

Universidade do Minho
Escola de Engenharia

Mafalda Ramôa

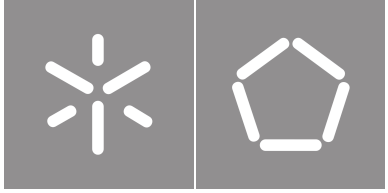
On the Quest for Shallow, Accurate, and Cost-Efficient Ansätze for Variational Quantum Algorithms

February, 2026

On the Quest for Shallow, Accurate, and Cost-Efficient Ansätze for Variational Quantum Algorithms
Mafalda Ramôa

UMinho | 2026

REPÚBLICA PORTUGUESA
CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR
fct Fundação para a Ciência e a Tecnologia



Universidade do Minho
Escola de Engenharia

Mafalda Ramôa

On the Quest for Shallow, Accurate, and Cost-Efficient Ansätze for Variational Quantum Algorithms

Doctorate Thesis

Work developed under the supervision of:
Sophia E. Economou
Luís P. Santos
Ernesto F. Galvão

February, 2026

COPYRIGHT AND TERMS OF USE OF THIS WORK BY A THIRD PARTY

This is academic work that can be used by third parties as long as internationally accepted rules and good practices regarding copyright and related rights are respected.

Accordingly, this work may be used under the license provided below.

If the user needs permission to make use of the work under conditions not provided for in the indicated licensing, they should contact the author through the RepositoriUM of Universidade do Minho.

License granted to the users of this work



Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International
CC BY-NC-SA 4.0

https://creativecommons.org/licenses/by-nc-sa/4.0/deed.en

This document was produced with the (pdf/Xe/Lua)LaTeX typesetting system and the NOVAthesis template (v7.8.0) .

J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/main/template.pdf.

Acknowledgements

Above all, I would like to thank my advisors for all the support throughout my PhD. The guidance of Professor Sophia Economou has profoundly impacted not only this project but also who I aim to be as a researcher. Her expertise, constructive criticism, and belief in my work have shaped not only my experience but also my perception of academia. This PhD would also not have been possible without Professor Luis Paulo Santos, to whom I am indebted for his constant support and valuable feedback, and Professor Ernesto Fagundes Galvão, who was always available to share his knowledge and believed in this project from the start. Finally, I would like to thank Professor Edwin Barnes, who, despite not being an official advisor, played the role of a great one, and Professor Nicholas Mayhall, for all his guidance and support.

While it is not possible to individually name everyone who positively impacted these four years, I have to express a heartfelt thank you to those without whom this period of my life would not have been the same. To Bennett, for always having my back. To my twin sister, for how much (good and bad) we have been through together. To everyone in my family who was there for me in spite of the ocean that research placed between us. To Inês, Juliana, Diego, Vítor, and Wagner, for so much shared laughter. To Stephanie, for our adventures. To Filipa, for all we learned together. To all my colleagues at the University of Minho, for all the lively lunchtime hours. To the group at the International Iberian Nanotechnology Laboratory, especially Manisha and Anita. To Mitra, for a beautiful friendship. To the group at Virginia Tech, especially Bela, Karunya, Yanzhu, Hisham, Bharath, Vagelis, and Peter, for how quickly they went from colleagues to friends. To my roommates, especially Sara, Lwam, and Sara, for their kindness and warmth. To Susana Pinto, for the start of my now fifteen-year-long journey in physics.

Finally, I thank Virginia Tech, the Virginia Tech Center for Quantum Information Science and Engineering (VTQ), the International Iberian Nanotechnology Laboratory (INL), the University of Minho, the Institute for Systems and Computer Engineering, Technology and Science (INESC TEC), and the High-Assurance Software Laboratory (HASLab) for hosting my research activities. I acknowledge support from Wellcome Leap as part of the Quantum for Bio Program, from FLAD (Luso-American Development Foundation), and from FCT (Fundação para a Ciência e a Tecnologia), under PhD research scholarship 2022.12333.BD. This work is in part financed by National Funds through the Portuguese funding agency, FCT, within project LA/P/0063/2020. DOI 10.54499/LA/P/0063/2020 | https://doi.org/10.54499/LA/P/0063/2020.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the Universidade do Minho.

_____, _____

(Place) (Date)

_____

(Mafalda Ramôa)

Resumo

Sobre a Procura por Ansätze Pouco Profundos, Exatos e Eficientes para Algoritmos Variacionais Quânticos

Desde a primeira vez que computação quântica foi concetualizada por Feynman, simulação quântica tem estado na vanguarda das possíveis aplicações de interesse. Os custos proibitivos de simular sistemas quânticos com computadores clássicos são uma forte motivação para a investigação em computação quântica. Ao longo da última década, tem-se observado um progresso notável no desenvolvimento de hardware quântico. No entanto, os obstáculos entre nós e a era de vantagem quântica útil mantêm-se essencialmente os mesmos: A informação quântica é frágil, e protegê-la é um desafio.

Visto que os dispositivos quânticos atuais são ruidosos e têm um número limitado de qubits, desenvolver algoritmos que possam ser executados neles é uma área de investigação importante. Por resultarem em circuitos de preparação de estado (ansätze) pouco profundos e treináveis, algoritmos quânticos variacionais (VQAs) adaptativos são uma opção promissora. O primeiro algoritmo nesta classe, o ADAPT-VQE, pode ser utilizado para problemas de estrutura eletrónica, otimização combinatória, matéria condensada, sistemas periódicos no tempo, entre outros.

Nesta tese, são propostas várias melhorias para VQAs adaptativos, com o objetivo de diminuir os custos e aumentar a viabilidade a curto prazo. Propõe-se um otimizador quasi-Newton para VQAs adaptativos que reduz os custos de medição necessários para implementar estes algoritmos em hardware quântico, bem como o tempo necessário para os simular classicamente. Propõe-se ainda um novo conjunto de operadores—'operatores de intercâmbio acoplado"(CEOs)—que resulta no menor número de portas quânticas e na menor profundidade de circuito entre todas as propostas na literatura. Combina-se vários desenvolvimentos-chave e mostra-se uma redução dramática dos recursos computacionais quânticos necessários em comparação com as primeiras versões do algoritmo ou com ansätze estáticos para VQE. Adicionalmente, propõe-se uma versão codesenhada do ADAPT-VQE (Co-ADAPT-VQE) onde o hardware quântico é tido em consideração na construção do ansatz. Este paradigma pode ser utilizado para otimizar circuitos de preparação de estado em qualquer dispositivo, colmatando limitações como conetividade limitada, tempos de coerência curtos, e erros nas portas quânticas. Finalmente, propomos estratégia para mitigar 'vales de gradientes', um problema de convergência específico ao ADAPT-VQE.

Palavras-chave: Computação Quântica · Simulação Quântica · Eigensolver Variacional Quântico

Abstract

On the Quest for Shallow, Accurate, and Cost-Efficient Ansätze for Variational Quantum Algorithms

From the very first time quantum computing was conceptualized by Feynman, quantum simulation has been at the forefront of the realm of applications. The prohibitive costs of simulating quantum-mechanical systems using classical computers are a strong motivation for quantum computing research. Over the past decade, we have witnessed remarkable progress in the development of quantum hardware. Yet, the obstacles between us and the era of useful quantum advantage remain largely the same: quantum information is fragile, and protecting it is a challenge for both theorists and experimentalists.

As current quantum devices are noisy and have a limited number of qubits, the development of algorithms that can be executed on them has become a significant area of research. Due to their shallow and trainable state preparation circuits (ansätze), adaptive variational quantum algorithms (VQAs) are strong contenders in the quest for quantum advantage with near-term quantum computers. While the first such algorithm, ADAPT-VQE, was proposed for finding molecular ground states, the framework has since been generalized to combinatorial optimization, condensed matter problems, time-periodic systems, and more.

In this thesis, we propose multiple improvements to adaptive VQAs, aimed at decreasing costs and improving near-term viability (by reducing circuit depth). We implement an improved quasi-Newton optimization protocol specifically tailored to adaptive VQAs and show that this protocol significantly reduces the measurement costs of implementing adaptive VQAs on quantum hardware, as well as the runtime of their classical simulation. We propose a novel operator pool–which we term the Coupled Exchange Operator (CEO) pool–that results in the lowest gate counts/depths across all proposals in the literature. We combine CEO-ADAPT-VQE with other key improvements and show a dramatic reduction of quantum resource requirements compared to the early versions of the algorithm and to the most widely used static VQE ansätze. Additionally, we propose a co-designed variant of ADAPT-VQE (Co-ADAPT-VQE) where the quantum hardware is taken into account in the construction of the ansatz. This framework can be readily used to optimize state preparation circuits for any device, addressing shortcomings such as limited connectivity, short coherence times, and variable gate errors. Finally, we propose strategies to mitigate gradient troughs, an ADAPT-VQE-specific convergence problem.

Keywords: Quantum Computing · Quantum Simulation · Variational Quantum Eigensolver

Contents

List of Figures **x**

List of Tables **xviii**

List of Algorithms **xix**

Acronyms **xx**

1 Introduction **1**

1.1 Motivation 1

1.2 Thesis Structure and Overview 3

1.3 List of Publications 4

2 Literature Review **5**

2.1 Quantum Algorithms for Optimization in the FTQC Era 5

2.1.1 Adiabatic Quantum Computing and Quantum Annealing 6

2.1.2 Quantum Phase Estimation 7

2.1.3 Grover's Algorithm 9

2.2 Quantum Algorithms for Optimization in the NISQ Era 10

2.2.1 VQAs and VQE 10

2.2.2 UCC 13

2.2.3 ADAPT-VQE 15

2.2.4 L-VQE 21

2.2.5 E-VQE 22

2.2.6 VAns 25

2.2.7 DISCO-VQE 27

2.2.8 QNP 29

2.2.9 tUPS 29

2.2.10 QAOA 31

2.2.11 ADAPT-QAOA 34

3 Recycling the Hessian: An ADAPT-VQE-Tailored Quasi-Newton Optimizer 36

3.1 Motivation 36
3.2 Analysis of ADAPT-VQE Measurement Costs 37
3.3 Gradient-Based Optimization Methods 39
3.3.1 Gradient Descent 39
3.3.2 Newton's Method 39
3.3.3 Quasi-Newton Methods 40
3.3.4 Convergence Rates 41
3.4 BFGS Algorithm for ADAPT-VQE 42
3.5 Results 44
3.5.1 Measurement Costs 44
3.5.2 Distance to the Exact Hessian 47
3.5.3 Evolution of the Optimization 49
3.6 Discussion 52

4 CEO-ADAPT-VQE: Improving the Hardware-Efficiency of Problem-Tailored Operator Pools 54

4.1 Motivation 54
4.2 QE Circuit Implementation 55
4.3 CEO-ADAPT-VQE 59
4.3.1 Multiple Variational Parameters MVP-CEOs 60
4.3.2 One Variational Parameter (OVP)-CEOs 61
4.3.3 The CEO-ADAPT-VQE Algorithm 64
4.4 Results 67
4.4.1 Convergence Plots: CEO- vs QEB- vs Qubit-ADAPT-VQE 68
4.4.2 Comparison with UCCSD-VQE through Bond Dissociation Curves 71
4.4.3 ADAPT-VQE Evolution and State of the Art 73
4.4.4 Comparison with Fixed-Structure Ansätze 77
4.5 Discussion 79

5 Co-ADAPT-VQE: Co-Designing State Preparation Circuits 82

5.1 Motivation 82
5.2 Excitation Circuits in ATA and LNN Connectivities 83
5.3 Co-ADAPT-VQE 86
5.4 Using Co-ADAPT-VQE to Minimize Entangling Gate Counts for LNN Connectivity . . . 87
5.5 Results 93
5.6 Discussion 98

6 Overcoming Gradient Troughs: Exploring Alternative Operator Positions **100**
6.1 Motivation . . . 100
6.2 Example of a Gradient Trough . . . 101
6.3 Detecting Gradient Troughs . . . 101
6.4 Mitigating Gradient Troughs . . . 104
6.5 Measurement Cost Analysis . . . 107
6.6 Results . . . 109
6.7 Discussion . . . 111

7 Conclusions and Future Work **114**

Appendices

A Evaluating Gradients on Quantum Hardware **116**

B BFGS Optimizer **117**
B.1 Line Search Subroutine . . . 117
B.2 BFGS Update Rule . . . 118
B.3 Properties of BFGS . . . 118

C Hessian Recycling for Other Molecules **120**

D Hessian Heatmaps for Other Iterations **122**

E Evolution of the 50th QEB-ADAPT-VQE Iteration for H6 **124**

F Variants of CEO-ADAPT-VQE **126**

G Combining CEO-ADAPT-VQE with Transversal Proposals **130**

H Orbital Optimization **133**

I Hamiltonian Grouping **135**

J Generalized Gradient Formula **137**

K Relation Between Gradient Troughs and Gradients Across the Ansatz **139**

L Co-ADAPT-VQE for ATA connectivity **141**

M Co-ADAPT-VQE for Higher Accuracy **143**

N Penalty Scaling **145**

O Results for LiH **147**

Bibliography **149**

List of Figures

2.1 Flowchart for a Variational Quantum Algorithm (VQA). 11

2.2 Flowchart for the Adaptive Derivative-Assembled Problem-Tailored (ADAPT)-Variational Quantum Eigensolver (VQE) algorithm. 16

3.1 Comparison of the error (top row), total number of line searches (middle row), and total number of function evaluations (bottom row) per iteration of the ADAPT-VQE algorithm with and without Hessian recycling for H_6. We consider the Qubit Excitation (QE) pool for (a) equilibrium and (b) stretched geometries, as well as (c) the qubit pool for a stretched geometry. The red curve is often not visible in the error plots due to it lying below the blue one. The region shaded blue is the region of chemical accuracy, defined as an error below 1 kcal/mol. . 45

3.2 Heatmaps showing the difference between the initial approximate inverse Hessian in the optimization process and the initial exact inverse Hessian, with (bottom row) and without (top row) Hessian recycling, for the 50th iteration of Qubit-Excitation-Based (QEB)-ADAPT-VQE. The plots show the element-wise difference between these two matrices. Due to it being strongly correlated and the hardest to simulate, here we consider the test molecule H_6 at different interatomic distances. 48

3.3 Frobenius distance between the initial approximate inverse Hessian in the optimization process $H_0^{(\text{opt})}$ and the initial exact inverse Hessian $H_0^{(\text{exact})}$, with and without Hessian recycling, for the first hundred QEB-ADAPT-VQE iterations. We consider two cases: one for which the Hessian recycling works particularly well (H_6 at 1 Å) and one for which it does not (H_6 at 3 Å). 49

3.4 Frobenius distance between the exact initial $H_0^{(\text{exact})}$ and final $H_{\text{final}}^{(\text{exact})}$ inverse Hessians, for the first hundred QEB-ADAPT-VQE iterations. The systems are the same as those from Fig. 3.3. 49

3.5 Evolution of the 75th QEB-ADAPT-VQE optimization with and without Hessian recycling, for the same two systems as in Fig. 3.3: H_6 at 1 Å (top row) and 3 Å (bottom row). The panels show (left to right) the error; the distance between the quasi-Newton $p_k^{(\mathrm{QN})}$ and Newton $p_k^{(\mathrm{N})}$ directions; and the distance between the last row of the approximate inverse Hessian $H^{(\mathrm{opt})}$ and the exact inverse Hessian $H^{(\mathrm{exact})}$. 50

3.6 Relevant quantities for the study of the convergence rate of the 75th QEB-ADAPT-VQE optimization with and without Hessian recycling, for the same two systems as in Fig. 3.3: H_6 at 1 Å (top row) and 3 Å (bottom row). (a) The distance between the approximate and the exact Hessian along the search direction, (b) the step size α at the end of the line search, and (c) the ratio between successive errors are shown. 51

4.1 Circuit implementation of the QE evolution $U_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped. While the sign is necessary for the circuit to correspond exactly to this operator, it becomes irrelevant when θ is optimized variationally. 58

4.2 Circuit implementation of the QE evolution $U_{\alpha_2\beta_1\to\alpha_1\beta_2}$. The circuit for $U_{\alpha_1\beta_2\to\alpha_2\beta_1}$ is identical, but with the rotation angle flipped. 58

4.3 Explicit implementation of the qubit excitation evolution $U_{\alpha_1\beta_1\to\alpha_2\beta_2}$. 58

4.4 Circuit implementation of the unitary generated by any linear combination of Pauli strings in a QE, $e^{\frac{i}{8}(\theta_0 XXXY+\theta_1 XXYX+\theta_2 YXYY+\theta_3 YXXX+\theta_4 YYXY+\theta_5 YYYX+\theta_6 XYYY+\theta_7 XYXX)}$. 61

4.5 Circuit implementation of $U^{(OneVariationalParameter(OVP)-CEO,+)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta)$. 63

4.6 Circuit implementation of $U^{(OVP-CEO,-)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta)$. 63

4.7 Decomposition of a two-control rotation. Any controlled gate can be used, as long as it imposes on the target (qubit 0) a reflection with respect to the Y axis, so that the sense of the rotation is inverted if the control is in state $|1\rangle$. We further note that the roles of the control qubits can be interchanged; this particular choice was made so that one controlled gate will be eliminated via the identity in Fig. 4.8. 63

4.8 Equivalence used to reduce the CNOT count. 63

4.9 Explicit implementation of the circuit in Fig. 4.5 using only CNOTs and single-qubit gates. 64

4.10 Convergence of the Coupled Exchange Operator (CEO)-, QEB-, and Qubit-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at bond distances close to equilibrium. The error is plotted against the iteration number (top), parameter count (middle), and CNOT count (bottom). The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). All algorithms terminate when the energy change falls below 10^{-10} Hartree. 69

4.11 Convergence of the CEO-, QEB-, and Qubit-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at stretched bond distances. The error is plotted against the iteration number (top), parameter count (middle) and CNOT count (bottom). The termination criterion is the same as in Fig. 4.10. 70

4.12 Convergence of the CEO-, QEB-, and Qubit-ADAPT-VQE algorithms for the triangular H_6 molecule at interatomic distance 2 Å. This system is particularly challenging to simulate due to corresponding to a spin-frustrated lattice. The error is plotted against the iteration number (top left), parameter count (top right), CNOT count (bottom left), and CNOT depth (bottom right). The termination criterion is the same as in Fig. 4.10. 71

4.13 Energy, error and parameter count of CEO-ADAPT-VQE against the bond distance for **a** LiH, **b** linear H_6, **c** BeH_2. Four different values of the gradient convergence threshold ϵ are considered: 10^{-k} for $k \in \{2, 3, 4, 5\}$. The Hartree-Fock, (untrotterized) Unitary Coupled Cluster Singles and Doubles (UCCSD) and Full Configuration Interaction (FCI) solutions are also included for comparison. The UCCSD ansatz includes all occupied-to-virtual single and double excitations. The corresponding parameter count is fixed at 92, 117, and 204 for the three molecules in the order that they appear. 72

4.14 Convergence of the CEO- and Generalized Singles and Doubles (GSD)-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at various bond distances. The symbol * signals that the CEO-ADAPT-VQE algorithm represents the state of the art, being implemented *in tandem* with other recent proposals: Tiling Efficient Trial circuits with Rotations Implemented Simultaneously (TETRIS) [10], optimized gradient measurements [9] and Hessian recycling (Ch. 3). In contrast, GSD-ADAPT-VQE follows the original protocol with a vanilla measurement strategy, representing the state of the art at the time ADAPT-VQE was first proposed [72]. The UCCSD ansatz is also included as a reference point. The corresponding circuit was implemented with one Trotter step and lexical ordering (with the excitations' source/target orbital indices as primary/secondary ordering criteria, respectively). The error is plotted against the iteration number, CNOT count, and CNOT depth. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively. 74

4.15 Convergence of the CEO- and GSD-ADAPT-VQE algorithms against the parameter count of the ansatz and measurement costs, which are given as multipliers for the cost of one energy measurement. The conditions are the same as in Fig. 4.14. 75

4.16 Convergence of CEO-ADAPT-VQE for linear H_6 at interatomic distance 1.5 Å, as compared to a variety of state-of-the-art fixed-structure ansatze: tiled Unitary Product State (tUPS), Orbital Optimization (OO)-tUPS, Perfect Pairing (PP)-tUPS, Quantum Number Preserving (QNP), and k-UpCCGSD-VQE. UCCSD-VQE (implemented with one Trotter step and lexical ordering) is also included for reference. The error is plotted against the number of layers (or iterations, in the case of ADAPT-VQE), CNOT count, parameter count, and measurement costs (given as multipliers for the cost of one energy measurement). The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). All algorithms terminate when chemical accuracy is reached for the first time, except for k-Unitary Paired Coupled Cluster Generalized Singles and Doubles (UpCCGSD), the error of which stagnates before reaching this threshold. 78

5.1 Circuit implementation of the unitary $U^{(QE)}_{\alpha_1\to\alpha_2}(\theta)$, generated by qubit excitation $T^{(QE)}_{\alpha_1\to\alpha_2}$. The same circuit can be used for $U^{(QE)}_{\beta_1\to\beta_2}$. The circuit for $U^{(QE)}_{\alpha_2\to\alpha_1}$ (or $U^{(QE)}_{\beta_2\to\beta_1}$) is identical, but with the rotation angle flipped. 85

5.2 Circuit implementation of the unitary $U^{(FE)}_{\alpha_1\to\alpha_2}(\theta)$, generated by qubit excitation $T^{(FE)}_{\alpha_1\to\alpha_2}$. The same circuit can be used for $U^{(FE)}_{\beta_1\to\beta_2}$. The circuit for $U^{(FE)}_{\alpha_2\to\alpha_1}$ (or $U^{(FE)}_{\beta_2\to\beta_1}$) is identical, but with the rotation angle flipped. The vertical dots indicate that there is a ladder of CNOT gates between qubits q_{z_1} to $q_{|\mathcal{Z}_{JW}|}$. 85

5.3 Circuit implementation of the unitary $U^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}(\theta)$, generated by qubit excitation $T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U^{(QE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped. 85

5.4 Circuit implementation of the unitary $U^{(FE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}(\theta)$, generated by qubit excitation $T^{(FE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U^{(FE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped. The vertical dots indicate that there is a ladder of CNOT gates between qubits q_{z_1} to $q_{|\mathcal{Z}_{JW}|}$. 86

5.5 Error *versus* CNOT count for the best performing variants of ADAPT-VQE: CEO-, QE- and Qubit-ADAPT-VQE. The molecule considered is H_6 at interatomic distance 3 Å in the STO-3G basis set. The blue shading represents the chemical accuracy region (error below 1 kcal/mol). 88

5.6 Implementation of the qubit single excitation circuit in Fig. 5.1 for Linear Nearest-Neighbor (LNN) connectivity using swap-based routing. Unmarked qubits are irrelevant to the excitation, but they are part of the path from the original to the final physical qubit representing qubit α_1. 89

5.7 Decomposition of a SWAP gate into CNOT gates. 89

5.8 Decomposition of an FSWAP gate into CNOT gates. 89

5.9 Implementation of the single fermionic excitation circuit in Fig. 5.2 for LNN connectivity. It is assumed that the mapping of fermionic modes to physical qubits is consistent with the ordering used in the Jordan-Wigner transform. 90

5.10 ADAPT-VQE error *versus* iteration number, CNOT count for LNN connectivity, and parameter count for linear (top) and triangular (bottom) H_6 (12 qubits) with interatomic distance 3 Å. The pools considered are CEO (see Ch. 4), QE [182] and GSD [72]. For each of these pools, we consider Co-ADAPT-VQE with the SWAP, FSWAP, and Qiskit-transpiler (QT) -based penalties. The canonical version of ADAPT-VQE transpiled to the same (LNN) connectivity is also included for comparison. For the case of CEO and QE pools, the FSWAP-based version of Co-ADAPT-VQE implements the fermionic version of these operators, which is obtained by exchanging SWAP gates for FSWAP. This exchange amounts to adding Jordan-Wigner anti-commutation strings to the corresponding generators. All algorithms terminate at chemical accuracy. In the bottom plots, the QE and CEO with FSWAP-based penalties overlap. Note that the GSD pool has no associated curve for SWAP- and FSWAP-based penalties. This is because the QE pool with FSWAP gates already implements GSD operators (with FSWAPs + QE circuit constructions), and implementing a fermionic operator using SWAP rather than FSWAP gates results in unnecessary CNOT gate overheads. 94

5.11 Error *versus* CNOT count for CEO-ADAPT-VQE for All-to-All (ATA) connectivity as compared to LNN connectivity, for H_6 at interatomic distance 3 Å. The LNN CNOT counts are obtained in two ways: By transpiling the ATA circuit from the canonical CEO-ADAPT-VQE (the best performing pool in the literature), and by using our Co-ADAPT-VQE protocol with the CEO pool and the FSWAP-based penalty. All algorithms terminate at chemical accuracy. . . . 95

5.12 Error *versus* CNOT count for GSD -ADAPT-VQE for ATA connectivity as compared to LNN connectivity. The details are the same as for Fig. 5.11(a), with the only difference being the pool (GSD vs CEO). In particular, the curve associated with this work corresponds to Co-ADAPT-VQE with the GSD pool and the FSWAP-based penalty, such that operators are implemented as cascades of FSWAP gates and qubit excitation circuits. 96

5.13 Evolution of the penalty and operator gradients for the Co-ADAPT-VQE algorithm. We consider the FSWAP-based variant with the CEO pool, applied to the linear H_6 molecule at interatomic distance 3 Å. The top panel compares the average penalty (equivalently, CNOT count) with the penalty of the selected operator for each iteration. The middle panel shows the evolution of the gradient norm and the penalty for a particular double excitation, and the bottom panel shows the same results for a particular single excitation. Black vertical bars indicate iterations in which the operator was chosen. 97

6.1 Noise-free simulation of ADAPT-VQE for a linear H_6 molecule with 4 Å interatomic distance until iteration 50. Between iterations 20 and 29, a significant gradient trough is evident, accompanied by a substantial decrease in gradient norm and a convergence plateau in energy, as is typical for them. The error is defined with respect to the FCI energy. 102

6.2 Gradient norm against operator position for a regular execution of ADAPT-VQE (i.e., appending operators). Panel (a) shows an iteration within a gradient trough, while the right panel considers an iteration at convergence. Within the gradient trough, a clear decrease in the gradient norm from the prepending to the appending position can be observed. For the case at convergence, there is no such consistent decrease visible: Instead, we can observe that the gradient norms are mostly uniform across positions. 102

6.3 Gradient norm across ansatz positions for ADAPT-VQE iterations 1–40. The gradient trough is visible across iterations 20–29 as a decrease in gradient norm as the position moves towards the end of the ansatz (with appending being the extreme case). 104

6.4 Workflow of the enhanced ADAPT-VQE algorithms, where red indicates our changes to the protocols and blue is the original ADAPT-VQE algorithm (see Ch. 2). Unlike the original ADAPT-VQE, our enhanced algorithms check for gradient troughs. If a gradient trough occurs, the protocol measures gradients at positions that are most likely to yield higher gradients than appending. In the next step, the operator and position that yield the largest magnitude gradient are identified, and the operator is inserted accordingly into the ansatz. All three algorithms are based on this principle. For an exact description see Sec. 6.4. The function $E(\theta_\mu, p_s, A_\mu)$ refers to the energy expectation value of the system as defined in Eq. J.2, where in the given case θ_μ is the variational parameter for the operator A_μ, p_s refers to the position of the newly added operator A_μ in the ansatz and A_μ with $\mu \in [1, N_{\text{Pool}}]$ is defined to be an operator within a given pool of size N_{Pool}. 106

6.5 Results of applying the four new protocols to ADAPT-VQE. The protocols start being applied three iterations after the first occurrence of a gradient trough, ensuring the algorithm is within the trough. The simulation uses the typical ADAPT-VQE (appending protocol) until iteration 22. From iteration 23 on, the enhanced protocols are activated until the algorithm converges. 110

6.6 Results of applying each enhanced protocol for 30 iterations of ADAPT-VQE, from iteration 23 to iteration 53 (within the occurrence of a gradient trough). After these iterations, the algorithm continues using the normal appending protocol. 112

C.1 Equivalent plots to Fig. 3.1 for LiH. 120

C.2 Equivalent plots to Fig. 3.1 for BeH_2. 121

D.1 Heatmaps showing the difference between the initial approximate inverse Hessian in the optimization process and the initial exact inverse Hessian, with (bottom row) and without (top row) Hessian recycling, for the 40th iteration of QEB-ADAPT-VQE. The plots show the element-wise difference between these two matrices. We consider H_6 at equilibrium and stretched geometries. 122

D.2 The equivalent of Fig. D.1 for the 60th optimization. 123

E.1 Relevant quantities for the study of the convergence rate of the 50th QEB-ADAPT-VQE optimization for H_6 at 1 Å. 124

F.1 Comparison of CEO-ADAPT-VQE variants at equilibrium bond distances. The subfigures depict the convergence of different variants of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at bond distances close to equilibrium. The QEB-ADAPT-VQE algorithm is also included for reference. The error is plotted against the iteration number (top), parameter count (middle), and `CNOT` count (bottom). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively. 127

F.2 Comparison of CEO-ADAPT-VQE variants at stretched bond distances. The subfigures depict the convergence of different variants of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at stretched bond distances. The QEB-ADAPT-VQE algorithm is also included for reference. The error is plotted against the iteration number (top), parameter count (middle), and CNOT count (bottom). The convergence criterion is the same as in Fig. F.1. 129

G.1 Combining CEO-ADAPT-VQE with transversal improvements. The subfigures depict the convergence of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at various bond distances. The algorithm is implemented *in tandem* with other recent proposals: Hessian Recycling (HR) (Ch 3), TETRIS [10], Optimized Gradient Measurements (OGM) [9], and all three (all). The baseline case, which uses the CEO pool while following the original ADAPT-VQE protocol [72] with a vanilla measurement strategy, is also included for reference. The error is plotted against the iteration number, CNOT count, CNOT depth, and measurement costs. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively. The curves for CEO and CEO (OGM) overlap on all plots except those pertaining to measurement costs. 131

H.1 Combining CEO-ADAPT-VQE with orbital optimization. The subfigures depict the convergence of CEO-ADAPT-VQE*, as defined in the Ch. 4, with and without orbital optimization. GSD-ADAPT-VQE is also included for reference. We consider H_6 at an interatomic distance of 1.5 Å. CEO-ADAPT-VQE*, GSD-ADAPT-VQE and UCCSD-VQE are defined in the main text (Chapters 2 and 4). The error is plotted against the iteration number, CNOT count, CNOT depth, and measurement costs. The measurement costs are given as multipliers for the cost of one energy measurement. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively. 134

I.1 Impact of k on the k-commutativity grouping. The plots show the evolution of the number of Pauli string collections and $\hat{R}$ against k, which defines the granularity of the commutativity considered in the observable grouping [37], for **a** LiH, **b** H_6, **c** BeH_2. While it is expected that fewer collections will be associated with lower measurement costs, covariances and varying coefficient magnitudes result in a nonlinear relation between the two. The metric $\hat{R}$ [35] approximates the ratio of measurement costs with and without grouping. The point $k = 0$ represents no grouping, such that $\hat{R}$ is one and the number of collections is the total number of Pauli strings in the Hamiltonian. 135

K.1 We compare the normalized gradient magnitudes before (iteration 16), during (iteration 25), and after (iteration 32) a gradient trough for the specified operators at all ansatz positions during a regular ADAPT-VQE execution (appending) for the top 5 operators for the prepending and appending positions. Top operators for prepending are labeled with '...[source orbitals] → [target orbitals]' and the ones for appending with '[source orbitals] → [target orbitals] ...'. Operators denoted as '[source orbitals] → [target orbitals]' are within the top for appending and prepending. The green boxes highlight the highest gradient magnitude position for the operator investigated. 140

K.2 Iterations considered in Fig. K.1 are highlighted. The gradient trough occurs in a regular ADAPT-VQE execution (appending operators). 140

L.1 Error *versus* ADAPT-VQE iteration, CNOT count, and parameter count for linear H_6 at interatomic distance 3 Å. Two variants are considered: CEO-ADAPT-VQE and the co-designed version, where the gradients are divided by the corresponding CNOT count for ATA connectivity. Both algorithms terminate when chemical accuracy is reached. 141

M.1 ADAPT-VQE error *versus* iteration number, CNOT count for LNN connectivity, and parameter count for linear H_6 (12 qubits) with interatomic distance 3 Å. All algorithms terminate when the gradient norm falls below 10^{-4}. 143

N.1 Impact of scaling the penalty via varying exponentials on the performance of Co-CEO-ADAPT-VQE for the linear H_6 molecule. A darker curve corresponds to a lower exponent. 145

O.1 Error *versus* the iteration number, LNN CNOT count, and parameter count for multiple variants of ADAPT-VQE applied to the LiH molecule (12 qubits) at interatomic distance 1.5 Å. We consider three different pools (CEO, QE, and GSD), three different variants of Co-ADAPT-VQE (SWAP-based, FSWAP-based, and Qiskit-transpiler-based), and the canonical ADAPT-VQE algorithm. 147

List of Tables

3.1 Total measurement costs incurred by the QEB-ADAPT-VQE algorithm in the gradient measurement and VQE subroutines (steps 8 and 14 in Algorithm 1), for the studied test cases. The costs are given as multipliers for the cost of a naive energy evaluation, so that unit cost implies $O(N^4)$ measurements. For the gradient measurement step, we consider the worst-case cost under the leading measurement strategy for the QE pool ($8N$ per iteration, where N is the number of spin-orbitals/qubits) [9]. 47

4.1 Minimal resource requirements of GSD-ADAPT-VQE and CEO-ADAPT-VQE* to reach chemical accuracy (defined as an error below 1 kcal/mol). The algorithms and costs are defined in the main text (see, e.g., Fig. 4.14). 55

5.1 Comparison of the CNOT counts required by CEO-ADAPT-VQE (see Ch. 4) to reach chemical accuracy in ATA and LNN connectivity. For the latter, we consider the previous state-of-the-art (obtained from the ATA circuit using the Qiskit (version 2.2.1) transpiler set to the maximum optimization level [137]) as well as an instance of Co-ADAPT-VQE specifically tailored to reduce the CNOT count of the final circuits. The percentages in parentheses represent the ratio of the CNOT count of our method to the CNOT count of the previous one. Three different systems are considered: linear H_6 (12 qubits), linear H_7 (14 qubits), and triangular H_6 (12 qubits). 83

5.2 Number of CNOT gates necessary to implement the unitaries generated by fermionic ($a_0 a_{99}^\dagger$ - h.c.) and qubit ($Q_0 Q_{99}^\dagger$ - h.c.) excitations. These unitaries act on a system with 100 spin-orbitals (equivalently, qubits), exchanging fermions from spin-orbitals 0 and 99. The table compares the two types of circuit implementations discussed in the main text under all-to-all and linear nearest neighbor connectivity. For the latter, two routing strategies are considered: one using SWAP gates and one using FSWAP gates (when applicable). Further, since it is possible to keep track of the qubit layout classically, we consider a setting where the final qubit layout is variable (i.e., equivalent to the initial one up to a permutation). All the circuits are transpiled using the respective techniques with no further optimization. While circuit optimization methods might produce slight improvements, they do not meaningfully change the trends observed in the table. 91

List of Algorithms

1 ADAPT-VQE 38
2 BFGS 41
3 BFGS for ADAPT-VQE Iteration n 43

Acronyms

$(SC)^2$ Surrogate-Constructed Scalable-Circuits *(p. 2)*

ADAPT Adaptive Derivative-Assembled Problem-Tailored *(pp. x–xix, 2–4, 10, 15–22, 25, 27, 28, 34, 35, 37, 40, 42–52, 54, 55, 58–60, 64, 65, 67–84, 86–88, 90, 92–102, 104–109, 111–115, 118, 120–122, 124, 126–131, 133, 134, 137, 139–145, 147, 148)*

ADAPT Adaptive Derivative-Assembled Pseudo-Trotter *(pp. 15, 16)*

AQC Adiabatic Quantum Computation *(pp. 5–7)*

ASP Adiabatic State Preparation *(p. 9)*

ATA All-to-All *(pp. xiv, xvii, xviii, 83, 87, 88, 90, 95, 96, 141, 142)*

BCH Baker–Campbell–Hausdorff *(pp. 14, 113)*

BP Barren Plateau *(pp. 1, 2, 12–14, 82)*

CEO Coupled Exchange Operator *(pp. xi–xviii, 2, 3, 54, 55, 59–62, 64–81, 83, 84, 86, 88, 90, 92, 94–97, 114, 126–131, 133, 134, 145, 147)*

CNOT Controlled NOT *(pp. 59, 61, 80)*

CVaR Conditional Value at Risk *(p. 33)*

DISCO Discretely Optimised *(pp. 10, 27–29)*

DLA Dynamical Lie Algebra *(p. 113)*

DVE Decision via Energy *(pp. 126–128)*

DVG Decision via Gradient *(pp. 126–128)*

E Evolutionary *(pp. 10, 22–25)*

FCI Full Configuration Interaction *(pp. xii, xiv, 72, 102, 109)*

FD Finite Difference *(p. 116)*

FTQC Fault-Tolerant Quantum Computation *(pp. 5, 10)*

GASP Genetic Algorithm for State Preparation *(p. 23)*
GSD Generalized Singles and Doubles *(pp. xii–xiv, xvi–xviii, 16, 18, 54, 55, 73–76, 78, 84, 94–96, 134, 147)*

HEA Hardware-Efficient Ansatz *(pp. 1, 12, 82)*
HR Hessian Recycling *(pp. xvi, 73, 76, 77, 130–132)*

L Layer *(pp. 10, 21, 22, 25)*
LNN Linear Nearest-Neighbor *(pp. xiii, xiv, xvii, xviii, 4, 83, 87–90, 93–96, 99, 141, 143, 147, 148)*

MVP Multiple Variational Parameter *(pp. 59–61, 64–67, 76, 80, 86, 90, 126–128, 130)*

NISQ Noisy Intermediate-Scale Quantum *(pp. 1, 5, 10, 31, 59, 67, 76, 81, 116)*

OGM Optimized Gradient Measurements *(pp. xvi, 73, 75–77, 130, 131)*
OO Orbital Optimization *(pp. xii, 30, 73, 78, 133)*
OVP One Variational Parameter *(pp. xi, 59, 62–67, 80, 86, 126, 127)*

PP Perfect Pairing *(pp. xii, 31, 78)*
PSR Parameter-Shift Rule *(p. 116)*

q-UCCD Quantum Unitary Coupled Cluster Doubles *(p. 13)*
QA Quantum Annealing *(pp. 5, 7)*
QAA Quantum Adiabatic Algorithm *(pp. 6, 7, 31, 32, 34)*
QAOA Quantum Alternating Operator Ansatz *(p. 32)*
QAOA Quantum Approximate Optimization Algorithm *(pp. 10, 22, 28, 31–35, 52)*
QE Qubit Excitation *(pp. x, xi, xiii, xvii, xviii, 18, 38, 44–47, 55, 56, 58–68, 70, 75, 80, 84, 90, 94, 109, 113, 116, 126, 128, 147)*
QEB Qubit-Excitation-Based *(pp. x–xii, xv, xvi, xviii, 18, 46–51, 58, 59, 65, 67–71, 77, 80, 84, 122, 124, 127–129)*
QNP Quantum Number Preserving *(pp. xii, 10, 29, 78, 79)*
QPE Quantum Phase Estimation *(pp. 5, 8)*

SAGSD Spin-Adapted Generalized Singles and Doubles *(p. 16)*
SC Scalable-Circuits *(p. 2)*
SCF Self-Consistent Field *(pp. 133, 134)*

SCGSD Spin-Complemented Generalized Singles and Doubles *(p. 17)*

SGSD Singlet Generalized Singles and Doubles *(pp. 16, 17)*

TETRIS Tiling Efficient Trial circuits with Rotations Implemented Simultaneously *(pp. xii, xvi, 19, 73, 74, 76, 77, 80, 130, 131)*

TSP Travelling Salesman Problem *(p. 10)*

tUPS tiled Unitary Product State *(pp. xii, 10, 29–31, 78, 79)*

UCC Unitary Coupled Cluster *(pp. 10, 13)*

UCCGSD Unitary Coupled Cluster Generalized Singles and Doubles *(p. 13)*

UCCSD Unitary Coupled Cluster Singles and Doubles *(pp. xii, xvi, 1, 3, 13, 14, 68, 71–73, 76–78, 80, 134)*

UpCCGSD Unitary Paired Coupled Cluster Generalized Singles and Doubles *(pp. xii, 13, 78, 79)*

VAns Variable Ansatz *(pp. 10, 25, 27)*

VQA Variational Quantum Algorithm *(pp. x, 3, 10–12, 24, 31, 33, 36, 37, 39, 52, 118)*

VQE Variational Quantum Eigensolver *(pp. x–xix, 1–4, 10, 12, 14–25, 27–29, 34, 37, 40, 42–44, 52, 54, 55, 58–60, 64, 65, 67–84, 86–88, 90, 92–102, 104–109, 111–115, 118, 120–122, 124, 126–131, 133, 134, 137, 139–145, 147, 148)*

1 Introduction

1.1 Motivation

Quantum computers are expected to be capable of efficiently simulating quantum systems even when this task is intractable for classical computers [53]. In the era of Noisy Intermediate-Scale Quantum (NISQ) computers, much research effort has been dedicated to the development of algorithms that are amenable to near-term hardware. One of the leading options for molecular simulations is the VQE proposed in Ref. [133]. In this hybrid algorithm, a classical computer is used to minimize a cost function evaluated on a quantum computer. The hope is that the iterative nature of VQE will allow it to employ shallow, NISQ-friendly circuits, in contrast with the long sequences of gates required to execute fully quantum algorithms such as Kitaev's quantum phase estimation [123]. VQE can be employed to find the ground state of many-body systems by taking the energy as the cost function. The expectation value of a Hamiltonian transformed by a many-body unitary is hard to evaluate classically [115], which suggests that VQE is a good contender for the first demonstration of practical quantum advantage.

