.

(81) Bittel, L.; Kliesch, M. *Phys. Rev. Lett.* **2021**, *127*, 120502.

(82) Kempe, J.; Kitaev, A.; Regev, O. *SISC* **2006**, *35*, 1070–1097.

(83) Babbush, R.; McClean, J.; Wecker, D.; Aspuru-Guzik, A.; Wiebe, N. *Physical Review A* **2015**, *91*, 022311.

(84) Shen, Y.; Zhang, X.; Zhang, S.; Zhang, J.-N.; Yung, M.-H.; Kim, K. *Physical Review A* **2017**, *95*, 020501.

(85) Sim, S.; Johnson, P. D.; Aspuru-Guzik, A. *Advanced Quantum Technologies* **2019**, *2*, 1900070.

(86) Choquette, A.; Di Paolo, A.; Barkoutsos, P. K.; Sénéchal, D.; Tavernelli, I.; Blais, A. *Physical Review Research* **2021**, *3*, 023092.

(87) Holmes, Z.; Sharma, K.; Cerezo, M.; Coles, P. J. *PRX Quantum* **2022**, *3*, 010313.

(88) Rubin, N. C.; Babbush, R.; McClean, J. *New Journal of Physics* **2018**, *20*, 053020.

(89) Crawford, O.; van Straaten, B.; Wang, D.; Parks, T.; Campbell, E.; Brierley, S. *Quantum* **2021**, *5*, 385.

(90) Gonthier, J. F.; Radin, M. D.; Buda, C.; Doskocil, E. J.; Abuan, C. M.; Romero, J. *Physical Review Research* **2022**, *4*, 033154.

(91) Gokhale, P.; Angiuli, O.; Ding, Y.; Gui, K.; Tomesh, T.; Suchara, M.; Martonosi, M.; Chong, F. T. *arXiv preprint arXiv:1907.13623* **2019**.

(92) Scalmani, G.; Barone, V.; Kudin, K. N.; Pomelli, C. S.; Scuseria, G. E.; Frisch, M. J. *Theoretical Chemistry Accounts* **2004**, *111*, 90–100.

Chapter 4

A fully differentiable quantum phase estimation

Background and personal contribution

In this final chapter, I present work started during an internship in the algorithms team at Xanadu Quantum Technologies, led by Dr. Juan Miguel Arrazola.

Here, I consolidate all the aspects highlighted in the Introduction: (i) a focus on molecular properties beyond energies (as we develop a method to compute forces in molecules), (ii) emphasis on software development (as it represents the first example of automatically differentiable Quantum Phase Estimation), and (iii) attention to hardware by comparing the cost analysis with specifically devised fault-tolerant algorithms.

Once again, I present the results of this work in the form of a paper.

I developed the smoothing strategy and conducted the numerical and formal analysis. Throughout the code development, I received assistance from Dr. Soran Jahangiri (Xanadu). Finally, I drafted the paper in its original version with the help of Dr. Agostino Migliore who also assisted me during part of the derivation presented in Sec. 3 of this work.

Abstract

We develop a strategy to integrate the quantum phase estimation algorithm within a fully differentiable framework with a focus on quantum chemistry applications. This is accomplished generalizing a smooth estimator associated with the parent distribution induced by an exact input eigenstate to arbitrary initial states. We provide analytical expressions to characterize the statistics and algorithmic cost of this generalized circular estimator. Furthermore, we numerically prove that the estimation accuracy is retained when an arbitrary state is considered and that it exceeds the one of standard majority rule. Finally, we explicitly link this procedure to physically meaningful quantities for the molecular Hamiltonian and prove our implementation performing ground state (singlet and triplet) geometry optimization with simulations up to 19 qubits. This work paves the way for future strategies aimed to merge interference methods and quantum differentiable programming.

4.1 Introduction

Increasing demand for large-scale computational needs[1] in the upcoming decades will be fostered by advances of big data information processing[2] and sophisticated multiscale modeling[3, 4] required by cutting-edge research.

This pressing need is already fuelling a paradigm shift in how we think computation. As on-chip power dissipation of available semiconductor technologies hampers the development of standard processing architectures, it will be more likely to witness a slowdown of Moore's law. Along the same line, diversification of computing paradigms, both architecturally and algorithmically speaking, could be beneficial for a more sustainable growth in energy and material utilisation[5].

To some degree, such a transition has already taken place in the context of electronic structure theory. Indeed, very active research lines are improving performances and costs of simulating molecular Hamiltonians by developing implementations on Graphical Processing Units (GPUs)[6–8] and, more recently, Tensor Processing Units (TPUs)[9]. These contributions concern not only improvements w.r.t. accuracy and time-to-solution in obtaining molecular energies but also regard molecular properties. The latter, from a chemist point of view, can be even more important when it comes to simulating real-life experiments. Indeed, it is well known that many molecular properties can be expressed in terms of energy derivatives[10] and can be used for a variety of applications ranging from electronic to magnetic responses or determination of minimum energy paths to understand reaction mechanisms and drug-protein interaction for pharmaceuticals purposes and materials discovery[11, 12]. For these reasons, much effort has been devoted during the years to develop efficient methods to work out analytical expression for gradients and higher order derivatives in many fields of quantum chemistry[13–15].

With the same spirit, in this work we aim to contribute developing the first example of automatically differentiable Quantum Phase Estimation geared to compute energy derivatives for molecular Hamiltonians.

Automatic differentiation (AD), a clever combination of the chain rule and dynamic programming, is the workhorse of deep learning techniques[16] and in the last years has been proven very valuable also when applied to quantum chemistry[17–19] and, more broadly, scientific computation[20–22]. With the advent of NISQ devices, variational algorithms[23], in which the circuit is trained to optimize parameters defining the action of gates tasked with a specific goal, have spurred the development of extensions of automatic differentiation tailored for quantum computing. Much progress has been done to build AD techniques in quantum computing that account for programs with[24] or without[25, 26] any control flow. These works have led to several implementations of quantum differentiable frameworks[27, 28].

On the other hand, the application of these methods to fault-tolerant algorithms, such as the QPE, has not yet been explored. Despite several works aimed to reduce the cost of quantum simulations[29–31] or to extract physically relevant quantitites[32, 33] other than single point energies, very little has been done to study the presence of a parametric dependence within the phase estimation circuit. Along this line, in a recent work by O'brien *et al.*[34], the authors focused on calculating derivatives in a fault-tolerant setting and proposed different options: (i) finite difference approximations, (ii) overlap estimation via Hellman-Feynman theorem, and (iii) direct estimation of the gradient function via the application of the algorithm developed in Ref. [35]. We refer to Sec. 4.4.2 where we describe in more detail these other options.

Here, we specifically focus on the aspect of differentiating through the Quantum Phase Estimation circuit as described in the standard textbook by Nielsen and Chuang[36]. We note the importance of a smooth estimator for the eigenvalue and emphasize the necessity of a formal analysis to quantify the cost associated with incorporating such an additional feature.

This study is organized as follows. Section 4.2 is devoted to recall the Quantum Phase Estimation (QPE) algorithm and sketch the proposed quantum differentiable pipeline. In Section 4.3 we discuss the main contribution of this work: building on the original paper of Cruz et al. [37] we develop a smooth estimator for the QPE to be used with arbitrary input states of the system register. This allows to seamlessly integrate the QPE algorithm in a fully differentiable workflow. We discuss its properties and derive error bounds to estimate how the smoothing procedure affects the computational cost of eigenvalue evaluation and of its derivatives (see Sec. 4.4). As previously mentioned, Sec. 4.4.2 provides a summary of the main options[34] devised to compute energy derivatives in a Fault-Tolerant setting. Further, in Sec. 4.5 we apply this method to two examples: H_3^+ and CH_2O performing geometry optimization for the ground singlet and first triplet state.

In this work, we develop a technique for post-processing measurements out of the quantum phase estimation circuit that allows it to be integrated into an automatically differentiable software. These findings may be relevant on their own giving insights into how one can reduce the cost of QPE, which although asymptotically efficient, is still impractical for demonstrating a clear and scalable quantum advantage[38–41]. The results we show indicate that to obtain a differentiable routine one must pay an additional computational cost. In the conclusions, we indicate potential roadblocks and possible ways around this issue.

4.2 Preliminaries and notation

To begin with, we revisit the fundamental components of the Quantum Phase Estimation algorithm in its standard formulation[42]. We will not go into the details of the different simulation techniques used within the QPE circuit as they go beyond the purposes of this study. Additionally, the second crucial element of this work involves quantum differentiable programming. Consequently, we will underscore the main features needed for our development and elucidate why a smooth estimator of the eigenvalues is imperative in this particular context.

4.2.1 Quantum Phase Estimation

The QPE algorithm aims to find the eigenvalues of an arbitrary unitary U with precision ϵ. In this work we will focus on the problem of quantum simulation, i.e. for us the eigenvalues of the unitary will be directly related to the energies $\{E_i \,|\, i = 1, 2, \dots R\}$ of a quantum system encoded in a qubit register (referred to as system register) of n qubits. This is done choosing the unitary U in such a way that it shares the same eigenstates $\{|i\rangle\}$ with the Hamiltonian H of the system and that their eigenvalues are related by an invertible function $\phi_i = f(E_i)$ as shown in Refs.[36, 42].

The QPE circuit (Fig. 4.1) is implemented using two qubit registers, (i) the system register of n qubits encoding the wavefunction of the simulated system and (ii) the readout register of t qubits which are measured at the end of the circuit execution to estimate the phases ϕ_i.

We can notice from the circuit that the algorithm consists of four main steps. First the system register is prepared into the state $|\psi\rangle = \sum_u c_u |u\rangle$ and the readout register into the state $H^{\otimes^t}|0\rangle = \frac{1}{2^t} \bigotimes_{k=1}^t (|0_k\rangle + |1_k\rangle)$. After this stage the QPU is in the state

$$|\Psi\rangle = \frac{1}{2^t} \sum_u c_u \bigotimes_{k=1}^{t} (|0_k\rangle + |1_k\rangle)|u\rangle \,. \tag{4.1}$$

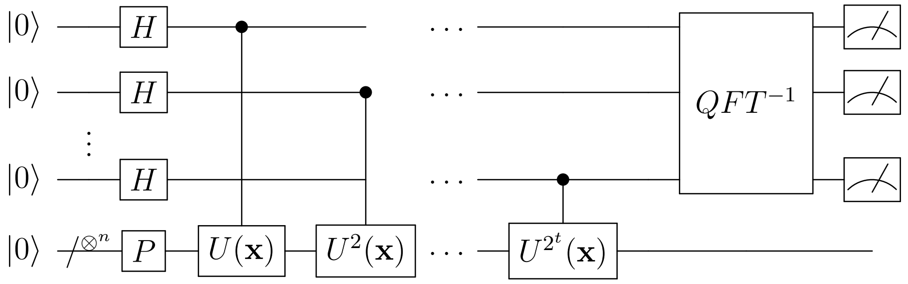


Figure 4.1: Schematic representation of of the Quantum Phase Estimation circuit. The readout register comprises t qubits each initialized in the state $|0\rangle$ while the system register is specified by n qubits to which is apply a generic state preparation routine P setting up the state $\psi = \sum_u c_u|u\rangle$. We highlighted that the unitary U can be dependent on the set of parameters $\mathbf{x}$. In this work we focus on the task of differentiating the eigenvalues estimated from this circuit w.r.t. $\mathbf{x}$.

We recall that here the label u runs over the eigenstates of the Hamiltonian and t is the number of qubits in the readout register.

Then the unknown phases are encoded into the quantum circuit performing a sequence of controlled evolutions of the system register conditioned on the states of the readout register. This brings the QPU into the state

$$|\Psi\rangle = \frac{1}{2^t}\sum_u c_u \bigotimes_{k=1}^{t}(|0_k\rangle + e^{i2\pi 2^{k-1}\phi_u}|1_k\rangle)|u\rangle\,. \tag{4.2}$$

This state is achieved as the application of a unitary evolution on the system register gives $U|\psi\rangle = \sum_u c_u e^{i2\pi\phi_u}|u\rangle$.

After all the phases are encoded into the readout register their values are transferred from the amplitudes to the computational basis spanned by the readout register applying an inverse Quantum Fourier Transform. This step is crucial for the whole algorithm. Here the digitalization of the encoded phase occurs; we can think of the computational basis spanned by the readout register as a grid of 2^t points evenly spaced that we can use to represent discretely the Fourier transform of the amplitudes encoded into the state of Eq. 4.2. Now the state of the QC reads:

$$|\Psi\rangle = \sum_u c_u|u\rangle \sum_{b_{1,u}=0}^{1} \sum_{b_{2,u}=0}^{1} \cdots \sum_{b_{t,u}=0}^{1} (\sum_k^{2^t} \frac{e^{i2\pi(\phi_u - 0.\phi_u)(k-1)}}{2^t})|b_{1,u}\rangle|b_{2,u}\rangle \dots |b_{t,u}\rangle \tag{4.3}$$

Where the binary fraction $0.\phi_u = 0.b_{1,u}b_{2,u} \dots b_{t,u}$ is a shorthand for $0.b_{1,u}b_{2,u} \dots b_{t,u} = \sum_k^t \frac{b_{k,u}}{2^k}$.

Finally, measuring the state of Eq. 4.3 on the computational basis we get a parent distribution given by $P(0.\phi) = \sum_u P_{\phi_u}(0.\phi)$ where each P_{ϕ_u} is the probability of measuring the bitstring $0.\phi$ induced by the eigenphase ϕ_u:

$$P_{\phi_u}(0.\phi) = \begin{cases} 1 & \Delta\phi = \phi_u - 0.\phi = 0 \\ \frac{g(0.\phi)}{2^{2t}} = \frac{1}{2^{2t}} \frac{sin^2(2^t\pi(\phi_u - 0.\phi))}{sin^2(\pi(\phi_u - 0.\phi))} & \Delta\phi = \phi_u - 0.\phi \neq 0 \end{cases} \tag{4.4}$$

We will analyze in detail the shape of the probability distribution in Sec. 4.3 as this will be the starting point to generalize the circular estimator developed in Ref.[37]. At this stage we only highlight that the result of this type of functional form is to have a superposition of peaks that is the narrower the more dense the grid with which we operate the discretization (e.g. the more readout qubits). Each of these peaks corresponds to an eigenstate $|u\rangle$ with non-zero overlap with the initial state $|\psi\rangle$ and is centered on the string that best approximates the phase ϕ_u. As a consequence the usual strategy for estimating the eigenvalue of the unitary U is to prepare a state with high overlap and estimate the corresponding phase ϕ_u as $\phi_u \approx \text{argmax}(P(0.\phi)) = 0.\phi_u$. This estimator is often referred to as majority rule estimate (in the following, ME). Its accuracy is bounded by the number of discretizing points used to represent $P(0.\phi)$, particularly the associated error of the estimate is $|\Delta\phi| = |\phi_u - 0.\phi_u| \leq \frac{1}{2^{t+1}}$.