In the context of VQE, the *ansatz* is a circuit that applies a parameterized unitary $U(\vec{\theta})$ to a reference state $|\psi_{\text{ref}}\rangle$. Typically, $|\psi_{\text{ref}}\rangle$ is a simple unparameterized state, such as the all-zero state or a classical approximation to the solution, and which can be prepared with a constant-depth circuit. $U(\vec{\theta})$ determines not only the circuit depth and gate count but also the optimization costs, since it impacts the optimization landscape. The seminal work of Ref. [133] proposed the use of the UCCSD ansatz. The corresponding unitary is the exponential of a linear combination of single and double fermionic excitations, with the parameter vector $\vec{\theta}$ consisting of the weights of the excitation operators. In contrast, Ref. [88] proposed the Hardware-Efficient Ansatz (HEA), which takes inspiration from device-specific (rather than problem-specific) information to construct the state preparation circuit. The entangling structure of the HEA is based on the connectivity of the device. This ansatz was later shown to suffer from Barren Plateaus (BPs) [110], a phenomenon characterized by the exponential (in the system size) concentration of the cost landscape towards the mean value, calling into question the trainability of such ansätze. This problem has since been extensively studied, with results evolving from case-specific proofs of occurrence to a unified theory of BPs [59, 139, 44]. Several approaches have been proposed to avoid BPs, namely using shallow circuits and local cost functions [163, 134], initializing the ansatz to identity [68], and leveraging problem-specific

symmetries in the circuit design [112]. However, Ref. [29] showed that all these strategies collaterally open the door to classical simulation algorithms: The same restrictions on the search space that make variational quantum algorithms provably BP-free can be leveraged to 'dequantize' them and dispense with the quantum-classical optimization loop altogether.

One of the few variational quantum algorithms that seems to combine the enticing attributes of being BP-free and not classically simulable is ADAPT-VQE [72]. While not rigorously proved, the absence of BPs is suggested by both theoretical arguments and empirical evidence [71, 93, 29]. The special feature of this algorithm is that the ansatz is constructed dynamically, by iteratively appending to a reference circuit parameterized unitaries generated by elements selected from an operator pool. The screening of generators is based on their energy derivatives (usually referred to simply as 'gradients'), such that at each step the choice of unitary depends on the variational state as well as on the molecular Hamiltonian. This problem- and system-tailored approach leads to remarkable improvements in circuit efficiency, accuracy, and trainability with respect to fixed-structure ansätze [72, 71]. These advantages have instigated the community to further understand and develop the algorithm. Active research topics include reducing measurement costs [10, 100, 105, 126, 120], seeking minimal complete pools (i.e., pools of minimal size that enable convergence) [153], decreasing circuit depth [10, 102, 52, 54], constructing more circuit-efficient pools [160, 182], studying the effect of noise [38], bypassing the optimization process [159, 66], and generalizing the algorithm to excited states [180, 186, 127] or to other problems [148, 92, 188, 171, 145, 176, 99, 128, 183, 107]. Recently, a variant of ADAPT-VQE termed Scalable-Circuits (SC)-ADAPT-VQE was used to prepare a 100-qubit vacuum state of the Schwinger model on a superconducting-qubit quantum computer [50]. This variant was later combined with a classically tractable surrogate construction approach that decreases the size of the pool, resulting in Surrogate-Constructed Scalable-Circuits $((SC)^2)$-ADAPT-VQE [78].

Despite this intense research following up on ADAPT-VQE, and the state-of-the-art improving dramatically since the first paper [72], it is still not understood how close we are to a demonstration on near-term hardware that can rival classical simulation. In fact, there is significant skepticism from the community that pre-fault-tolerant quantum algorithms can ever demonstrate quantum advantage. Often-cited concerns include the large number of measurements associated with VQE [67], the very low error rates that are required to reach sufficient accuracy at current gate counts [38], and the ADAPT-VQE-specific problem of gradient troughs [71].

In this thesis, we propose a suite of improvements to ADAPT-VQE that addresses all of these problems: (1) a Hessian recycling strategy that induces the first ADAPT-VQE-tailored optimization method in the literature; (2) a novel operator pool, based on a new class of operators we term CEOs, that outperforms all previously proposed pools; (3) a co-designed protocol where considerations about the features and shortcomings of the quantum hardware are directly incorporated into the ADAPT-VQE workflow; and (4) a strategy to overcome gradient troughs by exploring the addition of operators in alternative ansatz positions. These contributions result in a decrease by orders of magnitude in measurement costs, gate counts, and circuit depths.

Additionally, we combine our improvements with others proposed in the literature and compare the

resulting state-of-the-art ADAPT-VQE with the original proposal to gauge the evolution of the algorithm since its inception, showing a drastic improvement in costs and near-term viability. Further, to assess whether ADAPT-VQE is currently competitive with other VQEs, we implement five leading fixed-structure ansätze proposed in the literature. Our numerical simulations show that in the best-case scenario, fixed-structure ansätze required roughly 10^5 more quantum computer calls than state-of-the-art ADAPT-VQE for the same accuracy, contradicting the common belief that the adaptive ansatz construction incurs a measurement cost overhead.

1.2 Thesis Structure and Overview

The goal of this thesis is to characterize and improve upon the state-of-the-art of adaptive VQAs. In Chapter 2, we review relevant literature. The following four chapters present the main contributions of this work, as listed below. All code developed for this thesis is publicly available at `https://github.com/mafaldaramoa/ceo-adapt-vqe`.

Chapter 3, 'Recycling the Hessian: An ADAPT-VQE-tailored Quasi-Newton Optimizer' We implement a quasi-Newton optimizer where an approximation to the inverse Hessian matrix is continuously built and grown across the iterations of an adaptive VQA. The resulting algorithm has the flavor of a continuous optimization where the dimension of the search space is augmented when the gradient norm falls below a given threshold. We demonstrate that this inter-optimization exchange of second-order information consistently brings the approximate Hessian in the state of the optimizer closer to the exact Hessian. As a result, our method achieves a superlinear convergence rate even in situations where the typical implementation of a quasi-Newton optimizer converges only linearly. Our protocol reduces the measurement costs of implementing adaptive VQAs on quantum hardware, as well as the runtime of their classical simulation.

Chapter 4, 'CEO-ADAPT-VQE: Creating Hardware-Efficient and Problem-Tailored Operator Pools' We introduce a novel operator pool—which we term CEO pool—and show that it outperforms all previous proposals in the literature in terms of gate counts, gate depth, and measurement costs. We combine this pool with other key improvements to assess the cost of running state-of-the-art ADAPT-VQE on hardware. We show a dramatic reduction of quantum computational resource requirements compared to the early versions of the algorithm: `CNOT` count, `CNOT` depth, and measurement costs are reduced by up to 88%, 96%, and 99.6%, respectively, for molecules represented by 12 to 14 qubits (LiH, H_6, and BeH_2). We also find that our state-of-the-art CEO-ADAPT-VQE outperforms the UCCSD ansatz, the most widely used static VQE ansatz, in all relevant metrics, and offers a five order of magnitude decrease in measurement costs as compared to other static ansätze with competitive `CNOT` counts.

Chapter 5, 'Co-ADAPT-VQE: Co-Designing State Preparation Circuits' We propose a co-designed variant of ADAPT-VQE (Co-ADAPT-VQE) where the quantum hardware is taken into account in the construction of the ansatz. This framework can be readily used to optimize state preparation circuits for any device, addressing shortcomings such as limited connectivity, short coherence times, and variable gate errors. We exemplify the impact of Co-ADAPT-VQE by creating state preparation circuits for devices with LNN connectivity. We show a reduction of the CNOT count of the final circuits by up to 97% for 12-14 qubit systems, with the impact being greater for larger and more strongly correlated systems. The circuits created by Co-ADAPT-VQE provide an over 70% CNOT count reduction with respect to the original ADAPT-VQE in all-to-all connectivity, despite being restricted to LNN qubit interactions.

Chapter 6 'Overcoming Gradient Troughs: Exploring Alternative Operator Positions' We develop strategies for detecting and mitigating gradient troughs in ADAPT-VQE. Leveraging the non-commutative algebra of the ansatz, we develop heuristics for determining where to insert new operators into the circuit. We find that gradient troughs are more likely to occur when the same locations are repeatedly used for new operator insertions. Our novel protocols, which add new operators in different ansatz positions, enable us to escape gradient troughs and thereby reduce the measurement cost of the algorithm. This approach achieves an effective balance between cost and efficiency, leading to faster convergence without compromising the low circuit depth and gate count of ADAPT-VQE.

We conclude in Chapter 4 with final remarks and a discussion of interesting venues for future work.

1.3 List of Publications

- *Reducing Measurement Costs by Recycling the Hessian in Adaptive Variational Quantum Algorithms.* **M. Ramôa**, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. Quantum Sci. Technol. 10 (2024) 015031

  (discussed in Chapter 3)

- *Reducing the Resources Required by ADAPT-VQE Using Coupled Exchange Operators and Improved Subroutines.* **M. Ramôa**, P. G. Anastasiou, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. npj quantum Inf 11, 86 (2025)

  (discussed in Chapter 4)

- *Co-Designed Adaptive Quantum State Preparation Protocols.* **M. Ramôa**, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. arXiv:2601.20681

  (discussed in Chapter 5)

- *Strategies for Overcoming Gradient Troughs in the ADAPT-VQE Algorithm.* J. Stadelmann, J. Übelher, **M. Ramôa**, B. Sambasivam, S. E. Economou. arXiv:2512.25004

  (discussed in Chapter 6)

2

Literature Review

In this chapter, we review multiple proposals of quantum (or hybrid quantum-classical) algorithms for optimization and ground state preparation. Section 2.1 will focus on algorithms that were designed with the Fault-Tolerant Quantum Computation (FTQC) era in mind; in contrast, Sec. 2.2 will focus on *hybrid* and *variational* approaches that were proposed to adjust to the constraints of the NISQ era. Because the latter topic is fundamental for this project, while the former is tangential, it will be discussed in more depth.

2.1 Quantum Algorithms for Optimization in the FTQC Era

This section covers three key algorithms for optimization which, on account of requiring the execution of too deep and/or wide circuits, are only expected to be viable when quantum computers have sufficiently many qubits and sufficiently low error rates that error correction protocols can be implemented.

The proposals are Adiabatic Quantum Computation (AQC) (and the less rigorous framework of Quantum Annealing (QA)), Quantum Phase Estimation (QPE), and Grover's algorithm. There is hope that the former two might offer some advantage - perhaps an exponential advantage - over classical alternatives in optimization tasks, but this remains unproven. The latter option offers some provable speed-ups, but they are sub-exponential (usually quadratic or even more modest).

We remark that even though it's often claimed that Quantum Phase Estimation (QPE) yields exponential quantum advantage for the ground state problem in quantum chemistry, such a claim is made under the assumption that it is possible to efficiently prepare a good starting state. This stems from the idea that the structure of the problem may help us in this step by allowing us to obtain a good starting point. However, it cannot be ruled out that this same structure would allow us to efficiently obtain the solution to the full problem with a classical computer. For exponential quantum advantage to be reached, it does not suffice that a quantum computer finds a good solution at polynomial cost; in addition to that, we must have no classical heuristics capable of the same. Considering all of this, an exponential speed-up in quantum chemistry with QPE remains hypothetical [96].

Regarding *proven* quantum speed-ups in the realm of optimization, their backbone is usually Grover's algorithm (or its generalizations)—which means that these speed-ups are at most quadratic. We note

that while quadratic speed-ups are often relevant, both in efficiently solvable and in NP-Hard problems, the advanced state of classical computers leads classical operations to be performed with much more ease than quantum ones. A consequence of this is that the prefactors appearing in the costs, which are irrelevant for asymptotic complexity, may restrict the reaping of the fruits of quantum speed-ups to very large problem sizes.

Let us assume that a classical algorithm for a given task has cost $C_C n$ (where C_C is the cost of each classical operation and n relates to the size of the problem instance), while a quantum algorithm has cost $C_Q\sqrt{n}$ for the same task (where C_Q is the cost of each quantum operation). Then, we only obtain practical advantage using a quantum computer if $n > \left(\frac{C_Q}{C_C}\right)^2$. It has been estimated that $\frac{C_Q}{C_C}$ is of the order 10^{10} or higher at the moment [174], implying $n > 10^{20}$ is necessary. This places the possibility of benefiting from quantum resources rather far into the future, since large-scale quantum computers will be required to solve such huge problem instances.

2.1.1 Adiabatic Quantum Computing and Quantum Annealing

2.1.1.1 Adiabatic Quantum Computing

Adiabatic Quantum Computation (AQC) [49] (also known as the Quantum Adiabatic Algorithm (QAA)) was proposed for combinatorial search problems, such as satisfiability problems. The concept is rooted in the adiabatic theorem [23], which we loosely state as [33]:

Suppose that at $t = 0$, a physical system is in the nondegenerate ground state of a time-dependent Hamiltonian $H(t)$. Then if $H(t)$ changes slowly enough, the system will remain in its instantaneous ground state for any $t > 0$.

Adiabatic Quantum Computation (AQC) resorts to this theorem to find the ground state of a Hamiltonian H_P which encodes our problem. The computation will start in the ground state of a simple Hamiltonian H_B, after which the state is evolved under a time-dependent Hamiltonian $H(t)$ which interpolates between the simple Hamiltonian and our problem Hamiltonian:

$$H(t) = (1 - \frac{t}{T})H_B + \frac{t}{T}H_P, \tag{2.1}$$

where T is the total runtime.

If the transition is done *slowly enough*, the adiabatic theorem guarantees that the state we have at each time is the ground state of the current Hamiltonian. Thus, when we terminate the computation at time T, we have $H(T) = H_P$, and the final state encodes the solution to our problem. Note that we do not assume any knowledge about the solution, and that the initial state we prepare is easy to find and prepare; we can choose H_B to ensure this is the case.

A crucial point is the requirement that the change in the Hamiltonian occurs *slowly enough* (or equivalently, adiabatically). As is evident, this is a condition on the run time T: the slower the evolution must be, the longer the execution of the algorithm will take. This will depend on the problem at hand. Specifically,

it will depend on the minimum energy gap g_{min} between the ground state and the first excited state of $H(t)$ for any $0 \leq t \leq 1$ as

$$T = O\left(\frac{1}{g_{min}^2}\right). \tag{2.2}$$

There is an intuitive explanation for this dependency on g_{min}: If the transition is too fast, the system will jump across the gap, going from the ground state to the first excited state (causing the algorithm not to work as desired).

In general, it is not possible to estimate g_{min}, which prevents us from estimating the run time of the QAA. Evidently, if we do not have this information, the procedure is not well-defined: In that case, we do not know how slowly we must change the Hamiltonian (nor do we know if the runtime T is polynomial on the size of the system). Further, the problems where we can estimate g_{min} usually coincide with those that are classically tractable, since both tasks (estimating the gap and solving the problem) are aided by a regular structure.

Without any knowledge about g_{min}, the only guarantee we have from the adiabatic theorem is that as the total runtime T goes to infinity, the overlap with the desired state goes to one. However, this does not guarantee the efficiency of the algorithm in any way, since the regimen in which the solution is good enough may not be achievable within sub-exponential run times. Furthermore, the success probability does not generally increase monotonically with T, meaning that a slower evolution (higher runtime) will not necessarily improve the solution.

It has been proved that AQC is universal and polynomially equivalent to the circuit model of quantum computation [3].

2.1.1.2 Quantum Annealing

The difficulties in the practical implementation of adiabatic dynamics have led to the development of the more relaxed model of Quantum Annealing (QA) [87, 69]. This model is based on the same principles as AQC, but contemplates the occurrence of non-adiabatic effects. This implies forgoing the guarantee that the system remains in an eigenstate throughout the whole computation; the hope is that the energy of the state remains close to an extreme, and that the solution we obtain is satisfactory (even if it is not the optimal one).

QA is a non-universal heuristic framework; as an example, Shor's algorithm cannot be implemented on a quantum annealer. Whether QA is capable of providing a speed-up with respect to classical computers in optimization tasks remains an open question [138].

2.1.2 Quantum Phase Estimation

AQC and QA were proposed for *classical* problems with diagonal Hamiltonians. These cases have the particularity that the output is a classical bit string, and the value of the cost function can be evaluated

efficiently in a classical computer. If we have a non-diagonal Hamiltonian (from, e.g., an electronic structure problem), we can still resort to the adiabatic theorem; however, the algorithms we discussed before do not give us the eigenvalue, only the eigenstate.

The QPE routine [89] offers a way of solving this problem [173]. Given a unitary operator U and an input state $|\psi\rangle$ such that $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$ (i.e., $|\psi\rangle$ is an eigenstate of U with eigenvalue $e^{2\pi i\theta}$), this algorithm allows us to estimate θ [123].

The QPE circuit applies the transformation

$$|0\rangle^{\otimes t}|\psi\rangle \rightarrow \left|\widetilde{2^t\theta}\right\rangle|\psi\rangle, \tag{2.3}$$

where $\left|\widetilde{2^t\theta}\right\rangle$ is a state whose measurement (in the computational basis) yields a good (base-2) integer approximation to $2^t\theta$ with high probability. The number of qubits we use in the leftmost register affects the precision of our estimation of θ as well as our success probability.

If we wish to estimate the ground energy of a physical system, we apply the QPE routine with the ground state of the system as $|\psi\rangle$ and the time evolution operator e^{iHt} as U. The eigenstates of this operator are the same as those of the Hamiltonian H: If $H|u_i\rangle = E_i|u_i\rangle$, then $e^{iHt}|u_i\rangle = e^{iE_it}|u_i\rangle$. Thus, the output of the QPE routine will give us the ground energy of H (and similarly for excited states). Of course, we can only obtain the ground energy this way if we have access to the ground state; finding this ground state might be part of the problem.

Assume we have, not the ground state, but rather a linear superposition of eigenstates of the Hamiltonian. Note that any generic state may be expressed this way, since the eigenstates of a Hermitian operator form a complete basis for the corresponding dimension [123]. By linearity, if we have that $|\psi\rangle$ is a linear combination of eigenstates of U, i.e., $|\psi\rangle = \sum_i a_i|u_i\rangle$ with $U|u_i\rangle = e^{iE_it}|u_i\rangle$, then the output of QPE will be

$$|0\rangle^{\otimes t}\sum_i a_i|u_i\rangle \rightarrow \sum_i a_i\left|\widetilde{\frac{2^t}{2\pi}E_i}\right\rangle|u_i\rangle. \tag{2.4}$$

Upon measurement of the leftmost register, we have a probability $|a_i|^2$ of obtaining a good approximation to E_i with high probability. If we repeat the state preparation and measurement enough times, we will obtain the same approximation to the ground energy as if we had been able to prepare the exact ground state. However, as compared to that case, the probability of success will be multiplied by $|a_0|^2$. This means that the number of protocol repetitions will be increased $\frac{1}{|a_0|^2}$-fold; thus, we need to be able to prepare a state with large $|a_0|^2$. More precisely, for the algorithm to be efficient, we must be able to prepare states whose overlap with the ground state does not decrease exponentially with the system size.

In quantum chemistry, one possibility is to use classical approximations such as the Hartree-Fock state. However, it is not known whether the overlap of such approximations with the solution is exponentially vanishing with the dimension. As such, the preparation of a state with good overlap with the ground

state is a problem in itself [96]. Adiabatic State Preparation (ASP) has been regarded as a possible solution, in spite of the lack of guarantees discussed in Sec. 2.1.1. We note that g_{min} may be estimated by conventional methods in quantum chemistry [14].

2.1.3 Grover's Algorithm

Finally, we discuss the protocols that give us the less charming, but more reliable quadratic speed-up: those based on Grover's algorithm (also known as the "Quantum Search Algorithm") [75], which finds the marked solution in an unstructured database.

This algorithm can be applied when we have a black box function

$$f : \{0, 1, ..., N-1\} \longrightarrow 0, 1$$

such that

$$\begin{cases} f(x) = 1 & x = w \\ f(x) = 0 & x \neq w, \end{cases} \tag{2.5}$$

and we want to find the "marked" item w.

Using only $O(\sqrt{N})$ evaluations of the function (encoded in a quantum oracle), Grover's algorithm produces a state which, upon being measured on the computational basis, collapses to w with probability at least 50%. Classically, the number of function evaluations must scale like $O(N)$, since we must check half of the domain of the function on average; hence, we obtain a quadratic speed-up using quantum computers for this task.

Optimization algorithms offering polynomial speed-ups do not usually apply Grover's protocol directly. Rather, they re-design some (often the best) classical algorithm in a way that allows for the application of this quantum routine (or similar ones, such as quantum amplitude amplification [25]—the more general idea behind Grover's algorithm, and the backbone for the quantum quadratic speed-up).

Let us take the problem of finding the shortest path between two vertices in an n-vertex graph as an example. Using quantum amplitude amplification in the selection of the cheapest border edge in Dijkstra's (classical) algorithm, the complexity can be improved from $O(n^2)$ to $O(n^{\frac{3}{2}})$ [46].

A quantum speed-up may also be obtained in the task of finding the ϵ-spectral sparsifier of a weighted graph with n vertices and m edges: Using a quantum computer, this can be done in time $\tilde{O}\left(\frac{\sqrt{mn}}{\epsilon}\right)$ [12], versus the classically optimal $\tilde{O}(m)$ (where the tilde in $\tilde{O}$ indicates that logarithmic factors are ignored). For dense graphs, this amounts to an improvement from $\tilde{O}(n^2)$ to $\tilde{O}(n^{\frac{3}{2}})$ (which is less than quadratic, but was proved optimal up to polylog factors).

The two examples above showcase quantum speed-ups for problems that are efficiently solvable with a classical computer, but it is also possible to obtain polynomial speed-ups for NP-hard problems using similar ideas. For example, applying Grover's algorithm to the brute-force approach to the well-known NP-hard boolean satisfiability problem, we may improve the complexity from $\tilde{O}(2^n)$ to $\tilde{O}(\sqrt{2^n})$; and

applying it to the exhaustive search over all possible routes for the Travelling Salesman Problem (TSP), we may obtain a quadratic speed-up also, in this case from $O^*(n!)$ to $O^*(\sqrt{n!})$ (where the star in O^* indicates that polynomial factors are ignored).

Of course, more clever classical algorithms may outperform the application of Grover to such naive strategies: e.g., an improved variant of Schöning's algorithm [143, 149] solves 3-SAT problems in $O^*(1.329...^n)$ time, and the Bellman-Held-Karp algorithm [20, 82] finds the solution for the TSP in $O^*(2^n)$ time. However, quantum algorithms inspired by these classical protocols may improve upon their complexity also, and in fact they do: A quantum-assisted version of Schöning's algorithm is able to obtain the solution in $O(\sqrt{1.329...^n})$ repetitions using quantum amplitude amplification [7, 39], and a Bellman-Held-Karp-inspired application of Grover's search runs in $O^*(1.728^n)$ time [8]—a quadratic and subquadratic speed-up, respectively. Whether quantum computers may allow for a quadratic speed-up over their classical counterparts for the TSP remains an open question.

2.2 Quantum Algorithms for Optimization in the NISQ Era

The previous section provided an overview of algorithms for optimization with FTQC in mind. However, as discussed in the introduction, we currently find ourselves in the NISQ era, and the hardware requirements of the aforementioned proposals are far too demanding to be met by current quantum computers. In this section, we provide an overview of some near-term algorithms.

We will start with a description of the typical outline of VQAs and, more specifically, VQE. We discuss topics relevant to VQE, such as fermion-to-qubit mappings, ansätze, and barren plateaus. After that, we cover multiple examples of such algorithms, including dynamic options where the circuit structure is constructed on the fly (ADAPT-VQE, Layer (L)-VQE, Evolutionary (E)-VQE, Variable Ansatz (VAns), Discretely Optimised (DISCO)-VQE, and ADAPT-Quantum Approximate Optimization Algorithm (QAOA)), as well as static options where the circuit structure is fixed (Unitary Coupled Cluster (UCC), QNP, tUPS, and QAOA). Special attention will be paid to ADAPT-VQE, including a discussion of its different variants, optimization landscape, and convergence.

2.2.1 VQAs and VQE

The steps of a VQA are summarized in Fig. 2.1. The quantum computer prepares parameterized states $\left|\psi(\vec{\theta})\right\rangle$ and measures the corresponding energy, while the classical computer updates the parameter vector $\vec{\theta}$ to minimize this energy. The parameters are updated iteratively until a convergence criterion is met (e.g., the energy change from a parameter update falls below a given threshold). In the particular case where we are targeting eigenstates of a physical system, we use the acronym VQE (representing a subclass of VQAs).

The goal of a VQA is to improve near-term viability by replacing a single coherent circuit execution with a hybrid quantum-classical feedback loop where the role of the quantum computer is limited to evaluating expectation values. This decreases the depth of the circuits, which in turn reduces the coherence time

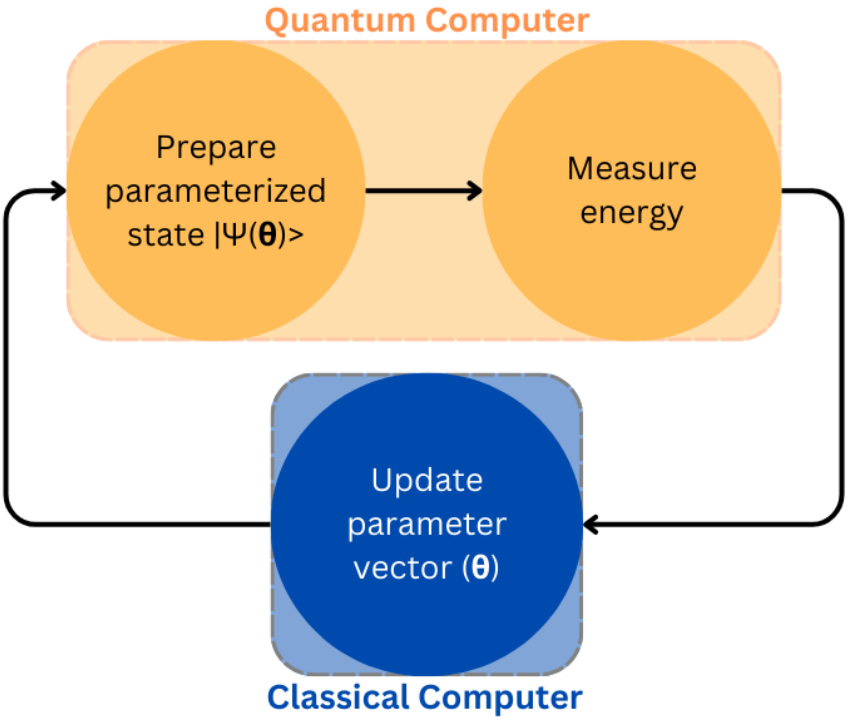


Figure 2.1: Flowchart for a VQA.

requirements, making these algorithms more amenable to current hardware than algorithms designed for fault-tolerant quantum computers. Hence, circuit depth is a common metric for gauging a circuit's hardware readiness. Since two-qubit gates typically have significantly higher error rates than single-qubit gates, another common metric is the number of two-qubit gates or the two-qubit gate depth. Common examples of two-qubit gates are `CX`s and `CZ`s, also known as `CNOT`s and `CP`/`CPhase`, respectively. However, many other options exist, including parameterized entangling rotations.

The main drawbacks of VQAs are the lack of provable guarantees, stemming from their heuristic nature, and the measurement overhead they incur. Implementing the classical optimization of a quantum cost function requires multiple calls to a quantum computer to measure its value (and possibly its gradients, depending on the optimization method). This is aggravated by the fact that meeting the accuracy requirements of certain applications, such as quantum chemistry, demands a very high number of circuit repetitions (also known as shots) for a single cost function evaluation [67].

2.2.1.1 Fermion-to-Qubit Mapping

In this thesis, we focus on quantum chemistry applications—in particular, we are interested in the electronic structure problem. To solve this problem using a quantum computer, we need a fermion-to-qubit mapping. A popular choice is the Jordan-Wigner transform [86]. This transform represents the occupation of each spin-orbital in the state of one qubit, which means that Slater determinants are mapped to computational basis states. This mapping transforms fermionic ladder operators as

$$
\begin{aligned}
a_i^\dagger &\rightarrow \frac{1}{2} \prod_{k=1}^{i-1} Z_k \cdot (X_i - iY_i), \\
a_i &\rightarrow \frac{1}{2} \prod_{k=1}^{i-1} Z_k \cdot (X_i + iY_i),
\end{aligned}
\tag{2.6}
$$

where Z_k, X_i, and Y_i are Pauli operators acting on the qubits labeled by the respective indices. The operator $a_i^\dagger$ is a *creation* operator, which creates a fermion in orbital i; a_i is an *annihilation* operator, which removes a fermion from orbital i. These operators naturally arise from the second quantization formalism.

2.2.1.2 Ansatz

In the first VQE demonstration, the state preparation circuit (also known as the ansatz) was implemented using integrated photonics technology. The variational parameters were angles of phase shifters. Since the ansatz used in the experiments prepared generic states within the 2-qubit Hilbert space, six phase shifters were necessary. This was a very simple proof-of-principle demonstration; evidently, searching through the entire Hilbert space is not feasible in general, as it would imply an exponential scaling of the number of parameters with the system size. The fact that the choice of ansatz dictates both solution quality and near-term viability has made this one of the primary research topics concerning VQEs.

Because of their convenience, hardware-dictated ansätze became a trend. These ansätze are built from gates that are native in the target hardware, usually by alternating layers of parameterized single-qubit rotations and unparameterized nearest-neighbor two-qubit entangling gates, and are commonly referred to as HEAs [88].

2.2.1.3 Barren Plateaus

The extensive analysis that was dedicated to VQA in the aftermath of the VQE proposal led to the conclusion that the 'randomness' in the structure of HEA-type ansätze leads to BPs, phenomena characterized by exponentially vanishing gradients. When a BP occurs, using a polynomial (on the size of the system) number of measurements will cause gradient descent to amount to a random walk with exponentially small probability of finding the solution [110]. The problem has been generalized to gradient-free optimization methods [13]. Intuitively, we can think that using a hardware-efficient ansatz in a variational algorithm is kindred to performing an uninformed search in an exponentially big space, starting at a random point: It seems unlikely that we will find the solution in polynomial time.

Hardware noise was shown to be capable of causing a similar problem: For a general class of local noise models, the gradients of the cost function vanish exponentially with circuit depth, regardless of the noise strength [170]. While they are different in nature from the previous BPs, these *noise-induced* variants also highlight the importance of the choice of state preparation circuit.

Hence, we have arrived at the main problem of ansatz design: It is not clear how to choose the circuit structure, and choosing it randomly (or according to the restrictions of our device) has proved not to be a good choice. Intuitively, the best choice should be problem-dependent—firstly, because we can hardly ever expect a one-size-fits-all solution for problems which vary greatly in nature; secondly, because the literature has repeatedly shown that tailoring the ansatz to the problem aids convergence; and thirdly, because the BP phenomenon seems to be strongly tied with problem-agnostic circuit structures. In fact, when the circuit structure is problem-specific and we have good initialization heuristics, proofs of existence of BPs immediately break down, as they are average statements over randomly sampled points in parameter space.

2.2.2 UCC

The UCC family of ansätze is motivated by classical variational methods for quantum chemistry [19, 146]. We define the UCC ansatz with excitations of order up to k as

$$|UCC(k)\rangle = e^{T^{(k)} - T^{(k)\dagger}} |HF\rangle , \tag{2.7}$$

where $|HF\rangle$ is the Hartree-Fock reference state, and $T^{(k)}$ is the truncated cluster operator

$$\begin{aligned} T^{(k)} &= \sum_{i=1}^{k} T_i, \\ T_1 &= \sum_{i,a} t_a^i a_a^\dagger a_i, \\ T_2 &= \sum_{i>j, a>b} t_{ab}^{ij} a_a^\dagger a_b^\dagger a_i a_j, \\ &\vdots \end{aligned} \tag{2.8}$$

where the lower-case t are real-valued expansion coefficients, which will be our variational parameters. The indices i, j and a, b run over orbitals that are occupied and unoccupied (virtual) in the reference state, respectively. $a^\dagger$, a are the fermionic creation/annihilation operators, as described in 2.2.1.1.

To implement a UCC ansatz, we must decide on the truncation order. A popular option is UCCSD, the instance of the UCC ansatz for $k = 2$. We note that even within UCCSD, multiple variants exist, for example Unitary Coupled Cluster Generalized Singles and Doubles (UCCGSD), Quantum Unitary Coupled Cluster Doubles (q-UCCD) [155], or k-UpCCGSD [95].

The dynamics of the UCCSD operator are similar to time evolution under a quantum Hamiltonian. We can write the ansatz as

$$|UCCSD\rangle = e^{(T_1+T_2)-(T_1^\dagger+T_2^\dagger)} |HF\rangle = e^{iH_{UCCSD}} |HF\rangle , \tag{2.9}$$

where $H_{UCCSD} = -i\left((T_1 + T_2) - (T_1^\dagger + T_2^\dagger)\right)$ is a Hermitian operator. We can think of $|UCCSD\rangle$ for a specific set of parameters $\vec{t}$ as the Hartree-Fock state after evolution under an 'effective' Hamiltonian $H_{UCCSD}(\vec{t})$ for a unit time period.

While there is reason to believe that the UCCSD ansatz does not suffer from BPs [30], this ansatz has its problems. Given its unitarity, it can be implemented on a quantum computer; but how to implement on a digital, gate-based quantum computer is not evident. We know how to implement simple operators of the form $e^{i\theta P}$, for P a Pauli string $\bigotimes_j \sigma_j$ (where σ_j, $j \in \{0, z, x, y\}$, are the Pauli matrices); but H_{UCCSD} cannot in general be written as θP, rather as a linear combination of operators which admit such form. For two non-commuting operators A and B,

$$e^{A+B} \neq e^A e^B \tag{2.10}$$

in general. In fact, for two operators A, B, we can calculate $e^A e^B$ as

$$e^A e^B = e^{A+B+\frac{1}{2}[A,B]+\frac{1}{12}[A,[A,B]]-\frac{1}{12}[B,[A,B]]+\ldots}. \tag{2.11}$$

Equation 2.11 is known as the Baker–Campbell–Hausdorff (BCH) formula. We can see that the product of the exponential series is the exponential series of the sums only in the special case of commuting operators ($[A, B] = 0$). Hence, implementing the UCCSD operator is not a trivial task. However, we can take advantage of the similarity of the UCCSD operator with time evolution operators and borrow tools from quantum simulation. Namely, we can use the Trotter formula [81, 123]

$$\lim_{n\to\infty}\left(e^{\frac{iAt}{n}} e^{\frac{iBt}{n}}\right)^n = e^{i(A+B)t}, \tag{2.12}$$

which remains valid even when A and B do not commute, and can be easily generalized to linear combinations of an arbitrary number of operators.

Circuit depth scales with the number of Trotter steps n (also called the *Trotter number*), as does the accuracy of the approximation. If we were allowed infinite depth, we could exactly approximate the UCCSD ansatz; however, in practice, we want to reduce circuit depth as much as possible. There is hope that the variational nature of VQE would partly mitigate the error of the approximation, and some results in the literature suggest that taking n to be 1 is sufficient to reach chemical accuracy [130]. However, the ambiguity associated with the choice of ordering (which arises from taking a finite Trotter decomposition) has been shown to cause the ansatz not to be chemically well-defined. Differences in the operator ordering (which should be irrelevant in practice) lead to energy variations that are relevant on the chemical scale. The difference in the error between different orderings can reach the order of magnitude of the absolute error associated with the original (untrotterized) ansatz [73].

Further, even with $n = 1$, implementing the UCCSD ansatz in today's quantum computers is not yet possible for systems of interest: In spite of the depth being polynomial ($\mathcal{O}(N^4)$) on the number of orbitals N, it is still impracticably large. One might think that the whole circuit is necessary to achieve a good solution, but it has been shown that this is not the case: The UCCSD ansatz has an unnecessarily

large number of parameters and terms [51, 111], implying that the circuits are unnecessarily long and the optimization is unnecessarily hard. This is not surprising if we consider that this ansatz includes *all* single and double excitations. It is predictable that they do not all bear the same importance in converging to the ground state. Further, while this ansatz is tailored to electronic structure problems in general, it is not tailored to each specific molecule (aside from the choice of spin-orbitals in the basis set). This motivates the idea of selecting only the most convenient excitations for the molecule under study to obtain a problem-tailored ansatz with reduced resource requirements.

The first proposal to explore this idea was Adaptive Derivative-Assembled Pseudo-Trotter (ADAPT)-VQE [72], and this class of algorithms with adaptively-built ansätze has remained an active area of research since. Major advances and improvements to the algorithm have been made, and competing proposals of differently built dynamic ansätze have emerged as well. We will now review the relevant literature regarding ADAPT-VQE, and follow with other ansatz proposals. Because ADAPT-VQE is the subject of this thesis, the other algorithms will be implicitly and explicitly compared against it.

2.2.3 ADAPT-VQE

ADAPT-VQE, first proposed in Ref. [72], is a special instance of the VQE algorithm where the ansatz is created dynamically instead of being defined *a priori*. This protocol has been shown to outperform static VQE variants in circuit depth and trainability [72]. We represent the protocol in Fig. 2.2.

The algorithm starts with a simple reference state. This is typically the Hartree-Fock ground state obtained from classical mean-field calculations, which corresponds to a computational basis state. Unitary operators are then added to the ansatz iteratively. These operators are of the form $e^{\hat{A}_k \theta_k}$, where the $\hat{A}_k$ are anti-hermitian generators belonging to a pre-selected operator pool.

To decide which operator is added in each iteration, a gradient-based selection criterion is used. More specifically, the candidate operator with the energy derivative of highest magnitude at point zero is appended to the ansatz in each iteration. At iteration n, the gradient associated with a generator $\hat{A}_k$ can be evaluated as

$$\left.\frac{\partial E^{(n)}}{\partial \theta_k}\right|_{\theta_k=0} = \left\langle \psi^{(n-1)} \right| \left[\mathcal{H}, \hat{A}_k\right] \left| \psi^{(n-1)} \right\rangle , \tag{2.13}$$

where $\left|\psi^{(n-1)}\right\rangle$ is the final state from the previous iteration, and $\mathcal{H}$ is the Hamiltonian of the target system. The gradient criterion reflects the expectation that operators with a high impact on the energy locally will also have a high impact once a full optimization is performed. After each addition, the ansatz is fully optimized; the optimized parameters carry over as initial parameters for the next iteration. The new parameter is initialized at zero, which results in trainability benefits [71]. The algorithm ends when the norm of the vector formed from all gradients falls below a given threshold; throughout this work, we use the L2-norm to determine convergence.

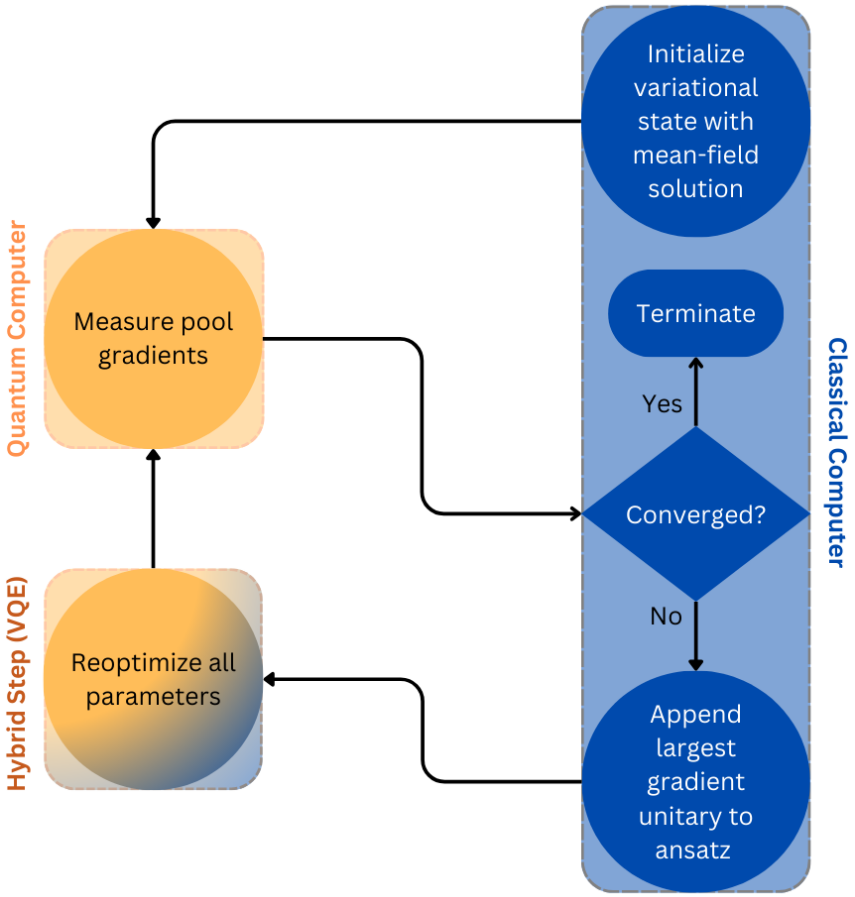


Figure 2.2: Flowchart for the ADAPT-VQE algorithm.

2.2.3.1 Fermionic-ADAPT-VQE

The original proposal of ADAPT-VQE [72] suggested using fermion-inspired operator pools. We exemplify three such pools.

The GSD pool consists of all single and double excitations of the form

$$\begin{aligned} \hat{\tau}_1 &= |a\rangle\langle b| - h.c., \\ \hat{\tau}_2 &= |ab\rangle\langle cd| - h.c., \end{aligned} \tag{2.14}$$

where a, b, c, d run over all sets of spin-orbitals that ensure preservation of Z spin projection, and $h.c.$ stands for *hermitian conjugate*. This pool has $O(N^4)$ operators, each of which consists of a linear combination of at most 8 Pauli strings. The average string length and depth of the circuit implementation of each operator are $O(N)$.

The Singlet Generalized Singles and Doubles (SGSD) (also Spin-Adapted Generalized Singles and Doubles (SAGSD)) pool contains operators of the form

$$
\begin{aligned}
\hat{\tau}_1 &= |p\rangle\langle q| + |\bar{p}\rangle\langle \bar{q}| - h.c., \\
\hat{\tau}_{2,A} &= -\frac{2}{\sqrt{12}} \left(|rs\rangle\langle pq| + \frac{1}{2} \left(|r\bar{s}\rangle + |\bar{r}s\rangle \right) \left(\langle p\bar{q}| + \langle \bar{p}q| \right) + |\bar{r}\bar{s}\rangle\langle \bar{p}\bar{q}| \right) - h.c., \\
\hat{\tau}_{2,B} &= -\frac{1}{2} \left(|r\bar{s}\rangle - |\bar{r}s\rangle \right) \left(\langle p\bar{q}| - \langle \bar{p}q| \right) - h.c.,
\end{aligned}
\tag{2.15}
$$

where p, q, r, s are indices of spatial orbitals, and the overbar indicates β (down) spin while the absence of it indicates α (up) spin. This pool has $\mathcal{O}(N^4)$ operators, each of which consists of a linear combination of at most 48 Pauli strings. The average string length and depth of the circuit implementation of each operator are $\mathcal{O}(N)$.

Finally, we write the operators of the Spin-Complemented Generalized Singles and Doubles (SCGSD) pool as

$$
\begin{aligned}
\hat{\tau}_1 &= |p\rangle\langle q| + |\bar{p}\rangle\langle \bar{q}| - h.c., \\
\hat{\tau}_{2,A} &= - \left(|rs\rangle\langle pq| + |\bar{r}\bar{s}\rangle\langle \bar{p}\bar{q}| \right) - h.c., \\
\hat{\tau}_{2,B} &= - \left(|r\bar{s}\rangle\langle p\bar{q}| + |\bar{r}s\rangle\langle \bar{p}q| \right) - h.c., \\
\hat{\tau}_{2,C} &= - \left(|r\bar{s}\rangle\langle \bar{p}q| + |\bar{r}s\rangle\langle p\bar{q}| \right) - h.c..
\end{aligned}
\tag{2.16}
$$

This pool has $\mathcal{O}(N^4)$ operators, each of which consists of a linear combination of at most 16 Pauli strings. Once again, the average string length and depth of the circuit implementation of each operator are $\mathcal{O}(N)$.

All three operator pools preserve particle number and Z spin projection, with SGSD and SCGSD additionally preserving the expectation value of the total spin squared, $\langle S^2 \rangle$, in theory. However, in practice, unitaries generated by these operators require Trotterization to be implemented in a digital quantum computer [123], and they do not preserve $\langle S^2 \rangle$ at finite circuit depths.

2.2.3.2 Qubit-ADAPT-VQE

In 2020, Qubit-ADAPT-VQE was proposed in [160] as a variant of ADAPT-VQE with a more hardware-efficient (but still problem-tailored) pool, termed *qubit pool*.

As we have seen, the operators in fermionic pools consist of a linear combination of up to 48 Pauli strings. Using the ladder-of-`CNOT`s implementation of Pauli exponentials [123], the circuit for a single operator may already have hundreds of `CNOT`s when as few as a dozen spin-orbitals are used to describe a molecule.

In the qubit pool, the operators are broken down into the individual Pauli strings, and the Jordan-Wigner string (composed of Pauli-Zs) is removed. This pool then contains length 2 or 4 Pauli strings with only Pauli-X and Pauli-Y operators (an odd number of each). The size of the pool is still $\mathcal{O}(N^4)$ (albeit with a larger prefactor), but now each operator is a single Pauli string. The average string length and the depth of the circuit implementation of each operator are $\mathcal{O}(1)$.

Qubit-ADAPT-VQE produces shallower ansatze for the same accuracy, while increasing the measurement costs. This pool does not preserve particle number, Z spin projection, or $\langle S^2 \rangle$.

2.2.3.3 QEB-ADAPT-VQE

The QEB-ADAPT-VQE uses yet another pool, comprised of QEs. QEs are essentially GSD operators, with the Jordan-Wigner string removed. An example of one such operator is

$$\begin{aligned}\hat{\tau} =& i(-X_q X_p X_s Y_r - X_q X_p Y_s X_r + X_q Y_p X_s X_r - X_q Y_p Y_s Y_r \\ &+ Y_q X_p X_s X_r - Y_q X_p Y_s Y_r + Y_q Y_p X_s Y_r + Y_q Y_p Y_s X_r),\end{aligned} \tag{2.17}$$

where p, q, r, s label spin-orbitals (equivalently, qubits). This pool has $\mathcal{O}(N^4)$ operators, each of which consists of a linear combination of at most 8 Pauli strings. The average string length is $\mathcal{O}(1)$.

It is possible to implement double qubit excitations with a `CNOT` count of 13 and depth of 11 [179], against the 48 `CNOT` count and depth of the ladder-of-`CNOT`s [123] method. This pool preserves particle number and Z spin projection, but it does not preserve $\langle S^2 \rangle$.

2.2.3.4 Gradient Measurements

We have seen that evaluating the pool gradients for the ADAPT-VQE operator selection step amounts to measuring the commutators of the pool generators with the problem Hamiltonian (Eq. (2.13)). The number of quantum computer calls required for these measurements varies depending on the strategy employed.

Typically, each pool operator is a linear combination of Pauli strings whose number remains constant with the system size. Further, we make the reasonable assumption that the Hamiltonian itself has been decomposed into a sum of polynomially many Pauli strings. The commutator $\left[\hat{H}, \hat{A}_i\right]$ is then an observable that can be efficiently measured in a quantum computer. Thus, the extra measurements of the ADAPT-VQE algorithm do not, in principle, hinder efficiency, as long as the scaling of the operator pool is not exponential on the system size. Of course, the gradient measurements may still represent a considerable measurement overhead as compared to the original VQE algorithm.

Näively, for most pools, evaluating the gradients of the form of Eq. (2.13) would require $\mathcal{O}(N^8)$ shots, where N is the number of orbitals (equivalently, qubits). This is because both the number of Pauli strings in the electronic Hamiltonian and the number of operators in the operator pool (namely, the fermionic, qubit, and qubit excitation pools) scale as $\mathcal{O}(N^4)$.

However, Ref. [9] proposed an optimized measurement grouping protocol that leverages the structure of the best performing operator pools (the qubit and qubit excitation pools) to group them into $\mathcal{O}(N)$ commuting sets, resulting in a remarkably lower $\mathcal{O}(N^5)$ cost for measuring all candidate operator gradients.

2.2.3.5 TETRIS-ADAPT-VQE

TETRIS-ADAPT-VQE [10] was proposed as a more circuit-depth-efficient algorithm for building adaptive ansätze. Unlike previous variants, this one does not introduce a new operator pool. Instead, the key change is to add several commuting operators per iteration (instead of adding only one as in the original algorithm). The operators to be added are the ones with the highest gradients, as long as they all act on disjoint qubits and have non-zero gradients. In the best-case scenario, the layer added in a given iteration acts on all qubits (like a winning Tetris game).

Reference [10] tests the algorithm with the qubit pool and the qubit excitation pool. These choices are convenient because the corresponding operators act on a very small number of qubits; we would likely not see a significant improvement if we used this strategy with fermionic pools. Since the leading pools in the literature have this desirable feature, the TETRIS protocol is generally applicable.

The TETRIS variant of ADAPT-VQE effectively reduces the circuit depth, the number of gradient measurements, and the number of optimizations compared to the original version. It does not seem to preclude trainability—in fact, it seems to maintain the same optimization perks as the original ADAPT-VQE.

2.2.3.6 Minimal Pools

Minimal pools are defined to be the minimum sized pools which allow ADAPT-VQE to converge. The size of these pools has been proven to scale linearly with the size of the system [153]. Such pools are not unique. Two examples for the case of dense, random Hamiltonians were given in [160]. These pools are unfit for physical Hamiltonians. Symmetry-adapted minimal pools that can be used for molecular Hamiltonians were presented in [153].

All of these pools have $\mathcal{O}(N)$ operators, each of which consists of a single Pauli string. Further, minimal pools exist such that each string (and thus also the circuit implementation of the corresponding operator) has length $\mathcal{O}(1)$. However, we note that the reduced size of minimal pools reduces the variational freedom, leading to slower convergence. In general, these pools are not the optimal choice in terms of circuit depth. Further, given the optimal measurement protocol for non-minimal pools discussed in Sec. 2.2.3.4, the gradient measurement step for pools of size $\mathcal{O}(N^4)$ generally requires the same asymptotic costs as for minimal pools.

2.2.3.7 Pool Tiling

Operator pool tiling [164] was proposed as a way of creating pools for problems with simple repeated lattice structures, of which there are many examples in condensed matter physics. The idea is to divide the computation into two steps:

1. Run ADAPT-VQE for a small problem instance of size L_1 with a large, highly expressive pool. The size may be exponential on $L1$. For example, we can use a pool consisting of all possible Pauli strings, except for the trivial one ($I...I$). The size of this pool is $4^{L1} - 1$.

2. Use the operators that were chosen by ADAPT-VQE for this small instance on a larger problem instance, by "tiling" them: From each operator $\hat{O}_i$ acting on $L1$ qubits, create $L_2 - L_1 + 1$ operators as

$$\hat{O}_i \otimes \hat{I}^{\otimes(L_2-L_1)},\ \hat{I} \otimes \hat{O}_i \otimes \hat{I}^{\otimes(L_2-L_1-1)},\ \ ...,\ \hat{I}^{\otimes(L_2-L_1-1)} \otimes \hat{O}_i \otimes \hat{I},\ \hat{I}^{\otimes(L_2-L_1)} \otimes \hat{O}_i$$

Note that the tiling is done in an overlapping fashion, so the operators do not commute.

The size of this pool is not exponential in L_2; in fact, assuming that L_2 is independent of L_1 (i.e. the size of the small problem instance does not grow if we tackle larger problem instances), the size of the pool grows linearly with L_2, as we have a total of $(L_2 - L_1 + 1)C$ operators, where C is the number of operators selected in the small instance (which is independent of L_2).

2.2.3.8 Trainability

The warm-started parameters from the previous iteration have been shown to benefit the optimization landscape of ADAPT-VQE in Ref. [71]; particularly, it was shown that the algorithm is able to *burrow* towards the exact solution (by adding new operators) even when caught in a local minimum. Further, the same reference provides reasons to believe that the algorithm does not suffer from barren plateaus.

2.2.3.9 Gradient Troughs

Gradient troughs are one of the main challenges for ADAPT-VQE. This is a convergence issue that occurs even in completely noise-free simulations. It is characterized by non-monotonic behavior of the norm of the vector consisting of the gradients of pool operators: This norm will see a meaningful drop, which is (if the algorithm proceeds) followed by a rise a few iterations later [21, 71]. The drop in the gradient norm is accompanied by the stalling of the energy.

Gradient troughs are problematic for two reasons: (1) Since ADAPT-VQE terminates when the gradient norm falls below a given threshold, a non-monotonic behavior may lead to wrongful convergence; and (2) lower gradients imply that a higher precision is needed for operator selection and for parameter optimization, leading to higher measurement costs.

Because both relate to low gradients, there is a common misconception that gradient troughs in ADAPT-VQE are conceptually equivalent to barren plateaus in VQE. This is not the case. Barren plateaus are characterized by gradients that vanish exponentially with the system size on average for random points in parameter space [110]. They hinder the ability to scale up the optimization to larger system sizes and threaten the practical viability of VQE, as they lead to exponential optimization costs. Barren plateaus stem from random ansatz structures and a lack of good heuristics for parameter initialization.

In ADAPT-VQE, we have good initialization techniques (based on the Hartree-Fock solution and parameter recycling) and the ansatz is not random; rather, it's informed both by the general problem (through fermion-tailored pools) and the particular system at hand (through the Hamiltonian-driven selection criterion). Furthermore, the algorithm is designed to consistently add operators with high gradient magnitudes.

We have theoretical and numerical reasons to believe that the VQE subroutine in ADAPT-VQE does not suffer from barren plateaus [71].

One might ask if gradient troughs are not a manifestation of the same problem. It is conceivable that instead of observing exponential vanishing gradients during the optimization, we would observe them during the operator selection step. However, chemistry-based considerations related to size-consistency suggest that gradient troughs do not deepen with system size [71]. Gradient troughs are observed numerically, not proved to exist analytically; they only arise for a subset of molecular systems, and are related to low-lying energy eigenstates rather than particular characteristics or shortcomings of the ansatz and parameter initialization.

2.2.3.10 Other Areas of Application

Finally, we note that while the focus of this thesis is on molecular ground state preparation, ADAPT-VQE has since its inception been extended to cover a wide range of applications. Variants of the algorithm have been proposed for preparing mixed and exited states [148, 70, 171, 181, 127, 60, 152, 187, 16, 15, 91, 31, 187, 187, 184, 79], studying dynamics [177, 185, 66, 99, 65], preparing ground states of time-periodic Hamiltonians [92], subspace diagonalization [119], classical optimization [188, 32, 157, 175, 162], preparing ground states of the Hubbard model [79], solving nuclear structure problems [144, 76, 128], simulating multi-orbital impurity models [118], materials design [18], studying vibrational properties [106], calculating the berry phase [116], solving linear equations [131], generating probability distributions [98], function fitting and data classification using Kolmogorov-Arnold networks [168], and many other tasks.

2.2.4 L-VQE

L-VQE [101] is a proposal of dynamic ansatz for k-community detection problems. To a certain degree, it is a hardware-efficient version of ADAPT-VQE. We follow with a superficial step-by-step description of the algorithm.

1. Initialize a circuit with a $R_Y(\theta_i)$ gate in each qubit i, where $\vec{\theta}$ is chosen randomly. Optimize these variational parameters.
2. Append to the ansatz a hardware-efficient type parameterized layer initialized to identity.
3. Optimize all variational parameters.
4. If the desired number of layers has not been reached, go to 2.

For the sake of clarity, the above procedure disregards the fact that all optimizations but the last one terminate after a fixed number of iterations (instead of continuing until convergence). The authors claim that this improves resilience against local minima. However, they did not conduct a thorough assessment of this claim and the number of iterations is selected empirically.

Unlike ADAPT-VQE, L-VQE does not use a selection criterion to choose new operators and grow the circuit. Instead, it iteratively grows the ansatz by adding the same basic circuit unit until a certain circuit depth l is reached. This basic unit is of the hardware-efficient type, consisting of Y rotation gates and nearest-neighbor CNOTs. When the parameters are at the initialization point (zero) this acts as the identity. This property seems to be associated with improved trainability. In Ref. [68], the authors attempted to limit the initial effective depth of ansätze by imposing parameter restrictions that forced the circuit to consist of a sequence of shallow blocks, each acting as the identity. The results demonstrated that this prevents the variance of the gradients at the start of the algorithm from decreasing exponentially with the system size—i.e., the strategy avoids starting the optimization in a barren plateau. This seems to be the only motivation for the design of the basic unit circuit of L-VQE; the choice is not problem-tailored, and the authors do not explore other possibilities.

The evolution of L-VQE is completely deterministic: W know in advance which will be the structure of the circuit. The only reason why we need to grow the ansatz from scratch, instead of just inserting the l layers at once, is that we need to perform each optimization $(1, 2, ..., l-1)$ in order to know the initial parameters for the next one. Thus, this proposal seems closer to an initialization/training strategy than an ansatz creation procedure. In fact, we can view L-VQE as a fixed-structure ansatz with l layers, which we optimize with a layerwise training strategy: First, we train the parameters of layer 1 fixing to zero the parameters from layer $l > 1$; then, we train the parameters from layers 1 and 2 using the respective final parameters from the last iteration as initial parameters, and fixing to zero the parameters from layer $l > 2$; and so on. This strategy has also been successfully applied to QAOA [97], in which case the authors introduced it as a training strategy ("parameters fixing strategy") rather than as a new algorithm.

Due to the hardware-efficient circuit structure and the absence of problem- and system-tailoring, L-VQE is not competitive with ADAPT-VQE. We summarize its shortcomings as follows:

- Arbitrariness in the choice of ansatz unit
- Predetermined circuit structure
- No motivated specification of circuit size (number of layers l is given as a hyperparameter)
- Narrow application (k-community detection)

2.2.5 E-VQE

E-VQE [142] was one of the earliest dynamic ansätze, along with ADAPT-VQE. It is a genetic algorithm [114] which mirrors the processes of natural selection, and was proposed for all problem domains (quantum chemistry, combinatorial optimization, etc.).

In this algorithm, the fundamental building blocks for the circuit are viewed as *genes*. Genes are essentially parameterized circuit layers with a certain structure, but unbound parameters. Our possible ansätze are *genomes*, constituted by *gene instances* in sequence. These gene instances are genes with bound parameters. The same gene can appear multiple times in a genome, with different parameters

(corresponding to different gene instances). A concept of genetic distance, or similarity, is defined so that similar genomes correspond to similar circuit structures.

We describe the essential steps of the algorithm below.

1. Create a population of P genomes and grow them by adding a gene.
2. Divide the genomes into species (grouping them by similarity). Randomly select one genome to represent each and, with these elements, create a *species representative set* S.
3. Optimize the last gene of each genome.
4. Calculate the fitness of each genome.
5. Select a genome according to probabilities proportional to their fitness. Repeat P times, with replacement. The P selected genomes will be the parents of the new generation.
6. To each of the P selected parent genomes, apply *mutations* (topological search, parameter search, or removal) with some probability.
7. Assign a species to each genome: Go through S and as soon as an element is similar enough, assign it to the same species. If no such element exists, add the genome to S (creating a new species with this genome as its representative).
8. If G generations have passed, terminate, returning the fittest genome. Otherwise, go to 2.

In contrast with most dynamic VQEs, we don't keep a single ansatz, rather a *population* of them. This allows us to explore different circuit structures in parallel, but it also introduces an optimization overhead.

Earlier (non-variational) algorithms had applied the ideas behind genetic algorithms to the quantum setting with the goal of synthesizing circuits corresponding to certain unitaries or preparing certain states [45, 104, 147]. Unlike such proposals, E-VQE is a variational, multi-purpose, and asexual approach, where evolution is achieved by means of *mutations*. This is in contrast with sexual schemes, which use *crossover* between two genomes to produce a child sharing characteristics with both parents (e.g., composing the first half of genes from one parent with the second half from the other). The benefits of crossover in the quantum setting are unclear: Since the value of the cost function evaluated on a given circuit tells us nothing about its value on parts of the circuit, there is no indication that the offspring produced by two fit genomes will also have a high fitness. Further, there is no reason to believe that the optimal parameters for a gene in a parent will be optimal for the same gene in their child; due to this, crossover is likely to result in sharp variations in fitness and parameter values between generations. The same reason might cause fitness to decrease from one generation to the next and lead to cold-started parameters for continuous optimization. We note that recently, a sexual reproduction scheme was applied to variational state preparation with some success in an algorithm termed Genetic Algorithm for State Preparation (GASP) [36]; however, it is unclear whether the algorithm actually benefited from crossover, as GASP was not benchmarked against any asexual variant.

In view of the shortcomings of sexual reproduction schemes in the context of VQAs, E-VQE creates new generations by applying mutations to some of the genomes from older populations; i.e., we create the ansätze we consider at iteration $i + 1$ from ansätze we were considering at iteration i, by applying some changes probabilistically. Most mutations are designed to at least maintain fitness; for example, whenever a new gene is added to a genome (one of the possible mutations), its parameters are chosen so that they implement the identity operation. This bypasses the problems we mentioned above in the context of crossover.

The *fitness* is a score for the genomes (ansätze), which will inform our choice of parents for the next generation: *Fitter* species have a greater chance of reproducing. Despite this, the authors aim to prevent a highly fit species from growing uncontrollably and outnumbering all others, thereby hindering the exploration of different circuit structures. Since a less fit genome may later become fitter, it is positive to foster genetic variety. In light of this, the authors chose the fitness to be species-adjusted, i.e., divided by the size of the species.

Another distinctive point in the definition of fitness of Ref. [142] is that it considers not only the value of the cost function but also the depth of the circuit and the number of two-qubit gates. However, the relative weights α and β of the latter two factors with respect to the former are left as hyperparameters. The authors do provide a guideline for choosing them, suggesting that $\alpha D + \beta K$, where D is the depth of the circuit and K is the number of two-qubit gates, should correspond to the desired precision. However, this guideline is vague, and even in demonstrations of the original paper (including numerical simulations as well as executions on hardware) the choice varied widely; for example, α ranged from 5×10^{-6} to 5×10^{3}. The largest value was chosen when running the algorithm in quantum hardware (IBMQ's Vigo [84]) and does not seem to follow the guidelines proposed by the authors, since it would correspond to a precision of the order of over 10^{3} for non-trivial circuits. This raises concerns regarding the selection of such hyperparameters.

Furthermore, we note that when running the algorithm on quantum hardware, the cost function will naturally be penalized by noise. Let us assume a new ansatz layer lowers the value of the cost function in an ideal machine, but leads to an increase in the presence of hardware noise. This will naturally penalize excessive circuit depth: In practice, the information obtained from the measurement indicates that the cost value has increased. It is unclear whether explicitly enforced penalties offer an advantage, especially considering that this complicates the protocol by introducing additional hyperparameters with no known optimal ranges.

In the proposal of E-VQE, not much attention is paid to the choice of genes that end up dictating the structure of the circuit. Genes are created by selecting gates randomly (for each qubit) from three possibilities: the identity gate, a generic single-qubit rotation, or a controlled version of the latter. No alternative options are tested or suggested, nor is a justification provided for this choice. The main driving factor appears to be the usual hardware limitations (namely, nearest neighbor connectivity); yet, within these, there are still many other possible choices. We remark that controlled versions of generic single-qubit rotations are not, in general, native operations.

Regarding the mutations (step 6) that drive the evolution of our genomes, they are three: topological

search, which amounts to adding a new gene (i.e., growing the ansatz); parameter search, which optimizes the genome layer-by-layer in a random order; and removal, which removes an arbitrary number of genes from the end of the genome. We note that mutations are the only way by which genomes can increase or decrease in size, and also the only way by which gene instances within the ansatz can be optimized.

Finally, we have the termination condition: having gone through a certain number of generations G, which is left as a hyperparameter. The authors provide no heuristics for choosing this number and no discussion on its scaling with the system size. It is unclear whether this indicates that we've reached a good-quality solution.

In light of the discussion above, we consider the following to be shortcomings of E-VQE:

- Arbitrariness in the choice of genes, genomes and mutations
- Circuit depth and two-qubit gate count penalties lack justification and guidelines
- No motivated termination criterion

2.2.6 VAns

Another proposal is the VAns [22], an ansatz created dynamically from problem-agnostic blocks. Because the final structure depends indirectly on the cost function, the ansatz is problem-driven, as happens in E-VQE (but not L-VQE). The protocol can be summarized as follows:

1. Choose an initial circuit.
2. Stochastically insert a new block from a dictionary of parameterized gates, with parameters initialized such that it corresponds to identity.
3. Simplify the resulting circuit.
4. Reoptimize the parameters and decide whether to accept the change.
5. If the convergence criterion has not been met, go to 2.

There are many free choices within the protocol, namely the initial circuit, the dictionary of parameterized gates, the simplification rules, the acceptance condition, and the convergence criterion. The particular instantiation of the algorithm they consider in the paper is merely an example; while there is some reasoning behind the choices, there is no reason to believe they are close to optimal. While the authors do provide some guidelines on how to choose these elements, they are vague; as such, there is no clean-cut procedure to construct the ansatz, and the VAns might be better regarded as a broad class of algorithms with common features. In fact, we could roughly view any ADAPT-VQE variant as a VAns with particular choices. We predict that unless the formulation becomes more precise in future work, there will be few performance guarantees regarding this algorithm.

We now succinctly describe the guidelines for selection of the unlatched elements of VAns. We begin with the initial circuit configuration. While the authors note that this can impact convergence, they do not

provide recommendations on what constitutes a good option. Two choices tested in the simulations are left as an example for the reader, but they are not based on solid heuristics. Further, problem-specific symmetries are not considered. It has been shown that starting off in a symmetry-broken state hinders convergence [21].

We follow with a discussion of the dictionary of parameterized gates that are used as the building block of the ansatz. First, we note that what they mean by *parameterized gates* is broader than usual and encompasses more than the typical rotations about an axis. The examples they provide consist of multi-parameter unitaries acting on one or two qubits; however, in principle, these parameterized gates may act on the entire system. The only condition imposed on the elements of the dictionary is that they should implement the identity for some parameters. Aside from this detail, the choice of gates is not specified. Finally, the interplay between the initial circuit and the dictionary of parameterized gates is not explored. It is expected that these choices will have a considerable influence on one another.

The fact that the parameters in each block are initialized to the values that implement the identity operation is related to the improved trainability that we discussed earlier. We note that, in general, there are many ways to choose the parameters so that the operator amounts to the identity; therefore, this guideline does not fully specify the initialization.

Regarding the simplification rules, most amount to using circuit identities (canceling back-to-back `CNOT`s, eliminating redundant rotations, etc.), which are more akin to compilation strategies than algorithmic features. The distinctive simplification rule is the removal of parameterized gates (the blocks considered in the dictionary). It is suggested that this removal operation may be applied to circuit blocks that are predicted not to have a sizable impact on the cost function by taking cues from the inner optimization loop (e.g., low gradients from a gradient-based optimization). Once an operator is removed, the cost function is remeasured. To avoid a heavy impact from flawed predictions, the operator is only effectively removed if its impact on the energy is below a certain threshold.

It is unclear how many times the simplification rules should be applied. The circuit compilation steps may be repeated until they no longer apply, but the removal operation could in principle be attempted for all ansatz blocks, at the expense of an increased number of optimizations (and measurement costs). Conceivably, one must either choose some *threshold* criterion and attempt to remove *all* operators which satisfy it, or choose some *ranking* criterion and attempt to remove the k operators with the highest/lowest score, where k is a fixed number. The choice in the original implementation is not made clear in Ref. [22].

A crucial point to note is that when a gate removal is attempted, no parameter re-optimization occurs. This means that the algorithm might attempt to remove an operator, see the value of the cost function increase too much, and choose to keep it; while if we had reoptimized the parameters, the energy could be kept constant (it cannot be lower, since, for some parameters, the operator amounts to identity).

After insertion/simplification, parameters are finally re-optimized to obtain the new value of the cost function. At this point, it should be decided whether to accept the update or not, as dictated by the acceptance condition. The authors chose to accept an operator if it maintains or lowers the value of the cost function, and accept it with an exponentially decaying probability otherwise. One could imagine an acceptance criterion that factors in circuit-related costs: e.g., among circuits that correspond to a similar

increase in the cost function value, favor shallower ones to promote the creation of a more compact ansatz. However, this possibility is not contemplated in the original protocol.

Finally, although the existence of a convergence criterion is assumed, it is left unspecified. The authors allude to termination being triggered by a maximum number of iterations; however, this choice is not motivated, nor is there an indication of how it should be adapted to different problem types and sizes.

The main shortcomings of VAns are thus:

- Arbitrariness in the choice of ansatz units
- Arbitrariness in choice/number of removal attempts
- Acceptance criterion for updates does not factor in highly varying circuit-related costs
- No motivated termination criterion

2.2.7 DISCO-VQE

DISCO-VQE [28] is one of the few dynamic ansätze that uses similar circuit units to ADAPT-VQE. We leave a simplified description of the algorithm below.

1. Initialize the ansatz with M operators acting on the Hartree-Fock state. These operators are randomly selected from a pool of size N_{pool}.
2. Optimize the M variational parameters using basin-hopping steps.
3. For each possible cyclic permutation acting on the current ansatz, create a new ansatz. Optimize all parameters for each $M-1$ ansätze. Keep the best option.
4. For each ansatz operator i from 1 to M, create $N_{pool}-1$ ansätze by replacing it with each pool operator. Optimize all parameters for each option. Select one of the changes based on a 'temperature-dependent' procedure and accept it based on a Metropolis criterion.
5. For each ansatz operator i from 1 to M, create $M-1$ ansatze by swapping it with each other ansatz operator. Optimize the parameters for each option. Select one of the changes based on a 'temperature-dependent' procedure and accept it based on a Metropolis criterion.
6. If convergence has not been reached, go to 2.

Taboo criteria may be applied to prevent, e.g., revisiting a certain configuration too often. We purposefully omitted this detail to avoid overloading the steps with information.

This algorithm differs from the previous ones in that it does not *grow* the ansatz on the fly; instead, the ansatz consists of a fixed number of operators M. The *dynamic* element is that the ansatz operators can be changed around between themselves, or replaced with others from a pool. The chosen pool operators are well-suited for quantum chemistry and were shown to allow for convergence within the

DISCO-VQE framework. In contrast, no motivation is provided for the choice of ansatz dynamics: The discrete optimization over operator ordering is done via cyclic permutations and swaps, and further adding to that freedom is the possibility of mutations that exchange an ansatz operator with another operator from the pool.

A big shortcoming of the algorithm is that no reason is given to believe that this selection of discrete steps is optimal or near-optimal. In fact, the authors mention that in later stages of the optimization, one of these steps (cyclic permutations) rarely improves the solution and may actually harm convergence. They tackle this problem by never accepting cyclic permutations unless they lower the energy, which is an imperfect solution: Under this approach, such changes will only ever improve convergence, but their often purposeless cost remains hidden.