We now move to discussing quantum gradients and differentiability as they will give us the reason to go beyond the majority rule estimator in Sec. 4.3.

4.2.2 Differentiating through the Quantum Phase Estimation circuit

Differentiable programming is a programming paradigm that leverages the chain rule to get the derivative of a computer program with respect to an input parameter. Underlying this computational paradigm is automatic differentiation, a method to compute exact numerical derivatives which underlies many machine learning techniques such as the renowned backpropagation algorithm[43]. The basic idea behind automatic differentiation is that the computer, regardless of the complexity of the function to evaluate, executes very simple operation that can be all tracked down and stored in memory. Such a sequence of operations is usually called evaluation trace and can be represented on a computational graph, see Fig. 4.2.

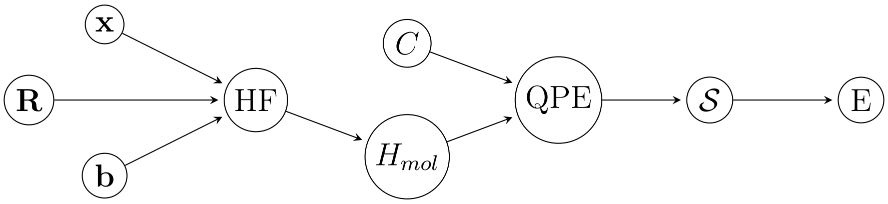


Figure 4.2: Coarse representation of the hybrid computational graph traversed to compute the molecular energy (and its derivatives) with the QPE algorithm. Each node represents deeply involved functions and is, in pratice wrote down much more accurately. As an example, we can consider the quantum circuit of Fig.4.1 as specific representation of the QPE node. Please notice as, in this representation, the initial state C fed into the QPE circuit is explicitly an independent (differentiable) variable. For an interpretation of this AD pipeline in terms of electronic structure theory quantities, please refer to Appendix 4.6. This work is mainly devoted to the development of $\mathcal{S}$ such that it provides a smooth estimator of the molecular energy E.

The example depicted in figure illustrates a coarse-grained representation of the computational graph, corresponding to the calculation of molecular electronic states' energies. In Sec. 4.5, we present outcomes related to derivatives concerning atomic positions ($\mathbf{R}$). We have distinguished the input variables to explicitly convey that other possibilities are natural extensions of this work. For instance, derivatives with respect to parameters of the basis set ($\mathbf{b}$) or external perturbations ($\mathbf{x}$) are potential avenues for exploration. The latter, in particular, will be explored in greater detail in a future work, given their capacity to enable the calculation of molecular properties relevant to spectroscopy.

A differentiable program is able to compute the derivative of the evaluation trace by traversing the computational graph through all its paths and gathering the partial derivative associated with each node. Specifically, during this operation, referred to as Jacobian accumulation, the program calculates the derivative of the function using the chain rule, which is composed of the derivatives of all the nodes (i.e., the fundamental operations executed by the computer to evaluate a function f) in the computational graph. It is crucial to emphasize for our purposes that, in order to maintain differentiability (and thus, the viability of the algorithm) of the evaluation trace, all individual steps must themselves be differentiable.

We refer to quantum differentiable programming when computational graphs include expressions evaluated on both classical and quantum hardwares. Here, we developed a fully differentiable QPE algorithm and applied it to quantum chemistry, the latter being one of the most promising fields of applications for this algorithm. From this standpoint, the PennyLane package represents a natural choice offering end-to-end differentiability thanks to the differentiable Hartree-Fock solver integrated within the library[44] and the possibility to include quantum functions into the computational graph.

Generally speaking, quantum functions take as input instructions for the quantum circuit and output results of its execution (i.e., measurements). For our particular purpose we can explicitly write the QPE parent distribution as a result of a parameterized expectation value:

$$QPE(\mathbf{x}) = \langle 0^{\otimes t}|\langle\psi|\mathrm{QFT}^{\dagger}\mathcal{U}_t^{\dagger}(\mathbf{x})\mathrm{QFT}\sigma_z^{\otimes t}\mathrm{QFT}^{\dagger}\mathcal{U}_t(\mathbf{x})\mathrm{QFT}|\psi\rangle|0^{\otimes t}\rangle \tag{4.5}$$

Where $\mathcal{U}_t$ is the sequence of controlled evolutions used to encode the phase on the readout register $\mathcal{U}_t(\mathbf{x}) = \prod_k^t(|0_t\rangle\langle 0_t| \otimes \mathbf{1}^{\otimes^n} + |1_t\rangle\langle 1_t| \otimes U_k(\mathbf{x}))$, and the initial layer of Hadamard gates has been identified as a QFT on the readout register in the product state $|0\rangle^{\otimes^t}$.

Repeated calls to the QPE($\mathbf{x}$) function allow to reconstruct the parent distribution P($0.\phi$) (Eq. 4.4) which is now parameterically dependent on $\mathbf{x}$ here used as arbitrary independent variables.

To perform Jacobian accumulation of hybrid computational graphs it is necessary to develop techniques to compute gradients of quantum functions. Extensive literature focuses on evaluating derivatives of expectation values of parameterized quantum circuits[45–47]. Most relevant for our purposes are the works of Refs.[48–52] as they provide general parameter-shift rules to compute derivatives of gates like $U(\theta) = e^{i\theta H}$ with arbitrary H. In Sec. 4.4 we will make use of their results to evaluate the number of calls needed to compute gradients differentiating through a QPE circuit.

Finally, now that we have discussed the basics of differentiable programming we can notice that if one wants to build up a program that estimates the phase out of the parent distribution induced by a QPE circuit also the classical postprocessing that evaluates an estimator for the phase out of the $P(0.\phi)$ must be differentiable. By definition the majority rule estimator discussed in the previous section does not fulfil this requirement. Therefore, in the next section we will focus on the main result of this work: the development of a smooth estimator for QPE.

As anticipated in the introduction, we propose an extension of the work of Cruz et al. [37] for an estimator to which we refer as Generalized Circular Estimator (GCE), whose features will be analyzed in the next section.

4.3 Generalized circular estimator

In this section we present a general smooth estimator for the QPE algorithm, characterize its statistical properties and discuss its application within a differentiable programming framework meant for evaluating energy derivatives of the molecular

Hamiltonian.

Our idea is to combine the Softmax-Gumbel approach [53], used in machine learning to efficiently differentiate through categorical distributions, and the idea of Cruz et al. [37] of estimating the phase from a circular average[54] of the parent distribution (Eq. 4.4). We will start recalling the results of Ref. [37] which are valid when the input state corresponds to an eigenstate $|u\rangle$ of the unitary U.

The authors propose to use as estimator of the energy the mean phase direction μ of the first trigonometric moment of the QPE final measurement distribution. Particularly, the first trigonometric moment θ reads

$$\theta = \sum_{0.\phi} P_{\phi_u}(0.\phi) e^{i2\pi 0.\phi} = |\theta| e^{i2\pi\mu} \tag{4.6}$$

Where $0.\phi$ is a label for the binary fraction, ranging from 0 to 1, represented by each bitstring of the computational basis spanned by the readout register. We recall that P_{ϕ_u} is defined by Eq. 4.4.

As the original authors point out a nice way to picture the meaning of θ it is that is equivalent to taking a vector sum in the complex plane where each vector has a direction specified by $e^{i2\pi 0.\phi}$ and modulus given by $P_{\phi_u}(0.\phi)$. This representation explains another important feature of θ (that we will recover in our generalization), i.e. its accuracy exceeds that of the majority rule estimator. This stems from the fact that each string over- or under-represents the true phase ϕ_u because of the discretization in binary fractions, then averaging among several strings weighted by their probabilities allows to get a more accurate estimate due to error cancellation effects.

Subsequently, the mean phase direction μ of θ is given by:

$$\mu = \arg(\theta) \tag{4.7}$$

Plugging into the definition of θ Eq. 4.4, the authors get an explicit expression for μ, θ and $|\theta|$.

$$\theta = \frac{2^t-1}{2^t}e^{i2\pi\phi_u} + \frac{1}{2^t}e^{-i(2^t-1)2\pi\phi_u} \tag{4.8}$$

$$\mu = \frac{1}{2\pi}\mathrm{arctan}\left(\frac{(2^t-1)\mathrm{sin}(2\pi\phi_u) - \mathrm{sin}((2^t-1)2\pi\phi_u)}{(2^t-1)\mathrm{cos}(2\pi\phi_u) + \mathrm{cos}((2^t-1)2\pi\phi_u)}\right) \tag{4.9}$$

$$|\theta| = \sqrt{4^{-t}(4^t - 2^{t+1} + 2 + 2(2^t-1)\mathrm{cos}(2^{t+1}\pi\phi))} \tag{4.10}$$

Both from Eq. 4.8 and Eq. 4.9 we can see that for large values of t the estimator μ approaches the true value of ϕ_u. Further, looking at Eq. 4.10 we can see that in the same regime (i.e., large t) the first trigonometric moment's modulus approaches $|\theta| = 1$. Further, this value is approached exponentially fast. As we shall see later on, from an algorithmic point of view it means that the additional sampling cost presented by the definition of this estimator becomes negligible as we increase the number of readout qubits.

All the statements done so far about the circular average estimator hold if the input state of our QPE algorithm is an eigenstate of our Hamiltonian. We now extend them to an arbitrary input state.

First of all it is important to understand why we can not use directly Eq. 4.8 when $|\psi\rangle$ is not an eigenstate of U. As recalled in Sec. 4.2.1, the probabilities $P(0.\phi)$ are due to the contributions of all the eigenstates with non-zero overlap with the initial state.

Thus if we applied straightforwardly the definition of first trigonometric moment on this new distribution we would get:

$$\theta' = \sum_u |c_u|^2 \sum_{0.\phi} P_{\phi_u}(0.\phi)e^{i2\pi 0.\phi} = \sum_u |c_u|^2\theta_u = \rho' e^{i2\pi\mu'} \tag{4.11}$$

The issue with this expression is that now we are taking a circular average which accounts for contributions due to different eigenstates. Interestingly, we could show that the last expression leads to an estimator for $\langle\psi|H|\psi\rangle$ as $\langle H\rangle \hat{=} \mathrm{arg}(\theta')$. We will not go through this as goes beyond the purpose of this study. Nonetheless we point out that comparing the cost of this estimator with existing algorithms[33] for evaluating expectation values in a fault-tolerant setting could be a matter of interest per se.

Eq. 4.11 is important not only because it provides us with an estimator for an expectation value of a generic $|\psi\rangle$ state, but also because it gives us a clue as to what needs to be done in order to generalize Eq. 4.6. Indeed, if we were able to filter out from the parent distribution all the spurious contributions due to the presence of the states $|u'\rangle \neq |u\rangle$ we could use again the original definition of θ. The strategy we adopt here is as follows: (i) we develop a method to obtain a new distribution analogous to the initial distribution except for the contribution of all the $|u'\rangle$s not of interest, (ii) we apply the original definition of θ to this new distribution to obtain a Generalized Circular Estimator (GCE) $\tilde{\theta}$. This steps are summarized in Fig. 4.3.

To accomplish the first step we assume that our initial state has the highest overlap $|c_u|^2$ with the eigenstate of interest $|u\rangle$, this is a standard assumption as we are usually interested in minimizing the cost which for the majority rule estimator scales as $\mathcal{O}(\frac{1}{|c_u|^2})$.

Then we can identify the region of the parent distribution induced by $|u\rangle$ using a tempered softmax function S_T. This function amplifies differences in the original distribution depending on a parameter T (often referred to as temperature due to the equivalence of this function to the Boltzmann distribution) according to the following rule:

$$P'(0.\phi) = S_T(P(0.\phi)) = \frac{e^{-\frac{P(0.\phi)}{T}}}{\sum_{0.\phi} e^{-\frac{P(0.\phi)}{T}}} \tag{4.12}$$

The shape of $P'(0.\phi)$ depends on the chosen temperature T and on the initial state $|\psi\rangle$ in the very same fashion of the Boltzmann distribution in statistical mechanics[55]. Particularly, when the temperature is very low we tend to get a one-hot distribution[56] peaked on the most sampled bitstring of the distribution, i.e. $0.\phi_u$. Subsequently, we can extract the $0.\phi_u$ value weight-averaging the

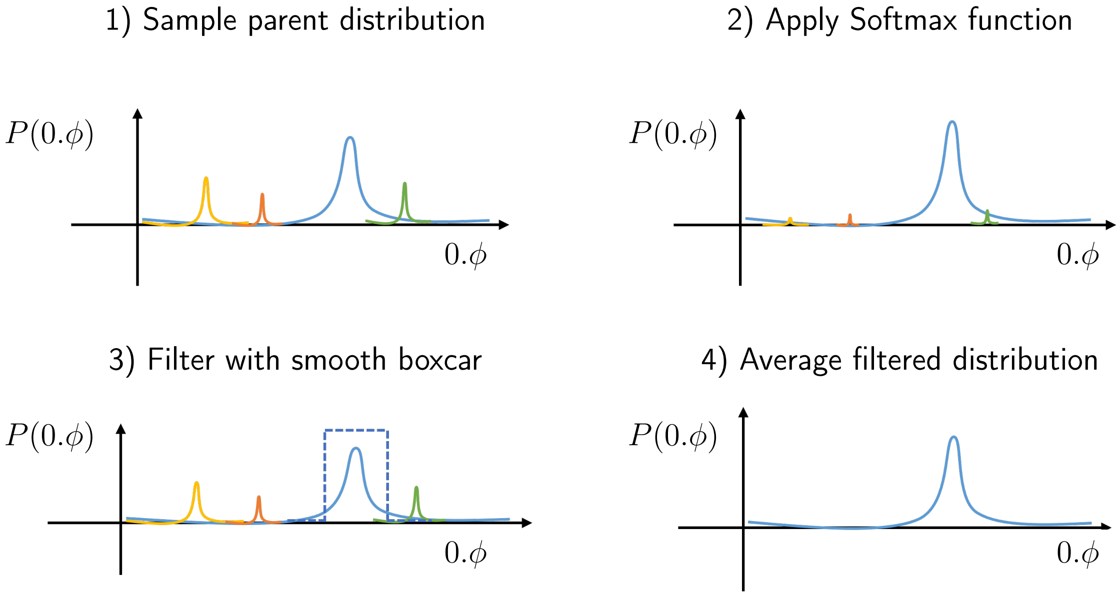


Figure 4.3: Schematic representation of the postprocessing applied to compute the GCE. Lineshapes of the distributions do not reproduce quantitatively Eq. 4.4. (1) different colors indicate bitstrings samplings induced by different eigenstates, please notice that, depending on the associated eigenvalues the degree of overlap of two peaks may differ. (2) The effect of the tempered Softmax (Eq. 4.12) is to spot the position of the most sampled peak by suppressing all the other contributions. (3) Based on the distribution in (2) we are able to build a smooth boxcar around the peak of interest. (4) Finally, we can estimate the GCE making a convolution between the parent distribution and the boxcar filter.