Further, this suggests that the discrete modifications to the ansatz lead to troubles in the continuous parameter optimizations, something we could already have speculated based on the fact that the naturally warm-started parameters in each ADAPT-VQE iteration seem to improve trainability and lead to smooth optimization landscapes [71, 10], something that has also been shown for QAOA [97] and which bears some resemblance with the layerwise training strategy that has been suggested to help in preventing barren plateaus [154].

One final shortcoming of DISCO-VQE is that the protocol makes no mention of how many operators the ansatz should include. This is not a trivial question, since, in a realistic run, we do not have access to the solution. Additionally, no termination criterion is specified, although it seems likely that the algorithm terminates when the energy change between macro-cycles falls below a given threshold.

We thus sum up the shortcomings of DISCO-VQE as:

- Arbitrariness in the choice of discrete steps
- No motivated specification of circuit size or termination criterion
- Cost overhead from discrete optimization
- Narrow application (electronic structure problems)
- Discrete steps may lead to non-smooth landscapes in the continuous parameter optimizations

The first two shortcomings lead to skepticism about whether the substantial cost overhead resulting from the large number of optimizations required by the discrete optimization of the ansatz is being effectively utilized, which reinforces the weight of the third shortcoming. In total, each macro-cycle of DISCO-VQE implies close to $M^2 + MN_{Pool} + M$ optimizations, versus the 1 optimization per iteration of ADAPT-VQE.

While the numerical results section of the paper compares the DISCO-VQE and ADAPT-VQE ansätze for the same number of operators M and concludes that the former is superior, we note that this comparison seems unfair given the difference in optimization counts: $\approx k(M^2 + MN_{Pool} + M)$ optimizations of dimension M versus M optimizations of dimension $1, ..., M$, where k is the number of macro-cycles

DISCO-VQE goes through. This is further aggravated by the fact that there is no mention to the value of k; if this number is large enough, DISCO-VQE could be finding the best ansatz by brute-force, which is not a scalable approach.

Regarding the last point we include in the shortcomings, we should note that it needn't *necessarily* be a shortcoming, since the stochasticity from the discrete steps may actually prevent the optimizer from falling in local traps. However, there is no reason to believe that it is not possible to reach a good solution with the iterative growth procedure that leads to warm started parameters. If it is possible to smoothly converge to a solution using such procedure, then the cost overhead the cold-started optimizations impose on DISCO-VQE (via added optimization difficulty) would prove unnecessary.

2.2.8 QNP

The QNP gate fabric is an ansatz that preserves particle number, Z spin projection, and $\langle S^2 \rangle$, while also taking hardware considerations into account by using local operators. It is created by alternating two building blocks:

- $\text{QNP}_{\text{OR}}(\theta)$: A four-body operator equivalent to two single excitations whose parameters are coupled to ensure preservation of fermionic quantities.
- $\text{QNP}_{\text{PX}}(\theta)$: A four-body operator that is equivalent to a double excitation, provided the Jordan-Wigner mode order is specifically selected to ensure the anticommutation Z string amounts to identity.

The authors propose two distinct initialization techniques for this ansatz: the Hartree-Fock state, or the Hartree-Fock state with a pair of coupled one-body exchange gates acting on it.

We list the main shortcomings of the QNP gate fabric below.

- No heuristics for choosing the number of layers.
- Initialization heuristics do not produce good results for all molecules.

2.2.9 tUPS

The tUPS ansatz builds off of the fact that an arbitrary wavefunction with the same particle number and spin symmetry as $|\psi_0\rangle$ can be represented as the product of M fermionic operators acting on $|\psi_0\rangle$ [28]:

$$|\psi(\theta)\rangle = \prod_{I=1}^{M} \exp\left\{\theta_I \hat{\kappa}_{\mu_I}\right\} |\psi_0\rangle\,, \tag{2.18}$$

where $\hat{\mu}_I$ represents a discrete parameter, and $\hat{\kappa}$ represents either a spin-adapted one-body operator or a paired two-body operator defined as follows:

$$\begin{aligned} \hat{\kappa}_{pq}^{(1)} &= \hat{E}_{pq} - \hat{E}_{qp}, \\ \hat{\kappa}_{pq}^{(2)} &= \hat{E}_{pq}^2 - \hat{E}_{qp}^2. \end{aligned} \tag{2.19}$$

These operators are constructed from the singlet excitation operator $\hat{E}_{pq}$, which conserves spin statistics when applied to a state. It is defined as

$$\hat{E}_{pq} = \hat{p}^\dagger \hat{q} + \hat{\bar{p}}^\dagger \hat{\bar{q}}, \tag{2.20}$$

where p, q denote spatial orbitals, presence/absence of overbar indicates β/α spin part, and we define $i \equiv a_i$, $i^\dagger \equiv a_i^\dagger$ to simplify the notation.

Importantly, Eq. (2.18) is only able to represent an arbitrary (in the relevant symmetry subsector) wavefunction in the limit $M \longrightarrow \infty$. The tUPS ansatz truncates this expression and, building on the QNP [11] fabric, defines the parameterized wave function to be [27]

$$\left|\psi_{\text{tups}}(\vec{\theta})\right\rangle = \prod_{m=1}^{L}\left(\prod_{p=1}^{A} \hat{U}^{(m)}_{2p+1,2p} \prod_{p=1}^{B} U^{(m)}_{2p,2p-1}\right)|\psi_0\rangle\,, \tag{2.21}$$

where L represents the number of layers, $A = \lfloor \frac{N-1}{2} \rfloor$, and $B = \lfloor \frac{N}{2} \rfloor$, where $\lfloor x \rfloor$ denotes the floor function. Each U operator has three distinct parameters, one attached to each operator, and is defined as

$$\hat{U}^{(m)}_{pq} = \exp\left\{\theta^{(m)}_{pq}\hat{\kappa}^{(1)}_{pq,1}\right\} \exp\left\{\theta^{(m)}_{pq,2}\hat{\kappa}^{(2)}_{pq}\right\} \exp\left\{\theta^{(m)}_{pq,3}\hat{\kappa}^{(1)}_{pq}\right\}. \tag{2.22}$$

The tUPS ansatz preserves key fermionic quantities like $\hat{N}$, $\hat{S}_z$, and $\hat{S}^2$, which ensures that the ansatz state lives in the same symmetry subsector as the ground state [27]. However, imposing such strong restrictions on the ansatz leads to an energy landscape filled with local minima. A consequence of this is that classical optimizers will typically converge to a shallow local minimum after starting, leading to high errors. In order for the tUPS ansatz to converge to low errors in the original proposal, basin-hopping parallel tempering (BHPT) was used with 8 parallel replicas and 1000 steps [27, 28], amounting to 8000 total optimizations. This was the number of optimizations necessary to find the solution for small molecules such as H_6 (12 qubits), with the expectation of even worse requirements for larger molecules. Hence, a major downside of tUPS is that finding the optimal parameters requires very significant measurement costs.

Considering the discussion above, we highlight the following shortcomings of the tUPS ansatz:

- No heuristics for choosing the number of layers.
- Difficult optimization landscape filled with local traps.

Finally, we note that two main variants of tUPS exist, which we describe below.

OO-tUPS A variant that uses orbital optimization to improve the solution. The orbital update can be implemented classically as a transformation of molecular integrals. Computing the gradients for the optimization requires at most the two-body reduced density matrix, which is measured for the energy already. As such, there is no increased measurement cost or number of gates.

PP-tUPS A variant inspired by PP, a form of valence bond theory. Since tUPS only includes excitations between sequential spatial orbitals, it is helpful to alternate the initial register between occupied and empty spatial orbitals. If we use the Hartree-Fock as the reference, the impact of the first layer is suboptimal, because operators exchanging electrons between same-occupancy spin-orbitals will have trivial action. The PP wave function is a product of wave functions for each electron pair in two spatial orbitals (one occupied and one virtual). The first part of each layer of tUPS implements a layer of single excitations that can exactly solve each subproblem (2 electrons in 2 orbitals) independently, so it can yield a direct product of the optimized two-orbital waveforms, much like the PP wave function.

2.2.10 QAOA

So far, we have explored near-term algorithms specifically aimed at finding eigenstates of physical Hamiltonians. In contrast, the QAOA algorithm [47] was proposed as a hybrid quantum-classical algorithm to produce approximate solutions for combinatorial optimization problems. It can be viewed as an attempt to make the QAA more amenable to NISQ hardware.

We note that, unlike other VQAs, QAOA is often framed as an algorithm aiming to *maximize* a function. This is a flexible choice, since we can turn any maximization problem into a minimization problem (and vice versa) without distorting the optimization landscape by simply flipping the sign of the cost function.

The algorithm fits within the structure of a VQA (figure 2.1). The distinctive point is the structure of the parameterized portion of the ansatz $U(\vec{\theta})$, namely, the fact that it explicitly depends on the cost Hamiltonian H_C.

The reference state is the uniform superposition state $|\psi_{\text{ref}}\rangle = |+...+\rangle$, which can be prepared with a single layer of Hadamard gates. Then, for each layer $i = 0, ..., p-1$, we do:

1. Time evolution under the cost Hamiltonian $e^{i\gamma_i H_C}$ (total 1 variational parameter, variable depth and gate count)

2. Time evolution under the mixer Hamiltonian $e^{i\beta_i H_B}$ (total 1 variational parameter, unit depth, N gate count)

The number of layers p is tunable, and it fixes the number of variational parameters ($2p$) as well as the circuit depth.

The structure of the cost Hamiltonian layer depends on the problem and on the problem instance. Let us take *Maxcut* in a graph with N vertices as an example [47]. In the worst case scenario (dense problem graph, i.e., nearly all-to-all vertex connectivity), the number of edges will be $O(N^2)$, so there will be $O(N^2)$ ZZ terms in the Hamiltonian and hence also $O(N^2)$ two-qubit rotations in the corresponding layer. The depth will be $O(N)$.

The mixer Hamiltonian is defined as $H_B = \sum_{i=1}^{N} X_i$. The key thing to note is that the starting state $|\psi_{\text{ref}}\rangle$ is the maximum energy eigenstate of this operator; this is related to the inspiration of this ansatz on the QAA, which we will briefly explain in 2.2.10.

We note that an extension of QAOA that allows for different types of mixers, the Quantum Alternating Operator Ansatz (QAOA), was proposed in [80].

Relation with the Adiabatic Theorem If p is large enough, the QAOA circuit is guaranteed to create a state in which the expectation value of the cost function is as close to the maximum as we want for some set of angles (variational parameters). This is because alternated application of time evolution under the mixer Hamiltonian H_B and the cost Hamiltonian H_C can apply an approximation to an adiabatic evolution leading $|s\rangle = |+...+\rangle$ (the starting state of QAOA and the maximum eigenvalue eigenstate of the mixer Hamiltonian H_B) to the solution (the maximum eigenvalue eigenstate of the cost Hamiltonian H_C). Of course, this does not mean that the solution can be found *efficiently*: p may need to grow exponentially with the problem size. While it serves as inspiration for the algorithm, the adiabatic theorem does not provide any guarantees regarding QAOA. The hope is that the added variational freedom might allow one to reach the solution with shallower circuits: Perhaps the flexibility in the choice of parameters can compensate for the approximations made with respect to the QAA circuit that would implement the ideal, adiabatically slow evolution.

A strong point of QAOA is that the approximation improves as p increases: The optimization for p layers can be viewed as the optimization for $p' > p$ layers, with the added constraint that the parameters of all layers after p should be zero. As such, the QAOA search space with p layers is contained within the QAOA search space with $p' > p$ layers. This is in contrast with the QAA (which we explored in subsection 2.1.1), in which the quality of the solution is not guaranteed to increase for increasing run times—which, in a circuit-based quantum computer, would amount to increasing circuit depths. In the paper where QAOA was originally proposed [47], the authors provided an example where QAA is trapped in a local minimum for sub-exponential runtimes, while the QAOA output state is concentrated near the true minimum for large problem sizes even with $p = 1$. In this case, QAA fails and QAOA succeeds.

Cost Function A peculiarity of QAOA is that given the nature of combinatorial problems, H_C is diagonal and the solution is a bit string, not a superposition state. In each iteration, state preparation with the current parameters is repeated, and bit strings are sampled from the output distribution (computational basis measurements). The value of the cost function is evaluated classically for each bit string using parity checks. This allows one to calculate the expectation value of H_C. The string output by QAOA is the best one among all those that were measured. The output value for the algorithm is the value of the objective function for the best bit string, not the expectation value of H_C in the superposition state; however, it is the expectation value of H_C that is maximized by the parameter optimization.

Ideally, QAOA will produce a superposition state where a good solution is likely to be measured. While its success (or lack thereof) stems from the overlap with the optimal string (or nearly-optimal ones), the QAOA optimization is led by the expectation value $\langle H_C \rangle$. We note that the relation between this expectation value and the quality of the best sampled bit string is not straightforward.

To illustrate this, consider a state whose measurement yields the optimal bit string, with objective value 10, with a 10% probability, and the worst bit string, with objective value -10, with a 90% probability.

In this state, we have $\langle H_C \rangle = -8$, far from the optimal value (10). As such, this state is unlikely to be output by QAOA. However, this would actually be a good output state, since 1 out of 10 measurements would yield the optimal bit string. In contrast, a state with 100% probability of sampling a bit string with objective value 5 would never yield the optimal solution, but would be more likely to be found by the optimizer given the higher value of $\langle H_C \rangle$ in this state ($5 > -8$).

We could think of using the minimum value of $\langle H_C \rangle$ among all observed bit strings in a given iteration as the objective function. Unfortunately, this is likely to lead to an ill-behaved optimization landscape whose minimum is harder to find. An interesting proposal is using the Conditional Value at Risk (CVaR) [17]. The $\text{CVaR}_\alpha(X)$ is the expected value of the lower α-tail of the distribution of X (i.e., the average value of the lowest $\alpha\%$ values of X: for example, $\text{CVaR}_{0.1}(X) = 0.5$ means that the average value of the 10% lowest values of X is 0.5).

When $\alpha \longrightarrow 0$, $\text{CVaR}_\alpha(X)$ tends to the minimum of X; when $\alpha = 1$, it is the expected value of X. Intermediate values of α interpolate between the two extremes. The benefit of CVaR is that it puts a greater emphasis on the best observed samples (the lower α-tail of the distribution) than the sample mean, while having a smoother behavior than the minimum.

The downside is that in order for the empirical CVaR estimator (determined with a finite shot count) to have the same variance as the sample mean with K samples, $\frac{K}{\alpha}$ samples are required. If we choose $\alpha = 0.1$, this implies a 10-fold increase in the number of samples required for identical accuracy. However, if we choose α as a constant (independent of system size), the multiplier on the number of shots does not grow with the system size (for a fixed accuracy).

Parameters There are some interesting results regarding the behavior of the optimal values for the variational parameters in QAOA.

In the paper that proposed QAOA [47], it was shown that if p is fixed (i.e. does not grow with the size of the system) and each bit is in (at most) a fixed number of clauses of the cost function, then there is a classical algorithm that efficiently optimizes the expectation value over the parameters. In this case, to obtain the result, we just need to prepare the corresponding state with a quantum computer, and sample from it (bypassing the typical optimization loop of VQAs, Fig. 2.1). With $m \log m$ repetitions, there is a $1 - \frac{1}{m}$ probability of sampling a bit string with expectation value $C^* - 1$ (C^* the maximum, i.e. optimal, value).

Another interesting result is that the optimal QAOA parameters exhibit concentration across different problem sizes for a fixed number of layers p [4]. More precisely, they seem to concentrate as an inverse polynomial on the problem size. If this is the case, we can train a QAOA circuit on a fraction of the qubits, and assert that these parameters are nearly optimal for the full instance, reducing optimization (and thus measurement) costs. However, we note that these results have the limitation that they only apply to the optimal parameters, and there is no description of the landscape surrounding them.

A different, and more general, type of concentration was shown in [24]: the authors found that the *optimization landscape* is identical across different problem instances if these instances come from a reasonable distribution (e.g. MaxCut over all 3-regular graphs). In this case, keeping the same parameters

will lead to identical cost function values, not only for the optimal parameters, but for all others too. An evident advantage of these is that we can reduce training costs by 'recycling' the QAOA parameters from one problem instance to another.

While these results are interesting, they only apply to very specific scenarios and do not tell us how to train the QAOA ansatz in general. Concerns have been raised that, while increasing p will cause the variational form to contain an increasingly better solution, the optimizer may not be able to find the solution due to being trapped in local maxima. Parameter initialization strategies are key to avoiding this; we refer to [97] for an example. The authors propose growing the ansatz from $p = 1$ to the desired number of layers, recycling the parameters from one iteration to the other. They compare this parameter fixing strategy with random initialization. While the latter barely improved the approximation ratio as layers were added after the first few, the former consistently improved the solution as layers were added, and output quite higher approximation ratios. However, the results were obtained using a single optimizer (Nelder-Mead), only one problem type (MaxCut), and only a few different problem instances (Erdos–Rényi and 3-regular graphs). Further research is necessary to establish the performance of the strategy in general.

Non-Simulability It was proved in Ref. [48] that, based on reasonable assumptions, even single-layer QAOA is not efficiently simulatable with a classical computer. More precisely, the authors prove that if it is possible for a classical computer to efficiently reproduce the output distribution of QAOA with $p = 1$, then the polynomial hierarchy collapses at its third level, a result widely considered to be unlikely in theoretical computer science.

Additionally, the authors argue that it may be easier to classically sample from the output of QAA. Specifically, if the Hamiltonian governing the evolution is stoquastic and QAA can be realized efficiently, then, under the same complexity-theoretic assumptions, QAA is easier to simulate than QAOA. A stoquastic Hermitian matrix is one which has real non-positive off-diagonal elements. While not required for the success of QAA, this condition on the Hamiltonian is often used as a way of guaranteeing that the gap g_{min} is non-vanishing throughout the evolution (which is necessary for the success of the algorithm).

2.2.11 ADAPT-QAOA

ADAPT-QAOA was proposed as a dynamic version of QAOA. As happens in ADAPT-VQE, the ansatz is grown iteratively from scratch; the difference is that in this case, the circuit block we add in each iteration resembles a QAOA layer.

While the cost Hamiltonian layer is exactly the same, the mixer Hamiltonian layer is allowed to differ: In each iteration, we choose H_B from an operator pool, instead of always using $\sum_i X_i$ as in the original algorithm. Much like what happens in other ADAPT algorithms, the choice of operator is dictated by the gradient rule.

The authors consider two possible *mixer pools*:

- P_{single}, consisting of X_i and Y_i on any qubit, as well as $\sum_i X_i$ and $\sum_i Y_i$.

- P_{multi}, which consists of the union of P_{single} with the set of all length 2 Pauli strings.

In numerical simulations, ADAPT-QAOA with P_{multi} was shown to result in a significant reduction in the `CNOT` count (with respect to QAOA) to reach a comparable error. The paper relates this improvement to the concept of *shortcuts to adiabaticity* or *counterdiabatic driving*.

ADAPT-QAOA has been used along with QAOA and symmetry-broken QAOA to study the relation between entanglement in the QAOA wave function and convergence / solution quality [32]. The motivation behind this study is that, in this algorithm, both the starting state and the optimal state are separable. This raises the question of how much entanglement we need. If we are interested in a separable state, producing a highly entangled state may not be desirable and can make the optimization more difficult. Numerical simulations seem to show that ADAPT-QAOA is able to entangle and disentangle qubits with more flexibility than QAOA, reaching the optimal level of entanglement entropy with greater ease.

3

Recycling the Hessian: An ADAPT-VQE-Tailored Quasi-Newton Optimizer

This chapter is associated with the following authored publication:

Reducing Measurement Costs by Recycling the Hessian in Adaptive Variational Quantum Algorithms. **M. Ramôa**, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. Quantum Sci. Technol. 10 (2024) 015031

3.1 Motivation

The choice of which classical optimizer to employ in a VQA is crucial, as it affects both the algorithm's costs and the solution quality. BFGS, a quasi-Newton optimizer which uses approximate second derivatives of the cost function to inform the search direction while avoiding the (costly) requirement of explicitly measuring them [85, 43, 58, 125], is a popular choice [77, 103, 72, 182, 108]. A typical implementation of this algorithm will initialize an approximate inverse Hessian H, a matrix with all second derivatives, at the identity. Curvature information is gathered as the iterations proceed by performing rank-2 updates on this matrix, which are based solely on changes in the gradient and parameter vectors. As H is updated, the search direction evolves from a vanilla gradient descent into an approximation to Newton's direction.

In a typical implementation of an adaptive VQA with the BFGS optimizer, the approximate inverse Hessian is reinitialized to the identity matrix each time the ansatz is extended. The re-initialization of H seems discordant with the recycling of the parameters: The initial state at iteration n is exactly the same as the final state from iteration $n-1$, and thus all previous approximate second derivatives remain as accurate as before.

In this chapter, we propose a strategy to recycle the Hessian in adaptive VQAs. At the end of each VQE subroutine, we save the collected curvature information and use it to inform the initialization of H for the next optimization. With our protocol, the state of the optimizer, with all its knowledge of the cost landscape, is transferred between optimizations instead of being erased and restarted from scratch. We show that as a result, the search direction aligns more quickly with Newton's direction, and superlinear convergence is attained more often. Surprisingly, recycling old second derivatives allows the optimizer to

better approximate new ones, too, allowing new correlations to be captured earlier in the optimization. Our strategy leads to a significant decrease in the number of function evaluations required per optimization, which translates to a decrease in the number of calls to the quantum processor. For the molecules we studied, with 12 and 14 qubits, our protocol reduces the total measurement cost of the algorithms by an order of magnitude. We expect the impact to become more significant for larger system sizes.

The measurement costs are one of the most critical limitations of adaptive VQAs. They may be alleviated in two ways: by decreasing the number of measurements required to obtain the pool gradients, or by decreasing the number of measurements required in the optimization. Previous works have focused on the former [153, 9], such that the cost bottleneck now resides in the optimization process. To our knowledge, the strategy we propose is the first to tackle this source of costs, and the first to propose an optimizer specifically tailored to adaptive VQAs. We note that its relevance is not limited to implementations on quantum hardware—it also reduces the runtime of classical simulations of these algorithms, which can take hours even for relatively small molecules such as H_6 on a minimal basis set (12 qubits).

This chapter is organized as follows. In Sec. 3.3, we cover the relevant gradient-based optimization methods. In Sec. 3.4 we introduce our main proposal, a BFGS algorithm tailored to ADAPT-VQE. In Sec. 3.5, we present results from numerical simulations proving that our method is capable of significantly reducing the total number of calls to the quantum computer required by ADAPT-VQE. We additionally examine quantitative markers along specific optimizations to justify the remarkable speed-up in convergence, and seek to understand under which conditions the advantage is the greatest by analyzing the distance between the initial approximate inverse Hessian and the initial exact inverse Hessian, as well as the difference between the latter and the final exact inverse Hessian. We conclude the chapter with a discussion in Sec. 3.6.

3.2 Analysis of ADAPT-VQE Measurement Costs

To facilitate the understanding of the similarities of adaptive VQEs with gradient-based optimization methods, we will rely on pseudo-code to describe the relevant protocols in this section. The pseudo-code for the canonical ADAPT-VQE algorithm is presented in Alg. 1.

As discussed, one of the main limitations of adaptive VQAs is the scaling of the measurement costs, which come from two components: the VQE subroutine (step 14) and the measurement of the pool gradients (step 8). A brief discussion of these costs follows.

The molecular Hamiltonian of a system represented by N spin-orbitals (qubits) will have $O(N^4)$ terms, so this is the worst-case measurement cost of an energy evaluation. However, empirical evidence shows that the Pauli strings in molecular Hamiltonians may be grouped into commuting sets of linear size, resulting in an $O(N^3)$ cost for each energy evaluation (which seems unlikely to be further decreased) [63, 178]. The total measurement cost of one optimization is the cost of one energy evaluation multiplied by the total number of energy evaluations, which might come directly from energy measurements or indirectly from gradient measurements (measuring a length n gradient vector comes at a cost of $2n$

Algorithm 1: ADAPT-VQE

Input:
$\left|\psi^{(ref)}\right\rangle$, $\{\hat{A}_k\}_K$, $\hat{\mathcal{H}}$ ▷ Problem specification
ϵ, L ▷ Hyperparameters
Output:
$|\psi^*\rangle$, x^*, E^*

1 $n \leftarrow 0$;
2 $x_0 \leftarrow \{\}$;
3 $\left|\psi^{(n)}\right\rangle \leftarrow \left|\psi^{(ref)}\right\rangle$;
4 $E_n \leftarrow$ measure_energy $\left(\hat{\mathcal{H}}, \left|\psi^{(n)}\right\rangle\right)$;
5 **while** $n < L$ **do**
6 $\quad n \leftarrow n + 1$;
7 $\quad$ **for** $k \leftarrow 1...K$ **do**
8 $\quad\quad$ Measure $g_k = \left\langle\psi^{(n-1)}\right| \left[\hat{\mathcal{H}}, \hat{A}_k\right] \left|\psi^{(n-1)}\right\rangle$; ▷ Measure pool gradients
9 $\quad i \leftarrow k$ s.t. $g_k = \max(\{|g_k|\}_K)$ ▷ Select new operator
10 $\quad G \leftarrow \|\{g_1, ..., g_K\}\|_F$; ▷ Calculate gradient norm
11 $\quad$ **if** $G > \epsilon$ **then**
12 $\quad\quad \left|\psi^{(n)}\right\rangle \leftarrow e^{\theta_i \hat{A}_i} \left|\psi^{(n-1)}\right\rangle$; ▷ Grow ansatz
13 $\quad\quad x_n \leftarrow \{x_{n-1}, 0\}$; ▷ Grow parameter vector
14 $\quad\quad E_n, x_n \leftarrow \text{VQE}(\hat{\mathcal{H}}, \left|\psi^{(n)}\right\rangle, x_n)$; ▷ Minimize energy
15 $\quad$ **else**
16 $\quad\quad$ Return $\left|\psi^{(n-1)}\right\rangle$, x_{n-1}, E_{n-1}; ▷ Successful convergence
17 Return $\left|\psi^{(n)}(x)\right\rangle$, x_n, E_n; ▷ Convergence target unmet

energy evaluations; see Appendix A for a review on how to measure gradients on hardware).

As for the cost of measuring the gradients (step 8), it is $\mathcal{O}(N^5)$ for both the qubit and QE pools [9]. This means that the bound on the optimization costs is higher than the bound on the gradient measurement costs if the number of energy measurements per optimization grows faster than quadratically with N. Typical implementations of line searches require evaluating the gradient vector at least once (see Appendix B.1 for a description of a line search algorithm). In this case, the optimization costs will dominate if the number of optimizer iterations (line searches) grows faster than linearly with N. We verify this is the case in numerical simulations.

As such, we believe that strategies to expedite the optimization—such as ours—tackle the biggest source of measurement costs in this algorithm as of now. We confirm this numerically in the results section (Table 3.1).

3.3 Gradient-Based Optimization Methods

The choice of classical optimizer is pivotal in a VQA: it impacts not only the quality of the output solution but also the costs, as different optimizers will require varying numbers of calls to the quantum computer. We refer to Refs. [161, 122] for an overview of the topic.

We focus on numerical optimization methods that explicitly use the gradient vector when setting the search direction. We refer to Appendix A for a discussion on how to evaluate the gradient when the cost function is evaluated using a quantum computer.

Gradient-based optimization methods employ a succession of line searches, using the final point of each as the initial point for the next. p_k, the search direction at iteration k, is determined from data collected at x_k, the initial point. Such data includes, but is not necessarily limited to, the gradient vector. The optimizer then seeks a step size α that (perhaps approximately) minimizes the cost function f along p_k. The next iterate is set as

$$x_{k+1} = x_k + \alpha p_k. \tag{3.1}$$

In the next sections, we discuss different strategies to choose p_k.

3.3.1 Gradient Descent

Vanilla gradient descent [125] is a first-order optimization method where the search direction at iteration k is opposite to the gradient at x_k, i.e.,

$$p_k^{(GD)} = -\nabla f(x_k). \tag{3.2}$$

This means that steps are taken in the direction of steepest descent. Note that there always exists an $\alpha > 0$ such that this step direction produces a lower value of f at the next iterate (Eq. (3.1)). Since the gradient is not suggestive of a particular value for α, heuristics must be used.

Gradient descent usually takes significantly longer to converge than more sophisticated alternatives.

3.3.2 Newton's Method

Newton's method [125, 58, 56] employs a second-order modification to the gradient descent direction using $\nabla^2 f$, the Hessian of f (a matrix containing all its second derivatives). Newton's direction,

$$p_k^{(N)} = -\nabla^2 f^{-1}(x_k)\nabla f(x_k), \tag{3.3}$$

is the vector pointing at the minimum of a quadratic model of f at x_k, as given by a second-order Taylor expansion,

$$\begin{aligned} f(x_k + \Delta x) \approx & f(x_k) + (\nabla f(x_k))^T \Delta x \\ & + \frac{1}{2} (\Delta x)^T \nabla^2 f(x_k) \Delta x. \end{aligned} \tag{3.4}$$

If $\nabla^2 f(x_k)$ is positive definite, it admits an inverse and induces a convex quadratic model. Therefore, the positive definiteness of the Hessian guarantees that Eq. (3.4) has a minimum. If this condition does not hold, $p_k^{(N)}$ might be undefined or correspond to an ascent direction. This is more likely in regions farther away from a minimum, where a bowl-shaped approximation is bound to be less adequate. In such cases, a positive-definite modification of the Hessian will typically be used in its place.

Since a unit step size would lead to the minimum if f were quadratic in its variables, most practical implementations set α to one (with possible adjustments depending on the observed decrease of the cost function).

The use of second-order derivatives enables Newton's method to enjoy a quadratic convergence rate [117]. However, this comes at considerable cost: the evaluation of $O(n^2)$ second derivatives and the inversion of an $n \times n$ matrix, where n is the optimization dimension.

3.3.3 Quasi-Newton Methods

Quasi-Newton methods [125, 58, 56, 85] collect and use second-order information without explicitly evaluating the second derivatives of f. In general, they converge more slowly than Newton's method, but faster than gradient descent. Despite never explicitly computing the Hessian matrix, they often achieve superlinear convergence rates. In these methods, the search direction is given by

$$p_k^{(QN)} = -H_k \nabla f(x_k), \tag{3.5}$$

where H_k is an approximation to the inverse Hessian (the actual Hessian is usually denoted B_k in numerical optimization literature). This approximation is built and updated along the optimization, using available information that does not directly include second derivatives.

There are many quasi-Newton methods. We will focus on BFGS, since it is considered the most efficient [125] and has many desirable properties (see App. B.3). This method was named after Broyden [26], Fletcher [55], Goldfarb [64] and Shanno [151] who proposed it independently in 1970. We provide pseudo-code for this optimizer in Algorithm 2. While many variants exist, we choose to remain as close as possible to the implementation in Scipy's [167] numerical optimization submodule `optimize`, as it is widely used in practice.

f and its gradient ∇f must be supplied as callables, such that the optimizer can evaluate them for any parameter vector. Common choices for the initial point x_0 are all-zero, random, and problem-specific initializations.

The hyperparameters $M \in \mathbb{N}$, $\epsilon_o \in \mathbb{R}^+$ correspond respectively to the maximum number of iterations (line searches) and to a convergence threshold on the gradient norm. We use the subscript o to distinguish this threshold from the ADAPT-VQE convergence threshold in Algorithm 1. The optimization is stopped when the number of iterations reaches M or when the magnitude of the gradient vector falls below ϵ_o, whichever happens first. Typical values for M, ϵ_o are on the order of 10^2 to 10^4, 10^{-6} to 10^{-8} respectively.

`line_search` is an algorithm which seeks the α that minimizes $f(x_{k+1})$, with x_{k+1} given by Eq. (3.1). We provide more details about this subroutine in App. B.1.

Algorithm 2: BFGS

Input:
f, ∇f, x_0 ▷ Problem specification
M, ϵ_o ▷ Hyperparameters
Output:
minimizer x^*, $f(x^*)$

1 $n \leftarrow \text{length}(x_0)$
2 $H_0 \leftarrow I_{n\times n}$ ▷ Initialize inverse Hessian
3 $k \leftarrow 0$
4 **while** $k < M$ **do**
5 $\quad p_k \leftarrow -H_k \nabla f(x_k)$; ▷ Update search direction
6 $\quad x_{k+1}, f(x_{k+1}), \nabla f(x_{k+1}) \leftarrow \texttt{line_search}(p_k, f, \nabla f, x_k, f(x_k), \nabla f(x_k))$;
7 $\quad$ **if** $\|\nabla f(x_{k+1})\|_F > \epsilon_o$ **then**
8 $\quad\quad s_k \leftarrow x_{k+1} - x_k$;
9 $\quad\quad y_k \leftarrow \nabla f(x_{k+1}) - \nabla f(x_k)$;
10 $\quad\quad H_{k+1} \leftarrow \texttt{update_h}(H_k, s_k, y_k)$; ▷ Update inverse Hessian
11 $\quad\quad k \leftarrow k + 1$;
12 $\quad$ **else**
13 $\quad\quad$ Return $x_{k+1}, f(x_{k+1})$; ▷ Successful convergence
14 $\quad$ **end**
15 **end**
16 Return $x_k, f(x_k)$; ▷ Convergence target unmet

In the first iteration of BFGS, the direction calculated in step 5 is opposite to the gradient (just like in gradient descent algorithms). This is due to the choice of setting H_0 to the identity. In principle, this could be any symmetric positive definite matrix; however, in general, heuristics for how to choose it are lacking. The identity matrix is the standard, unbiased option, and it is usually assumed by numerical optimizers (this is the case in SciPy's implementation [167]).

As the algorithm proceeds, H_k will be updated to better reflect the curvature of f, thus refining the search direction. These updates depend only on the parameter and gradient vectors. For more details, we refer to App. B.2.

3.3.4 Convergence Rates

A brief discussion of convergence rates follows. We refer to Ref. [125] for details.

In this thesis, we will define convergence rates based on ratios of errors. This is referred to as 'Q-convergence' (from *quotient*) and is standard in numerical optimization literature.

The iterates $\{x_k\}$ are said to converge Q-linearly to the solution x^* if there is a constant $r \in (0, 1)$ such that

$$\frac{\|x_{k+1} - x^*\|_F}{\|x_k - x^*\|_F} \leq r \tag{3.6}$$

for all large enough k. They are said to converge Q-superlinearly if

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|_F}{\|x_k - x^*\|_F} = 0\,, \tag{3.7}$$

and Q-quadratically if there is a constant $M \in \mathbb{R}^+$ such that

$$\frac{\|x_{k+1} - x^*\|_F}{\|x_k - x^*\|_F^2} \leq M \tag{3.8}$$

for all large enough k.

Q-linear convergence is typical of gradient descent, while Newton's method usually enjoys Q-quadratic convergence. Quasi-Newton methods are in between, often converging Q-superlinearly. More specifically, if the quasi-Newton Hessian B_k and search direction p_k obey

$$\lim_{k\to 0} \frac{\|(B_k - \nabla^2 f(x^*))p_k\|_F}{\|p_k\|_F} = 0\,, \tag{3.9}$$

then there will be an index k_0 such that the unit step length will be accepted for all iterations $k > k_0$, and the iterates $\{x_k\}$ will converge to the solution Q-superlinearly (see Theorem 3.6 in Ref. [125]).

The difference between the approximate and exact Hessians along the search direction going to zero as the iterations proceed (equation (3.9)) is both necessary and sufficient for quasi-Newton methods to be Q-superlinearly convergent.

As the Q-convergence definition will be used at all times, we will omit the 'Q-' when referring to convergence rates in what follows.

3.4 BFGS Algorithm for ADAPT-VQE

Having discussed the BFGS optimizer as well as the ADAPT-VQE algorithm, we now propose an ADAPT-VQE-tailored BFGS optimizer in Algorithm 3.

In labeling the inverse Hessian H, we use a single subscript k to denote the kth optimization iteration, and $n \times n$ to denote the nth ADAPT-VQE iteration (since this sets the dimension of the matrix). When either iteration label is clear from context, we omit the corresponding subscript to simplify the notation.

We note that aside from the callables f, ∇f, all inputs to the algorithm at iteration n are outputs from iteration $n-1$. The point $x^*_{(n-1)}$, as well as the corresponding gradient $\nabla f(x^*_{(n-1)})$ and inverse Hessian $H^*_{n-1\times n-1}$, pertain to the state of the optimizer at the end of iteration $n-1$.

The novelty of Algorithm 3 is in step 3, where we use the inverse Hessian resulting from the $(n-1)$th ADAPT-VQE optimization to initialize the inverse Hessian at the nth optimization. We note that because the final inverse Hessian will be used as an input for the next iteration, we update it in line 10, before the convergence check. This is in contrast with Algorithm 2, where the Hessian is not updated in the iteration where convergence is reached.