$P'(0.\phi)$ distribution with the binary fraction represented on the computational basis, $0.\phi_u = \sum_{0.\phi} 0.\phi \, P'(0.\phi)$.

Once this is done, we want to identify a subset of bitstrings around $0.\phi_u$ to be used for circular averaging exactly as in the starting case of Eq. 4.6. The reason why we assume we can decouple the contributions of different eigenvalues in populating the same string lies in the definition of $P_{\phi_u}(0.\phi)$. As we can see in Fig. 4.4a, where we plot Eq. 4.4 for different values of t, $P_{0.\phi_u}$ is a function that shrinks exponentially with the number of readout qubits having a central lobe Full Width at Half Maximum (g_{FWHM}) dependency on the readout qubits given by[57]

$$g_{\mathrm{FWHM}} = \frac{1}{2^t} \,. \tag{4.13}$$

This feature is essential for our filtering procedure as it allows to identify a subset of binary strings $0.\phi$ for each eigenstate $|u\rangle$ where the following condition holds:

$$P(0.\phi) \approx |c_u|^2 P_{\phi_u}(0.\phi) \tag{4.14}$$

The meaning of this statement is that when two eigenphases are far apart then the contribution of one of the two to the sampling probability of the same bitstring will be negligible. In other words, given a sufficient number of readout qubits we can always find a bitstring $0.\phi$ for which $P(0.\phi) - \frac{|c_u|^2}{2^{2t}} P_{\phi_u}(0.\phi) < \gamma$, where γ is an arbitrary small number.

These arguments allow us to propose an extension of θ to arbitrary initial input states, which we label $\tilde{\theta}$, as:

$$\tilde{\theta} = |c_u|^2 \sum_{0.\phi \in \mathcal{G}} P_{\phi_u}(0.\phi) e^{i2\pi 0.\phi} \tag{4.15}$$

Here we have introduce the set of relevant bitstrings $\mathcal{G} = \{0.\phi_u - h, 0.\phi_u + h\}$. Where h is a width parameter defining how many bitstrings to include in the circular average.

In practice, when running the QPE circuit we can only access measured probabilities, to get a distribution similar to Eq.4.15 we apply a smooth boxcar filter centered at $0.\phi_u$:

$$\tilde{\theta} = \sum_{0.\phi} B_k^h(0.\phi) P(0.\phi) e^{i2\pi 0.\phi} \tag{4.16}$$

Where $B_k^h(0.\phi)$ is given by:

$$B_k^h(0.\phi) = \frac{1}{2}\Big\{\tanh[k\,(0.\phi - (\phi_u - h)] - \tanh[k\,(0.\phi - (0.\phi_u + h)]\Big\} \tag{4.17}$$

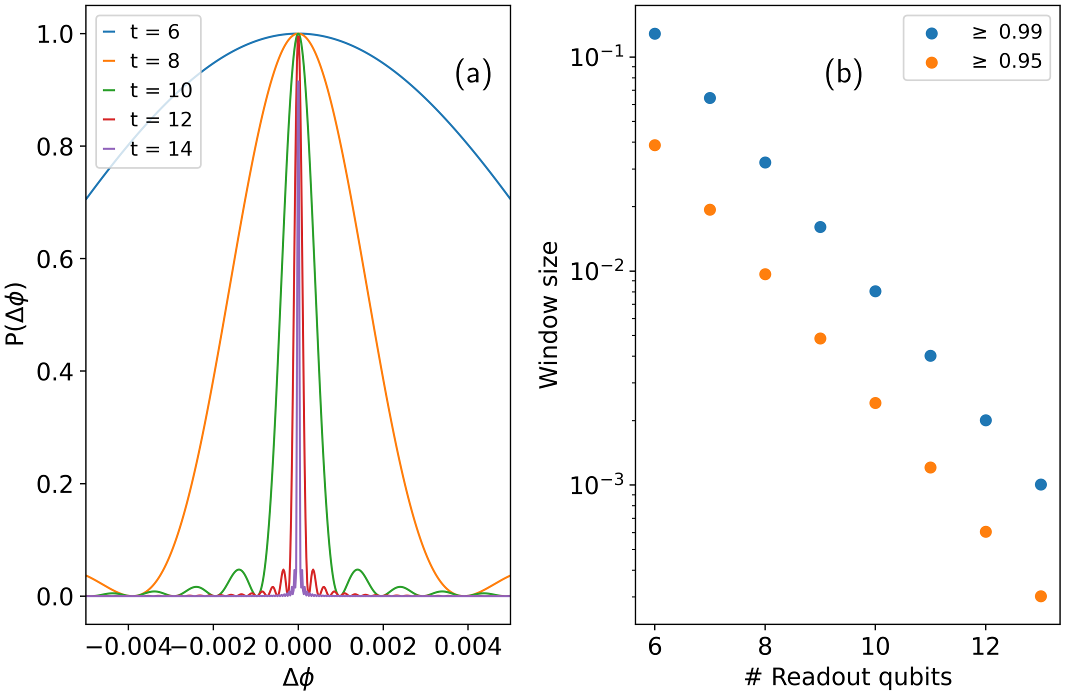


Figure 4.4: a) Plot of the quantum phase estimation distribution kernel as per Eq. 4.4. As we can observe, the main lobe exponentially shrinks as the number of readout qubits increases. b) Size of the window filter as a function of the number of readout qubits. The values are obtained such that the area integrated by the peak underlying the filter is 95% (orange dots) and 99% (blue dots) of the integral over the entire domain, respectively.

With k being a tunable parameter which accounts for the steepness of the window function. As we can see (Fig.4.5), an appropriate choice of the value of k is necessary to keep the assumptions underlying Eq. 4.15 valid.

Finally, we obtain the following expressions for the first trigonometric moment $\tilde{\theta}$ and its mean phase direction $\tilde{\mu}$ (which we recall is our estimator for the target eigenvalue):

$$\tilde{\theta} = |c_u|^2 \sum_{0.\phi \in \mathcal{G}} P_{\phi_u}(0.\phi) e^{i2\pi 0.\phi} \approx \sum_{0.\phi} B^h_k(0.\phi) P(0.\phi) e^{i2\pi 0.\phi} = |c_u|^2 \rho e^{i2\pi\tilde{\mu}} \qquad (4.18)$$

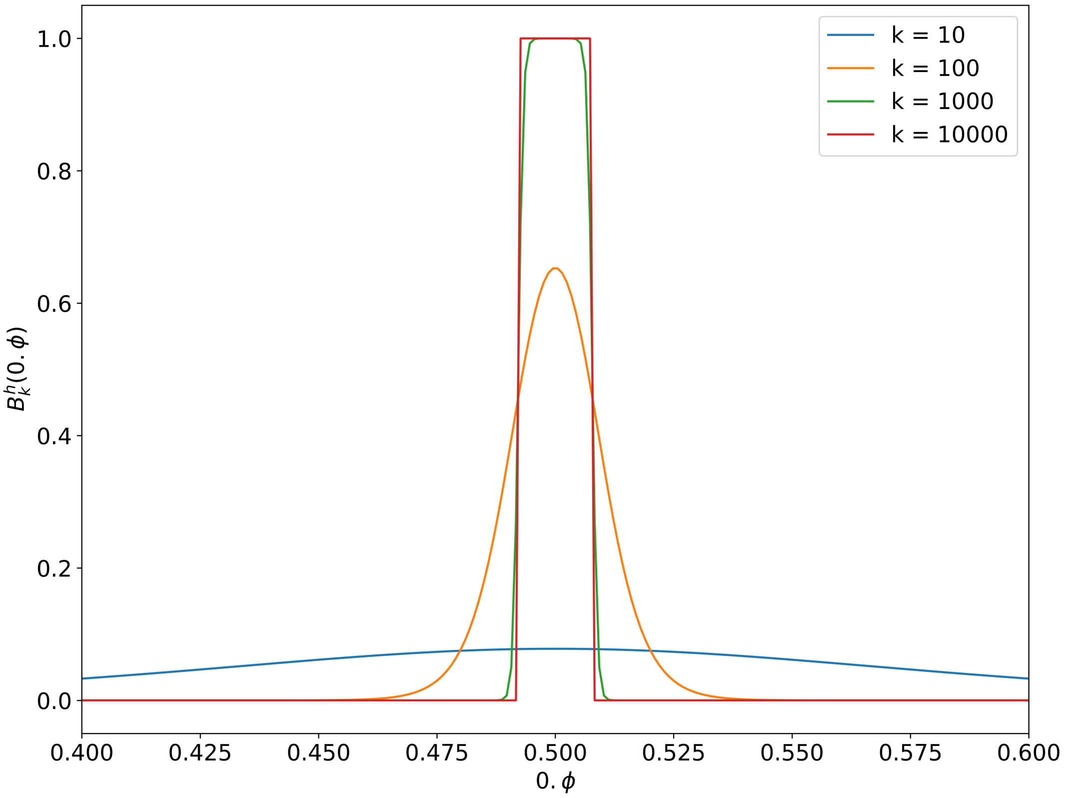


Figure 4.5: Smooth boxcar filter for different values of k as per Eq. 4.17. The values reported are obtained for a t=10 qubits readout grid with a window width of $|\mathcal{G}| = 16$ bitstrings corresponding to $h \approx 0.015$.

$$\tilde{\mu} = \arg(\tilde{\theta}) = \frac{1}{2\pi|\tilde{\theta}|}\arctan(\frac{Im(\tilde{\theta})}{Re(\tilde{\theta})}) \tag{4.19}$$

At this stage we need to understand how to choose a possible value for h. This should be done balancing two factors: (i) including the greatest number of bitstring possible, (ii) avoiding spurious contributions from other eigenstates' distributions. A possible heuristic to obtain a reasonable value for h is to consider a window size such that the ratio $\frac{I_h(g)}{I(g)}$ between the integrated area of $P(\Delta\phi)$ in the window spanned by h and the integral over the complete domain $I(g)$ is at least 0.95, see Fig. 4.4b. Such a strategy may prioritize the first constraint over the exclusion of other eigenstates' distribution from the circular average. To better understand this a formal analysis is needed.

In the next section we will provide analytical expressions for $\tilde{\mu}$ which will allow to provide a rationale of this heuristics and to demonstrate the accuracy of the generalized circular estimator. Finally, we will also be able to quantitatively account for the sampling cost of the measurement distribution.

4.3.1 Statistical analysis of the GCE

In this section we derive analytic expressions for the GCE to understand its numerical and statistical properties.

The first step is to recognize that by substituting Eq. 4.4 in the definition of θ, in the limit of large number of readout qubits the grid resolution with which we estimate the phase ϕ becomes infinitesimal and we can write

$$\tilde{\theta} = \frac{|c_u|^2}{2^t} \int_{0.\phi_u - h}^{0.\phi_u + h} \frac{sin^2(2^t \pi(\phi - x))}{sin^2(\pi(\phi - x))} e^{i2\pi x} dx \tag{4.20}$$

Where we have dropped the label u to indicate that the phase ϕ_u refers to the eigenstate $|u\rangle$ and changed the variable $0.\phi \rightarrow x$ as it is now continous. We also remark that $0 \leq h \leq \frac{1}{2}$.

It is important to notice that w.r.t. the original formulation of Cruz et al., here we will obtain only the first term of Eq. 4.8 as we do not perform explicit summation but integrate over a continous variable. This choice does not affect the accuracy of our analysis provided we are considering approximatively $t \geq 10$ qubits.

After some algebra we rewrite Eq. 4.20 as

$$\tilde{\theta} = \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t} \int_{\pi(\Delta\phi - h)}^{\pi(\Delta\phi + h)} \sum_{n,n'=0}^{2^t - 1} e^{2i(n+n'-2^t)u} du \tag{4.21}$$

Where we have used Euler equations to manipulate the integral function and changed the variable to $u = \pi(\phi - x)$. Further we have introduced $\Delta\phi = \phi - 0.\phi_u$ as a shorthand for the phase mismatch between the real eigenphase and the center of the filtering box.

To solve Eq. 4.21 we consider separately the terms $n+n' = 2^t$ and $n+n' \neq 2^t$:

$$\begin{aligned}
\tilde{\theta} &= \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}\Big[(2^t-1)\int_{\pi(\Delta\phi-h)}^{\pi(\Delta\phi+h)} du + \int_{\pi(\Delta\phi-h)}^{\pi(\Delta\phi+h)} \sum_{n,n'=0\,,\,n+n'\neq 2^t}^{2^t-1} e^{2i(n+n'-2^t)u} du\Big] \\
&= \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}|\Big[(2^t-1)u - i\sum_{n,n'=0\,,\,n+n'\neq 2^t}^{2^t-1} \frac{e^{2i(n+n'-2^t)u}}{2(n+n'-2^t)}\Big]|_{\pi(\Delta\phi-h)}^{\pi(\Delta\phi+h)} \\
&= \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}\Big[2\pi(2^t-1)h - i\sum_{n,n'=0\,,\,n+n'\neq 2^t}^{2^t-1} \frac{e^{2i(n+n'-2^t)\pi(\Delta\phi+h)} - e^{2i(n+n'-2^t)\pi(\Delta\phi-h)}}{2(n+n'-2^t)}\Big]
\end{aligned} \tag{4.22}$$

This expression is exact and contains two terms arising from different contributions. We are interested in studying the limiting cases of this formula: (i) when the box spans the all possible values of the phase (i.e., $h = \frac{1}{2}$) and (ii) when the box is very narrow and comparable with the grid spacing (i.e., $h = \frac{|\mathcal{G}|}{2^t}$ with $|\mathcal{G}|$ being the number of bitstrings included in $\mathcal{G}$). The first one is interesting because we can see that we are able to recover the original picture of Cruz et al.[37], on the other hand the latter case is useful as represents the tipical regime of use for the GCE.

Starting with the case of $h = \frac{1}{2}$ it is straightforward to show that $\theta = \frac{|c_u|^2 e^{i2\pi\phi}}{2^t}(2^t - 1)$ with modulus

$$|\tilde{\theta}| = |c_u|^2 \frac{2^t-1}{2^t}\ . \tag{4.23}$$

Which is exactly the result of Ref.[37] when $|c_u|^2 = 1$ (i.e., the input state is an eigenstate) and the limit of large t is considered.

We now move to the case of $h = \frac{|\mathcal{G}|}{2^t}$. We start expanding the complex exponentials in the sum around $\Delta\phi = 0$ and $h = 0$, which holds when the number of readout qubit is reasonably high:

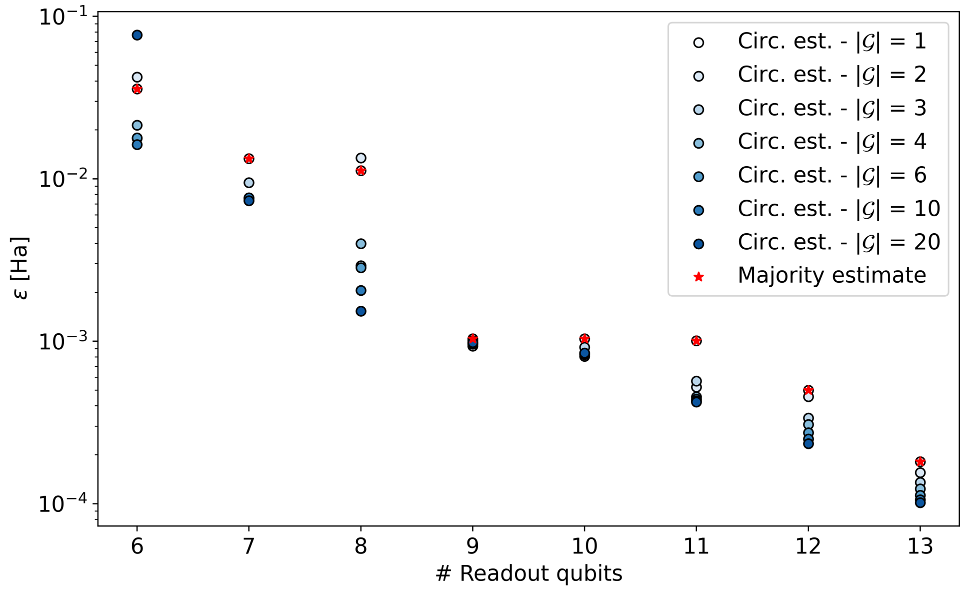


Figure 4.6: Estimation accuracy of the generalized circular estimator. We report the error ϵ w.r.t. the FCI solution for the H_3^+ molecule at the ground state geometry varying the number of readout qubits. Red stars show the error obtained with the majority estimator (i.e. considering only the most sampled bitstring), dots display the error obtained with the estimator proposed in this work varying the window width (shades of blue).

$$\begin{aligned}
\tilde{\theta} &\approx \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}\Big[2\pi(2^t-1)h - \frac{i}{2}(i4\pi h \sum_{n,n'=0}^{2^t-1} 1 - 8\pi^2\Delta\phi h \sum_{n,n'=0}^{2^t-1}(n+n'-2^t))\Big] \\
&= \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}\Big[2\pi(2^t-1)h + 2\pi h 2^{2t} - i4\pi^2\Delta\phi h 2^{2t}\Big] \\
&= \frac{|c_u|^2 e^{i2\pi\phi}}{\pi 2^t}\Big[2\pi(2^t-1)\frac{|\mathcal{G}|}{2^t} + 2\pi|\mathcal{G}|2^t - i4\pi^2|\mathcal{G}|\Big] \\
&= \frac{2|c_u|^2|\mathcal{G}||B|}{2^t}e^{i(2\pi\phi-\alpha)} \\
&\approx 2|\mathcal{G}||c_u|^2 e^{i2\pi\phi}
\end{aligned} \tag{4.24}$$

Where moving from the second to the third line we have used the fact that $h = \frac{|\mathcal{G}|}{2^t}$ and that $\Delta\phi$ is at most $\frac{1}{2^t}$. In the fourth line we have introduced B as a shorthand for

$$B = |B|e^{-i\alpha} = 2^t + \frac{2^t - 1}{2^t} - i2\pi \ , \tag{4.25}$$

finally, the last step is obtained keeping only the leading term in 2^t. The additional terms originate as an artifact of the smoothing procedure, but it is easy to show that for large t values $|B| = \mathcal{O}(2^t)$ whence $\alpha \equiv \arcsin(\frac{2\pi}{|B|}) \approx 0$.

This expression is useful as shows that the GCE introduces an exponentially vanishing bias into our estimator. This result comes from an expression obtained expanding Eq. 4.22 around small values of $\Delta\phi$ and h at the first order; in practice, considering the complete expression, the GCE accuracy can exceed the one of the standard majority estimator. This result was already found for the case of an exact input eigenstate[37], here we show that this feature holds even when an arbitrary input state is considered provided that the measurement distribution is reconstructed via a sufficient number of samples. Particularly, in Fig. 4.6, we report an example of the error ϵ made by the GCE in estimating the energy of an electronic state as a function of the number of readout qubits (dots); we compare this error to that made using the standard majority rule (stars). Further, we have also studied the effect of varying the number of bitstring included in the circular average (i.e., $|\mathcal{G}|$) (different shades of blue). First of all, we can notice that the GCE is performing better than the majority rule estimate for all values of t. More precisely the error of the latter scales as $\epsilon_{MR} = \frac{1}{2^{t+1}}$ while the circularly averaged estimated error scales as $\epsilon_{GC} = \frac{1}{2^{t+2}}$.

If we now consider the dependency on the window size, two effects become noticeable: (i) for smaller values of t, there are some values of h for which we make a larger error than expected using the standard estimator; (ii) contrary to expectations, for low values of t, it does not hold true that a wider window necessarily gives a more accurate result. The first effect is related to the final discussion of the last paragraph, i.e., including too many bitstrings when peaks corresponding to different eigenstates are not sufficiently separated may introduce spurious contributions. The second effect stems from the fact that the improved accuracy of the GCE is due to error cancellation effects. With a lower number of qubits, this error cancellation may undergo significant variation due to discretization effects on the averaged distribution $P(0.\phi)$.

We now conclude this section addressing the question: how many samples do we need to reconstruct the parent distribution and obtain the accuracy shown in Fig. 4.6? To answer we can compute the variance of the mean phase direction $\tilde{\mu}$ and use Chebyshev's inequality:

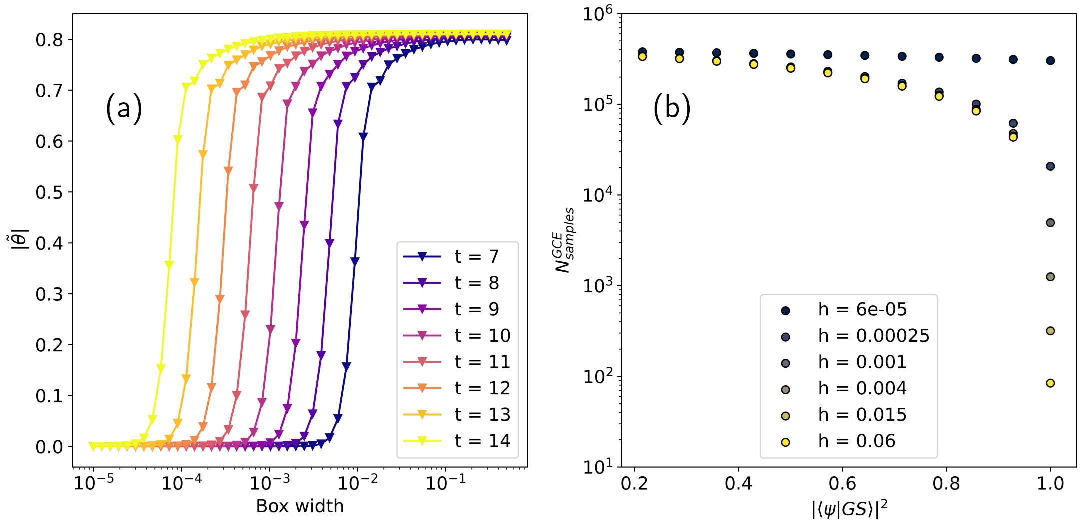


Figure 4.7: Sampling cost of the generalized circular estimator. a) Modulus of $\tilde{\theta}$ as a function of the box width varying the number of readout qubits (shown with different colors) according to Eq. 4.22. We set $\Delta\phi = \frac{1}{2^t}$. Further, we have considered an initial state with an overlap with the target state of $|c_u|^2 = 0.8$. b) Sampling cost of the generalized circular estimator $N^{GCE}_{samples}$, based on Eq. 4.26, as a function of the initial state overlap with a target ground state $|\langle\psi|GS\rangle|^2$ varying the window width h (shown with different colors).

$$N^{GCE}_{samples} = \frac{\mathrm{Var}(\tilde{\mu})}{\epsilon^2} \tag{4.26}$$

The variance of the mean phase direction is obtained appyling error propagation theory on $\tilde{\theta}$ (Eq. 4.27) assuming the latter is a proper random variable [58] and using the definition of variance for $\tilde{\theta}$ according to circular statistics [54] (Eq. 4.28):

$$\mathrm{Var}(\tilde{\mu}) = \frac{1}{2}\left[|\frac{\partial \tilde{\mu}}{\partial Re\{\tilde{\theta}\}}|^2 \mathrm{Var}(\tilde{\theta}) + |\frac{\partial \tilde{\mu}}{\partial Im\{\tilde{\theta}\}}|^2 \mathrm{Var}(\tilde{\theta}) \right] \tag{4.27}$$

$$\mathrm{Var}(\tilde{\theta}) = 1 - |\tilde{\theta}| \tag{4.28}$$

In Fig. 4.7 we study the cost of obtaining an estimate for the mean phase direction with an error of $\epsilon = 1\, mHa$. Particularly, in Fig. 4.7a we plot $|\tilde{\theta}|$ varying both the number of readout qubits and the box width h. We can notice that the effect of increasing the number of readout qubits is to recover the full information of the peak (i.e. reach the limit of Eq. 4.23) with a smaller window. From Eqs. 4.26-4.28 we can understand that this directly impacts the cost of the routine. Indeed, as we approach Eq.4.23 we tend to lower the cost of the estimation, in order to do so carefully (i.e. without including other peaks' contributions in the average) we need to increase the grid resolution. Similar considerations can be drawn looking at Fig.4.7b. Here we see the effect of the initial state overlap on the cost varying the window width at fixed number of readout qubits ($t = 13$). For a numerical assessment of these estimates we refer to the appendix 4.6.

Overall we see that the increased accuracy of $\tilde{\mu}$ has a considerable computational cost, especially if the input state has a small overlap with the eigenstate of interest. Further investigation is required to reduce such a computational overhead and thus enhance the scalability of the approach with respect to the size of the system. The improvement would be particularly relevant to the treatment of strongly correlated systems, where, even for modest system size, the initial state is generally expected to have a relatively small overlap with the target state[39]. Finally, we recall that beyond an improved accuracy this approach allows to obtain gradients of the estimated eigenvalues. This is a completely different task and in the next section we go through the costing of this routine and compare with existing methods in literature.

4.4 Cost of gradient estimation and related works

Building on the previous analysis, in this section we analyze the cost of evaluating the gradient with the quantum differentiable program sketched in Fig. 4.2 and compare our results with state-of-the-art methods to evaluate energy gradients for the molecular hamiltonian on a fault-tolerant quantum computer.

4.4.1 Computational cost analysis

The overall cost for the gradient evaluation with the differentiable QPE will be given by:

$$\tilde{C} = CNk \tag{4.29}$$

Where C is the cost of a single QPE circuit evaluation, N is the number of calls to the routine needed for evaluating $\tilde{\mu}$ with an ϵ-good estimate and k is the number of calls required for evaluating the gradient. In this section we estimate $\tilde{C}$ breaking down the different contributions.

It is important to note that, with regard to molecular systems, the scaling of the routine cost with the system size meant as number of electrons/spin-orbitals is included in the quantity C and here will not be considered. Concerning N, by definition we have $N = N^{GCE}_{samples}$ as defined in Eq. 4.26.

Thus, we are only left to estimate k. The number of calls to the QPE subroutine to estimate the gradient with a parameter-shift rule, following Wierichs *et al.*[51], is given by $k = RM$ where M is the number of cartesian coordinates (or more generally the number of variables) w.r.t. which differentiate and R is a factor counting either the number of parameter-dependent gates (decomposition based parameter-shift rule) or the number of unique positive differences in the parameterized circuit spectrum (Fourier based parameter-shift rule). Depending on the structure of the Hamiltonian the number of independent circuits executions needed for the parameter-shift rule can vary substantially. Since the eigenvalue distribution for a molecular Hamiltonian does not have an ordered structure it is safer to consider in the cost analysis a decomposition based strategy for the quantum gradient. If we indicate as G the number of gates where the parameters appear after the compilation of the simulation routine implemented in the quantum circuit we get:

$$k \approx \mathcal{O}(G\, t\, M) \tag{4.30}$$

Where t is the number of readout qubits and, thereby, the number of times we call the quantum simulation routine inside the phase estimation circuit.

The last step needed to estimate the overall cost $\tilde{C}$ is to obtain the number of shots required for the gradient estimation. As we focus on the quantum chemistry framework we will label the parameters w.r.t. which we are taking the gradient as $\mathbf{R}$ to indicate nuclei positions. Of course we recall that, as mentioned in the introduction, the application range of energy derivatives and automatic differentiation for quantum chemistry is much broader than just geometry optimization.

Within the parameter-shift framework (either decomposition-based or Fourier-based) we can write the force acting on the nucleus at the position R_j using the generalized circular estimator as:

$$\frac{\partial \tilde{\mu}}{\partial R_j} = \sum_{l}^{k} y_l \tilde{\mu}(\mathbf{R}_l) \tag{4.31}$$

Where k is the number of function evaluation required by the parameter-shift rule and $\mathbf{y}$ is a coefficient vector defined by the particular flavour of gradient rule used.