The motivation behind Algorithm 3 is that ADAPT-VQE uses the final point of one optimization as the initial point for the next (see Algorithm 1). Since at the start of the new optimization we have not

Algorithm 3: BFGS for ADAPT-VQE Iteration n

Input:

f, ∇f, $x^*_{(n-1)}$, $\nabla f(x^*_{(n-1)})$, $H^*_{n-1\times n-1}$ ▷ Problem specification

M, ϵ_o ▷ Hyperparameters

Output:

minimizer x^*, $f(x^*)$, $\nabla f(x^*)$, $H^*_{n\times n}$

$x_0 \leftarrow \{x^*_{(n-1)}, 0\}$

$g_0 \leftarrow \{\nabla f(x^*_{(n-1)}), \frac{\partial f}{\partial \theta_n}(x_0)\}$

$H_0 \leftarrow \begin{pmatrix} H^*_{n-1\times n-1} & 0 \\ 0 & 1 \end{pmatrix}$ ▷ Initialize inverse Hessian

$k \leftarrow 0$

while $k < M$ **do**

  $p_k \leftarrow -H_k \nabla f(x_k)$; ▷ Update search direction

  $x_{k+1}, f(x_{k+1}), \nabla f(x_{k+1}) \leftarrow \texttt{line_search}(p_k, f, \nabla f, x_k, f(x_k), \nabla f(x_k))$;

  $s_k \leftarrow x_{k+1} - x_k$;

  $y_k \leftarrow \nabla f(x_{k+1}) - \nabla f(x_k)$;

  $H_{k+1} \leftarrow \texttt{update_h}(H_k, s_k, y_k)$; ▷ Update inverse Hessian

  $k \leftarrow k + 1$;

  if $\|\nabla f(x_k)\|_F < \epsilon_o$ **then**

    Return $x_k, f(x_k), \nabla f(x_k), H_k$; ▷ Successful convergence

  end

end

Return $x_k, f(x_k), \nabla f(x_k), H_k$; ▷ Convergence target unmet

moved in the cost landscape, the curvature information we gathered during the previous optimization continues to approximately capture the shape of the parameter landscape, at least in regards to the parameters $\theta_0, ..., \theta_{n-1}$. As we have no second-order information concerning θ_n yet, we choose to expand $H^*_{n-1\times n-1}$ with a unit diagonal and zeros elsewhere. This unbiased choice evidently preserves the positive definiteness of the matrix, thus our modified algorithm also guarantees that the quadratic model has a minimum and p_k corresponds to a descent direction at all times.

It is simple to see that positive definiteness is also preserved by the act of removing a row and the corresponding column from H_k. This means that we have the freedom to select from which parameters we wish to preserve second-order information. As a result, our strategy generalizes to the case where we wish to freeze a subset $\{\theta_{f_1}, ..., \theta_F\}$ of F parameters that were active in the previous iteration. In this case, we simply remove from $H^*_{n-1\times n-1}$ the rows and columns corresponding to the indices $f_1, ..., F$. This was not explicitly included in Algorithm 3 for the sake of conciseness.

Equipped with our modified BFGS algorithm, ADAPT-VQE has the flavor of a continuous optimization of growing dimension. The search space is expanded when the ansatz gradient norm falls below ϵ_o, because we do not expect a significant energy decrease along the directions in the current cost landscape. The algorithm terminates when the pool gradient norm is below ϵ, because we do not expect a significant

energy decrease along the directions in which we can expand this space.

The selection criterion of ADAPT-VQE dictates that the direction in which to expand the parameter space leads to the steepest cost landscape. This is reminiscent of first-order optimization methods, with the difference being that here we are screening possible candidates, and so the choice of direction is made with respect to parameters not currently in the parameter vector.

For the sake of conciseness, we will refer to our approach as *recycling the Hessian*. Note that the inverse Hessian is unique and fully determines the actual Hessian, such that by recycling one, we implicitly recycle the other.

3.5 Results

In this section, we present numerical simulation results for different systems at various bond lengths. While we focus on linear H_6 (12 qubits) as a toy model for a strongly correlated system, we additionally consider LiH (12 qubits) and BeH_2 (14 qubits) as real molecules. In all cases, we use the STO-3G basis set and no frozen-core approximations. The pools we use are the QE pool [182] and the qubit pool [160], defined in Ch. 2. We set the ADAPT-VQE convergence threshold ϵ to 10^{-6} and 10^{-5}, respectively. The threshold for the qubit pool is set to a higher value because this pool is larger.

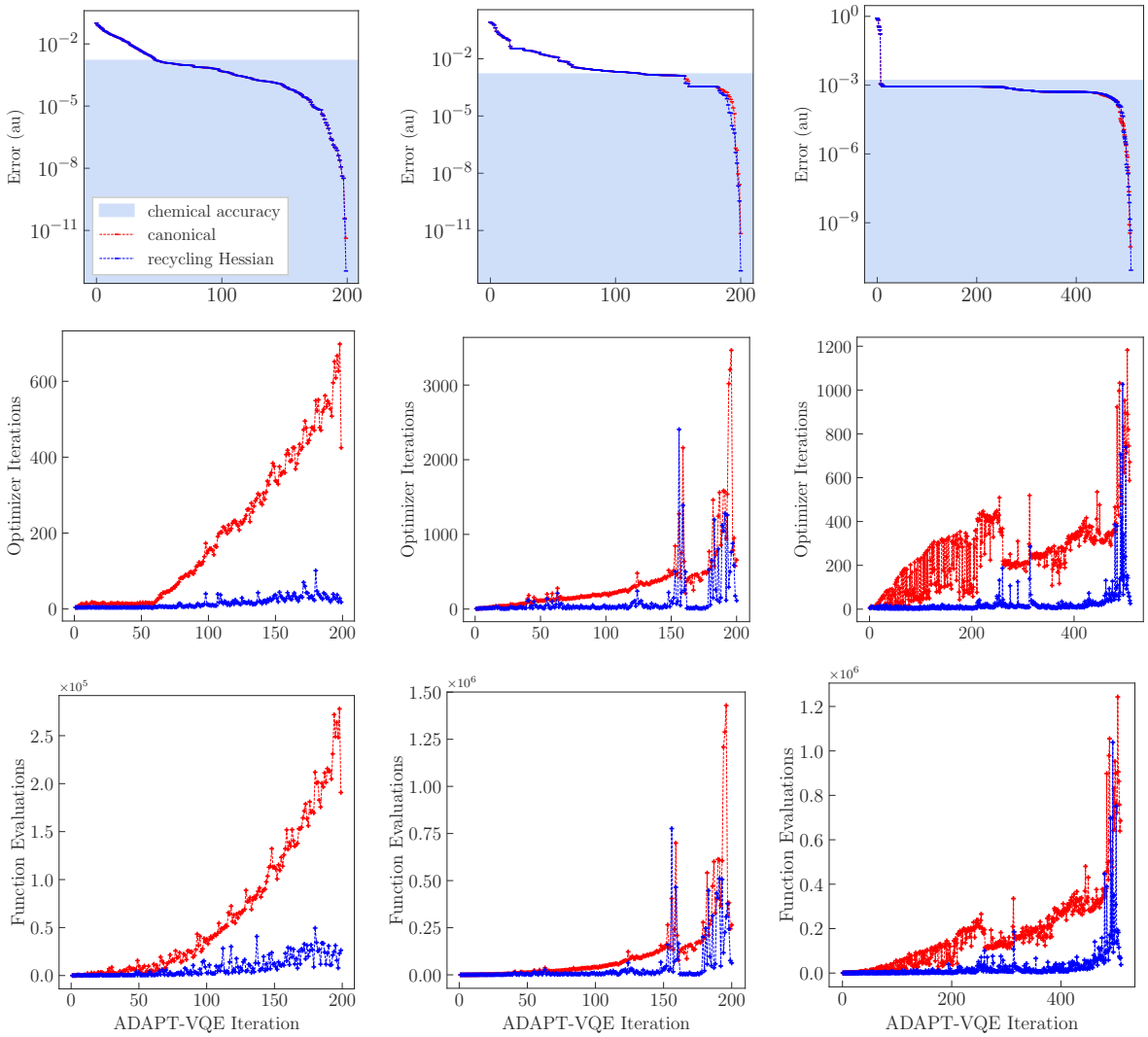
The code used for the numerical simulations has been made publicly available on GitHub (`https://github.com/mafaldaramoa/ceo-adapt-vqe`). OpenFermion [5] was used for manipulating fermionic operators, and the corresponding plugin with PySCF [6] for the underlying electronic structure calculations. All expectation values were calculated via matrix algebra. To calculate distances between matrices, we use the Frobenius norm, defined for an $m \times n$ matrix A as $\|A\|_F = \sqrt{\text{Tr}(AA^\dagger)} = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$. The hyperparameters for BFGS were chosen as $\epsilon_o = 10^{-6}$ and $M = 10000$ (see Sec. 3.3.3).

We now briefly describe how this section is organized. We begin by analyzing the impact of recycling the Hessian on the measurement costs of ADAPT-VQE for multiple molecules, interatomic distances, and pools in Sec. 3.5.1. In Sec. 3.5.2, we seek to understand when our strategy works the best by analyzing the distance between the approximate and exact (inverse) Hessians, as well as the evolution of the latter. Finally, in Sec. 3.5.3, we dive into specific optimizations to better understand the faster convergence of our method.

3.5.1 Measurement Costs

In this subsection, we assess the impact of recycling the Hessian on the measurement costs of the optimization process. We consider the key cost to be the number of function evaluations. Evaluating the energy corresponds to one function evaluation, while evaluating an element of the gradient vector corresponds to two (see Appendix A). The circuits for these two cases differ by a negligible number of gates, and the observable is always the Hamiltonian [91]. This means that each function evaluation requires measuring the same set of Pauli strings and implies a similar number of shots, up to differences

in the variance of the expectation value of the energy in the state. Note that we can place a state-independent upper bound on the number of shots required for a given error [172]. Based on these arguments, we expect the total number of function evaluations to be a good figure for assessing costs.



(a) QE pool, 1Å (b) QE pool, 3Å (c) Qubit pool, 3Å

Figure 3.1: Comparison of the error (top row), total number of line searches (middle row), and total number of function evaluations (bottom row) per iteration of the ADAPT-VQE algorithm with and without Hessian recycling for H_6. We consider the QE pool for (a) equilibrium and (b) stretched geometries, as well as (c) the qubit pool for a stretched geometry. The red curve is often not visible in the error plots due to it lying below the blue one. The region shaded blue is the region of chemical accuracy, defined as an error below 1 kcal/mol.

Figure 3.1 shows the results for the H_6 molecule at various bond distances, using the QE and qubit pools. The first thing to note is that the error curves overlap for nearly all iterations, except for regimes of very high accuracy where recycling the Hessian results in a lower absolute error. We do not expect this to be a benefit of our method, as this accuracy range is unlikely to be relevant in practice. Simulation data shows that up to such high accuracy regimes, the final ansätze are identical regardless of whether

we recycle the Hessian.

Despite the matched accuracy, the number of optimizer iterations is remarkably different between the two methods. When the initial estimate for the inverse Hessian is the identity, the number of BFGS iterations clearly increases faster than linearly with the ADAPT-VQE iteration number; in contrast, when we recycle the Hessian, the number of line searches is roughly constant across a large number of ADAPT-VQE iterations, despite the considerable increase in the dimension of the parameter space. Since the second-order information is not reconstructed from scratch in each optimization, accounting for parameter correlations does not necessarily imply a superlinear number of optimizer iterations. In fact, the search direction for the very first line search at ADAPT-VQE iteration n is already equipped with information concerning correlations between $n-1$ parameters, represented by $(n-1)^2$ second derivatives. The only missing information concerns a linear number of second derivatives, describing the correlation between each parameter and the last.

Interestingly, the impact of recycling the Hessian for H_6 at 3 Å is more significant on qubit-ADAPT-VQE (Fig. 3.1(c)) than on QEB-ADAPT-VQE (Fig. 3.1(b)). With the qubit pool, the Hessian recycling protocol decreases the total number of function evaluations by 84% — 20% more than the decrease with the QE pool. This suggests that the impact of the strategy is not only system- but also pool-dependent.

In general, for a fixed dimension, we expect a higher number of line searches to imply a higher number of function evaluations (see Appendix B.1). This is confirmed by the bottom panels of Fig. 3.1, which show that the Hessian recycling strategy succeeds in producing a significant decrease in measurement costs.

In Appendix C, we include additional plots for the LiH and BeH_2 molecules. In all cases, recycling the Hessian results in relevant savings in measurement costs. The savings are particularly notable for the H_6 molecule (especially at the equilibrium geometry). Among the three, this is the most difficult to simulate, requiring the most iterations and measurements. In all cases, the difference in costs seems to increase as the iterations advance. Thus, we can expect our method to become even more beneficial for larger systems.

Considering the same molecules at larger bond distances allows us to study the impact of our protocol as systems become more strongly correlated. While the savings in the number of line searches and measurements are maintained or even increased as we stretch LiH and BeH_2, they become less significant for H_6 at stretched geometries. Among the test cases we considered, this was the one where our strategy performed the worst. We investigate the causes in Sec. 3.5.2. Despite this, recycling the Hessian still results in a reduction of the total number of function evaluations by 64% across the whole execution.

We finish this section with quantitative examples of the measurement cost reduction achieved by our method. Table 3.1 includes the costs of the VQE step with and without Hessian recycling, as well as of the gradient measurement step, through complete executions of QEB-ADAPT-VQE for various systems. Note that the latter step concerns the measurement of the gradients of operators generated by pool elements, not of operators in the ansatz (which are included in the costs for VQE). We do not consider any grouping strategies for the Hamiltonian, as they are beyond the scope of this work.

The numerical data show that the measurement cost of the VQE subroutine is reduced by roughly one order of magnitude when our Hessian recycling strategy is employed. We additionally verify that,

			Molecule					
			LiH		H_6		BeH_2	
			1.5 Å	3 Å	1 Å	3 Å	1.3 Å	3 Å
Step	Gradient Measurement		5.2×10^3	5.4×10^3	1.9×10^4	1.9×10^4	1.2×10^4	1.3×10^4
	VQE	Canonical	2.5×10^5	2.4×10^5	1.3×10^7	2.2×10^7	4.2×10^6	3.6×10^6
		Recycling Hessian	6.0×10^4	3.2×10^4	1.7×10^6	7.9×10^6	9.4×10^5	5.6×10^5
		(reduced to)	(24%)	(13%)	(13%)	(36%)	(22%)	(16%)

Table 3.1: Total measurement costs incurred by the QEB-ADAPT-VQE algorithm in the gradient measurement and VQE subroutines (steps 8 and 14 in Algorithm 1), for the studied test cases. The costs are given as multipliers for the cost of a naive energy evaluation, so that unit cost implies $O(N^4)$ measurements. For the gradient measurement step, we consider the worst-case cost under the leading measurement strategy for the QE pool ($8N$ per iteration, where N is the number of spin-orbitals/qubits) [9].

as predicted by our analysis in Sec. 3.2, this subroutine is the bottleneck of the algorithm as far as measurement costs are concerned. In fact, the associated cost is always at least one order of magnitude higher than the cost of the gradient measurements, even when the Hessian recycling strategy is employed. As such, we confirm that our proposal addresses the most significant source of measurement costs of ADAPT-VQE, decreasing its *total* measurement costs by an order of magnitude (for the studied molecules—as discussed, we expect this decrease to become more significant for larger systems). For the majority of the molecules we considered, the reduction in the measurement costs of the optimization was in the 70-90% range.

3.5.2 Distance to the Exact Hessian

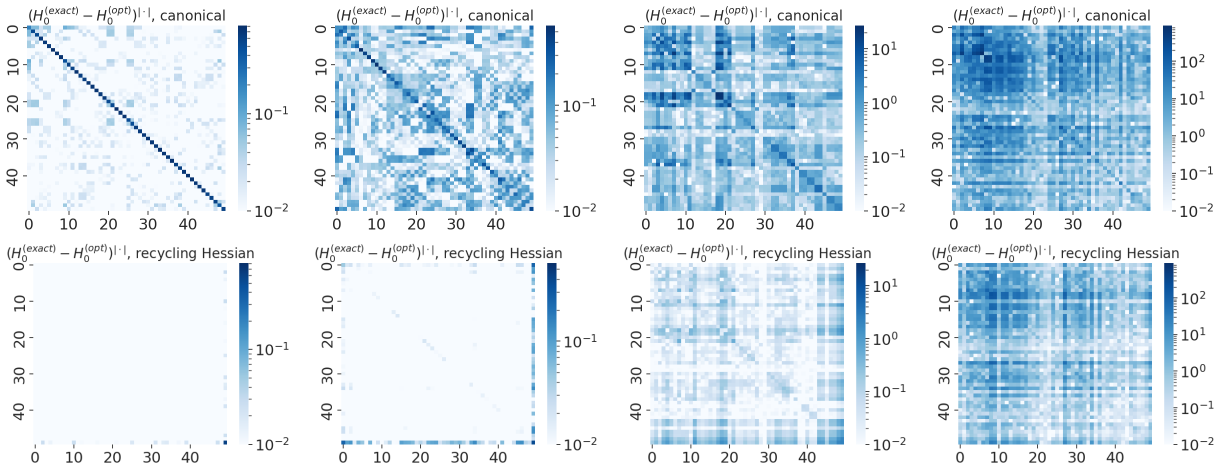
In this subsection, we analyze the distance between the approximate and exact inverse Hessians, $H^{(\text{opt})}$ and $H^{(\text{exact})}$, with and without Hessian recycling.

We begin with the heatmaps of Fig. 3.2, a visual representation of this distance at the beginning of the 50th optimization of ADAPT-VQE with the QE pool. We plot $(H_0^{(\text{exact})} - H_0^{(\text{opt})})^{|\cdot|}$, the element-wise distance between $H_0^{(\text{opt})}$ and $H_0^{(\text{exact})}$. We consider H_6 at four different bond distances, and we use the same color map for each bond distance. It should be noted that there is nothing unique about the 50th optimization. We chose this number because it is high enough for the optimization process to be interesting, but not so high that the calculations become intractable or the heatmaps become illegible. In Appendix D we show that the behavior generalizes by presenting equivalent heatmaps for other iterations.

We note that due to how we initialize H_k in Algorithm 3, the last row/column of the heatmaps is the same regardless of whether we recycle the Hessian or not. However, as expected, the inner second derivatives are better approximated by the previous iterations' derivatives than by the elements of the identity matrix. The difference is more significant for geometries near equilibrium.

Another interesting geometry-related trend can be observed in the upper heatmaps: for less correlated geometries (shorter bond distances), the approximation is the poorest for the diagonal elements [1]. As we

[1] One could hypothesize that this is a matter of scaling. However, the errors in the diagonal remain dominant even if we multiply the identity matrix by a scalar factor tuned to minimize the distance to the true values.



(a) H_6 at 1Å (b) H_6 at 1.5Å (c) H_6 at 2Å (d) H_6 at 3Å

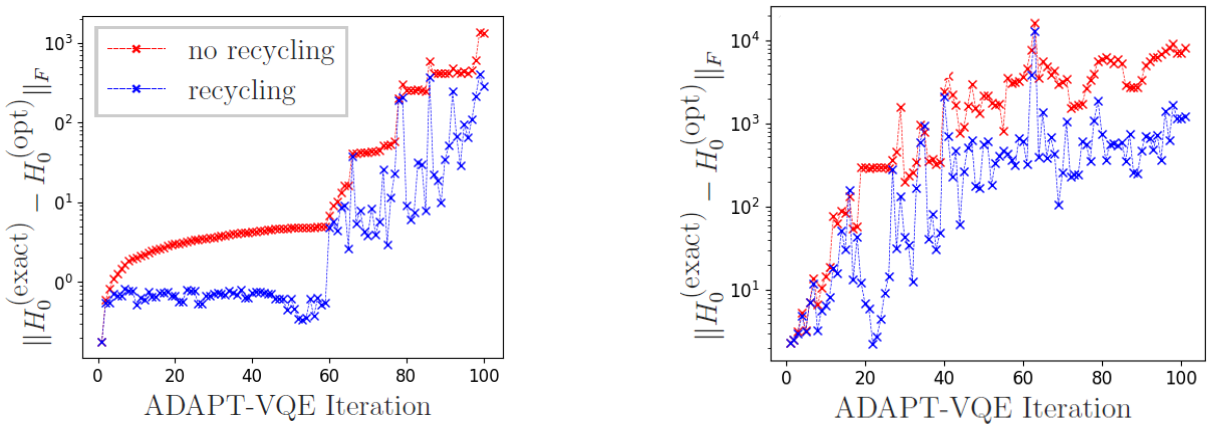
Figure 3.2: Heatmaps showing the difference between the initial approximate inverse Hessian in the optimization process and the initial exact inverse Hessian, with (bottom row) and without (top row) Hessian recycling, for the 50th iteration of QEB-ADAPT-VQE. The plots show the element-wise difference between these two matrices. Due to it being strongly correlated and the hardest to simulate, here we consider the test molecule H_6 at different interatomic distances.

increase the bond distance, there is a shift in behavior, as the magnitude of some off-diagonal elements in the difference matrix starts rivaling the magnitude of the diagonal elements. At 3 Å, the diagonal entries are no longer dominant. This is clearly symptomatic of a more complicated optimization: While for shorter bond distances the parameters can nearly be treated as uncorrelated, correlations between parameters play a bigger role for stretched geometries. Relatedly, the magnitude of the elements of the difference matrix changes from well below unity up to several hundred as we increase the bond distance.

While the heatmaps may help us gain intuition, the information they provide is limited, as they only concern one iteration. In Fig. 3.3, we plot the Frobenius distance between these same matrices for the first 100 iterations of QEB-ADAPT-VQE. We focus on the two extreme bond distances: 1 Å and 3 Å. As expected, the initial distance to the exact inverse Hessian is greater for the stretched bond distance, both when we recycle the Hessian and when we do not. Remarkably, the impact of recycling the Hessian on the initial distance is a similar multiplicative factor (close to 0.1) for both bond distances, despite the impact on costs being significantly larger for the 1 Å geometry (as we saw in Figs. 3.1(a), 3.1(b)).

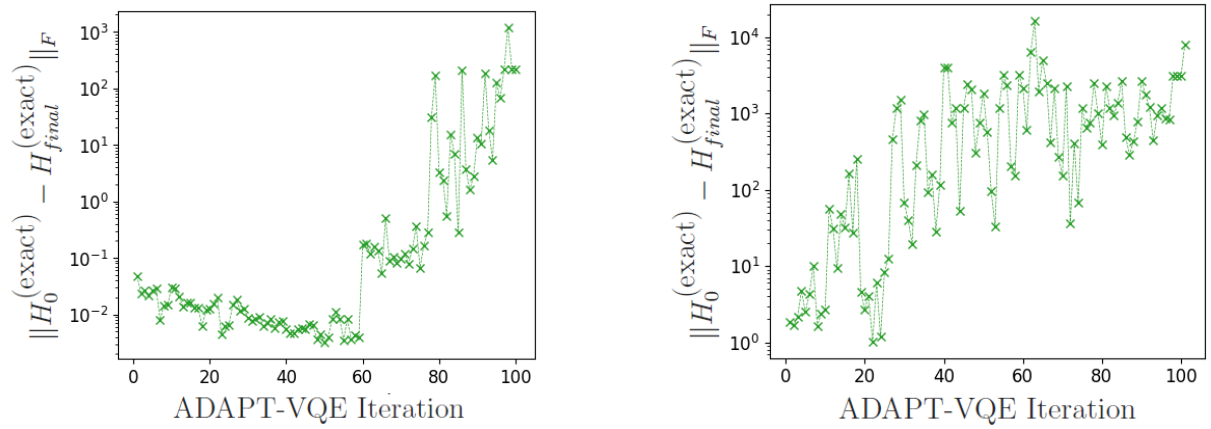
We hypothesize that, in large part, the difference in performance is not due to the initial matrix being a poor approximation, but rather due to the optimization process being more difficult. If the initial point is farther away from a minimum, and the curvature around these two points is vastly different, the optimizer will require more iterations to move along the cost landscape, update the inverse Hessian, and reach the minimum. For such lengthier optimizations, the cost for the optimizer to move across the landscape is more likely to surpass the cost of approximating the second derivatives around the initial point. To test this hypothesis, we plot the distance between the exact initial and exact final inverse Hessians in Fig. 3.4.

The plot confirms that there is a significant increase in the distance between the initial and final inverse Hessians when the bond distance is increased. For H_6 at 1 Å, the distance varies between 10^{-2} and 10^{3}



(a) H_6 at 1 Å (b) H_6 at 3 Å

Figure 3.3: Frobenius distance between the initial approximate inverse Hessian in the optimization process $H_0^{(\text{opt})}$ and the initial exact inverse Hessian $H_0^{(\text{exact})}$, with and without Hessian recycling, for the first hundred QEB-ADAPT-VQE iterations. We consider two cases: one for which the Hessian recycling works particularly well (H_6 at 1 Å) and one for which it does not (H_6 at 3 Å).



(a) H_6 at 1 Å (b) H_6 at 3 Å

Figure 3.4: Frobenius distance between the exact initial $H_0^{(\text{exact})}$ and final $H_{\text{final}}^{(\text{exact})}$ inverse Hessians, for the first hundred QEB-ADAPT-VQE iterations. The systems are the same as those from Fig. 3.3.

and only surpasses unity after 80 iterations. In contrast, at 3 Å, the distance varies between 10^0 and 10^4 and is above unity (and closer to the maximum value) throughout all the ADAPT-VQE iterations.

3.5.3 Evolution of the Optimization

In the previous subsections, we focused on data from the beginning or end of each optimization. This allowed us to characterize the behavior of our protocol throughout complete runs of the ADAPT-VQE algorithm, and analyze how the dimension of the optimization and the molecule under study impact the costs.

However, as the protocol we propose is in fact a numerical optimization method, it is interesting to zoom in on the optimization process and investigate the impact of recycling the Hessian in quantitative

aspects of it. Thus, in this section, we delve into a particular optimization. We consider the 75th iteration of QEB-ADAPT-VQE for H_6 at equilibrium (1 Å) and stretched (3 Å) bond distances (the same systems we analyzed in the previous section). The corresponding optimization has 75 parameters. Once again, we note that there is nothing unique to the 75th iteration; we chose it because it is complex enough to be relevant, but not so much that calculating the relevant data becomes a computational challenge. In Appendix E we consider a different iteration to show that the results generalize.

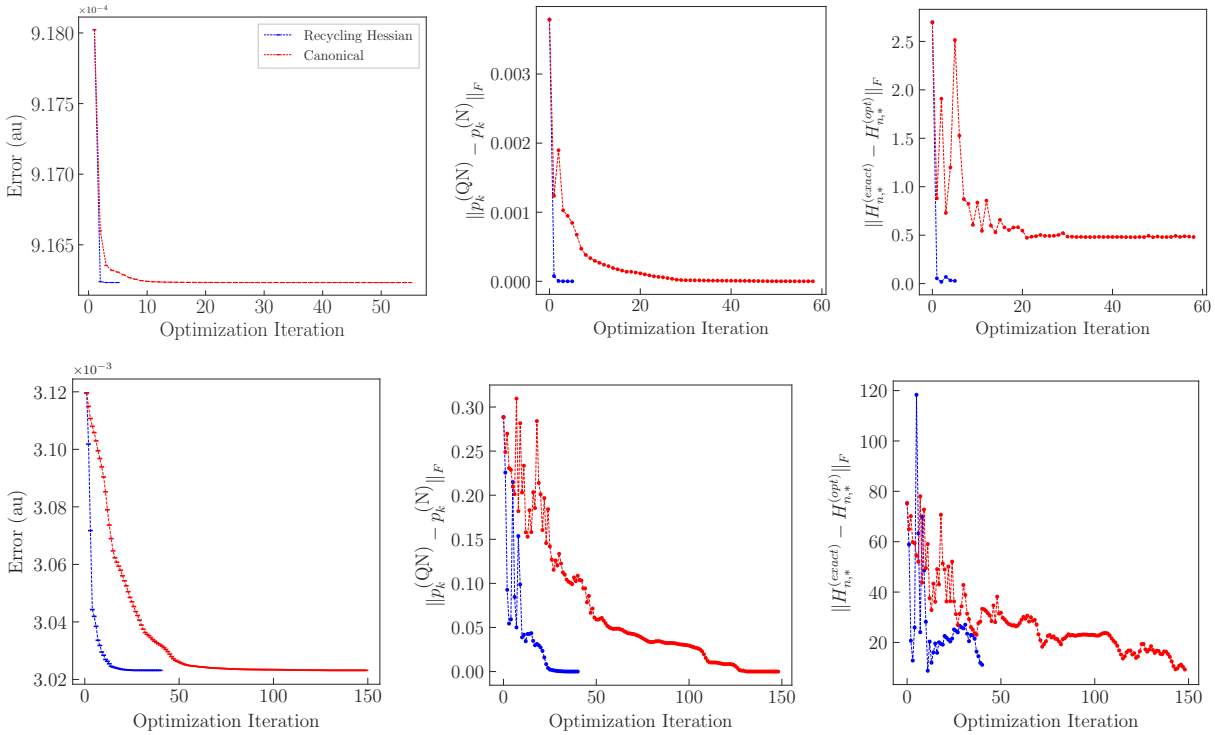


Figure 3.5: Evolution of the 75th QEB-ADAPT-VQE optimization with and without Hessian recycling, for the same two systems as in Fig. 3.3: H_6 at 1 Å (top row) and 3 Å (bottom row). The panels show (left to right) the error; the distance between the quasi-Newton $p_k^{(\mathrm{QN})}$ and Newton $p_k^{(\mathrm{N})}$ directions; and the distance between the last row of the approximate inverse Hessian $H^{(\mathrm{opt})}$ and the exact inverse Hessian $H^{(\mathrm{exact})}$.

In the first column of Fig. 3.5, we observe that, as expected, the optimization requires significantly fewer iterations to converge for the equilibrium geometry. In line with previous results, recycling the Hessian results in a similar final error for a lower number of iterations.

In the second column, we can see that the search direction (Eq. (3.5)) aligns much more quickly with the Newton direction (Eq. (3.3)) when the Hessian is recycled. We note that the initial distance to Newton's vector (which we plot as the value for iteration 0) is identical whether the Hessian is recycled or not: the $n-1$ interior parameters were previously optimized and thus have nearly zero gradients, and step 3 in Algorithm 3 is agnostic to any second-order information concerning the nth parameter. Despite being similar initially, once the parameters start changing, the second-order information contained within the recycled Hessian becomes relevant, allowing the direction to more quickly align with Newton's. This is remarkable: Unlike Newton's method, our optimization method never explicitly measures the Hessian,

yet it aligns with the Newton search direction in a fraction of the iterations needed for the canonical BFGS implementation.

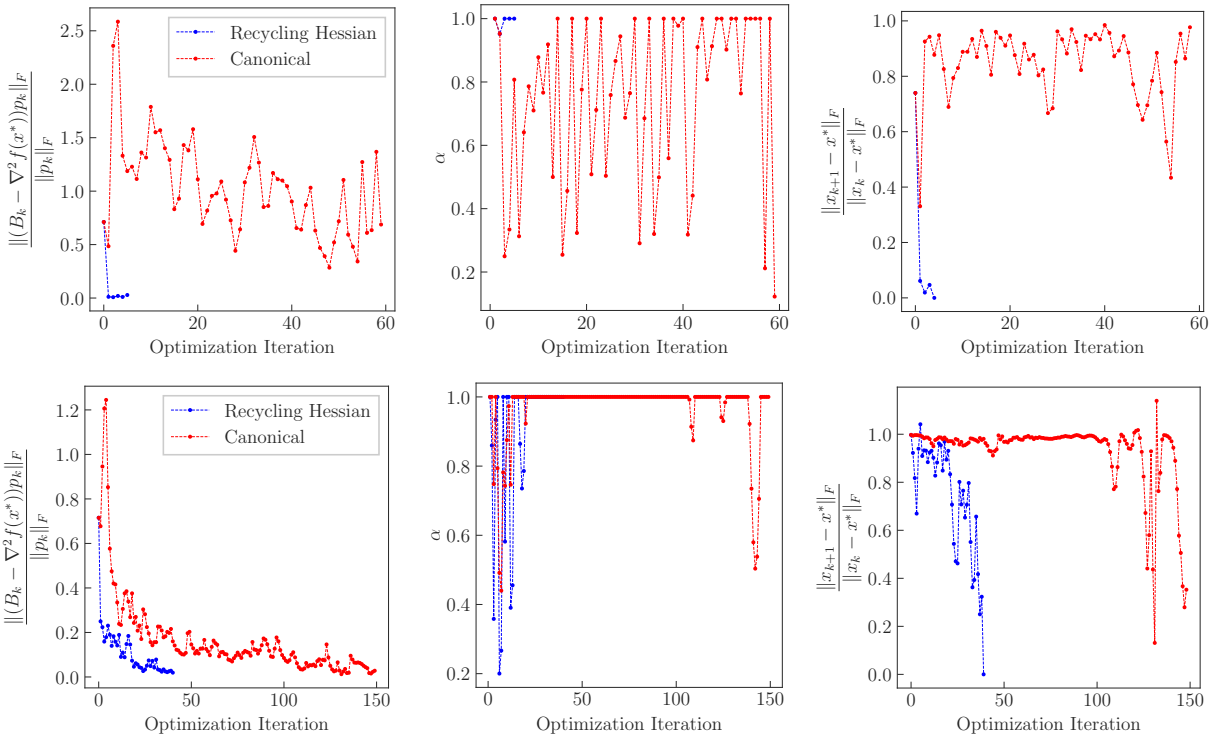


Figure 3.6: Relevant quantities for the study of the convergence rate of the 75th QEB-ADAPT-VQE optimization with and without Hessian recycling, for the same two systems as in Fig. 3.3: H_6 at 1 Å (top row) and 3 Å (bottom row). (a) The distance between the approximate and the exact Hessian along the search direction, (b) the step size α at the end of the line search, and (c) the ratio between successive errors are shown.

The third column of Fig. 3.5 shows the distance between the last row of the approximate and exact inverse Hessians. We denote the last row by $H_{n,*}$; recall that the matrix is symmetric, so that we have $H_{n,*} = H_{*,n}$. This is the vector of second-order derivatives involving the new parameter. For similar reasons to those explained in the discussion of Fig. 3.5(b), the initial distance, plotted at iteration 0, is identical regardless of whether the Hessian is recycled or not. Since the second-order derivatives being recycled only involve the parameters $\theta_1, ..., \theta_{n-1}$, naively, there seems to be no reason to expect that recycling the Hessian will lead to faster convergence of the last column/row (corresponding to θ_n). Surprisingly, this does happen; it seems that the initial curvature information concerning the interior parameters allows the optimizer to focus on exploring the new search direction, and enables a faster buildup of information regarding correlations between the new parameter and old ones. Even more surprisingly, the distance is still higher for the canonical BFGS when convergence is reached, despite the larger number of iterations. This is particularly visible for the 1 Å geometry.

Finally, we focus on the convergence rate of the optimizations. The relevant quantities (see Sec. 3.3.4) are plotted in Fig. 3.6. In the first column, we see that the difference between the approximate and exact Hessians along the search direction goes to zero in both optimizations when we recycle the Hessian, but

this does not happen when we do not. Further, the step size (second column) saturates to unity after roughly half of the iterations when we recycle the Hessian, but oscillates instead of stabilizing when we do not. Together, these results indicate that recycling the Hessian results in a superlinear convergence (Eq. (3.7)) that would otherwise not be achieved.

The third column of Fig. 3.6 confirms this. BFGS with Hessian recycling enjoys superlinear convergence, while canonical BFGS converges linearly with a convergence constant r close to 1—the worst possible scenario (see Eq. (3.6)). Such a convergence rate is expected of gradient descent in ill-conditioned problems, and constitutes an underwhelming performance for a quasi-Newton optimizer from which we hope to achieve superlinear convergence.

3.6 Discussion

In this chapter, we proposed to tailor BFGS, one of the most popular optimizers for VQAs, to ADAPT-VQE, one of the most popular such algorithms. In a typical implementation, each iteration of ADAPT-VQE performs an optimization where the state of the optimizer is initialized as if we possessed no knowledge about the curvature of the cost landscape, despite the fact that some knowledge was collected along the previous optimization. We develop a variant of the BFGS method that allows second-order information to naturally flow from one iteration to the next. By recycling the inverse Hessian matrix maintained by the optimizer, this protocol converges superlinearly even in situations where the canonical BFGS implementation does not, and achieves significant savings in the number of function evaluations required in each iteration of ADAPT-VQE. This specifically addresses the costs of the optimization process, which is the biggest source of measurement costs in such algorithms. In addition to decreasing the number of calls to the quantum computer in hardware implementations of adaptive VQAs, our strategy reduces the runtime of classically simulating them, thereby improving our ability to design and test these algorithms.

Since the impact of our Hessian recycling strategy seems to increase with the size of the system and the dimension of the optimization, we expect it to be even more beneficial for molecules for which performing classical simulations is infeasible.

While we focused on ADAPT-VQE for testing our proposed optimizer, our algorithm easily generalizes to other iterative state preparation protocols such as ADAPT-QAOA [188] or layerwise learning [154]. Freezing parameters or layering [10, 102] are also compatible with our strategy, requiring nothing more than a simple manipulation of the initial inverse Hessian (removing or adding subsets of rows and columns).

The noise-resilience of the proposed optimization algorithm is left as an open question. Adding realistic noise to the simulations incurs an overhead in computational costs, leading to prohibitively long simulation times for interesting molecules. Previous approaches to the noisy simulation of ADAPT-VQEs have averted this issue by growing the ansatz noiselessly and assessing the impact of noise in the final circuit exclusively [38, 102]. While this may provide insight regarding the near-term viability of such algorithms, it is evidently not applicable to the optimization process. Developing strategies for noisy simulation that address this issue is left for future work. We note that while there is a general expectation that optimizers designed to

be robust in the presence of noise (such as SPSA [156]) will be better suited for experiments in quantum hardware, BFGS was shown in Ref. [132] to be among the best algorithms for noisy optimizations, with a performance comparable to SPSA.

4

CEO-ADAPT-VQE: Improving the Hardware-Efficiency of Problem-Tailored Operator Pools

This chapter is associated with the following authored publication:

Reducing the Resources Required by ADAPT-VQE Using Coupled Exchange Operators and Improved Subroutines. **M. Ramôa**, P. G. Anastasiou, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. npj Quantum Inf 11, 86 (2025)

4.1 Motivation

As discussed in Ch. 1, despite recent progress, the gate counts and measurement costs of ADAPT-VQE remain a challenge for experimental demonstrations. The question of how close we are to a demonstration on near-term hardware that can rival classical simulation remains open.

In this chapter, we take a key step in addressing this question by introducing an improved operator pool, which we term the CEO pool, and merging it with recent progress in decreasing measurement costs and improving the hardware efficiency of the adaptive ansatz construction. We show that this novel variant of ADAPT-VQE, labeled CEO-ADAPT-VQE*, combines frugal measurement costs with shallow ansätze. We use this to gauge the evolution of ADAPT-VQE since its inception. Our simulations for a range of molecules show that our present version outperforms all previous ones, and the gate counts are reduced dramatically as compared to the original ADAPT-VQE. Further, the measurement costs incurred by our algorithm are five orders of magnitude lower than those incurred by static ansätze with comparable CNOT counts. These improvements bring us closer to the goal of demonstrating practical quantum advantage.

Table 4.1 showcases the evolution of ADAPT-VQE to date by comparing our novel algorithm with GSD-ADAPT-VQE for LiH (12 qubits), H_6 (12 qubits) and BeH_2 (14 qubits). As proposed in the original formulation of ADAPT-VQE, the latter uses a fermionic pool, consisting of GSD excitations. The table presents the CNOT count, CNOT depth and measurement costs (estimated as the total number of noiseless energy evaluations, which constitutes a lower bound) of the algorithms at the first iteration where they reach

		LiH at 3 Å	H_6 at 1.5 Å	BeH_2 at 2 Å
GSD-ADAPT-VQE	CNOT count	392	6896	2192
	CNOT depth	384	6586	1909
	Measurement Costs	50468	654899	520225
CEO-ADAPT-VQE*	CNOT count (reduced to)	107 (27%)	812 (12%)	288 (13%)
	CNOT depth (reduced to)	30 (8%)	282 (4%)	95 (5%)
	Measurement Costs (reduced to)	560 (1%)	10857 (2%)	2197 (0.4%)

Table 4.1: Minimal resource requirements of GSD-ADAPT-VQE and CEO-ADAPT-VQE* to reach chemical accuracy (defined as an error below 1 kcal/mol). The algorithms and costs are defined in the main text (see, e.g., Fig. 4.14).

chemical accuracy. We observe that CEO-ADAPT-VQE* reduces these costs dramatically—to 12-27%, 4-8% and 0.4-2%, respectively, as compared to the original ADAPT-VQE algorithm. These reductions bring us closer to the goal of demonstrating practical quantum advantage.

The chapter is structured as follows. Section 4.2 inspects the structure of qubit excitations to motivate the definition of the CEOs, which are then used to define the CEO-ADAPT-VQE algorithm in Sec. 4.3. Section 4.4 contains the results of numerical simulations comparing the newly proposed algorithm with previous ADAPT-VQE variants and state-of-the-art static ansätze. Section 4.5 ends the chapter with concluding remarks.

4.2 QE Circuit Implementation

We begin by delving into the structure of double QEs [182] and the corresponding circuit implementation.

Suppose we have a set of four qubits, two of which correspond (under the Jordan-Wigner mapping) to α-type spin-orbitals and two of which correspond to β-type ones. Accordingly, we label them α_1, α_2, β_1, β_2, where the numeric labels within the same spin-orbital type can be chosen arbitrarily. There will be two unique double QEs acting on these spin-orbitals:

$$T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} = Q^\dagger_{\alpha_2}Q^\dagger_{\beta_2}Q_{\alpha_1}Q_{\beta_1} - Q^\dagger_{\beta_1}Q^\dagger_{\alpha_1}Q_{\beta_2}Q_{\alpha_2}, \tag{4.1}$$

$$T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2} = Q^\dagger_{\alpha_1}Q^\dagger_{\beta_2}Q_{\alpha_2}Q_{\beta_1} - Q^\dagger_{\beta_1}Q^\dagger_{\alpha_2}Q_{\beta_2}Q_{\alpha_1}, \tag{4.2}$$

where

$$\begin{aligned} Q^\dagger_i &= \frac{1}{2}(X_i - iY_i), \\ Q_i &= \frac{1}{2}(X_i + iY_i), \end{aligned} \tag{4.3}$$

are the qubit creation and annihilation operators. They are equivalent to the fermionic creation and annihilation operators after removal of the Jordan-Wigner anticommutation string (see Eq. (2.6)).

Under the Jordan-Wigner transform, the state of each qubit represents the occupation number of a spin-orbital. Slater determinants are represented by computational basis states whose particle number is

given by the Hamming weight of the corresponding bit string, and whose spin quantum number is given by $\frac{1}{2}(N_\alpha - N_\beta)$, where N_α and N_β are the total occupation numbers of all qubits representing α and β spin-orbitals, respectively. It is easy to see that the operators in Eqs. (4.1) and (4.2) exchange two determinants in such a way that these quantities are preserved.

Operators $T^{(QE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$, $T^{(QE)}_{\alpha_1\beta_2\to\alpha_2\beta_1}$ are also valid QEs; however, they differ from the operators in Eqs. (4.1), (4.2) (respectively) only by a minus sign. The sign reflects which electronic transition we label as an excitation and which we label as a de-excitation. As this labeling becomes irrelevant once the operator is multiplied by a variational parameter, these operators are redundant with the above, and we can freely choose either option for each case. As for the two non-redundant QEs, the difference between them is that the operator in Eq. (4.1) exchanges $|0\rangle_{\alpha_2}|1\rangle_{\alpha_1}|0\rangle_{\beta_2}|1\rangle_{\beta_1}$ with $|1\rangle_{\alpha_2}|0\rangle_{\alpha_1}|1\rangle_{\beta_2}|0\rangle_{\beta_1}$, while the operator in Eq. (4.2) exchanges $|1\rangle_{\alpha_2}|0\rangle_{\alpha_1}|0\rangle_{\beta_2}|1\rangle_{\beta_1}$ with $|0\rangle_{\alpha_2}|1\rangle_{\alpha_1}|1\rangle_{\beta_2}|0\rangle_{\beta_1}$. All other Slater determinants are quenched.

Henceforth we ignore all qubits on which the operator action is trivial, and assume that the four relevant qubits are ordered as α_2, α_1, β_2, β_1 from the most to the least significant. We can thus omit the indices; e.g., $|0\rangle_{\alpha_2}|1\rangle_{\alpha_1}|0\rangle_{\beta_2}|1\rangle_{\beta_1}$ and $X_{\alpha_2}Y_{\alpha_1}X_{\beta_2}X_{\beta_1}$ are represented simply as $|0101\rangle$ and $XYXX$ using little endian ordering. This choice is merely for the sake of clarity and incurs no loss of generality.

In terms of Pauli strings, the operators can be written as

$$\begin{aligned} T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} =& \frac{i}{8}(XXXY - XXYX + XYXX + XYYY \\ & - YXXX - YXYY + YYXY - YYYX) \\ =& \frac{i}{8}XXXY(1 - IIZZ + IZIZ - IZZI \\ & - ZIIZ + ZIZI - ZZII + ZZZZ), \end{aligned} \tag{4.4}$$

$$\begin{aligned} T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2} =& \frac{i}{8}(XXXY - XXYX - XYXX - XYYY \\ & + YXXX + YXYY + YYXY - YYYX) \\ =& \frac{i}{8}XXXY(1 - IIZZ - IZIZ + IZZI \\ & + ZIIZ - ZIZI - ZZII + ZZZZ). \end{aligned} \tag{4.5}$$

We factored out the $XXXY$ Pauli string to emphasize the action of each of the operators. We note that the choice of which Pauli string to factor out is irrelevant, as any of the eight would result in the same expression in brackets (with a -1 multiplicative factor in half of the cases). It is easy to see that the last bracketed expression in Eq. (4.4) acts on $|0101\rangle$ and $|1010\rangle$ as

$$\begin{aligned} |0101\rangle &\to +8\,|0101\rangle, \\ |1010\rangle &\to +8\,|1010\rangle, \end{aligned} \tag{4.6}$$

while quenching all other computational basis states. Similarly, the last bracketed expression in Eq. (4.5) quenches all computational basis states except $|1001\rangle$ and $|0110\rangle$, on which it acts as

$$\begin{aligned} |1001\rangle &\longrightarrow +8\,|1001\rangle\,, \\ |0110\rangle &\longrightarrow +8\,|0110\rangle\,. \end{aligned} \tag{4.7}$$

On the other hand, $iXXXY$ applies a bit flip (in the computational basis) to each of the qubits it acts on, changing the occupation number of the corresponding spin-orbitals. Additionally, it imparts a phase ± 1 which depends on the value of the 0th (rightmost) qubit. This phase is a reflection of our choice in labeling excitations / de-excitations.

The operators that constitute the ansatz are parameterized and exponentiated versions of qubit excitations:

$$U^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} = e^{\theta T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}}, \tag{4.8}$$

$$U^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2} = e^{\theta T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2}}. \tag{4.9}$$

The unitary in Eq. (4.8) will rotate $|0101\rangle$, $|1010\rangle$ as

$$\begin{aligned} |0101\rangle &\longrightarrow \cos\theta\,|0101\rangle + \sin\theta\,|1010\rangle\,, \\ |1010\rangle &\longrightarrow \cos\theta\,|1010\rangle - \sin\theta\,|0101\rangle\,, \end{aligned} \tag{4.10}$$

while the one in Eq. (4.9) will rotate $|1001\rangle$, $|0110\rangle$ as

$$\begin{aligned} |1001\rangle &\longrightarrow \cos\theta\,|1001\rangle + \sin\theta\,|0110\rangle\,, \\ |0110\rangle &\longrightarrow \cos\theta\,|0110\rangle - \sin\theta\,|1001\rangle\,. \end{aligned} \tag{4.11}$$

All other Slater determinants are left unchanged.

An important question is how to implement the unitaries in Eqs. (4.8), (4.9) as quantum circuits. Since all the Pauli strings in the exponent commute, no Trotterization is required, and we can implement the exponentials of the eight strings in sequence using eight pairs of three-step `CNOT` ladders [123]. Naively, this would require 48 `CNOT`s. However, if the `CNOT`s between rotations are instead implemented such that they all share the same target, and we organize the Pauli strings such that two consecutive ones differ on two qubits, the `CNOT` count can be reduced to 13 [121, 169].

Another alternative, proposed in Ref. [179], is to leverage the fact that the operators are simply conditional rotations. More precisely, the rotations are applied to a computational basis state or not, depending on the parities of the states of some subsets of qubits. For example, the operator in Eq. (4.8) applies a rotation to a computational basis state $|x_3x_2x_1x_0\rangle$ if and only if $\overline{x_0 \oplus x_2} \wedge \overline{x_1 \oplus x_3} \wedge (x_0 \oplus x_1)$. The first (second) term guarantees that the occupation number of orbitals α_1 and β_1 (α_2 and β_2) is the same. The last term guarantees that if the former are occupied, the latter are unoccupied or vice-versa. Since `CNOT`s act as reversible XOR gates, it becomes evident that this operator can be implemented as in Fig. 4.1. Similarly, the operator in Eq. (4.9) can be implemented by the circuit in Fig. 4.2.

We can then rewrite the circuits in Figs. 4.1 and 4.2 in terms of single-qubit and `CNOT` gates. It was shown in Ref. [179] that a wisely chosen implementation of the multi-controlled rotation requires eight

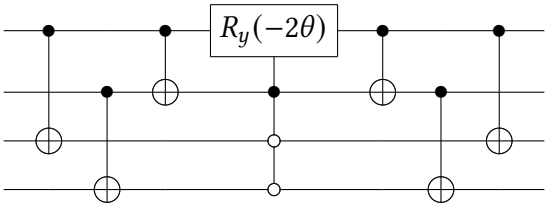


Figure 4.1: Circuit implementation of the QE evolution $U_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped. While the sign is necessary for the circuit to correspond exactly to this operator, it becomes irrelevant when θ is optimized variationally.

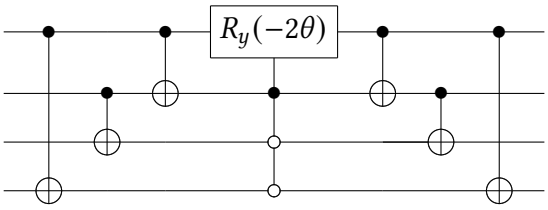


Figure 4.2: Circuit implementation of the QE evolution $U_{\alpha_2\beta_1\to\alpha_1\beta_2}$. The circuit for $U_{\alpha_1\beta_2\to\alpha_2\beta_1}$ is identical, but with the rotation angle flipped.

CNOTs, one of which cancels out with another one in the outer circuit. This results in circuits whose CNOT count of 13 matches that of the optimized product implementation of Refs. [121, 169], but whose CNOT depth is decreased to 11 (instead of 13). Figure 4.3 shows the result of using this strategy to decompose the rotation in the circuit of Fig. 4.1. A similar strategy can be applied to the circuit in Fig. 4.2, and more generally to any double QE evolution.

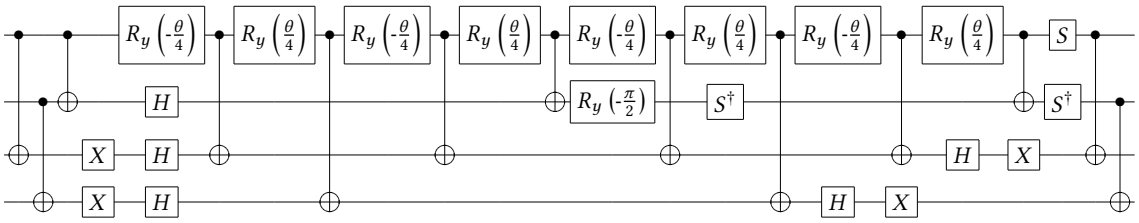


Figure 4.3: Explicit implementation of the qubit excitation evolution $U_{\alpha_1\beta_1\to\alpha_2\beta_2}$.

We recall that we are considering the special case of double QEs acting on four spin-orbitals which are equally divided between α-type and β-type. If all orbitals are of the same type, there are not two, but three unique double QEs: $T^{(QE)}_{\alpha_1\alpha_3\to\alpha_2\alpha_4}$, $T^{(QE)}_{\alpha_1\alpha_4\to\alpha_2\alpha_3}$, and $T^{(QE)}_{\alpha_1\alpha_2\to\alpha_3\alpha_4}$ for α-type, and similarly for β-type. Their structure and circuit implementation can easily be found using the same methods. Assuming they are ordered as α_4, α_3, α_2, α_1 (little endian), the unitary generated by $T^{(QE)}_{\alpha_1\alpha_3\to\alpha_2\alpha_4}$ can be implemented as in Fig. 4.1, and the one generated by $T^{(QE)}_{\alpha_1\alpha_4\to\alpha_2\alpha_3}$ as in Fig. 4.2. None of the circuits directly implements $T^{(QE)}_{\alpha_1\alpha_2\to\alpha_3\alpha_4}$, but the corresponding circuit implementation is easily derived from the others (e.g. we can simply exchange the roles of qubits 2 and 3 in Fig. 4.1). It is then straightforward to again obtain explicit circuits with a CNOT count of 13 and CNOT depth of 11.

Single and double QEs are the constituents of the pool used in QEB-ADAPT-VQE [182], the most circuit-efficient ADAPT-VQE protocol to date. We have not discussed single QEs. Evidently, they only

exist for pairs of spin-orbitals of the same type, and each such pair admits exactly one unique QE. The corresponding evolutions can be implemented using circuits with 2 CNOTs [182].

4.3 CEO-ADAPT-VQE

The choice of operator pool dictates the structure of the unitaries that will appear in the ansatz. Drawing inspiration from unitary coupled cluster theory, the first version of ADAPT-VQE [72] used a pool consisting of anti-Hermitian sums of fermionic excitations mapped into qubit operators via the Jordan-Wigner transform [86]. However, the corresponding evolutions (exponentials generated by these operators) are not convenient to implement with a quantum circuit. In order to preserve particle number and total Z spin projection (S_Z), they must correspond to a linear combination of several Pauli strings; and due to the nonlocality inherent to the antisymmetry of the fermionic many-body wave function, the strings must grow linearly with the number of orbitals in the basis set. This means that the number of entangling gates required to implement the unitary generated by each pool operator will grow linearly (on average) with the number of qubits. This was improved in Ref. [160], which proposed Qubit-ADAPT-VQE, a variant of the algorithm where the pool consists of individual Pauli strings. They showed that a pool of operators which do not respect fermionic anticommutation nor preserve particle number and S_Z can produce circuits with a significantly lower CNOT count than the original fermionic pool, at the expense of a higher number of variational parameters and measurements. Circuit efficiency was further improved in Ref. [182], which proposed a pool of QEs, operators that preserve particle number and S_Z symmetries despite not respecting anticommutation. The symmetry-preserving structure of QEs can be leveraged to create Controlled NOT (CNOT)-efficient circuits for the corresponding evolutions [179], and the resulting QEB-ADAPT-VQE algorithm stands as the most hardware-efficient variant as of today.

In this section, we propose CEO-ADAPT-VQE, an adaptive VQE based on CEOs. These operators consist of linear combinations of QEs acting on the same set of spin-orbitals, which may share one variational parameter (One Variational Parameter (OVP)-CEOs) or be independently parameterized (Multiple Variational Parameter (MVP)-CEOs). These operators are capable of simultaneously realizing exchanges between multiple pairs of Slater determinants, in contrast with QEs, which exchange exactly one pair of determinants each. We show that evolutions of MVP-CEOs consisting of up to three QEs can be implemented by circuits with the same CNOT count as the evolution of each individual excitation, while OVP-CEO evolutions can be implemented by circuits with roughly 30% *fewer* CNOTs. The CNOT counts are the key cost to consider when dealing with NISQ devices, as they are by far associated with the highest error rates. We provide explicit constructions for all circuits and numerically simulate CEO-ADAPT-VQE for multiple molecules. The results show that this algorithm reduces the CNOT count with respect to both QEB- and Qubit-ADAPT-VQE, with a decrease of roughly 50% and 65% (respectively) for the most difficult systems. The difference is expected to increase with the system size. This decrease does not entail any collateral increase in measurement costs. In fact, with respect to the other variants, our algorithm maintains or decreases the total number of gradient measurement rounds as well as of variational parameters.

We will motivate the introduction of these operators with a detailed analysis of QEs (Sec. 4.2). Secs. 4.3.1 and 4.3.2 introduce two different classes of CEOs. Finally, the complete CEO-ADAPT-VQE algorithm is proposed in Sec.4.3.3.

4.3.1 Multiple Variational Parameters MVP-CEOs

In the previous section we saw that each QE exchanges exactly two Slater determinants. In this section, we will define operators which are capable of doing all valid exchanges simultaneously. By 'valid exchanges' we mean those that preserve particle number and S_z. In the example of the previous subsection, this would mean exchanging $|0101\rangle \leftrightarrow |1010\rangle$ *and* $|1001\rangle \leftrightarrow |0110\rangle$.

As before, we consider a set of four spin-orbitals which is equally divided between α- and β-type. We do not enforce restrictions on the spin-orbitals; they may or not be occupied in the reference state. We label them α_1, β_1, α_2, β_2 and define the following family of parameterized operators, consisting of linear combinations of the two unique QEs acting on these spin-orbitals:

$$\begin{aligned} T^{(MVP-CEO)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta_1,\theta_2) &= \theta_1 T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} + \theta_2 T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2} \\ &= \theta_1 Q^\dagger_{\alpha_2} Q^\dagger_{\beta_2} Q_{\alpha_1} Q_{\beta_1} + \theta_2 Q^\dagger_{\alpha_1} Q^\dagger_{\beta_2} Q_{\alpha_2} Q_{\beta_1} - h.c. \end{aligned} \tag{4.12}$$

We note once again that the choice of whether to use, e.g., $T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$ or $T^{(QE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is irrelevant. A different choice might at most lead to a minus sign, which is absorbed by the variational parameters. Up to this irrelevant degree of freedom, the operator is unique. Thus, the unordered set of spin-orbital indices is enough to identify it unambiguously.

We call the operators of the type of Eq. (4.12) CEOs, because they combine the exchanges corresponding to multiple QEs in one operator. We use MVP (*multiple variational parameters*) to indicate that the different exchanges are independently parameterized.

The operator

$$U^{(MVP-CEO)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta_1,\theta_2) = e^{T^{(MVP-CEO)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta_1,\theta_2)} \tag{4.13}$$

will act as

$$\begin{aligned} |0101\rangle &\to \cos\theta_1 |0101\rangle + \sin\theta_1 |1010\rangle, \\ |1010\rangle &\to \cos\theta_1 |1010\rangle - \sin\theta_1 |0101\rangle, \\ |1001\rangle &\to \cos\theta_2 |1001\rangle + \sin\theta_2 |0110\rangle, \\ |0110\rangle &\to \cos\theta_2 |0110\rangle - \sin\theta_2 |1001\rangle , \end{aligned} \tag{4.14}$$

where we again assume the qubits to be labeled α_2, α_1, β_2, β_1 from the most to the least significant. All other Slater determinants are left unchanged. Unlike QEs, these operators act non-trivially on *all* Slater determinants where the underlying rotation preserves S_z and particle number. In what concerns single excitations, we see that CEOs and QEs are identical—in this case, there is only one valid exchange.

The question that remains is how to create a quantum circuit which implements the coupled exchange evolution in Eq. (4.13). Since QEs acting on the same set of spin-orbitals commute, we can implement

this unitary by concatenating two circuits with a similar structure to the one in Fig. 4.3, resulting in a circuit with a total of 26 CNOTs. However, this is not optimal. Plugging the expressions in Eqs. (4.1) and (4.2) into the definition of the CEOs, we see that they consist of a linear combination of eight Pauli strings:

$$\begin{aligned} &T^{(MVP-CEO)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta_1,\theta_2) = \\ &\frac{i}{8}[+(\theta_1+\theta_2)XXXY-(\theta_1+\theta_2)XXYX \\ &+(\theta_1-\theta_2)XYXX+(\theta_1-\theta_2)XYYY \\ &-(\theta_1-\theta_2)YXXX-(\theta_1-\theta_2)YXYY \\ &+(\theta_1+\theta_2)YYXY-(\theta_1+\theta_2)YYYX]. \end{aligned} \tag{4.15}$$

This happens because all QEs acting on the same set of spin-orbitals consist of uniformly weighted linear combinations of the same Pauli strings, with the difference residing in the signs of the coefficients.

The circuit implementation specific to QEs proposed in Ref. [179] does not apply to our CEOs, because it relies on the fact that all Pauli strings have the same weight. However, we can implement the unitary generated by the operator in Eq. (4.15) using the optimized circuits for exponentials of commuting Pauli strings proposed in Refs. [121, 169]. In Fig. 4.4 we provide a 13-CNOT circuit which implements the exponential of any linear combination of the eight Pauli strings we are concerned with, and which implements the unitary $e^{T^{(MVP-CEO)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta_1,\theta_2)}$ as a special case with only two independent parameters.

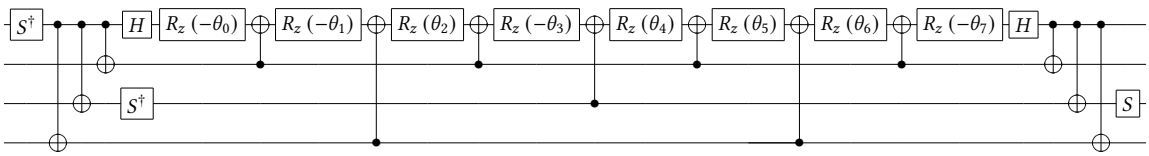


Figure 4.4: Circuit implementation of the unitary generated by any linear combination of Pauli strings in a QE, $e^{\frac{i}{8}(\theta_0 XXXY+\theta_1 XXYX+\theta_2 YXYY+\theta_3 YXXX+\theta_4 YYXY+\theta_5 YYYX+\theta_6 XYYY+\theta_7 XYXX)}$.

So far, we have focused on the case where we have four spin-orbitals which are equally divided between α-type and β-type (often referred to as "opposite-spin excitations"). If we instead we have a set of four spin-orbitals of the same type ("same-spin excitations"), we have not two, but three distinct QEs acting on this set. In this case, we have a CEO with three variational parameters, which simultaneously exchanges three different pairs of Slater determinants. Nevertheless, it still corresponds to a linear combination of the same eight Pauli strings, so that the corresponding unitary can be implemented by the 13-CNOT circuit of Fig. 4.4.

Since two-qubit QEs realize the one and only viable determinant exchange for the corresponding set of spin-orbitals, they trivially belong to the MVP-CEO set.

4.3.2 One Variational Parameter (OVP)-CEOs

In the preceding section we proposed CNOT-efficient circuits for MVP-CEOs. We will now show that it is possible to further decrease the CNOT count for specific values of the variational parameters θ_1, θ_2.

Specifically, if the two variational parameters have the same magnitude, we obtain the following two subfamilies of CEOs:

$$\begin{aligned} T^{(OVP-CEO,+)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta) &= \theta(T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} + T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2}) \\ &= \theta(Q^\dagger_{\alpha_2}Q^\dagger_{\beta_2}Q_{\alpha_1}Q_{\beta_1} + Q^\dagger_{\alpha_1}Q^\dagger_{\beta_2}Q_{\alpha_2}Q_{\beta_1}) - h.c. \end{aligned} \tag{4.16}$$

$$\begin{aligned} T^{(OVP-CEO,-)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta) &= \theta(T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} - T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2}) \\ &= \theta(Q^\dagger_{\alpha_2}Q^\dagger_{\beta_2}Q_{\alpha_1}Q_{\beta_1} - Q^\dagger_{\alpha_1}Q^\dagger_{\beta_2}Q_{\alpha_2}Q_{\beta_1}) - h.c. \end{aligned} \tag{4.17}$$

We call these operators OVP-CEOs because they each have one variational parameter. We additionally use the superscript $(+/-)$ to distinguish the two subfamilies. Note that the labeling is arbitrary—the signs are exchanged if we use $T^{(QE)}_{\alpha_1\beta_2\to\alpha_2\beta_1}$ instead of $T^{(QE)}_{\alpha_2\beta_1\to\alpha_1\beta_2}$. It can be readily seen that these operators consist of linear combinations of only four Pauli strings:

$$\begin{aligned} &T^{(OVP-CEO,+)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta) = \\ &\frac{i\theta}{4}\left(XXXY - XXYX + YYXY - YYYX\right) = \\ &\frac{i\theta}{4}XXXY\left(1 - IIZZ - ZZII + ZZZZ\right), \end{aligned} \tag{4.18}$$

$$\begin{aligned} &T^{(OVP-CEO,-)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta) = \\ &\frac{i\theta}{4}\left(XYXX + XYYY - YXXX - YXYY\right) = \\ &\frac{i\theta}{4}XXXY\left(IZIZ - IZZI - ZIIZ + ZIZI\right). \end{aligned} \tag{4.19}$$

As before, we define the corresponding unitaries as

$$U^{(OVP-CEO,\pm)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta) = e^{T^{(OVP-CEO,\pm)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta)}. \tag{4.20}$$

The terms in each operator commute and they can be organized such that two adjacent terms differ on exactly two qubits. Then, it is straightforward to see that, using once again the methods of Refs. [121, 169], the corresponding unitaries can be implemented with a total `CNOT` count and depth of 9.

We can further improve this if we understand *why* the CEOs correspond to a sum of a lower number of Pauli strings. This happens because, as compared to QEs, they are conditional on a simpler function of the parities of subsets of qubit states. We saw that $T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$ will only *not* quench a Slater determinant under three conditions: (i) α_1 and β_1 must have the same occupation number, (ii) equivalently for α_2 and β_2, and (iii) the occupation numbers in (i) must be opposite to those in (ii). These three conditions give rise to the three-control rotation in Fig. 4.1. In contrast, the CEO $T^{(OVP-CEO,\pm)}_{\alpha_1\beta_1\alpha_2\beta_2}$ will not quench those Slater determinants if (i) the total occupation number of the α orbitals is exactly one, and (ii) equivalently for the β orbitals. This observation leads us to the circuit implementations in Figs. 4.5 and 4.6.

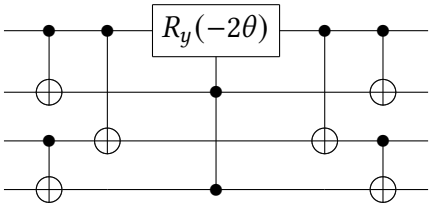


Figure 4.5: Circuit implementation of $U^{(OVP-CEO,+)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta)$.

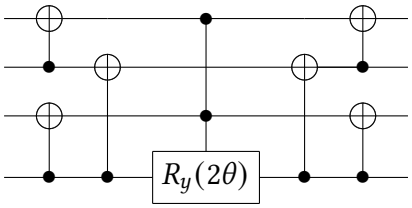


Figure 4.6: Circuit implementation of $U^{(OVP-CEO,-)}_{\alpha_1\beta_1\alpha_2\beta_2}(\theta)$.

To decompose the multi-controlled rotations, we follow a similar procedure to Ref. [179]. We implement them as in Fig. 4.7, convert the CZ gates into CNOT gates where the control is the 0th qubit, and use the identity in Fig. 4.8 to remove a CNOT. The final circuit for $U^{(OVP-CEO,+)}_{\alpha_1\beta_1\alpha_2\beta_2}$ is shown in Fig. 4.9. A similar decomposition can be obtained for $U^{(OVP-CEO,-)}_{\alpha_1\beta_1\alpha_2\beta_2}$. With this, we see that coupled exchange evolutions can be implemented with a CNOT count of 9 and a CNOT depth of 7. This should be contrasted with the circuits for double QE evolutions, for which these values are 13 and 11, respectively. As compared to the circuits for the 4-qubit generators in the qubit pool [160], with a CNOT count and depth of 6, the circuits we propose barely increase the CNOT depth. Yet, unlike them, our circuits fully conserve S_z and particle number.

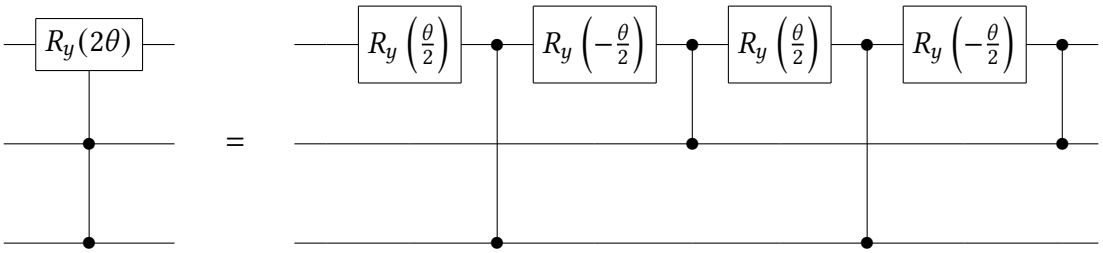


Figure 4.7: Decomposition of a two-control rotation. Any controlled gate can be used, as long as it imposes on the target (qubit 0) a reflection with respect to the Y axis, so that the sense of the rotation is inverted if the control is in state $|1\rangle$. We further note that the roles of the control qubits can be interchanged; this particular choice was made so that one controlled gate will be eliminated via the identity in Fig. 4.8.

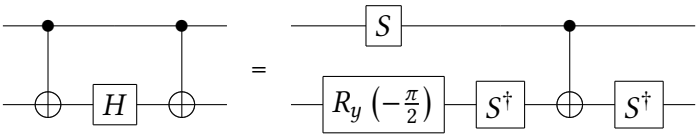


Figure 4.8: Equivalence used to reduce the CNOT count.

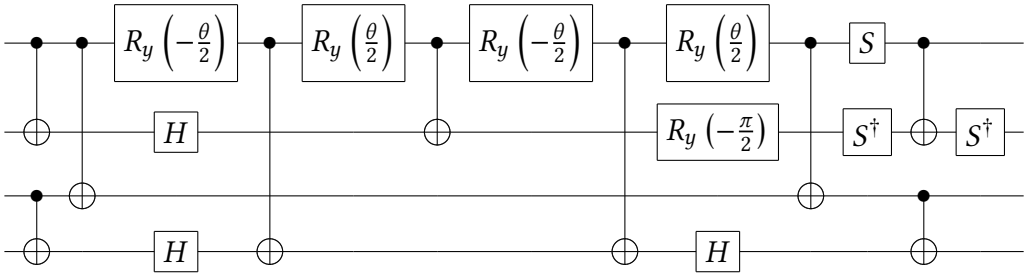


Figure 4.9: Explicit implementation of the circuit in Fig. 4.5 using only `CNOT`s and single-qubit gates.

As before, we focused our analysis on the case where the spin-orbitals are equally divided between α- and β- type. If all orbitals are of the same type, we consider all six unique OVP-CEOs that can be formed from sums and differences of pairs of these excitations. It is straightforward to generalize the circuit implementations. We additionally recall that all single QEs trivially belong to the OVP-CEO set.

4.3.3 The CEO-ADAPT-VQE Algorithm

We now propose the CEO-ADAPT-VQE algorithm, which makes use of the CEO operators defined in the previous subsections. This algorithm follows the usual ADAPT-VQE workflow (see Sec. 2.2.3), with modifications in the gradient measurement and operator selection steps. The need for these modifications stems from MVP-CEOs having multiple variational parameters, unlike the operators of any previously proposed pool. A consequence of this is that there is no unique gradient to use as the selection criterion.

We begin by specifying the adopted notation. We use $\{T_k^{(OVP-CEO)}\}$ to denote the pool formed from the set of OVP-CEOs [1]. For each $A \in \{T_k^{(OVP-CEO)}\}$, we define a set

$$M^{(QE)}(A) = \{B \in \{T_k^{(QE)}\} : \text{Supp}(B) = \text{Supp}(A)\}, \tag{4.21}$$

where 'Supp' denotes the support of an operator (the set of qubits on which it acts non-trivially). $M^{(QE)}(A)$ is the set of QEs acting on the exact same spin-orbitals (qubits) as A. We further define $M_{\neq 0}^{(QE)}(A)$ as the set obtained from $M^{(QE)}(A)$ by removing all elements associated with zero energy derivatives. We use # to denote the cardinality of a set, and define a function `max_g` which takes an operator pool and selects the element generating the unitary with the highest gradient magnitude (at the point where the parameter is zero).

Modified gradient measurement step:

- Evaluate the gradients of the unitaries generated by elements of the operator pool $\{T_k^{(OVP-CEO)}\}$. If the norm of the vector formed by these gradients is under a pre-defined convergence threshold ϵ, terminate. [2]

[1] Note that we have defined the pool operators to include the variational parameters. This is because our operator selection step might append to the ansatz an unitary with multiple variational parameters. As such, our protocol does not fit into the usual framework where an unparameterized generator is multiplied by a single variational parameter.

[2] This is the original gradient measurement step, with the specification that the pool is the set of OVP-CEOs.

Modified operator selection step:

- Define $T_n^{(OVP-CEO)} \equiv \texttt{max_g}(\{T_k^{(OVP-CEO)}\})$.
- If $\#M_{\neq 0}^{(QE)}(T_n^{(OVP-CEO)}) = 1$, add $e^{T_n^{(OVP-CEO)}}$ to the ansatz.
- Otherwise, add $e^{T_n^{(MVP-CEO)}}$ to the ansatz, where

$$T_n^{(MVP-CEO)} = \sum_{T_i^{(QE)} \in M_{\neq 0}^{(QE)}(T_n^{(OVP-CEO)})} \theta_i T_i^{(QE)}. \tag{4.22}$$

  Here θ_i is an independent variational parameter associated with the generator $T_i^{(QE)}$. Note that the new unitary contains either two or three variational parameters.

Note that gradient measurement step implies the same number of measurements for CEO-, QEB- and Qubit-ADAPT-VQE, because these operators are all formed from the same set of Pauli strings. The cost is $\mathcal{O}(N^5)$, where N is the number of spin-orbitals [9]. However, the total measurement cost will depend on the number of iterations. The modified operator selection step requires knowing the gradients of some QE evolutions, but such gradients are always linear combinations of gradients of OVP-CEO evolutions. Thus, all data required in the operator selection step of CEO-ADAPT-VQE was already collected in gradient measurement step.

In formulating a variant of ADAPT-VQE with CEOs, the main question is which operator to choose in each iteration. ADAPT-VQE typically chooses operators based on the gradient selection criterion; however, since MVP-CEOs do not have a unique variational parameter, this criterion does not straightforwardly apply to them. The criterion could be generalized to the sum of the gradient magnitudes of each independently parameterized constituent; for an MVP-CEO, this would correspond to the sum of the gradient magnitudes of all the QEs it contains, while for an OVP-CEO this would be the gradient magnitude as usual.

However, note that the gradient magnitude associated with $T_n^{OVP-CEO}$ is the sum of the gradient magnitudes of its constituent QEs. To see this, consider two OVP-CEOs acting on the same set of spin-orbitals, $T^{(OVP-CEO,+)}$ and $T^{(OVP-CEO,-)}$. By the triangle inequality, the gradient magnitude associated with each is bounded above by the sum of the magnitudes of the gradients associated with the constituent QEs. If the latter gradients have the same sign, $T^{(OVP-CEO,+)}$ saturates the bound; if the signs are opposite, $T^{(OVP-CEO,-)}$ does. Since $T_n^{OVP-CEO}$ is by definition the OVP-CEO associated with the highest gradient, its type (sum/difference) is the one which saturates the bound. Thus, the gradient associated with the selected OVP-CEO is the same as what we would obtain applying the generalized gradient criterion to the MVP-CEO formed from the same QEs. In summary, the three operators

1. OVP-CEO with the highest gradient magnitude among all OVP-CEOs (that we labeled $T_n^{(OVP-CEO)}$ in the gradient measurement step)
2. MVP-CEO whose QEs have the highest total sum of gradient magnitudes (‘generalized gradient’) among all MVP-CEOs

3. MVP-CEO acting on the same spin-orbitals as 1, minus the QEs with zero gradients (that we labeled $T_n^{(MVP-CEO)}$ in the gradient measurement step)

usually have equivalent gradients, and operators 2 and 3 are usually the same (this is not always the case due to the existence of MVP-CEOs with 3 QEs; however, the difference is, in general, not significant enough to impact the results). In light of this, we choose to consider 3 as the leading candidate among all MVP-CEOs.