The optimal budget shot allocation to obtain $\frac{\partial \tilde{\mu}}{\partial R_j}$ within a tolerable error ϵ is:

$$N_{\partial_j \tilde{\mu}} = \frac{\mathrm{Var}(\tilde{\mu})||\mathbf{y}||_1^2}{\epsilon^2} \tag{4.32}$$

Finally, we conclude this section estimating the number of queries to a circuit implementing the evolution operator U for estimating the gradient with a fully differentiable Quantum Phase Estimation as:

$$\tilde{C} = \mathcal{O}(G\,t\,MN_{\partial_j\tilde{\mu}}) \tag{4.33}$$

Interestingly, lowering the number of readout qubits affects simultaneously: (i) t, (ii) ϵ and (iii) $N_{\partial_j\tilde{\mu}}$ as we expect the 1-norm of the parameter-shift rule's coefficients to decrease as their number decreases. To this end methodologies as the ones developed in Ref.[59] may be extremely beneficial in this context.

4.4.2 Related works on gradient estimation

The goal of this section is to briefly summarize previous works on evaluating energy derivatives within the context of quantum computation. This account closely follows the work of O' Brien *et al.*[34] which reported current state-of-the-art methods to compute energy gradients in a fault-tolerant setting.

We remark that all our considerations are made for a fully variational wavefunction meaning that given a set of *external physical parameters* $\mathbf{x}$ and a set of *wavefunction parameters* λ we calculate the energy from the expression

$$E(\mathbf{x}) = E(\mathbf{x}, \lambda^*) \tag{4.34}$$

where the parameters λ^* represent the optimal value of λ and where the optimized energy function $E(\mathbf{x}, \lambda^*)$ satisfies the variational conditions for all values of the external parameters $\mathbf{x}$. This statement holds true also for the quantities extracted from QPE as, within the discretization error due to the finite number of readout qubits, we sample from the exact eigenbasis of the molecular Hamiltonian.

Hellmann-Feynman approach

This first approach is based on the seminal work of Feynman [60]. The idea is that we can compute energy derivatives from the expectation value of the Hamiltonian derivative operator:

$$\frac{dE}{d\mathbf{x}} = \langle\Psi|\frac{dH}{d\mathbf{x}}|\Psi\rangle \tag{4.35}$$

As we can notice Eq. 4.35 does not provide a recipe for evaluating energy gradients but only tells us the quantity that we need to evaluate. Among the various strategies proposed in literature a possible strategy is to use the overlap estimation algorithm which allows to compute Eq. 4.35 at the Heisenberg limit (i.e. with a cost scaling as $\mathcal{O}(\epsilon^{-1})$). This method provides an estimate of the energy gradient by directly calculating the expectation value with the algorithm proposed in Ref. [61]. As compared to the latter, since we are dealing with a real-valued expectation we only need to call the amplitude estimation algorithm two times instead of three as originally sketched in Ref.[61].

It is important to notice that this method implies to adopt a block-encoding routine for the energy derivative operator. Using tensor hyper-contraction the authors of Ref. [34] were able to show that, despite a larger prefactor, the asymptotic cost of block-encoding $\frac{dH}{d\mathbf{x}}$ equals the asymptotic cost of block-encoding the molecular Hamiltonian.

Direct gradient-based quantum estimation algorithm

As an alternative to the previous method, we can employ another ad-hoc approach developed for computing gradients of functions. In particular, building upon the algorithm developed in [62] and refined in [35], the authors of [34] demonstrate that nearly optimal scaling can be achieved for calculating the forces of molecular Hamiltonians.

Without going into the details the main idea is that using a single query to a state preparation routine U_ψ (i.e. we need to know how to prepare the exact ground state of the molecule) and querying $\mathcal{O}(\frac{\sqrt{M}}{\epsilon})$ (where M is the number of variables to differentiate) times the oracle

$$U_p : |x\rangle|0\rangle \rightarrow |x\rangle \otimes (\sqrt{(f(x))}|\psi_1(x)\rangle|1\rangle + \sqrt{(1-f(x))})|\psi_0(x)\rangle|0\rangle \tag{4.36}$$

it is possible to simultaneously evaluate a given $f(\mathrm{x})$ at differents points and computing a finite difference approximation of the function directly on the quantum computer.

This algorithm, which makes use of different qubits register, makes several calls to routines implementing operators like $U = e^{-it\frac{dH}{dR_k}}$ for which the same results on block-encoding discussed in the previous paragraph is used.

Finite differences approach

Finally, another possibility is to consider the QPE algorithm itself as a black-box which allows to sample the function at a given point in the parameter space. From this standpoint we can use it to compute gradients via classical finite differences approximations.

Concerning this method, again we follow Ref. [34] and consider a degree-$2m$ finite difference formula as:

$$\begin{aligned} \frac{dE^{(m)}}{d\mathbf{x}} &= \sum_{l=-m, l\neq 0}^{l=m} b_l^{(m)} E(\mathbf{x} + l\, d\mathbf{x} \cdot \mathbf{v}) \\ b_l^{(m)} &= \begin{cases} 0 & l = 0 \\ \frac{(-1)^{l-1}}{d\mathbf{x}\, l} \frac{\binom{m}{|l|}}{\binom{m+|l|}{|l|}} & l \neq 0 \end{cases} \end{aligned} \tag{4.37}$$

We will not analyze in detail the cost of this algorithm as it is already discussed by the same authors. Nonetheless, we want to highlight the difference in Eq. 4.37 when using the ME estimator or the GCE estimator. Particularly, propagating errors through the previous equations we get:

$$\left| \frac{dE}{d\mathbf{x}} - \frac{dE^{(m)}}{d\mathbf{x}} \right| = \sum_{l=-m, l\neq 0}^{l=m} \epsilon_{PE} |b_l|^2 + \epsilon_{FD} \tag{4.38}$$

In the last equation appear two sources of error: (i) the error due to the phase estimation procedure and (ii) the error due to the finite difference approximation.

At first, we focus on the error due to the phase estimation procedure. When using the ME estimator, $\epsilon_{PE} = \epsilon_{ME} = \frac{1}{2^{t+1}}$. On the other hand, if we consider the GC estimator we should use: $\epsilon_{PE} = \epsilon_{GC} = \frac{1}{2^{t+2}} + \sqrt{\frac{\mathrm{Var}^{h,t}(\mu)}{N}}$. From the last two expressions, we can deduce that the gradient approximation obtained with the GC estimator is more accurate when

$$\sqrt{\frac{\mathrm{Var}^{h,t}(\mu)}{N}} \leq \frac{1}{2^{t+2}} \ . \tag{4.39}$$

At this point it is important to stress that, when using the MR estimator, the step size used must be such that it satisfies

$$d\mathbf{x} : |\phi(\mathbf{x} + l\, d\mathbf{x} \cdot \mathbf{v}) - \phi(\mathbf{x} + l'\, d\mathbf{x} \cdot \mathbf{v})| > \frac{1}{2^t}\ \forall\, l, l' \tag{4.40}$$

The last condition stems from the fact that for a signal to be detected with the finite difference approximation not only the step size must be small enough to avoid truncation errors but should also be large enough that we are able to spot function variations at different points as per the QPE grid resolution. If these conditions are not met, as shown in Fig. 4.8, we are not able to approximate the gradient even though we use high-order degrees of approximation. In particular, Fig. 4.8a shows the gradient norm $|\nabla\langle H\rangle|$ (used as a metric of sensitivity) computed with a central finite difference approximation at different orders (blue stars) adopting a ME estimator. For comparison we reported with a solid blue line the analytical result at the CCSD level (which is equivalent to FCI for a two-electron system as the one considered in this case) and with a solid orange line the gradient norm corresponding to the result of the differentiable QPE. As we can see, even using a degree-8 finite difference approximation with a step size of $dx = 10^{-3}$Å the resolution enabled by a $t = 13$ grid does not allow to detect any signal. In Fig. 4.8b we compare the error of the gradient (as the gradient norm difference) estimated with a degree-2 finite difference method using the GCE (orange stars) and the ME estimator (blue stars). The black dashed line represents a threshold determined by the exact gradient norm: points situated below this

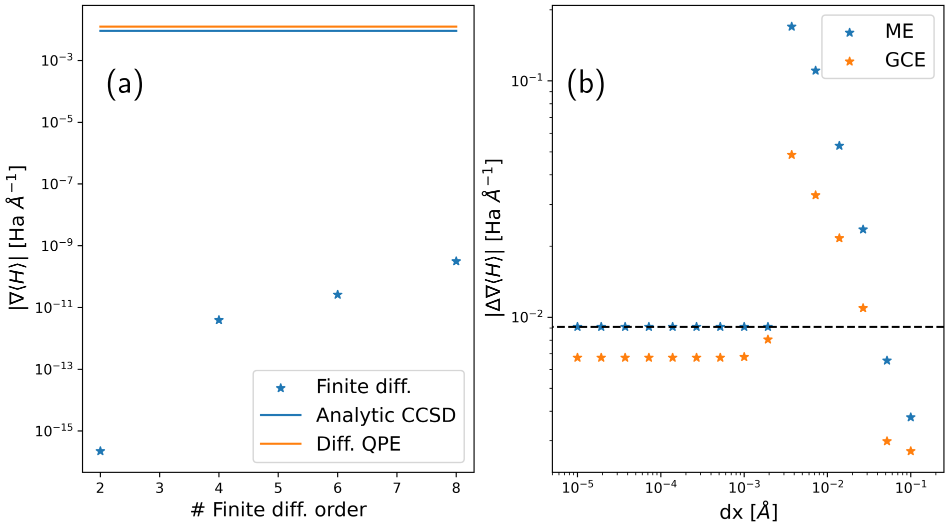


Figure 4.8: Finite discretization affects the evaluation of gradients through finite difference methods. a) Energy gradient norm with respect to the nuclei position for H_3^+ at the HF equilibrium geometry. Blue stars represent values obtained with a finite difference approximation at different orders using the majority rule estimator. The solid blue line corresponds to the analytically computed gradient at the CCSD level of theory, while the orange solid line corresponds to the gradient obtained with the fully differentiable pipeline described in Fig. 4.2. b) Error in estimating the gradient using a finite difference method (order 2) as a function of step size. The blue stars represent values obtained with the majority estimator, while the orange stars are obtained using the estimator developed in this work. All calculations have been performed using t=13 readout qubits.

line suggest a better approximation of the gradient. Conversely, points lying on (or above) the line are associated with null (or incorrect) gradients. The smooth GCE, as observed, enables the computation of gradient approximations even with very small step sizes, whereas the ME estimator fails to detect any signal. Upon increasing dx we see that the estimated gradient starts suffering from truncation errors. Eventually, at values of $dx \approx 0.1\text{Å}$ the estimated gradients becomes more accurate, nonetheless these values of step size have already proven unreliable in standard quantum chemistry settings where we usually aim to an error on the final predicted equilibrium bond lengths of around $\Delta r \approx 0.01$Å.

These results show that with a small scale fault-tolerant computer (e.g. using $t \approx 10 - 20$), despite the good asymptotic scaling shown in Ref.[34], using finite differences with the ME estimator may not be feasible.

To conclude this section we want to summarize the result of this comparison: our approach currently suffers from a finite-shot sampling cost which makes it unfavourable w.r.t. other strategies such as Ref. [63] and Ref. [35] which saturate the Heisenberg limit. On the other hand, the proposed strategy is more flexible when it comes to consider extensions of this algorithm towards higher-order derivatives. Indeed, both considering an Hellmann-Feynman kind of approach and the gradient-based estimation algorithm, the extension to higher order derivatives requires in the former case an exponentially higher[63] cost, in the latter a redesign of the algorithm itself. In addition, and perhaps more importantly, our algorithm does not necessitate the preparation of the exact ground state on the quantum computer, a requirement that, on the other hand, is mandatory in both previous cases. Such a condition could potentially limit the application of these methodologies to regimes where classical computation proves more efficient than quantum computation[38, 39].

The goal of a future work will be to restore optimal (Heisenberg) scaling of the approach proposed here by leveraging techniques such as compressed sensing[64] or amplitude amplification [65] tailored to an automatically differentiable framework. Another essential direction to align this strategy with existing literature proposals involves reducing the cost of differentiation itself. Currently, quantum differentiation methods exhibit backpropagation-like scalings[66] for only a specific class of circuits. Our cost analysis, based on a general differentiation rule, would greatly benefit from this improvement.

4.5 Numerical results

Here we report the numerical results of our implementation of the GCE described in Sec. 4.3 coupled to the automatically differentiable pipeline provided by PennyLane. The latter, thanks to the differentiable quantum chemistry[44] feature ensures end-to-end differentiability of the procedure and allowed us to apply this method to the evaluation of energy gradients of molecular Hamiltonians which is of utmost importance in the field of quantum chemistry.

In addition to numerically validating the algorithm proposed in this study, these calculations themselves represent a novelty. To the best of our knowledge, only two other papers (focusing on fault-tolerant algorithms) in literature report similar calculations: O'Brien *et al.*[63] calculated the equilibrium bond length of H_2 on an actual quantum processor in a minimal basis set, and more recently, Sugisaki *et al.*[67] developed a protocol for optimizing geometry using a Bayesian phase difference algorithm, applying it to different molecules with the largest system being N_2 with a Complete Active Space CI using 6 electrons and 6 orbitals (6e, 6o).

4.5.1 Computational details

All the calculations have been performed with a locally modified version of PennyLane[27] v0.24, the code generating all the data of this section can be found at [68]. Reference calculations have been performed with the Psi4[69] code for the calculations at the FCI level, CASCI calculations on the CH_2O molecule have been performed with the PySCF[70] code. All the calculations are performed adopting a STO-3G basis set; concerning the H_3^+ molecule this implies 6 qubits needed for encoding the wavefunction into the system register using a Jordan-Wigner encoding[71], for the CH_2O molecule the active space was chosen keeping active only two electrons and the four highest lying MOs. For all the calculations we set the value of $k = 1000$ for smoothing the boxcar function and a window width $h = \frac{8}{2^t}$. We set the temperature for the tempered-softmax distribution (Eq. 4.12) at $T = 0.0035$.