The next step is to decide between $T_n^{(OVP-CEO)}$ and $T_n^{(MVP-CEO)}$, since we have a tie in (generalized) gradients. OVP-CEOs have the benefit of being implemented with fewer CNOTs, while MVP-CEOs offer more variational freedom; the question is how to balance these factors. We choose to allow a QE to have an individual variational parameter if and only if it has a nonzero gradient. This is based on the expectation that, as a general rule, operators with zero gradients have no impact on the energy regardless of the value of the variational parameter, which we confirm in numerical simulations. Thus, if we consider an OVP-CEO of the form $\theta(T_1^{(QE)} \pm T_2^{(QE)})$ where $T_2^{(QE)}$ has zero gradient in the current variational state, the corresponding evolution $e^{\theta(T_1^{(QE)} \pm T_2^{(QE)})}$ is equivalent to $e^{\theta T_1^{(QE)}}$ when acting on this particular state. Despite being equivalent in practice, the former unitary has a more efficient implementation (Fig. 4.9) than the latter (Fig. 4.3), requiring 9 CNOTs instead of 13.

Let us take as a simple example the case of the H_2 molecule, represented by four qubits in a minimal basis set. We use little-endian ordering and assume the spin-orbitals to be ordered 1-4, where the first two are β-type, and the others are α-type. At the beginning of the first iteration, we represent the state by $|0101\rangle$ (the Hartree-Fock solution). For this system, we have only two distinct double QEs, $T^{QE}_{1,3\to2,4}$ and $T^{QE}_{1,4\to2,3}$. The first one acts as

$$\begin{aligned} |0101\rangle &\longrightarrow +|1010\rangle\,, \\ |1010\rangle &\longrightarrow -|0101\rangle\,, \end{aligned} \tag{4.23}$$

while the second acts as

$$\begin{aligned} |1001\rangle &\longrightarrow +|0110\rangle\,, \\ |0110\rangle &\longrightarrow -|1001\rangle\,. \end{aligned} \tag{4.24}$$

As before, all other determinants are quenched. It is then clear that the unitary $e^{\theta T^{QE}_{1,4\to2,3}}$ leaves the Hartree-Fock reference state unchanged for any value of θ (i.e., $e^{\theta T^{QE}_{1,4\to2,3}}|0101\rangle = |0101\rangle$). While $e^{\theta(T^{QE}_{1,3\to2,4}+T^{QE}_{1,4\to2,3})}$ and $e^{\theta T^{QE}_{1,3\to2,4}}$ are distinct unitaries, they are equivalent when acting on any state that is orthogonal to $|1001\rangle$ and $|0110\rangle$. Therefore, we can choose the one that has the most efficient circuit implementation. This choice will not impact the variational state.

This motivates us to choose an OVP-CEO as opposed to an MVP-CEO when only one of the constituent excitations has nonzero gradient; however, the question of which CEO is the best choice when multiple constituent QEs have nonzero gradients remains open. It could be beneficial to opt for $T_n^{(OVP-CEO)}$ instead of $T_n^{(MVP-CEO)}$ even when the gradients of the underlying excitations are both nonzero. $T_n^{(OVP-CEO)}$ contains the pair of same-support QEs for which the sum of gradient magnitudes is the highest. If the signs

of the gradients are the same, the operator $T_n^{(OVP-CEO)}$ is of the form $T_n^{(OVP-CEO,+)} = \theta(T_1^{QE} + T_2^{QE})$; otherwise, it is of the form $T_n^{(OVP-CEO,-)} = \theta(T_1^{QE} - T_2^{QE})$. Note that the sign of the gradient dictates the sign of the local minimizer found by optimizing the corresponding parameter (they are opposite). Since both the sum and the difference are included in the OVP-CEO pool, the gradient selection naturally favors coupling pairs of same-support excitations in such a way that the sign of the parameter can be matched with the sign of the minimizer for both operators. Of course, the magnitude is constrained to be the same for both. We could choose $T_n^{(OVP-CEO)}$ instead of $T_n^{(MVP-CEO)}$ hoping that the fact that the former has a more efficient circuit implementation compensates for the fact that the QEs are restricted to have the same parameter magnitude.

In App. F, we consider a variant of CEO-ADAPT-VQE where the choice between $T_n^{(OVP-CEO)}$ and $T_n^{(MVP-CEO)}$ is based on the energy change the corresponding unitaries are capable of producing, scaled by the CNOT count of the corresponding circuits. Surprisingly, despite making a choice that optimizes the energy change per unit CNOT in any given iteration, this energy-based decision criterion results in a higher CNOT count than the gradient-based one for the same error. This justifies our decision to always opt for $T_n^{(MVP-CEO)}$ as long as it contains more than one QE with nonzero gradient. We refer to App. F for a discussion concerning this seemingly unexpected result. For the sake of completeness, this section additionally considers variants of ADAPT-VQE which use either OVP- or MVP-CEOs exclusively.

4.4 Results

In this section, we compare the proposed CEO-ADAPT-VQE algorithm with the previous most hardware-efficient variants of ADAPT-VQE: QEB-ADAPT-VQE [182] and Qubit-ADAPT-VQE [160]. We first consider three systems in the STO-3G basis set: LiH (12 qubits) and BeH_2 (14 qubits) as examples of realistic molecules, and linear H_6 (12 qubits) as a proxy for strongly correlated systems whose classical simulation is not viable. We additionally consider triangular H_6, which is expected to be more difficult to simulate classically due to spin frustration. We use a higher convergence threshold ϵ for the largest system due to the corresponding pools being significantly larger.

Following the methods of Refs. [72, 160, 182] for pool benchmarking, we compute the expectation values via matrix algebra, without including shot noise or hardware noise. This prevents confounding factors in the comparison of the performance of the algorithms and allows for viable classical simulation runtimes. Since CNOT gates have the greatest physical error rates and error correction is out of bounds in the NISQ paradigm, we use the CNOT count of the circuits as the key figure to predict the impact of hardware noise in the algorithms [38]. Statistical noise is likely to significantly increase the measurement costs—hence, we expect our estimates to constitute a loose lower bound to the actual measurement costs in a realistic setting. The purpose of these estimates is not to provide an accurate estimation of these costs, but to get a sense of how they compare for the different algorithms.

The code used for the numerical simulations has been made publicly available on GitHub (https://github.com/mafaldaramoa/ceo-adapt-vqe). We use the Openfermion [5] and PySCF [6]

packages to create and manipulate fermionic operators. All errors are calculated with respect to the FCI energy. The classical optimizer employed on the VQE subroutine is BFGS as implemented in Scipy [167], which we locally modified to recycle the Hessian in order to decrease the simulation time (see Ch. 3).

This section is organized as follows. In Sec. 4.4.1, we compare the iteration, parameter, and CNOT counts of CEO-, QEB- and Qubit-ADAPT-VQE for various molecules at different geometries. Section 4.4.2 offers a comparison of CEO-ADAPT-VQE with UCCSD-VQE throughout the bond dissociation curves of the same molecules. In Sec. 4.4.3, we discuss enhancing CEO-ADAPT-VQE with other improvements proposed in the literature. To showcase the reduction in the resource requirements of ADAPT-VQE from the time it was first proposed, we compare this enhanced version with a fermionic variant of the algorithm as proposed in the original work of Ref. [72]. Finally, in Sec. 4.4.4 we compare our adaptive algorithm with the leading fixed-structure ansätze.

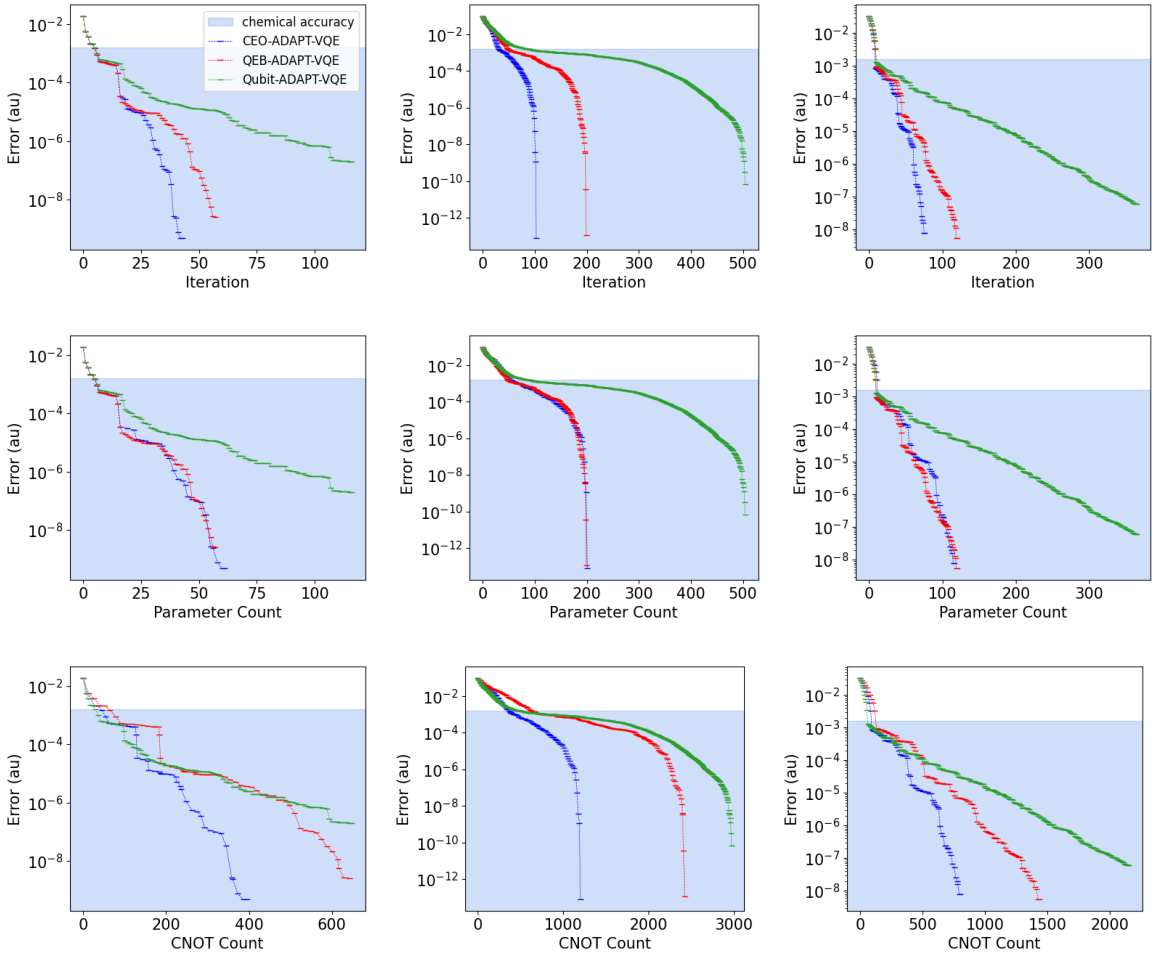
4.4.1 Convergence Plots: CEO- vs QEB- vs Qubit-ADAPT-VQE

Figure 4.10 shows the evolution of the three algorithms for three different molecules at bond distances close to equilibrium. In the upper panels, we can see that CEO-ADAPT-VQE is the fastest in decreasing the error with respect to the iteration number, which is equivalent to the number of optimizations and the number of gradient measurement rounds.

In the middle panels, we see that for any given accuracy, CEO-ADAPT-VQE requires roughly the same number of parameters as QEB-ADAPT-VQE and significantly fewer than Qubit-ADAPT-VQE. Together with the lower number of iterations, these results show that among the three protocols, ours is the most frugal in terms of measurement costs. We recall that in each iteration, CEO-ADAPT-VQE adds up to three new variational parameters which, on average, will be associated with lower gradient magnitudes than either QEB- or Qubit-ADAPT-VQE. As such, it is a surprising result that it does not require a higher number of variational parameters to achieve similar accuracy. We expect this to be related to nonlocal effects, as it suggests that operators with low gradient magnitudes added earlier may have an impact on the energy that is nearly as significant as that of operators with high gradient magnitudes added later on. It is plausible that an operator with a low gradient magnitude brings relevant Slater determinants into the superposition state, and/or interacts favorably with posteriorly added operators. As for the excess of variational parameters in Qubit-ADAPT-VQE as compared to CEO-/QEB-ADAPT-VQE, we attribute it to symmetry breaking.

Finally, the bottom panels show the most important cost: the CNOT count. We observe that CEO-ADAPT-VQE achieves a significant reduction with respect to either of the other two algorithms. For linear H_6, Fig. 4.10(b), it requires at the moment of convergence roughly a third of the CNOTs of Qubit-ADAPT-VQE, and a half of QEB-ADAPT-VQE. Further, the error it has achieved upon reaching the termination criterion is at least as low as that of the other two algorithms. It is notable that such a reduction in the CNOT count can be achieved without any setbacks in regards to measurement costs or optimization difficulty (as gauged by the number of parameters).

In Fig. 4.11 we consider the same molecules at stretched bond distances, where correlation effects should play a bigger role. Once more, we observe that CEO-ADAPT-VQE results in the most gate-efficient

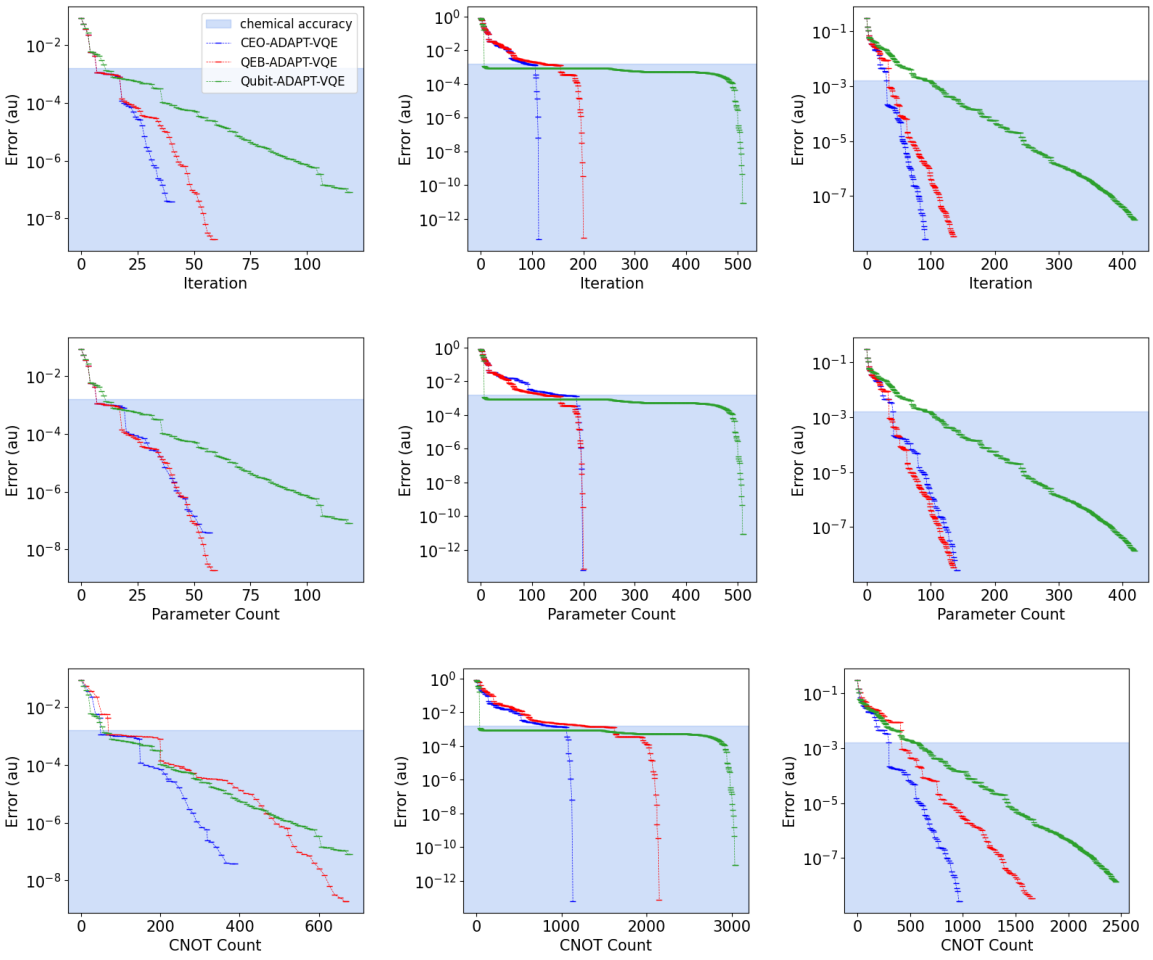


(a) LiH at 1.5Å (b) Linear H_6 at 1Å (c) BeH_2 at 1.3Å

Figure 4.10: Convergence of the CEO-, QEB-, and Qubit-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at bond distances close to equilibrium. The error is plotted against the iteration number (top), parameter count (middle), and CNOT count (bottom). The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). All algorithms terminate when the energy change falls below 10^{-10} Hartree.

circuits among all three algorithms, without any drawbacks in terms of the number of parameters, number of optimizations, or measurement costs—on the contrary, these costs are reduced.

We note that for stretched linear H_6 (Fig. 4.11), Qubit-ADAPT-VQE is the best-performing variant up to the boundary of the chemical accuracy region. It may seem that, in this case, Qubit-ADAPT-VQE is the best option if the goal is reaching chemical accuracy faster; however, the state found by this algorithm at that point is actually a low-lying excited state with an energy roughly 10^{-3}au higher than the ground state energy, which it has trouble steering away from for hundreds of iterations. Further, Qubit-ADAPT-VQE will generally converge faster in early iterations, where the state is likely less plagued by symmetry breaking (known to significantly affect convergence [21]). Because we are dealing with small, classically tractable systems, relatively few iterations suffice to reach chemical accuracy. However, a higher number of iterations will be required for larger molecules in general. Given that Qubit-ADAPT-VQE is typically outperformed by both CEO- and QEB-ADAPT-VQE after a few tens of iterations, we expect it not to be the



(a) LiH at 3Å (b) Linear H_6 at 3Å (c) BeH_2 at 3Å

Figure 4.11: Convergence of the CEO-, QEB-, and Qubit-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at stretched bond distances. The error is plotted against the iteration number (top), parameter count (middle) and CNOT count (bottom). The termination criterion is the same as in Fig. 4.10.

leading algorithm for systems that are not amenable to classical simulation. In fact, the visible trend in which the difference in the CNOT counts becomes more significant for more strongly correlated molecules and larger iteration numbers indicates that CEO-ADAPT-VQE will be particularly advantageous for systems which are larger and/or harder to simulate classically.

Finally, Fig. 4.12 shows the convergence of the error obtained by the three adaptive algorithms against the iterations, CNOT count/depth and number of parameters for a triangular H_6 molecule, which exhibits spin frustration. Among all the systems considered here, this is the most challenging, which is confirmed by the fact that a larger number of iterations, CNOTs and parameters are required to reach the same convergence criterion. Yet, CEO-ADAPT-VQE clearly remains the most efficient option, requiring less than half of the CNOTs of the alternatives. As before, the parameter count is roughly the same for the CEO and QE pools, while being higher for the qubit pool. This confirms that our protocol can be successfully applied to strongly correlated systems, the expected use cases of quantum algorithms.

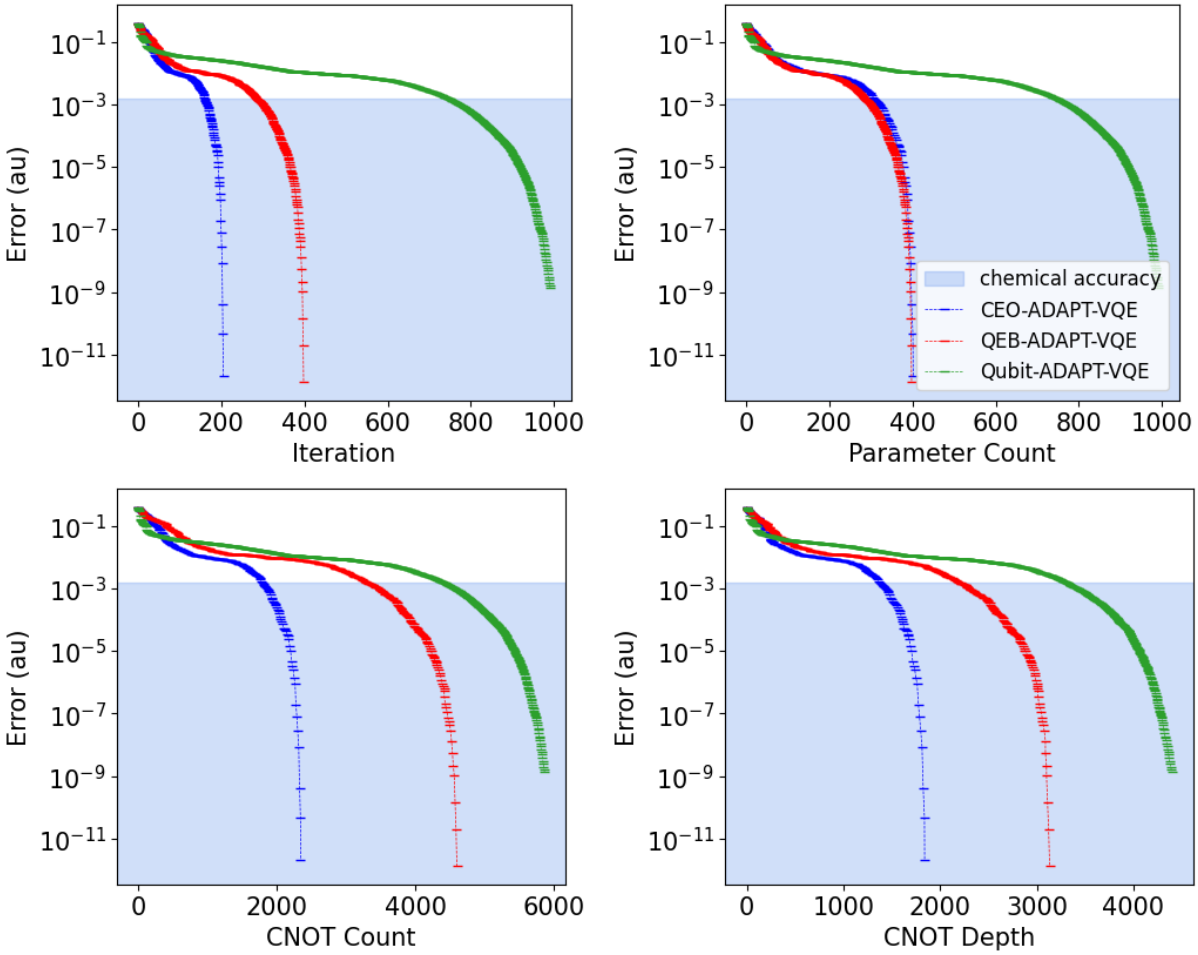


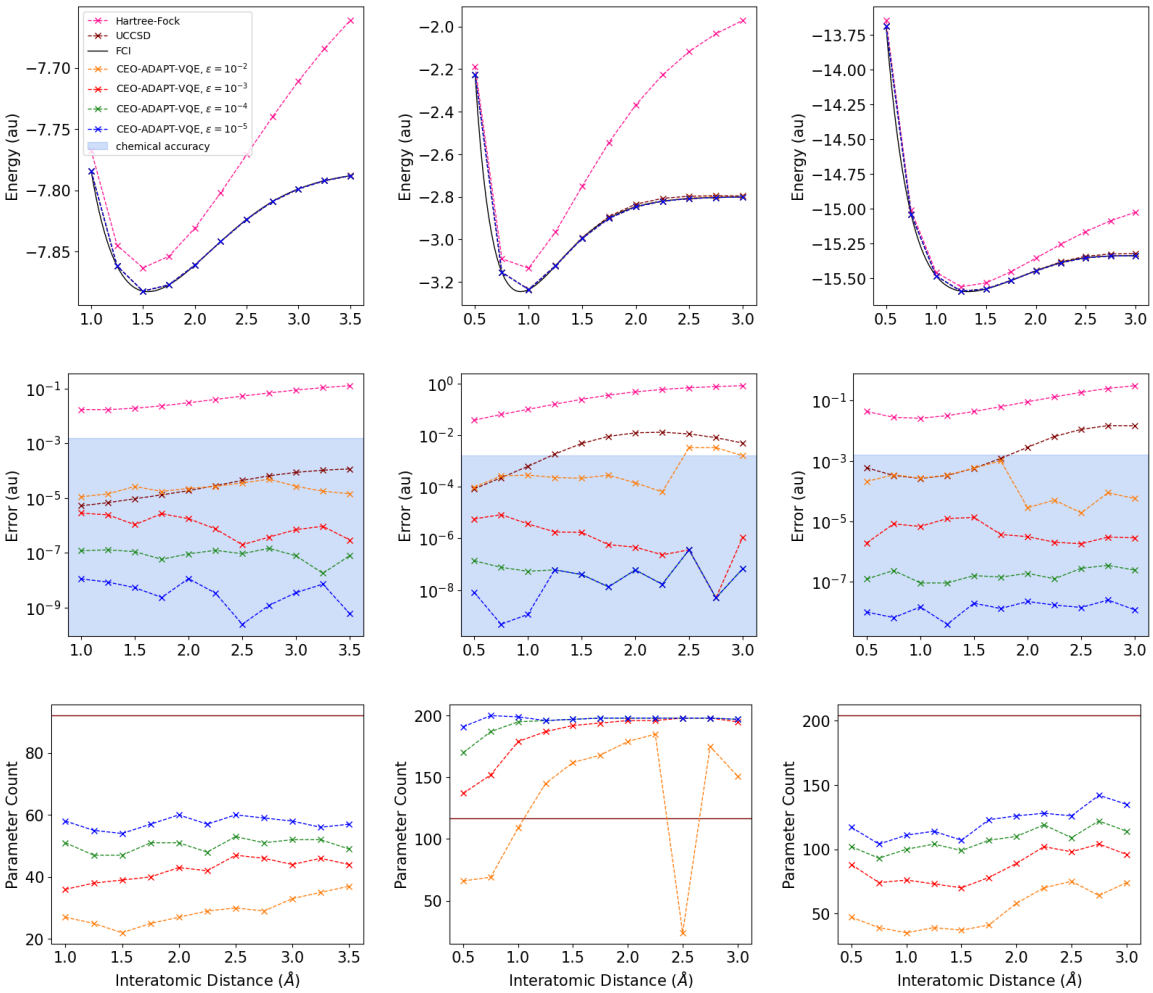
Figure 4.12: Convergence of the CEO-, QEB-, and Qubit-ADAPT-VQE algorithms for the triangular H_6 molecule at interatomic distance 2 Å. This system is particularly challenging to simulate due to corresponding to a spin-frustrated lattice. The error is plotted against the iteration number (top left), parameter count (top right), CNOT count (bottom left), and CNOT depth (bottom right). The termination criterion is the same as in Fig. 4.10.

4.4.2 Comparison with UCCSD-VQE through Bond Dissociation Curves

In this section, we compare CEO-ADAPT-VQE with (untrotterized) UCCSD-VQE for multiple values of the ϵ hyperparameter (the threshold on the gradient norm which defines convergence) on the energy, error, and parameter count.

Figure 4.13 shows these quantities as functions of the bond distance for all three molecules under study. In addition to UCCSD-VQE and CEO-ADAPT-VQE (with different values of ϵ), we include the Hartree-Fock and FCI solutions for reference. We recall that the former is the starting point, while the latter is the target.

In the upper and middle plots we observe that as expected, CEO-ADAPT-VQE always improves upon the Hartree-Fock solution by a significant margin. The middle plots show that as compared to UCCSD-VQE, it roughly matches or decreases the final error for all molecules and bond distances, and for all values of ϵ considered. Values of ϵ below 10^{-2} result in significant improvements with respect to UCCSD for all



(a) LiH (b) Linear H_6 (c) BeH_2

Figure 4.13: Energy, error and parameter count of CEO-ADAPT-VQE against the bond distance for **a** LiH, **b** linear H_6, **c** BeH_2. Four different values of the gradient convergence threshold ϵ are considered: 10^{-k} for $k \in \{2, 3, 4, 5\}$. The Hartree-Fock, (untrotterized) UCCSD and FCI solutions are also included for comparison. The UCCSD ansatz includes all occupied-to-virtual single and double excitations. The corresponding parameter count is fixed at 92, 117, and 204 for the three molecules in the order that they appear.

molecules and bond distances, with the error being lowered by a multiplicative factor of up to 10^6.

In the bottom panels, we can see that our adaptive protocol produces more compact parameter vectors than UCCSD-VQE. For LiH (Fig. 4.13(a)) and BeH_2 (Fig. 4.13(c)), the parameter count is brought down by a factor of over 50% on average, despite the increased accuracy. The trend is reversed for the case of H_6 (Fig. 4.13(b)), where CEO-ADAPT-VQE requires a larger parameter count for most bond distances—but in the middle panels, we can see that these are all cases where UCCSD-VQE fails to produce chemically accurate results, such that its performance is not satisfactory.

One more important observation is that lowering ϵ systematically results in a lower error. Thus, as compared to the other variants of ADAPT-VQE, CEO-ADAPT-VQE maintains the desirable feature that the final accuracy can be tuned by adjusting this hyperparameter.

4.4.3 ADAPT-VQE Evolution and State of the Art

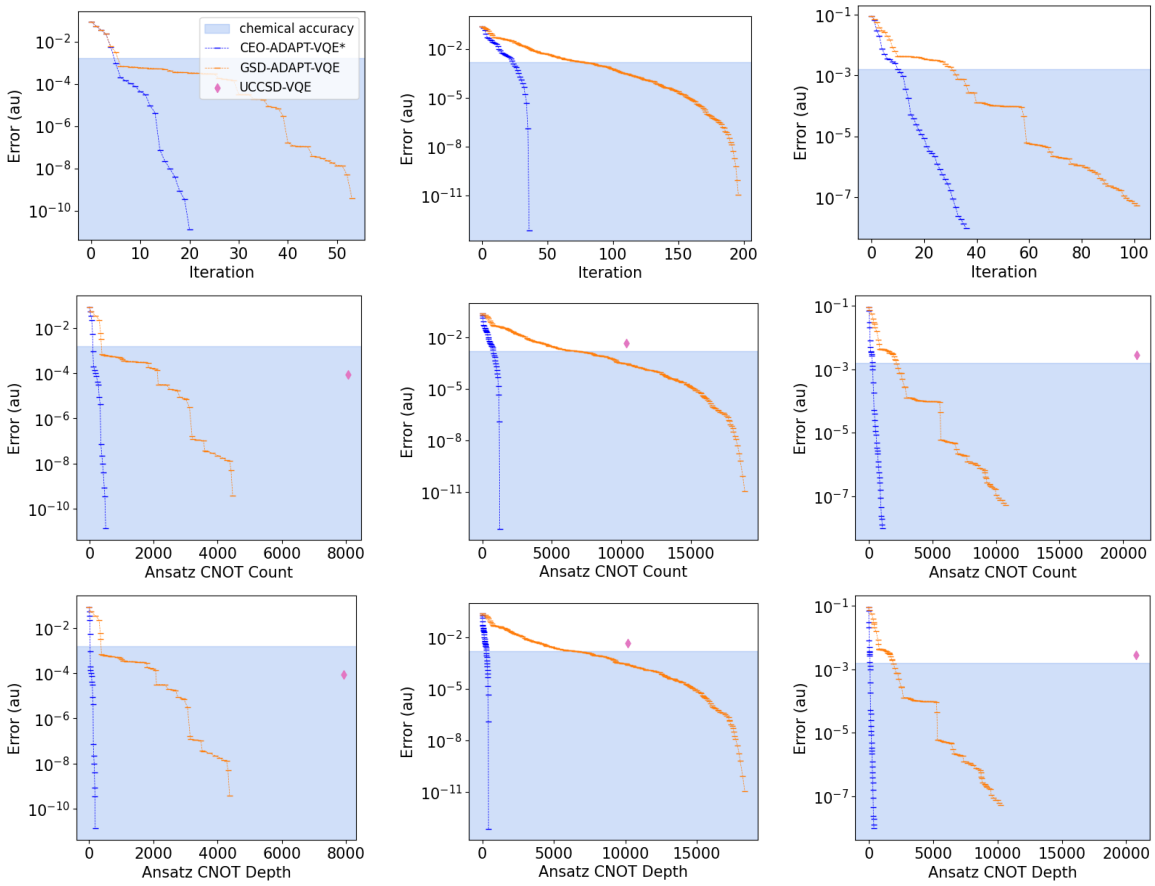
The previous sections focused on benchmarking the performance of our new CEO variant of ADAPT-VQE. In order to showcase exclusively the improvements brought forth by the CEO pool, we did not consider any other proposals of improvements to ADAPT-VQE. However, several techniques to improve the hardware efficiency and decrease the measurement costs of the algorithm are compatible with our protocol and can be readily incorporated into it. In this section, we consider an implementation of CEO-ADAPT-VQE which leverages such techniques, namely:

- TETRIS [10]
- OGM [9]
- HR

The two first techniques were discussed in the literure review (Ch. 2), while the last one was proposed in Ch. 3.

OO, as proposed in Ref. [54], is also compatible with our algorithm. However, while this technique may be advantageous when considering larger molecules and/or basis sets, its impact is negligible in the case of the minimal STO-3G basis set. Therefore, we do not consider it in this section. We refer to App. H for details about orbital optimization and its impact on CEO-ADAPT-VQE.

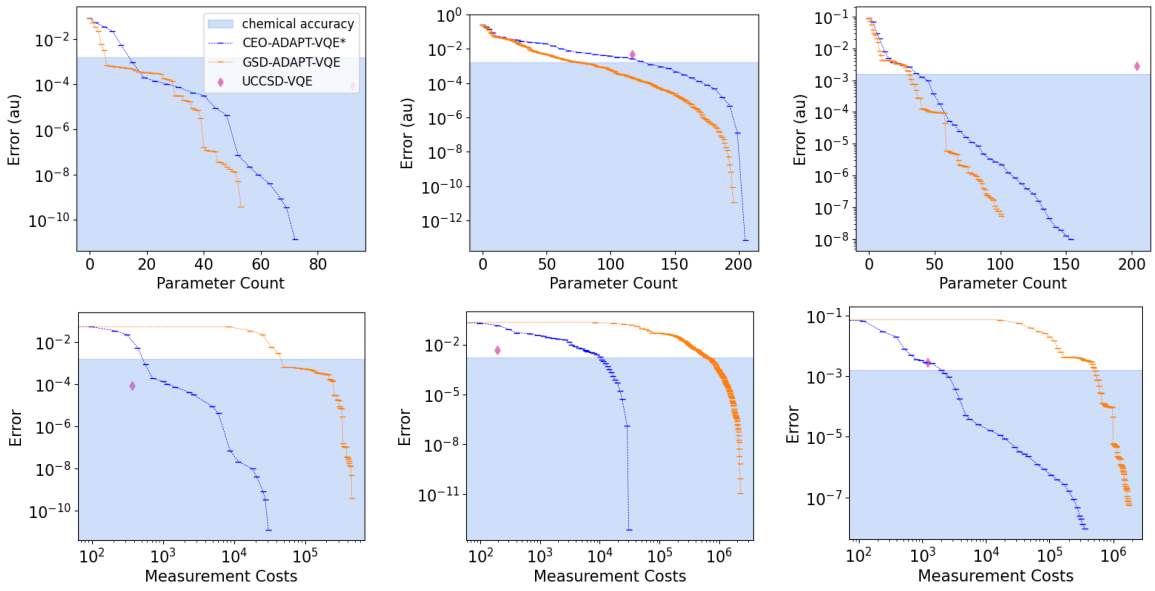
Figures 4.14 and 4.15 compare the current state of the art of ADAPT-VQE against the original protocol. For an analysis of the individual impact of each of these strategies on CEO-ADAPT-VQE, we refer to App. G. In the following, we consider an algorithm implementing all strategies simultaneously: TETRIS-CEO-ADAPT-VQE enhanced with OGM and HR (which we label CEO-ADAPT-VQE* for simplicity). This hardware-efficient and cost-frugal variant of ADAPT-VQE, reflecting the improvements of recent years, is compared against GSD-ADAPT-VQE and UCCSD-VQE (implemented with lexical ordering and a single Trotter step). In the spirit of the original algorithm of Ref. [72], the latter algorithm uses a pool of GSD excitations. In this case, we employ a vanilla approach where the gradient observables are not subject



(a) LiH at 3Å (b) Linear H_6 at 1.5Å (c) BeH2 at 2Å

Figure 4.14: Convergence of the CEO- and GSD-ADAPT-VQE algorithms for **a** LiH, **b** linear H_6, **c** BeH_2, at various bond distances. The symbol * signals that the CEO-ADAPT-VQE algorithm represents the state of the art, being implemented *in tandem* with other recent proposals: TETRIS [10], optimized gradient measurements [9] and Hessian recycling (Ch. 3). In contrast, GSD-ADAPT-VQE follows the original protocol with a vanilla measurement strategy, representing the state of the art at the time ADAPT-VQE was first proposed [72]. The UCCSD ansatz is also included as a reference point. The corresponding circuit was implemented with one Trotter step and lexical ordering (with the excitations' source/target orbital indices as primary/secondary ordering criteria, respectively). The error is plotted against the iteration number, `CNOT` count, and `CNOT` depth. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively.

to grouping, such that the cost of the gradient measurement step scales with $O(N^8)$. Further, we implement the circuits generated by GSD operators using ladders of CNOTs. In the case of doubles (which dominate costs), each circuit contains $16(n^{(df)} - 1)$ entangling gates (where $n^{(df)}$ denotes the number of qubits the operator acts on nontrivially). Note that as of today, schemes exist for measuring the GSD gradients at $O(N^6)$ cost [100] and for implementing double fermionic excitation evolutions with $2n^{(df)}+5$ CNOTs [179]. However, our implementation aims to reflect the knowledge at the time of the proposal of ADAPT-VQE.