We found that higher values (i.e. $k > 5000$) of k lead in some cases to numerical instabilities. We recall that a thorough analysis of the window size effects is given in Sec. 4.3 (Fig. 4.6). Further, we set the number of readout qubits $t = 13$ for the H_3^+ molecule and $t = 11$ for the CH_2O molecule. Altogether we simulated 19 qubits in both cases.

It is important to notice that QPE methods always involve a rescaling of the Hamiltonian needed to avoid aliasing for the recovered energy. In practice, it means working with a rescaled Hamiltonian $H_{SC} = \frac{H_{mol} - E_{min}\mathcal{I}}{\Delta E}$ with ΔE being the spectral range of interest. Unfortunately, such a procedure increases the cost making the overall scaling from $\mathcal{O}(\epsilon^{-1})$ to $\mathcal{O}(\Delta E \epsilon^{-1})$. In all our numerical simulations we mitigated this issue working with the purely electronic Hamiltonian (i.e. adding *a posteriori* the frozen core contribution and/or the nucleus-nucleus interaction terms). Finally, for the calculations on the trihydrogen cation, we used a gradient descent optimizer with stepsize of 0.01 as implemented in PennyLane; regarding the formaldehyde, we adopted an in-house automatically differentiable implementation of the BFGS[72] optimizer.

4.5.2 Trihydrogen cation (H_3^+)

As a first application of our differentiable pipeline, we show two examples of geometry optimization on the H_3^+ molecule. This system, being the simplest polyatomic molecule has been extensively studied for benchmarking *ab initio* models[73, 74] and it has also attracted attention for being one of the most common species in molecular hydrogen plasmas occurring in interstellar clouds and planetary ionospheres[75].

In Fig. 4.9 we report a convergence plot of the geometry optimization for the ground state which is stable if the hydrogen atoms are arranged in a triangular shape[76]. The initial state used for this calculation is the HF state which after Jordan-Wigner mapping reads $|HF\rangle = |110000\rangle$. The initial geometry has been set equal to an isosceles triangle with bond lengths slightly distorted from the HF equilibrium geometry. Please notice that in Fig.4.9b the dashed dotted lines refer to different bond lengths and that the green and yellow ones are superimposed.

As we can see, after 20 iterations the optimization is able to recover the correct equilatelar (D_{3h} symmetry) shape in agreement, within chemical accuracy, with the FCI optimized geometry. Interestingly, we can notice how the energy obtained with the QPE is slightly below the FCI reference (orange dashed line in Fig.4.9a), this is due to the fact that the QPE is not a variational method.

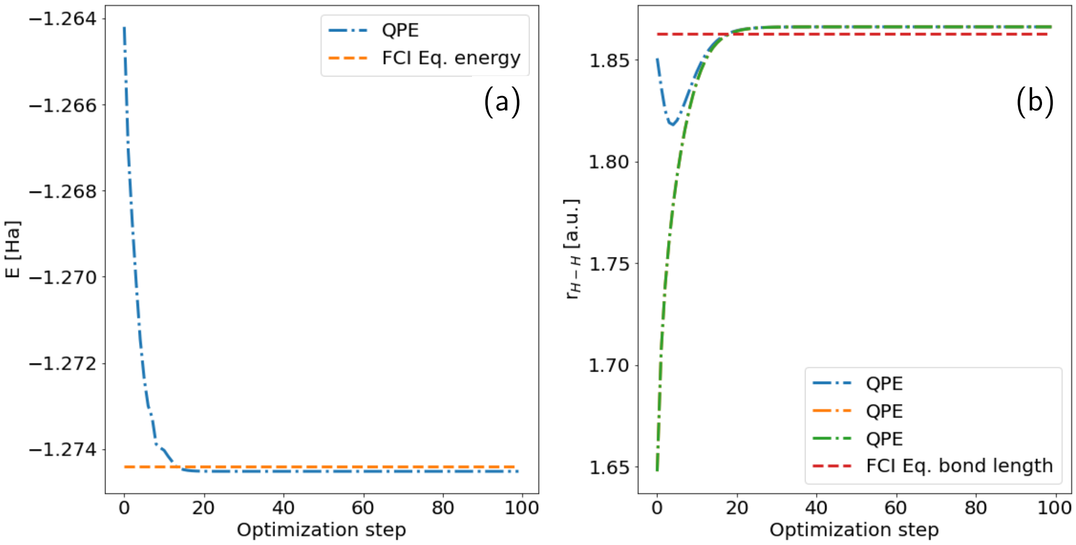


Figure 4.9: Ground state geometry optimization for the H_3^+ molecule. a) Convergence of the ground-state energy E and b) the bond length r_{H-H}. Dashed-dotted lines represent values obtained during the optimization at the QPE level, dashed lines are reference values obtained at the FCI level.

We consider a second example for the same molecule: the first excited triplet state. Here the initial state used is a spin-flipped doubly excited determinant built upon the HF configuration, i.e. $|\psi\rangle = t_{01}^{35}|HF\rangle = |0001010\rangle$. As already reported in literature[77], the triangular geometry is not stable under this excitation and undergoes photodissociation. On the other hand, a linear structure is a stable configuration for the $|T_0\rangle$ state; thus, we initialized the geometry in a linear shape with the two terminal hydrogens not equally distant from the central one (see Fig. 4.10b). Again we can see how the optimization allows to reach energy and bond length with errors below chemical accuracy (i.e. $\Delta E < 1\,\mathrm{m}Ha$ and $\Delta r < 0.01\,\text{Å}$).

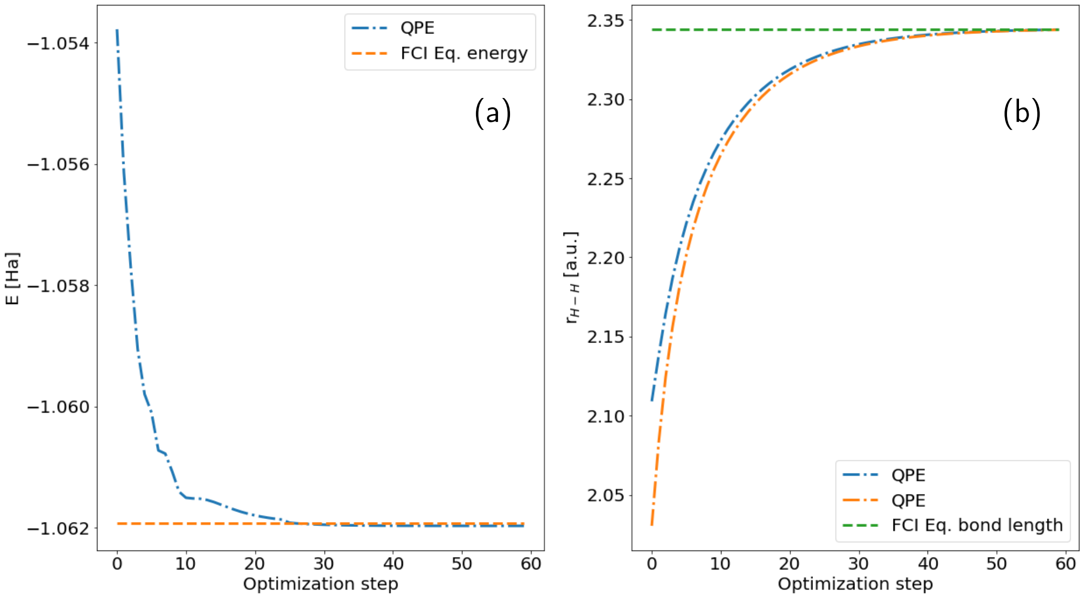


Figure 4.10: First triplet state ($|\mathrm{T}_0\rangle$) geometry optimization for the H_3^+ molecule. a) Convergence of the ground-state energy E and b) the bond length r_{H-H}. Dashed-dotted lines represent values obtained during the optimization at the QPE level, dashed lines are reference values obtained at the FCI level.

In the next section we showcase an application to a larger molecule to understand the impact of our procedure when considering a more elaborate electronic structure.

4.5.3 Formaldeyhde ($\mathrm{CH_2O}$)

Here we show two examples of geometry optimization carried out with our automatically differentiable QPE on a larger molecular system. We studied the formaldehyde molecule in its ground state and first triplet state ($|T_0\rangle$) corresponding to the spin-flip transition $^3(n \rightarrow \pi^*)$. These systems have been extensively studied both theoretically and experimentally[78–80] as representatives systems for the carbonyl-chemistry and, more broadly, for their role played in photochemistry.

In order to perform geometry optimization with the scheme proposed in Sec. 4.3, we differentiated throughout all the optimization an eigenstate distribution induced by the $|HF\rangle$ state which, for the active space considered, reads $|HF\rangle = |11000000\rangle$. In Tab. 4.1 we report the results of our optimization and compare them with a geometry optimization carried out at the CASCI level (within the same active space) and with experimental data from [81] as reported in [82].

As we can see the predicted CO double bond length is within chemical accuracy ($\Delta r < 0.01$ Å) both considering as a reference the theoretical calculations at the CASCI level and the experimental values. Also, considering the predicted angles, an error of $\Delta\theta \approx 2°$ represents a good agreement between theory (both methods) and experiments. Instead, looking at the predicted CH bond length we can see that a more significant mismatch is present between the theoretical methods ($\Delta r \approx 0.05$ Å). Such a difference may be ascribed to different factors: (i) biases induced by the external parameters T and k, (ii) possible numerical instabilities encountered in the AD pipeline or (iii) differences in the optimization procedure and the underlying Hartree-Fock solver. Notably, we can observe an error cancellation effect (with the active space truncation) as, looking at the experimental values, we can appreciate a better agreement with the result obtained with the differentiable QPE. Such an agreeement gets even better (i.e. within chemical accuracy) if we look at the experimental results reported by Ref. [83].

Table 4.1: Geometry optimization of the $|\mathrm{GS}\rangle$ electronic state of formaldehyde. Distances are expressed in Å, angles in degrees. Experimental errors are reported in brackets.

	CASCI(2e, 4o)	QPE(2e, 4o)	Exp.[81]
r_{CO}	1.212	1.211	1.21(1)
r_{CH}	1.055	1.106	1.12(1)
$\mathrm{O\hat{C}H}$	123.18	122.96	121(1)
$\mathrm{O\hat{C}H}$	123.18	122.96	121(1)
$\mathrm{H\hat{C}H}$	113.64	114.08	-

We now move on to the discussion of the results concerning the optimization of the first triplet state ($|T_0\rangle$). Here, all along the optimization procedure, we differentiated through a distribution induced by the triplet state built upon excitation of the HF determinant, $|\psi\rangle = t_0^2|HF\rangle + t_1^3|HF\rangle$ with t_i^j being a single excitation from spin-orbital i to j. We recall that in our convention for labeling the orbitals, α/β spins are interleaved and that spin-orbital labeling starts with the leftmost qubit being indexed with 0.

As we can notice in Tab. 4.2, bond lengths and angles are within chemical accuracy comparing the QPE results and the classical CASCI reference further proving that the GCE estimator coupled with the automatic differentiation pipeline of PennyLane is able to reproduce both total energies and forces. On the other hand, looking at the experimental reference, we can see that the active space chosen (regardless of the theoretical method employed) induces an error of about ≈ 0.02Å overestimating the CO bond length and underestimating the CH bond length. This is in agreement with other results reported in literature for calculations in restricted active space; for example, Angeli et al.[79] (including orbital optimization) report an error of 0.062 and 0.028 Å for the CO bond length and 0.005 and 0.028 Å for the CH bond, respectively for two different active spaces.

Table 4.2: Geometry optimization of the first triplet state ($|T_0\rangle$) electronic state of formaldehyde. Distances are expressed in Å, angles in degrees.

	CASCI(2e, 4o)	QPE(2e, 4o)	Exp.[80]
r_{CO}	1.320	1.325	1.307
r_{CH}	1.063	1.060	1.084
$O\hat{C}H$	117.36	119.75	-
$O\hat{C}H$	116.72	119.74	-
$H\hat{C}H$	125.92	120.51	121.8

These results demonstrate not only the effective application of the automatically differentiable QPE to a non trivial chemical system but also emphasize the importance of numerics alongside with theoretical cost analysis when proposing an algorithm. Indeed, these findings underscore the necessity of specific domain-knowledge and stress the importance of maturing practical experience, as similarly demonstrated in the literature for standard computational quantum chemistry, for achieving results that align with other methods and experiments. In particular, we found that to practically extend these experiments to the optimization of high-energy excited states it is mandatory to conduct on-the-fly assessments to understand how the displacement of atoms induces changes in electronic structure, and consequently, determining how to modify the input molecular state within the quantum circuit. The development of a cheap and fast algorithm to make this assessment practical is a possible avenue of this work.

4.6 Conclusions

In this work we have extended the smooth estimator proposed in Ref.[37] for the QPE to be used when the exact eigenstates of the unitary U are not known. This allowed the implementation of the phase estimation algorithm in a quantum differentiable framework applicable to non-trivial problems, such as molecular geometry optimization. These calculations represent a milestone unreported so far both concerning system size (number of atoms) and target state (we showcased optimizations beyond the singlet ground state).

Despite the various possibilities that this work offers, there are some aspects that limit its scalability to larger systems at present. Particularly, we have seen how reconstructing the measurement distribution requires a substantial computational overhead, that can be mitigated by improving the wavefunction guess.

As we mentioned in Sec. 4.4.2, a straightforward improvement may be the inclusion of a cosine or Kaiser window into the phase estimation circuit; this concentrates the information in the main lobe of the distribution and may lower the number of relevant bitstrings to sample for the circular average. As shown in Ref.[59] this has already proven a cubic speedup for the majority rule estimator and may provide similar speedups for the GCE. In addition to the rise in cost, it is important to note that the smoothing procedure we have reported results in a nontrivial dependence on both the temperature (T) and the boxcar hardness factor (k). Procedures to mitigate the bias arising from these factors are necessary for applying this technique to more intricate calculations, such as the optimization of excited-state geometry or higher-order derivatives. Possible strategies could imply resorting to recent variations of the quantum phase estimation such as[64, 84, 85].

Once a more stable smoothing procedure is found an important contribution would be showing the application of this pipeline to the simulation of spectroscopy experiments.

It should also be emphasized that when developing a quantum algorithm we are not only concerned with finding a solution to a useful problem but also about ensuring that the runtime of the algorithm is feasible and advantageous compared to other solutions, including classical ones. Despite the favorable scaling of QPE, the answer to this question in the field of quantum chemistry is still open[38, 40, 41, 86]. Considering the algorithm proposed in this work, there will undoubtedly a need for further work reaching cost-effective differentiable solutions. This study further underscores the importance of expanding the classes of circuits that allow for backpropagation scaling in derivative calculations[66].

Appendix A: The physical meaning of differentiating through the measurement distribution

In this section we derive analytic expressions for the energy derivative as defined by Eq. 4.15. These expressions allow us to understand the physical origin of the automatically computed gradient.

The main idea is that we can recast the whole differentiable QPE procedure in this expression:

$$E \approx \mathcal{S}\big[\int \langle\Psi|e^{-iHt}|\Psi\rangle e^{i\omega t}dt\big] = \mathcal{S}\big[I(\omega)\big] \tag{4.41}$$

Where $\mathcal{S}$ is a differentiable functional corresponding to the smoothing procedure outlined in Fig. 4.3 of the main text.

Accordingly, we can write explicitly the energy derivative differentiating the last expression:

$$\frac{\partial \mu}{\partial \mathbf{R}} = \frac{\delta \mathcal{S}}{\delta I} \frac{\partial I}{\partial \mathbf{R}} \tag{4.42}$$

Here, our focus is on the second factor $\frac{\partial I}{\partial \mathbf{R}}$, as the first does not provide any information about the physics of the system; instead, it pertains to the dependence of our estimator on the parameters denoted as T and k in the main text.

By expanding $|\Psi\rangle$ on the Hamiltonian eigenbasis we get:

$$I(\omega, \mathbf{R}) = \sum_i |c_i(\mathbf{R})|^2 \delta(\phi_i(\mathbf{R}) - \omega) \tag{4.43}$$

Where, as expressed in the main text introduction (Sec. 4.2.1), $\phi_i = f(E_i)$.

Differentiating and accounting for the finite precision of the quantum register grid we obtain:

$$\frac{\partial I}{\partial \mathbf{R}} = \sum_i \frac{\partial |c_i|^2}{\partial \mathbf{R}} \frac{\partial P}{\partial \phi_i} \frac{\partial \phi_i}{\partial \mathbf{R}} \tag{4.44}$$

Where P is the parent distribution kernel defined in Eq.4.4 of the main text and i is a label running over the electronic eigenstates.

In our approximation, i.e. Eq. 4.14, the previous Eq. reduces to:

$$\frac{\partial I}{\partial \mathbf{R}} \approx \frac{\partial |c_i|^2}{\partial \mathbf{R}} \frac{\partial P}{\partial \phi_i} \frac{\partial \phi_i}{\partial \mathbf{R}} \tag{4.45}$$

With $|i\rangle$ being the only eigenstate contributing to the probability P.

Again we focus on the terms which bring the molecular physics into the algorithm. As such we do not comment the term $\frac{\partial P}{\partial \phi_i}$ which can be obtained analytically from Eq. 4.4.

Applying the chain rule to the eigenphase derivative we can easily notice the emergence of the true molecular energy derivatives: $\frac{\partial \phi_i}{\partial \mathbf{R}} = \frac{\partial \phi_i}{\partial f} \frac{\partial f}{\partial E_i} \frac{dE_i}{d\mathbf{R}}$. The dependence on f is a purely algorithmic choice and will not be considered here. On the other hand, following Ref. [34] we can directly relate the force calculated w.r.t. the i-th eigenstate to molecular integrals derivatives as:

$$\frac{dE_i}{d\mathbf{R}} = \left(\sum_{pq} \gamma_{pq} \frac{dh_{pq}}{d\mathbf{R}} + \sum_{pqrs} \Gamma_{pqrs} \frac{dg_{pqrs}}{d\mathbf{R}} \right) - \left(\sum_{pqm} \gamma_{pq} h_{mq} \frac{dS_{mp}}{d\mathbf{R}} + 2 \sum_{pqrst} \Gamma_{pqrs} g_{tqrs} \frac{dS_{tp}}{d\mathbf{R}} \right) \tag{4.46}$$

Where γ_{pq} and Γ_{pqrs} are, respectively, the one- and two-electrons reduced density matrices[87] for the i-th eigenstate, h and g are the one- and two-electrons integrals representing all the interactions present in the molecular Hamiltonian (i.e., electronic motion, electron-nuclear interactions and electron-electron interactions). All indexes of the summation run over the spin-orbitals. Finally, S is the basis overlap matrix.

To obtain a complete expression we now consider the evaluation of $\frac{\partial |c_u|^2}{\partial \mathbf{R}}$. We consider a real-valued wavefunction (without any loss of generality as long as we are not neglecting the presence of external magnetic fields) for which we can write the first-order response of the wavefunction w.r.t. a perturbation (in our case being a nuclear displacement) as [88]:

$$\frac{\partial c_\mu}{\partial \mathbf{R}} = -\sum_{\nu} \mathcal{G}^{-1}_{\mu\nu} \frac{\partial \mathcal{F}_\nu}{\partial \mathbf{R}} \tag{4.47}$$

Where $\mathcal{G}$ and $\mathcal{F}$ are, respectively, the electronic Hessian and the electronic gradient defined as:

$$\mathcal{G}_{\mu\nu} = \frac{\partial^2 E}{\partial c_\mu \partial c_\nu} = 2(\langle \mu | H | \nu \rangle - E_{GS} \langle \mu | \nu \rangle) \tag{4.48}$$

$$\mathcal{F}_\mu = \frac{\partial E}{\partial c_\mu} = 2 \langle \mu | P H | \Psi \rangle \tag{4.49}$$

Previous Eqs. hold if we are considering derivatives of the ground state coefficients. Moreover, we point out that $|\mu\rangle$ and $|\nu\rangle$ are N-electrons Slater determinants. The expressions on the Hamiltonian eigenbasis can be recovered considering that the two sets of coefficients are related by the basis change $\mathbf{C}_{\{|u\rangle\}} = \mathbf{V}^\dagger \mathbf{C}_{\{|\mu\rangle\}} \mathbf{V}$ (where $\mathbf{V}$ is the Hamiltonian diagonalizing matrix). Finally, in Eq. 4.49 we have introduced the projection operator $P = \mathbf{1} - |\Psi\rangle\langle\Psi|$.

From Eq. 4.47 and $\frac{\partial |c_u|^2}{\partial \mathbf{R}} = 2c_u \frac{\partial c_u}{\partial \mathbf{R}}$ we can attribute a precise physical meaning to the overlap derivative term: in analogy with *Hooke's law* the wavefunction relaxation is proportional to the "force" $-\frac{\partial \mathcal{F}_\nu}{\partial \mathbf{R}}$ and inversely proportional to the "force constant" $\mathcal{G}$. Indeed, in absence of external perturbation (i.e. nuclear displacement) the electronic gradient is null and the wavefunction is stable.

Finally, adding the contribution of Eq. 4.46 we can reconstruct the meaning of Eq. 4.45. In essence, Eq. 4.45 tells us that the measurement distribution induced by the QPE circuit varies as nuclei are displaced by effect of the variations of the electronic energies (Eq. 4.46) and of the wavefunction coefficients (Eq. 4.47). These contributions are singled out and modulated by the functional $\mathcal{S}$, other algorithmic choices (e.g. $f \propto \cos^{-1}$ when employing a qubitization[89] based QPE), and the number of readout qubits entering the quantity $\frac{\partial P}{\partial \phi_i}$.

Appendix B: Effect of sampling noise

In Fig. 4.11, we illustrate the effect of sampling on the geometry optimization of H_3^+ in its singlet ground state using the strategy defined in the main text, see Sec. 4.2 and Sec. 4.3.

As we can observe, both in terms of energy error (ΔE, Fig. 4.11a) and bond length error (Δr, Fig. 4.11b), the estimate for the number of samples to be measured outlined in Fig. 4.7 is consistent with the findings reported here. Indeed, the initial state used for performing the geometry optimization is always the Hartree-Fock determinant, which, starting from the initial geometry (the same one used for the calculations presented in the main text), exhibits an overlap with the FCI solution $> 90\%$.

Interestingly, by looking at Fig. 4.11, we may be tempted to conclude that the generalized circular estimator variance does not actually depend on the number of readout qubits as very similar trends are observed w.r.t. the number of shots regardless of the number of readout qubits and also, the final errors w.r.t. the

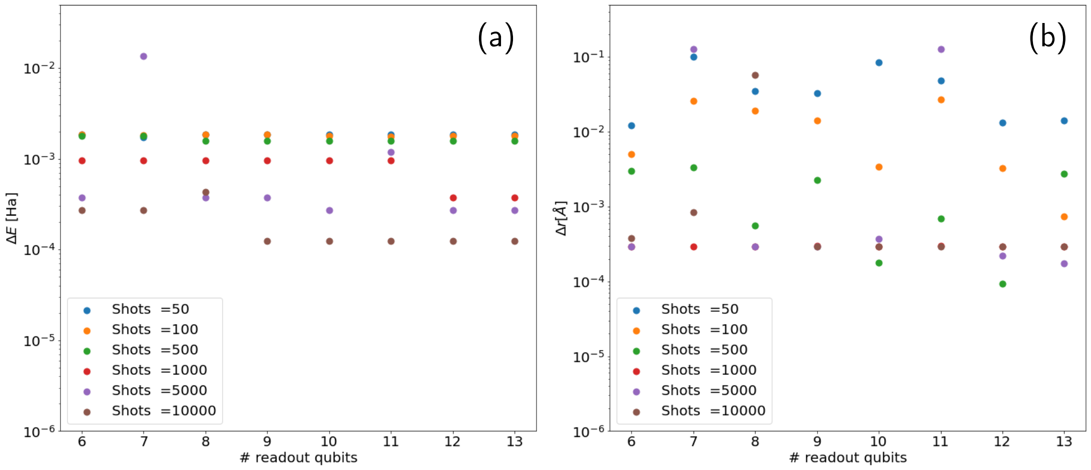


Figure 4.11: Ground state geometry optimization under sampling noise. (a) Energy error (ΔE) and bond length error (b) (Δr) w.r.t. the FCI optimized solution as a function of number of readout qubits varying the number of collected samples for reconstructing the parent distribution (shown with different colors). We remark that each point is an independent optimization run.

exact geometry look quite independent on t. As we have discussed in Sec.4.3.1, this is not the case. The results shown in Fig. 4.11 arise from the interplay of two effects: (i) the window sizes that we have used in these calculations shrink exponentially with the number of qubits as reported in Fig. 4.4; this implies that the number of bitstrings included in the circular average (for the calculations reported in Fig. 4.11) is always the same regardless of the number of readout qubits. Clearly, this approach works also with a lower number of qubits because the input state used allows to sample from a large number of bitstrings without adding information from spurious excited states. Moreover (ii) the results that we are showing do not refer to the evaluation of a single point energy but are obtained as a result of different independent optimization processes; to this extent the errors reported combine the errors on the function (and gradient) evaluations with the optimization process.

References

(1) Chang, C.; Deringer, V. L.; Katti, K. S.; Van Speybroeck, V.; Wolverton, C. M. *Nature Reviews Materials* **2023**, 1–5.

(2) Talia, D.; Trunfio, P.; Carretero, J.; Garcia-Blas, J. *Frontiers in Big Data* **2022**, *5*, 4.

(3) Mennucci, B.; Corni, S. *Nat. Rev. Chem.* **2019**, *3*, 315–330.

(4) Ciccotti, G.; Dellago, C.; Ferrario, M.; Hernández, E.; Tuckerman, M. *The European Physical Journal B* **2022**, *95*, 3.

(5) Di Felice, R.; Mayes, M. L.; Richard, R. M.; Williams-Young, D. B.; Chan, G. K.-L.; de Jong, W. A.; Govind, N.; Head-Gordon, M.; Hermes, M. R.; Kowalski, K., et al. *Journal of Chemical Theory and Computation* **2023**, *19*, 7056–7076.

(6) Ufimtsev, I. S.; Martinez, T. J. *Journal of Chemical Theory and Computation* **2008**, *4*, 222–231.

(7) Liu, F.; Luehr, N.; Kulik, H. J.; Martínez, T. J. *Journal of Chemical Theory and Computation* **2015**, *11*, 3131–3144.

(8) Seritan, S.; Bannwarth, C.; Fales, B. S.; Hohenstein, E. G.; Isborn, C. M.; Kokkila-Schumacher, S. I.; Li, X.; Liu, F.; Luehr, N.; Snyder Jr, J. W., et al. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2021**, *11*, e1494.

(9) Pederson, R.; Kozlowski, J.; Song, R.; Beall, J.; Ganahl, M.; Hauru, M.; Lewis, A. G.; Yao, Y.; Mallick, S. B.; Blum, V., et al. *Journal of Chemical Theory and Computation* **2022**, *19*, 25–32.

(10) Almlöf, J.; Taylor, P. R. *International Journal of Quantum Chemistry* **1985**, *27*, 743–768.

(11) Martis, E.; Radhakrishnan, R.; Badve, R. *Journal of Applied Pharmaceutical Science* **2011**, 02–10.

(12) Luo, S.; Li, T.; Wang, X.; Faizan, M.; Zhang, L. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2021**, *11*, e1489.

(13) Koch, H.; Jensen, H. J. A.; Jo/rgensen, P.; Helgaker, T.; Scuseria, G. E.; Schaefer III, H. F. *The Journal of Chemical Physics* **1990**, *92*, 4924–4940.

(14) Assaraf, R.; Caffarel, M. *The Journal of Chemical Physics* **2000**, *113*, 4028–4034.

(15) Wang, S.; Li, C.; Evangelista, F. A. *Journal of Chemical Theory and Computation* **2021**, *17*, 7666–7681.

(16) Baydin, A. G.; Pearlmutter, B. A.; Radul, A. A.; Siskind, J. M. *Journal of Marchine Learning Research* **2018**, *18*, 1–43.

(17) Battaglia, S. In *Quantum Chemistry in the Age of Machine Learning*; Elsevier: 2023, pp 577–616.

(18) Kasim, M. F.; Lehtola, S.; Vinko, S. M. *The Journal of Chemical Physics* **2022**, *156*.

(19) Tamayo-Mendoza, T.; Kreisbeck, C.; Lindh, R.; Aspuru-Guzik, A. *ACS Central Science* **2018**, *4*, 559–566.

(20) Bartlett, R. A.; Gay, D. M.; Phipps, E. T. In *Computational Science–ICCS 2006: 6th International Conference, Reading, UK, May 28-31, 2006, Proceedings, Part IV 6*, 2006, pp 525–532.

(21) Frank, S. A. *Frontiers in Ecology and Evolution* **2022**, *10*, 1010278.

(22) Xu, W.; Chen, X.; Coleman, T. F. *Journal of Computational Finance* **2016**, *19*.

(23) Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; Patrick, C. *Nat. Rev. Phys.* **2021**, 1–20.

(24) Zhu, S.; Hung, S.-H.; Chakrabarti, S.; Wu, X. In *Proceedings of the 41st ACM SIGPLAN Conference on Programming Language Design and Implementation*, 2020, pp 272–285.

(25) Fang, W.; Ying, M.; Wu, X. *arXiv preprint arXiv:2211.04507* **2022**.

(26) Matteo, O. D.; Izaac, J.; Bromley, T. R.; Hayes, A.; Lee, C.; Schuld, M.; Száva, A.; Roberts, C.; Killoran, N. *ACM Transactions on Quantum Computing* **2023**, *4*, 1–20.

(27) Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Ahmed, S.; Ajith, V.; Alam, M. S.; Alonso-Linaje, G.; AkashNarayanan, B.; Asadi, A., et al. *arXiv preprint arXiv:1811.04968* **2018**.

(28) Broughton, M.; Verdon, G.; McCourt, T.; Martinez, A. J.; Yoo, J. H.; Isakov, S. V.; Massey, P.; Halavati, R.; Niu, M. Y.; Zlokapa, A., et al. *arXiv preprint arXiv:2003.02989* **2020**.

(29) Hagan, M.; Wiebe, N. *Quantum* **2023**, *7*, 1181.

(30) Low, G. H.; Su, Y.; Tong, Y.; Tran, M. C. *PRX Quantum* **2023**, *4*, 020323.

(31) Babbush, R.; Huggins, W. J.; Berry, D. W.; Ung, S. F.; Zhao, A.; Reichman, D. R.; Neven, H.; Baczewski, A. D.; Lee, J. *Nat. Comm.* **2023**.

(32) Kang, C.; Bauman, N. P.; Krishnamoorthy, S.; Kowalski, K. *Journal of Chemical Theory and Computation* **2022**, *18*, 6567–6576.

(33) Steudtner, M.; Morley-Short, S.; Pol, W.; Sim, S.; Cortes, C. L.; Loipersberger, M.; Parrish, R. M.; Degroote, M.; Moll, N.; Santagati, R., et al. *arXiv preprint arXiv:2303.14118* **2023**.

(34) O'Brien, T. E.; Streif, M.; Rubin, N. C.; Santagati, R.; Su, Y.; Huggins, W. J.; Goings, J. J.; Moll, N.; Kyoseva, E.; Degroote, M., et al. *Physical Review Research* **2022**, *4*, 043210.

(35) Huggins, W. J.; Wan, K.; McClean, J.; O'Brien, T. E.; Wiebe, N.; Babbush, R. *Physical Review Letters* **2022**, *129*, 240501.

(36) Nielsen, M. A.; Chuang, I. L., *Quantum computation and quantum information*; Cambridge university press: 2010.

(37) Cruz, P. M.; Catarina, G.; Gautier, R.; Fernández-Rossier, J. *Quantum Science and Technology* **2020**, *5*, 044005.

(38) Lee, S.; Lee, J.; Zhai, H.; Tong, Y.; Dalzell, A. M.; Kumar, A.; Helms, P.; Gray, J.; Cui, Z.-H.; Liu, W., et al. *Nature Communications* **2023**, *14*, 1952.

(39) Fomichev, S.; Hejazi, K.; Zini, M. S.; Kiser, M.; Morales, J. F.; Casares, P. A. M.; Delgado, A.; Huh, J.; Voigt, A.-C.; Mueller, J. E., et al. *arXiv preprint arXiv:2310.18410* **2023**.

(40) Rubin, N. C.; Berry, D. W.; Malone, F. D.; White, A. F.; Khattar, T.; DePrince III, A. E.; Sicolo, S.; Küehn, M.; Kaicher, M.; Lee, J., et al. *PRX Quantum* **2023**, *4*, 040303.

(41) Zini, M. S.; Delgado, A.; dos Reis, R.; Casares, P. A. M.; Mueller, J. E.; Voigt, A.-C.; Arrazola, J. M. *Quantum* **2023**, *7*, 1049.

(42) Cleve, R.; Ekert, A.; Macchiavello, C.; Mosca, M. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **1998**, *454*, 339–354.

(43) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. *nature* **1986**, *323*, 533–536.

(44) Arrazola, J. M.; Jahangiri, S.; Delgado, A.; Ceroni, J.; Izaac, J.; Száva, A.; Azad, U.; Lang, R. A.; Niu, Z.; Di Matteo, O., et al. *arXiv preprint arXiv:2111.09967* **2021**.

(45) Li, J.; Yang, X.; Peng, X.; Sun, C.-P. *Physical review letters* **2017**, *118*, 150503.

(46) Mitarai, K.; Negoro, M.; Kitagawa, M.; Fujii, K. *Physical Review A* **2018**, *98*, 032309.

(47) Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; Killoran, N. *Physical Review A* **2019**, *99*, 032331.

(48) Izmaylov, A. F.; Lang, R. A.; Yen, T.-C. *Physical Review A* **2021**, *104*, 062443.

(49) Banchi, L.; Crooks, G. E. *Quantum* **2021**, *5*, 386.

(50) Kyriienko, O.; Elfving, V. E. *Physical Review A* **2021**, *104*, 052417.

(51) Wierichs, D.; Izaac, J.; Wang, C.; Lin, C. Y.-Y. *Quantum* **2022**, *6*, 677.

(52) Wiersema, R.; Lewis, D.; Wierichs, D.; Carrasquilla, J.; Killoran, N. *arXiv preprint arXiv:2303.11355* **2023**.

(53) Jang, E.; Gu, S.; Poole, B. Categorical Reparameterization with Gumbel-Softmax, 2017.

(54) Mardia, K. V.; Jupp, P. E., *Directional statistics*; John Wiley & Sons: 2009.

(55) Angeli, C.; Cimiraglia, R.; Dallo, F.; Guareschi, R.; Tenti, L. *Journal of Chemical Education* **2013**, *90*, 1639–1644.

(56) Zheng, A.; Casari, A., *Feature engineering for machine learning: principles and techniques for data scientists*; " O'Reilly Media, Inc.": 2018.

(57) Moore, A. J.; Wang, Y.; Hu, Z.; Kais, S.; Weiner, A. M. *New Journal of Physics* **2021**, *23*, 113027.

(58) Eriksson, J.; Ollila, E.; Koivunen, V. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp 3565–3568.

(59) Rendon, G.; Izubuchi, T.; Kikuchi, Y. *Physical Review D* **2022**, *106*, 034503.

(60) Feynman, R. P. *Physical Review* **1939**, *56*, 340.

(61) Knill, E.; Ortiz, G.; Somma, R. D. *Physical Review A* **2007**, *75*, 012328.

(62) Gilyén, A.; Arunachalam, S.; Wiebe, N. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2019, pp 1425–1444.

(63) O'Brien, T. E.; Senjean, B.; Sagastizabal, R.; Bonet-Monroig, X.; Dutkiewicz, A.; Buda, F.; DiCarlo, L.; Visscher, L. *npj Quantum Information* **2019**, *5*, 1–12.

(64) Yi, C.; Zhou, C.; Takahashi, J. *arXiv preprint arXiv:2306.07008* **2023**.

(65) Brassard, G.; Hoyer, P.; Mosca, M.; Tapp, A. *Contemporary Mathematics* **2002**, *305*, 53–74.

(66) Bowles, J.; Wierichs, D.; Park, C.-Y. *arXiv preprint arXiv:2306.14962* **2023**.

(67) Sugisaki, K.; Wakimoto, H.; Toyota, K.; Sato, K.; Shiomi, D.; Takui, T. *The Journal of Physical Chemistry Letters* **2022**, *13*, 11105–11111.

(68) https://github.com/davidecast/DiffQPE Accessed by date: 07/01/2023.

(69) Turney, J. M.; Simmonett, A. C.; Parrish, R. M.; Hohenstein, E. G.; Evangelista, F. A.; Fermann, J. T.; Mintz, B. J.; Burns, L. A.; Wilke, J. J.; Abrams, M. L., et al. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012**, *2*, 556–565.

(70) Sun, Q.; Berkelbach, T. C.; Blunt, N. S.; Booth, G. H.; Guo, S.; Li, Z.; Liu, J.; McClain, J. D.; Sayfutyarova, E. R.; Sharma, S., et al. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2018**, *8*, e1340.

(71) Jordan, P.; von Neumann, J.; Wigner, E. P. In *The collected works of Eugene Paul Wigner: Part A: The scientific papers*; Springer: 1993, pp 298–333.

(72) Nocedal, J.; Wright, S. J., *Numerical optimization*; Springer: 1999.

(73) Tennyson, J. *Reports on Progress in Physics* **1995**, *58*, 421.

(74) Polyansky, O. L.; Alijah, A.; Zobov, N. F.; Mizus, I. I.; Ovsyannikov, R. I.; Tennyson, J.; Lodi, L.; Szidarovszky, T.; Csaszar, A. G. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **2012**, *370*, 5014–5027.

(75) McCall, B. J.; Oka, T. *Science* **2000**, *287*, 1941–1942.

(76) Aguado, A.; Roncero, O.; Tablero, C.; Sanz, C.; Paniagua, M. *The Journal of Chemical Physics* **2000**, *112*, 1240–1254.

(77) Sanz, C.; Roncero, O.; Tablero, C.; Aguado, A.; Paniagua, M. *The Journal of Chemical Physics* **2001**, *114*, 2182–2191.

(78) Krylov, A. I.; Sherrill, C. D.; Head-Gordon, M. *The Journal of Chemical Physics* **2000**, *113*, 6509–6527.

(79) Angeli, C.; Borini, S.; Ferrighi, L.; Cimiraglia, R. *Journal of Molecular Structure: THEOCHEM* **2005**, *718*, 55–69.

(80) Clouthier, D.; Ramsay, D. *Annual Review of Physical Chemistry* **1983**, *34*, 31–58.

(81) Tables, J. T. *J. Phys. Chem. Ref. Data* **1985**, *14*.

(82) Schuetz, C.; Frenklach, M.; Kollias, A.; Lester Jr, W. *The Journal of Chemical Physics* **2003**, *119*, 9386–9392.

(83) Takagi, K.; Oka, T. *Journal of the Physical Society of Japan* **1963**, *18*, 1174–1180.

(84) Ni, H.; Li, H.; Ying, L. *arXiv preprint arXiv:2302.02454* **2023**.

(85) Ding, Z.; Lin, L. *PRX Quantum* **2023**, *4*, 020331.

(86) Delgado, A.; Casares, P. A.; Dos Reis, R.; Zini, M. S.; Campos, R.; Cruz-Hernández, N.; Voigt, A.-C.; Lowe, A.; Jahangiri, S.; Martin-Delgado, M. A., et al. *Physical Review A* **2022**, *106*, 032428.

(87) Helgaker, T.; Jorgensen, P.; Olsen, J., *Molecular electronic-structure theory*; John Wiley & Sons: 2013.

(88) Helgaker, T.; Jørgensen, P. *Advances in quantum chemistry* **1988**, *19*, 183–245.

(89) Low, G. H.; Chuang, I. L. *Quantum* **2019**, *3*, 163.

Chapter 5

Conclusions

This thesis work is inscribed within the field of quantum computation aimed at the simulation of molecular systems. Its organization reflects a progression of the algorithms developed to describe chemical systems: starting with a method for calculating ground state electronic energies in gas-phase proceeding to more complex systems and molecular properties.

The variety of tools adopted and possible scenarios to be considered also reflects the current state of hardware development. In particular, the uncertainty regarding the type of physical implementation, the type of architecture and (potentially) also the type of programming paradigm means that the development of useful and effective algorithms does not only involve efficient scaling but also flexibility due to the variations that this type of technology will inevitably undergo.

The insights developed over the past decades by the scientific community in studying the electronic structure problem suggest that there is still much room for improvement and extension in the field of quantum information.

In particular, in Chapter 2, by addressing the electronic structure problem with optimal control tools, we introduced a parameterization that explicitly acts by introducing time-varying effective nuclear charges. What is the effect of using localized basis functions in this scenario?

Similarly, Chapter 3 presents an extension of VQE for condensed-phase molecular systems. The analysis presented in this thesis focuses on the calculation of the electronic structure of the ground state. What are the best strategies, taking into account different solvent relaxation regimes, for the calculation of excited states properties in solution? More generally, the possibility of including a solvent description is a necessary step to bring these techniques closer to capturing the complexity of chemical phenomena. Thus, the development of quantum multiscale methods is, in general, a possible continuation of this thesis work.

The last chapter explored the possibilities offered by extending quantum phase estimation to an automatic differentiation paradigm and highlighting its possible limitations. Clearly, an algorithm capable of scaling backpropagation on universal circuits would be a game-changer beyond the work proposed in this thesis, but it would inevitably impact the cost of the proposed routine. Nevertheless, a future perspective for this work would be to devise more efficient methods for calculating gradients by reducing the number of circuits to independently measure. This, coupled with the development of alternative smoothing procedures and methods such as those explored in Chapter 2, could represent a significant step forward in the development of useful quantum algorithms on an industrial scale.

List of Publications

Publications related to this thesis:

1. Castaldo, D.; Rosa, M.; Corni, S. Fast-forwarding molecular ground state preparation with optimal control on analog quantum simulators. *in preparation*, **2023**.
2. Castaldo, D., Jahangiri, S., Delgado, A., & Corni, S. (2022). Quantum simulation of molecules in solution. Journal of Chemical Theory and Computation, 18(12), 7457-7469.
3. Castaldo, D.; Jahangiri, S.; Arrazola, J.M.; Corni, S. A fully differentiable quantum phase estimation algorithm. *in preparation*, **2023**.

Other publications:

1. Castaldo, D., Rosa, M., Corni, S. (2021). Quantum optimal control with quantum computers: A hybrid algorithm featuring machine learning optimization. Physical Review A, 103(2), 022613.
2. Pravatto, P., Castaldo, D., Gallina, F., Fresch, B., Corni, S., Moro, G. J. (2021). Quantum computing for classical problems: variational quantum eigensolver for activated processes. New Journal of Physics, 23(12), 123045.
3. Vanzan, M., Gil, G., Castaldo, D., Nordlander, P., Corni, S. (2023). Energy transfer to molecular adsorbates by transient hot electron spillover. Nano Letters, 23(7), 2719-2725.