(a) LiH at 3Å (b) Linear H_6 at 1.5Å (c) BeH_2 at 2Å

Figure 4.15: Convergence of the CEO- and GSD-ADAPT-VQE algorithms against the parameter count of the ansatz and measurement costs, which are given as multipliers for the cost of one energy measurement. The conditions are the same as in Fig. 4.14.

It was shown in Ref. [9] that the cost of measuring the constituent strings of QEs at the gradient measurement step of ADAPT-VQE is upper bounded by $8N$ times the cost of one naive energy evaluation (for the same error). This is our estimate for the cost of this step in CEO-ADAPT-VQE with OGM. By similar arguments, the cost of this step when the measurements are done naively is $4N_s$, where N_s is the number of distinct Pauli strings to be measured. This is our estimate for the cost of estimating the GSD gradients.

In what concerns the energy, a naive protocol (i.e., independently measuring each Pauli string in the Hamiltonian) would lead to an $O(N^4)$ evaluation cost. However, multiple strategies exist for grouping these strings into up to linear-sized commuting collections [63, 166, 34, 61]. It is important to note that such groupings were shown in Ref. [109] to lead to additional covariance, such that the ratio of the number of commuting groups to the total number of strings is not straightforwardly related to the decrease in measurement costs. Therefore, it is not trivial to decide which is the best strategy, or how the costs relate to those of a naive Hamiltonian measurement for a given desired error. In both cases, the answer will be state- and Hamiltonian-dependent. A complete study is outside of the scope of our work. We consider the grouping based on k-commutativity proposed in Ref. [37]. This grouping has the advantage

of allowing one to tune the depth of the measurement circuits: A lower value of k leads to shallower circuits, while a higher value of k leads to lower measurement costs. Given that the depth of the ansatz is expected to dominate over the (linear) depth of the measurement circuits [1, 62], we take the maximum value $k = n$, where n is the number of qubits (general commutativity). We include an analysis of the impact of k on the number of collections and measurement costs for the systems under study in App. I.

Finally, based on the parameter-shift rules proposed in Refs. [113, 150, 91], we consider the cost of one gradient evaluation to be twice the cost of one energy evaluation.

To estimate the measurement cost reduction achieved by grouping the Hamiltonian relative to a no-grouping scenario, we consider the metric

$$\hat{R} := \left[\frac{\sum_{i=1}^{N} \sum_{j=1}^{m_i} |c_{ij}|}{\sum_{i=1}^{N} \sqrt{\sum_{j=1}^{m_i} |c_{ij}|^2}} \right]^2 \tag{4.25}$$

proposed in [35]. This metric gauges the savings achieved by grouping a Hamiltonian into N collections, where collection i has m_i Pauli strings. c_{ij} designates the coefficient of the jth string in the ith set. $\hat{R}$ approximates the expected value of the measurement cost ratio M_u/M_g, where M_u and M_g represent the shot count required to achieve a given accuracy with and without grouping, respectively, over the uniform spherical measure. This metric assumes that the distribution of measurements among collections is chosen so as to minimize the sampling error.

The uppermost panels in Fig. 4.14 reveal a significantly faster convergence of CEO-ADAPT-VQE* compared to GSD-ADAPT-VQE when we consider the number of iterations. While the iteration count is not a straightforward indicator of relevant costs, a lower number implies fewer measurement rounds and optimizations in total. The error against the CNOT count, plotted in the second row of panels, is particularly interesting to benchmark NISQ algorithms. We observe a remarkable reduction in the CNOT count required to reach a given error for all molecules. This reduction is the greatest for the most strongly correlated molecule (H_6). In this case, the improvement of CEO-ADAPT-VQE* with respect to GSD-ADAPT-VQE is even more significant than the improvement of the latter with respect to UCCSD-VQE. The third row of panels shows the error against the CNOT depth, which, given the short coherence times of NISQ devices, is often viewed as the go-to figure of merit to assess near-term viability. Owing to the TETRIS compactification, the reduction in CNOT depth achieved by CEO-ADAPT-VQE* is even greater than the reduction in CNOT count, with respect to the original algorithm.

Figure 4.15 compares the parameter count (top) and measurement costs (bottom) of CEO-ADAPT-VQE* and GSD-ADAPT-VQE. We note that the former favors circuit-efficiency over parameter-efficiency; namely, MVP-CEOs and the TETRIS protocol are aimed at improving the energy reduction achieved by low-depth circuits without concern for the number of variational parameters. Yet, the number of additional parameters used in CEO-ADAPT-VQE* is very modest. Further, while a larger parameter vector is expected to cause the optimization to require more function and gradient evaluations due to being higher dimensional, the total measurement costs of CEO-ADAPT-VQE* are *reduced* with respect to GSD-ADAPT-VQE. We attribute this to the use of the OGM and HR strategies, which significantly alleviate the overhead

from the gradient measurement rounds and from the successive optimizations, respectively.

One more remarkable result we can observe in Fig. 4.15 is that for the same error, the total measurement requirements of UCCSD-VQE are only significantly lower than those of CEO-ADAPT-VQE* for H_6, which we attribute to the phenomenon of gradient troughs known to plague this system [71]. In this case, we observe a 29-fold increase in measurement costs (for the same accuracy), which nevertheless seems modest considering the widespread expectation of a prohibitive overhead and the fact that UCCSD-VQE is not a viable option for this system (given that it is not able to reach chemical accuracy). For the other systems, the measurement costs of CEO-ADAPT-VQE* are within an order of magnitude of those of UCCSD-VQE: They are increased only 4-fold for LiH, and are roughly matched for BeH_2. We note that the latter is the largest molecule, suggesting that CEO-ADAPT-VQE* may reduce measurement costs as we increase system size and approach classically intractable molecules, for which UCCSD has even more unnecessary parameters. This is a surprising result, considering that the measurement cost overhead incurred by the adaptive ansatz construction has been pointed out as a possible shortcoming of ADAPT-VQE, and a potential barrier to its practical viability. The reason for this expectation is two-fold: (i) the asymptotic cost of measuring the pool gradients required to adaptively build the ansatz is higher than the cost of measuring the energy, and (ii) ADAPT-VQE performs several optimizations per run, while static ansätze perform a single one. However, in practice, we observe that the total number of energy evaluations required by ADAPT-VQE throughout all optimizations until the UCCSD error is reached is actually *lower* than the total required by the UCCSD-VQE optimization. This can be explained by the fact that growing the ansatz from scratch leads to earlier optimizations being lower-dimensional, which in turn implies faster convergence and cheaper gradient evaluations. Additionally, it allows us to warm-start the parameters and inverse Hessian in each optimization. We can even view the sequence of ADAPT-VQE optimizations as a single adaptive optimization; such an optimization is particularly resource-efficient, because parameters are only included if or when they are actually relevant. Finally, we note that due to the lower parameter count and warm-starting of ADAPT-VQE optimizations, sampling noise is likely to be less detrimental for this algorithm than for UCCSD-VQE, which will further benefit the measurement costs of the former in a practical setting. This was demonstrated in Ref. [141], where even the simplest implementation of the qubit-ADAPT-VQE algorithm (without TETRIS, OGM or HR) required only 5% of the total shot count of UCCSD-VQE to reach chemical accuracy in the majority of the runs, when determining the ground state energy of molecular hydrogen (H_2) in the presence of finite sampling noise.

We remark that the improvements applied to CEO-ADAPT-VQE in this section could also be applied to QEB- and Qubit-ADAPT-VQE to obtain similar relative improvements in the `CNOT` depth and measurement costs. The relative performance of the three algorithms would remain similar to what we observed in Sec. 4.4.1, where none of the algorithms include the improvements.

4.4.4 Comparison with Fixed-Structure Ansätze

In this section, we compare our adaptive algorithm (CEO-ADAPT-VQE*) against five leading non-adaptive ansätze for the electronic structure problem that we introduced in Ch. 2, namely:

- tUPS [27]
- OO-tUPS [27]
- PP-tUPS [27]
- QNP gate fabric [11]
- k-UpCCGSD[95]

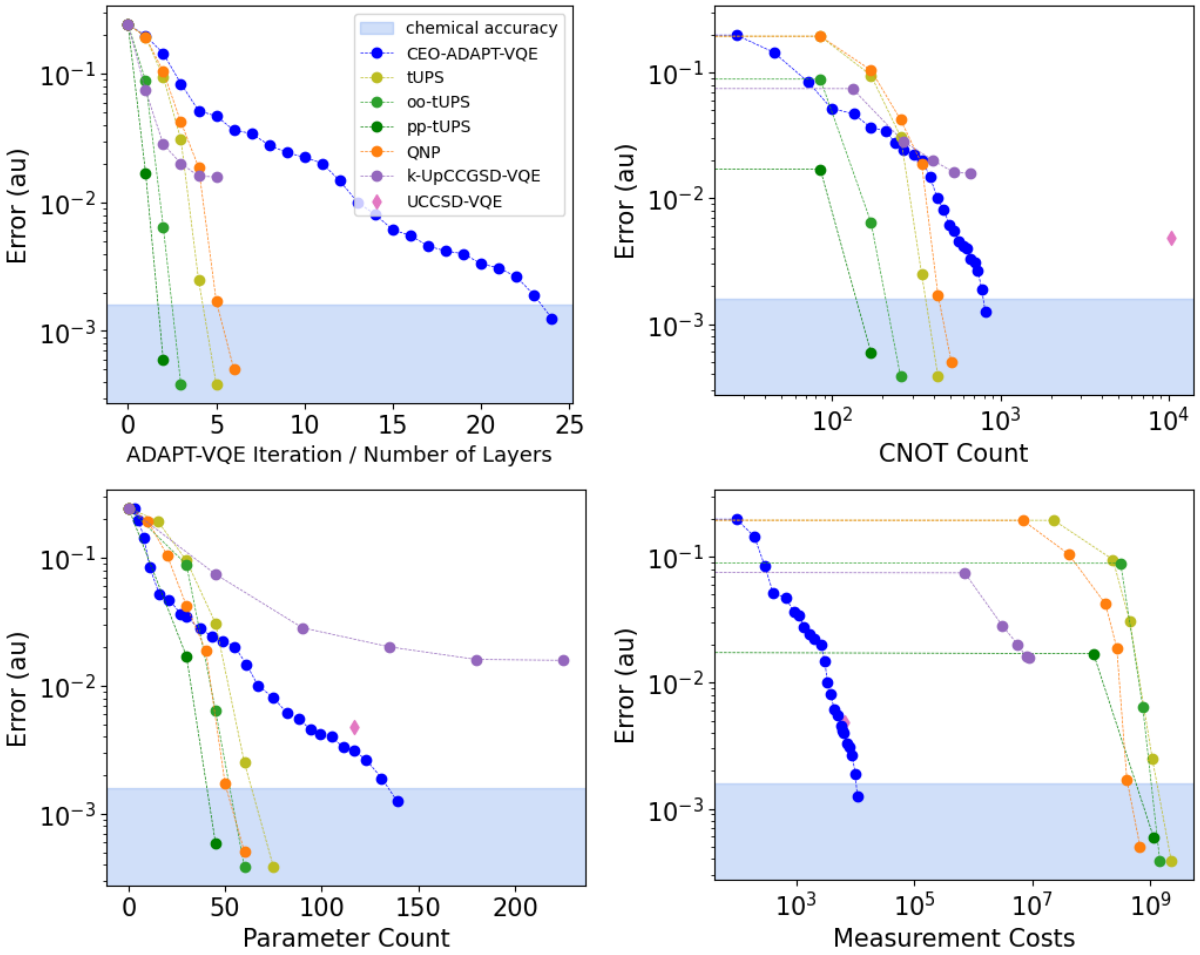


Figure 4.16: Convergence of CEO-ADAPT-VQE for linear H_6 at interatomic distance 1.5 Å, as compared to a variety of state-of-the-art fixed-structure ansätze: tUPS, OO-tUPS, PP-tUPS, QNP, and k-UpCCGSD-VQE. UCCSD-VQE (implemented with one Trotter step and lexical ordering) is also included for reference. The error is plotted against the number of layers (or iterations, in the case of ADAPT-VQE), CNOT count, parameter count, and measurement costs (given as multipliers for the cost of one energy measurement). The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). All algorithms terminate when chemical accuracy is reached for the first time, except for k-UpCCGSD, the error of which stagnates before reaching this threshold.

The optimization for the first four ansätze is done using a parallel-tempering basin-hopping approach [27]. In this method, local optimizations are followed by random perturbations of the parameters, such

that multiple regions of the optimization landscape are searched for a good minimum. This optimization technique requires a significantly larger number of function evaluations than local optimization methods, and is aimed at local-trap-riddled cost landscapes. We note that the original QNP proposal in Ref. [11] suggested two fixed-parameter initialization strategies followed by local optimization; however, in our simulations, the QNP gate fabric did not yield satisfactory accuracy when these strategies were applied. In the case of the k-UpCCGSD ansatz, as suggested in the original work of Ref. [95], we employed 30 optimizations with parameters randomly initialized between -π and π. For the CNOT count, we assumed this ansatz to be implemented using the fermionic swap networks proposed in Ref. [129].

The error from the basin-hopping optimizations is as provided in Ref. [27]. The function evaluations were taken as the average over 10 optimizations, multiplied by the 8000 optimizations employed in the basin-hopping approach. This is due to the fact that Ref. [27] did not make available the number of function/gradient evaluations, and the runtime of our Python implementation of the ansatz does not allow us to run thousands of optimizations in a reasonable time. As before, measurement costs include both energy and gradient evaluations, and all these evaluations are noiseless. We use the same optimizer (BFGS) and gradient convergence threshold (10^{-8}) in all cases.

Figure 4.16 compares the cost of the algorithms for linear H_6 at interatomic distance 1.5 Å in terms of CNOTs, parameters, and measurement costs until chemical accuracy is reached. We observe that all algorithms succeed in reaching this threshold except for k-UpCCGSD, whose error stagnates after a few layers. While this could likely be improved by increasing the number of optimizations, the number of parameters and CNOTs suggest that this ansatz is not a competitive option, as they are higher than those of the other ansätze for a given error. The different variants of the tUPS ansatz and the QNP gate fabric succeed in reaching chemical accuracy, and do so with a lower CNOT count than CEO-ADAPT-VQE*. However, while the improvement in the CNOT count is of less than an order of magnitude, the measurement costs are increased by roughly 5 orders of magnitude. This is due to the difficulty of optimizing the rough parameter landscapes created by these tiled circuit structures. While ADAPT-VQE succeeds using parameters initialized at zero and local optimizations, these fixed-structure ansätze do not produce good results under such conditions. Hence, finding a good minimum requires performing many local optimizations, which increases the measurement costs to an impractical extent. In addition to the high number of energy evaluations, no heuristics have been developed to determine the optimal number of optimizations required to achieve good accuracy, and it is unknown how the number scales with system size. Therefore, despite the low CNOT counts, we do not expect these ansätze to constitute a viable option for realistic implementations.

4.5 Discussion

In this chapter, we proposed a new variant of ADAPT-VQE based on a pool of coupled exchange operators (CEOs) and showed that its performance is superior to any previously proposed variants of the algorithm.

We further combined this protocol with other techniques aimed at decreasing the circuit depth and measurement costs of adaptively built ansätze. By uniting the CEO pool with optimized gradient measurement strategies [9], Hessian recycling (Ch. 3) and the TETRIS protocol [10], we showed a total reduction of up to 88%, 96% and 99.6% in the CNOT count, CNOT depth and measurement costs relative to the original ADAPT-VQE algorithm, for 12- to 14-qubit molecules. We additionally observed that in spite of the common belief that the adaptive ansatz construction incurs a significant measurement overhead, the total measurement costs of CEO-ADAPT-VQE* are actually comparable to those of UCCSD-VQE, a widely used static ansatz for the same problem. Further, we found numerical evidence of a decrease in the measurement costs of the former relative to the latter for larger molecules (Fig. 4.15), suggesting that our algorithm may actually offer a shot count reduction for classically intractable systems as compared to non-adaptive strategies. This conjecture was strengthened by a comparison with five leading fixed-structure ansätze, which were shown to either be unable to reach chemical accuracy or do so at the expense of a dramatic overhead in function evaluations as compared to our algorithm.

The new class of operators we introduced (CEOs) consists of linear combinations of qubit excitations (QEs) in which the QEs can be independently parameterized (MVP-CEOs) or share a single variational parameter (OVP-CEOs). We leveraged the structure of CEOs to implement the unitaries generated by them with similar or shallower circuits than those generated by QEs. We showed that such unitaries preserve particle number and spin symmetries, and that they can be implemented by circuits whose CNOT count is the same or lower as compared to any other circuits known to have these desirable symmetry preservation properties. Explicit CNOT-efficient circuit constructions were provided for all possible types of CEOs. We constructed circuits for MVP-CEO evolutions by optimizing a sequence of circuit implementations of exponentials of individual (commuting) Pauli strings, with a resulting CNOT count of 13—the same that is required to implement the evolution of a single one of its constituent QEs (which can be as many as three). In the case of OVP-CEO evolutions, we leveraged their structure as multi-controlled rotations to achieve an even lower CNOT count of 9.

The algorithm we propose, CEO-ADAPT-VQE*, is a variant of the quantum simulation algorithm ADAPT-VQE that makes use of the CEO operators to build a symmetry-adapted ansatz. We compared this algorithm with QEB- and Qubit-ADAPT-VQE, which to the best of our knowledge were the previous most hardware-efficient variants of ADAPT-VQE. Numerical simulations for various molecules show that CEO-ADAPT-VQE* significantly and systematically reduces the CNOT count of the ansatz as compared to these other variants. We found reductions by up to 65%, with greater reductions for more strongly correlated molecules, which are also the most interesting systems—stronger correlations are harder to capture classically, thus making such systems possible candidates for quantum advantage experiments. We further observed that the reduction in the CNOT count increased with the number of algorithmic iterations, suggesting that larger molecules will favor CEO-ADAPT-VQE even more than our small, classically simulatable test cases.

The improved hardware-efficiency of the proposed adaptive VQE comes at no added cost. In fact, our algorithm requires *fewer* iterations for the same accuracy than QEB- or Qubit-ADAPT-VQE. As such, it implies fewer optimizations and fewer gradient measurement rounds, effectively *lowering* the measurement

costs. What is more, the number of variational parameters it requires is either maintained or decreased with respect to other variants.

Considering all the advantages it offers, we expect CEO-ADAPT-VQE* to be a leading candidate for molecular simulations on near-term quantum computers. In combination with other recent improvements to adaptive ansätze proposed in the literature, this algorithm is able to achieve a remarkable reduction in CNOT count/depth and measurement costs as compared to ADAPT-VQE at the time of its inception, showing great progress in the path towards quantum advantage with NISQ devices.

5

Co-ADAPT-VQE: Co-Designing State Preparation Circuits

This chapter is associated with the following authored publication:

Co-Designed Adaptive Quantum State Preparation Protocols. **M. Ramôa**, L. P. Santos, N. J. Mayhall, E. Barnes, S. E. Economou. arXiv:2601.20681

5.1 Motivation

As discussed in Ch. 2, several types of ansätze have been proposed in the literature, which can be broadly classified as *hardware-tailored* or *problem-tailored*. The former are chosen to include only gates and interactions that are natively available in the target device, while the latter are based on knowledge about the problem at hand; they may, e.g., preserve system symmetries or draw inspiration from previous classical methods. An example in the former category is the HEA [88], which offers the alluring feature of producing circuits that are ready to be executed on hardware, such that no transpilation overheads are incurred. However, this ansatz suffers from BPs, which prevents it from scaling efficiently with the system size [110]. In the second category, we have the ADAPT-VQE algorithm [72], which is not only problem- but also system-tailored. However, in contrast with hardware-efficient ansätze, ADAPT-VQE, in its original formulation, is completely agnostic to the target hardware, which results in high transpilation overheads.

In this chapter, we propose to combine the benefits of hardware- and problem-tailored ansätze by co-designing ADAPT-VQE such that it takes into account architecture restrictions and device limitations. Our proposed algorithm, Co-ADAPT-VQE, modifies the selection criterion to penalize circuit components that are less suitable for the underlying quantum computer. This customizable penalty may be based on generic features, such as the depth or two-qubit gate count of the transpiled circuit, or on device-specific features, such as gate errors. This allows the creation of more compact and noise-resilient circuits using a protocol that is simultaneously hardware-, problem-, and system-tailored. Our algorithm maintains the optimization advantages of ADAPT-VQE while augmenting it with all the benefits stemming from hardware efficiency.

	CNOT Count for Chemical Accuracy		
	ATA Connectivity	LNN Connectivity - Previous	LNN Connectivity - This Work
Linear H_6	854	2945	348 (12%)
Linear H_7	437	1497	92 (6%)
Triangular H_6	1533	5323	184 (3%)

Table 5.1: Comparison of the CNOT counts required by CEO-ADAPT-VQE (see Ch. 4) to reach chemical accuracy in ATA and LNN connectivity. For the latter, we consider the previous state-of-the-art (obtained from the ATA circuit using the Qiskit (version 2.2.1) transpiler set to the maximum optimization level [137]) as well as an instance of Co-ADAPT-VQE specifically tailored to reduce the CNOT count of the final circuits. The percentages in parentheses represent the ratio of the CNOT count of our method to the CNOT count of the previous one. Three different systems are considered: linear H_6 (12 qubits), linear H_7 (14 qubits), and triangular H_6 (12 qubits).

We exemplify the potential of Co-ADAPT-VQE with a standard task in quantum simulation: finding molecular ground states using state preparation circuits. We consider a restriction to LNN connectivity and aim to minimize the number of two-qubit gates. Table 5.1 shows that the two-qubit gate counts suffer an over 3-fold increase under transpilation from ATA to LNN connectivity using state-of-the-art methods; yet, Co-ADAPT-VQE is able to reduce the LNN counts by up to 97%, resulting in an *improvement* over the ATA connectivity circuit, despite the additional interaction restrictions.

The rest of this chapter is organized as follows. In Sec. 5.2, we introduce the operators and circuits that are relevant for this chapter. Section 5.3 introduces Co-ADAPT-VQE. Section 5.4 features an example proposal where it is used to minimize the LNN CNOT count of the leading variants of ADAPT-VQE. The results of this proposal are presented in Sec. 5.5. Finally, Sec. 5.6 concludes the chapter with a discussion of the results.

5.2 Excitation Circuits in ATA and LNN Connectivities

As discussed in previous chapters, the operator pool is an important component of ADAPT-VQE, since it has an impact on the final circuit. The first proposal was to use a pool of fermionic excitation operators. An example single fermionic excitation is

$$T^{(FE)}_{\alpha_1\to\alpha_2} = a^\dagger_{\alpha_2}a_{\alpha_1} - a^\dagger_{\alpha_1}a_{\alpha_2} \to \frac{i}{2}\bigotimes_{z_k\in\mathcal{Z}_{JW}} Z_{z_k}(+X_{\alpha_2}Y_{\alpha_1} - Y_{\alpha_2}X_{\alpha_1}), \tag{5.1}$$

while an example fermionic double excitation is

$$\begin{aligned} T^{(FE)}_{\alpha_1\beta_1\to\alpha_2\beta_2} =& a^\dagger_{\alpha_2}a^\dagger_{\beta_2}a_{\alpha_1}a_{\beta_1} - a^\dagger_{\beta_1}a^\dagger_{\alpha_1}a_{\beta_2}a_{\alpha_2} \\ \to \frac{i}{8}\bigotimes_{z_k\in\mathcal{Z}_{JW}} Z_{z_k}(&+X_{\alpha_2}X_{\alpha_1}X_{\beta_2}Y_{\beta_1} - X_{\alpha_2}X_{\alpha_1}Y_{\beta_2}X_{\beta_1} + X_{\alpha_2}Y_{\alpha_1}X_{\beta_2}X_{\beta_1} + X_{\alpha_2}Y_{\alpha_1}Y_{\beta_2}Y_{\beta_1} \\ &- Y_{\alpha_2}X_{\alpha_1}X_{\beta_2}X_{\beta_1} - Y_{\alpha_2}X_{\alpha_1}Y_{\beta_2}Y_{\beta_1} + Y_{\alpha_2}Y_{\alpha_1}X_{\beta_2}Y_{\beta_1} - Y_{\alpha_2}Y_{\alpha_1}Y_{\beta_2}X_{\beta_1}), \end{aligned} \tag{5.2}$$

where $\mathcal{Z}_{JW}$ is the set of indices of qubits involved in the Jordan-Wigner anticommutation string arising from the Pauli Z operators in Eq. (2.6), which depends on the qubit ordering chosen for the transform. Taking $i, j \leftarrow \texttt{sorted}(\alpha_1, \alpha_2)$ and $p, q, r, s \leftarrow \texttt{sorted}(\alpha_1, \alpha_2, \beta_1, \beta_2)$, we have $\mathcal{Z}_{JW} = \{x \in \mathbb{N} : i < x < j\}$ and $\mathcal{Z}_{JW} = \{x \in \mathbb{N} : p < x < q \vee r < x < s\}$ for the single and double excitations, respectively. We call the set of fermionic excitations of the form of Eqs. (5.1) and (5.2) without index restrictions the GSD pool.

The chain of Pauli Z operators that arises when we apply the Jordan-Wigner transform to a fermionic excitation acts on a number of qubits that grows linearly on average with the size of the system. A consequence of this is that the unitaries generated by operators such as the ones in Eqs. (5.1), (5.2) require a linear number of entangling gates to be implemented. As discussed in Ch. 2, QEs were proposed to improve the near-term viability of ADAPT-VQE by removing the anticommutation string [182]. The operators in this pool are formulated in terms of the qubit ladder operators:

$$\begin{aligned} Q_i^\dagger &= \frac{1}{2}(X_i - iY_i), \\ Q_i &= \frac{1}{2}(X_i + iY_i), \end{aligned} \tag{5.3}$$

which are equivalent to the Jordan-Wigner-transformed fermionic ladder operators (Eq. (2.6)) up to the anticommutation string. With these operators, we can define a qubit single excitation as

$$T^{(QE)}_{\alpha_1 \to \alpha_2} = Q^\dagger_{\alpha_2} Q_{\alpha_1} - Q^\dagger_{\alpha_1} Q_{\alpha_2} = \frac{i}{2}(+X_{\alpha_2} Y_{\alpha_1} - Y_{\alpha_2} X_{\alpha_1}), \tag{5.4}$$

and a qubit double excitation as

$$\begin{aligned} T^{(QE)}_{\alpha_1\beta_1 \to \alpha_2\beta_2} =& Q^\dagger_{\alpha_2} Q^\dagger_{\beta_2} Q_{\alpha_1} Q_{\beta_1} - Q^\dagger_{\beta_1} Q^\dagger_{\alpha_1} Q_{\beta_2} Q_{\alpha_2} \\ =& \frac{i}{8}(+ X_{\alpha_2} X_{\alpha_1} X_{\beta_2} Y_{\beta_1} - X_{\alpha_2} X_{\alpha_1} Y_{\beta_2} X_{\beta_1} + X_{\alpha_2} Y_{\alpha_1} X_{\beta_2} X_{\beta_1} + X_{\alpha_2} Y_{\alpha_1} Y_{\beta_2} Y_{\beta_1} \\ & - Y_{\alpha_2} X_{\alpha_1} X_{\beta_2} X_{\beta_1} - Y_{\alpha_2} X_{\alpha_1} Y_{\beta_2} Y_{\beta_1} + Y_{\alpha_2} Y_{\alpha_1} X_{\beta_2} Y_{\beta_1} - Y_{\alpha_2} Y_{\alpha_1} Y_{\beta_2} X_{\beta_1}). \end{aligned} \tag{5.5}$$

Note that the operators in Eqs. (5.4) and (5.5) are equivalent to those in Eqs. 5.2 and 5.4 with the Pauli Z operators removed.

The QEB-ADAPT-VQE protocol, which uses these operators instead of their fermionic versions, has been shown to lead to a reduction of the CNOT count with respect to the original fermionic pool [182]. In Ch. 4, we proposed a variant of ADAPT-VQE based on CEOs that further decreases CNOT counts.

The unitaries generated by fermionic and qubit excitations allow for more efficient circuit implementations than unitaries generated by general linear combinations of Pauli strings [179]. Throughout this chapter, we will use U and T to denote unitaries and the corresponding generators, respectively, such that, e.g., $U^{(QE)}_{\alpha_1 \to \alpha_2}(\theta) = e^{\theta T^{(QE)}_{\alpha_1 \to \alpha_2}}$.

The most efficient circuit implementation for the qubit single excitation defined in Eq. (5.4) is shown in Fig. 5.1. This circuit implements a Y rotation if the states of the qubits representing the spin-orbitals involved in the excitation have odd parity, corresponding to an occupied/unoccupied spin-orbital pair where

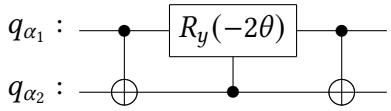


Figure 5.1: Circuit implementation of the unitary $U^{(QE)}_{\alpha_1\to\alpha_2}(\theta)$, generated by qubit excitation $T^{(QE)}_{\alpha_1\to\alpha_2}$. The same circuit can be used for $U^{(QE)}_{\beta_1\to\beta_2}$. The circuit for $U^{(QE)}_{\alpha_2\to\alpha_1}$ (or $U^{(QE)}_{\beta_2\to\beta_1}$) is identical, but with the rotation angle flipped.

an excitation is allowed to occur. The controlled rotation gate can be implemented using two CNOTs and single-qubit gates; however, it is possible to optimize the final circuit such that it requires a total of only two CNOTs .

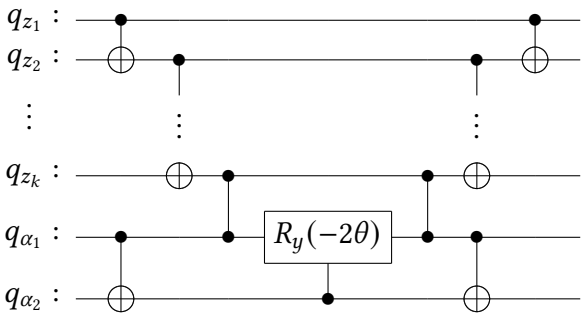


Figure 5.2: Circuit implementation of the unitary $U^{(FE)}_{\alpha_1\to\alpha_2}(\theta)$, generated by qubit excitation $T^{(FE)}_{\alpha_1\to\alpha_2}$. The same circuit can be used for $U^{(FE)}_{\beta_1\to\beta_2}$. The circuit for $U^{(FE)}_{\alpha_2\to\alpha_1}$ (or $U^{(FE)}_{\beta_2\to\beta_1}$) is identical, but with the rotation angle flipped. The vertical dots indicate that there is a ladder of CNOT gates between qubits q_{z_1} to $q_{|\mathcal{Z}_{JW}|}$.

In contrast, the circuit required to implement the single fermionic excitation in Eq. (5.1) is depicted in Fig. 5.2. As compared to the qubit single excitation circuit, we require additional CNOT gates to calculate the parity of the set $\mathcal{Z}_{JW}$ of qubits involved in the Jordan-Wigner anticommutation string, as discussed in the previous section. The number of CNOTs required to implement this circuit is $O(N)$, where N is the total number of qubits. The specific CNOT count depends on how many qubits appear between the qubits q_{α_1} and q_{α_2} in the ordering considered for the Jordan-Wigner transform.

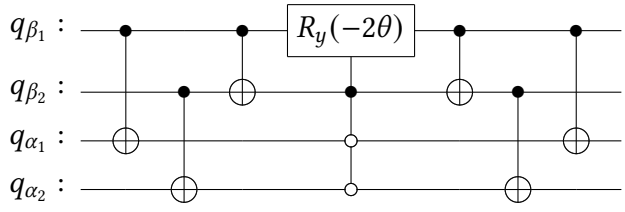


Figure 5.3: Circuit implementation of the unitary $U^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}(\theta)$, generated by qubit excitation $T^{(QE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U^{(QE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped.

While single excitations can be implemented using one-control Y rotations, double excitations require three-control Y rotations. The circuit implementation of the qubit double excitation in Eq. (5.5) is shown in Fig. 5.3. CNOTs are used to calculate the parities of the states of subsets (pairs) of qubits, such that the controlled rotation only occurs if spin-orbitals α_1, β_1 are occupied and α_2, β_2 are unoccupied, or vice-versa.

Once the multi-control rotation is decomposed into basic gates and circuit optimizations are applied, the circuit includes a total of 13 CNOTs.

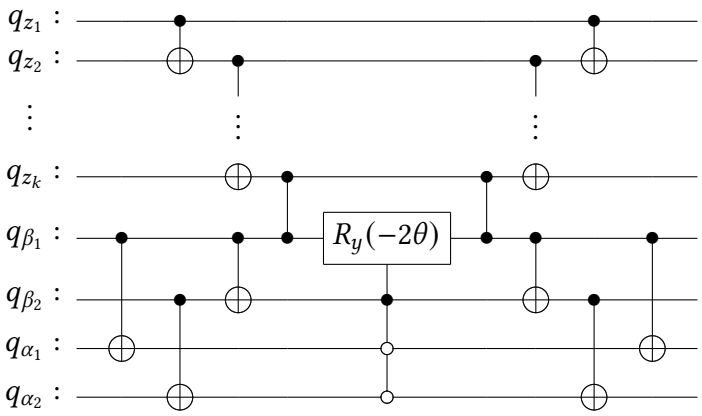


Figure 5.4: Circuit implementation of the unitary $U^{(FE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}(\theta)$, generated by qubit excitation $T^{(FE)}_{\alpha_1\beta_1\to\alpha_2\beta_2}$. The circuit for $U^{(FE)}_{\alpha_2\beta_2\to\alpha_1\beta_1}$ is identical, but with the rotation angle flipped. The vertical dots indicate that there is a ladder of CNOT gates between qubits q_{z_1} to $q_{|\mathcal{Z}_{JW}|}$.

Similar to the case of single excitations, the fermionic double excitation in Eq. (5.2) can be implemented using the qubit double excitation circuit with additional CNOTs to calculate the parity of the states of qubits involved in the Jordan-Wigner anticommutation string. The resulting circuit is depicted in Fig. 5.4. Once again, we have a $\mathcal{O}(N)$ circuit count, with the specific number depending on the spin-orbital ordering.

The operators we proposed in Ch. 4, CEOs, consist of linear combinations of QEs that allow for particularly efficient circuit implementations. They come in two flavors: OVP-CEOs, which consist of the sum or the difference of two double qubit excitations sharing a variational parameter, and MVP-CEOs, which consist of linear combinations of up to three double qubit excitations with independent variational parameters. We showed that the former require only 9 CNOTs to be implemented, offering a 30% decrease in two-qubit gate count as compared to a unitary generated by one of the constituent qubit excitations, while the latter require 13, which translates to a 67% improvement with respect to the 39 CNOTs required to implement the three unitaries generated by three generic double qubit excitations.

5.3 Co-ADAPT-VQE

The canonical ADAPT-VQE protocol (Fig. 2.2) appends to the ansatz the operator with the highest gradient in each iteration. This is an attempt to maximize the impact on the energy per operator. However, such a selection criterion is architecture-agnostic; it does not take into account important factors such as how hard it is to implement each operator in the target hardware, or how affected by noise each operator is. Operators that require interactions between distant physical qubits (in the target hardware's connectivity graph), or that act on qubits with higher noise rates and lower coherence times, may be poor choices even if the gradients are high; in some cases, the impact on the energy may not compensate the cost of

implementing this operator in terms of gate count, additional impact of noise, increase in circuit depth, or other factors.

As such, we propose to modify the ADAPT-VQE selection criterion to include information about the quantum computer on which the algorithm is meant to be executed. Specifically, we propose to scale the gradients as

$$\frac{\left.\frac{\partial E^{(n)}}{\partial \theta_i}\right|_{\theta_i=0}}{p} = \frac{\left\langle \psi^{(n-1)} \right| \left[\hat{H}, \hat{A}_k\right] \left| \psi^{(n-1)} \right\rangle}{p}, \tag{5.6}$$

where p is a hardware-dependent penalty. This penalty may be related to any relevant metric for the cost of implementing the operator in the target quantum computer, e.g., gate counts after transpilation into the target gate set and connectivity graph, average coherence times of the qubits involved in the operation, or summed average errors for all two-qubit interactions, etc. In these examples, the penalty for each operator may be calculated before the start of the protocol and stored for use in each ADAPT-VQE iteration; however, the penalty may also be recalculated dynamically as the algorithm evolves. An example is if we wish to minimize circuit depth; in this case, we may recalculate the penalty in each iteration such that it corresponds to the additional circuit depth we get when appending each operator to the current ansatz circuit. Note, however, that while polynomial, the classical overhead stemming from this dynamic calculation may be significant if we perform circuit optimization after the attempted addition of each candidate operator, since the total number of circuit optimizations per iteration will then be equal to the number of operators in the pool, which scales as $O(N^4)$ for the best performing pools, where N is the number of orbitals (or qubits) (see Ch. 2).

5.4 Using Co-ADAPT-VQE to Minimize Entangling Gate Counts for LNN Connectivity

We will exemplify the impact of our co-designed variant of ADAPT-VQE (Co-ADAPT-VQE) by using it to minimize the entangling gate count of the state preparation circuits run on hardware with LNN connectivity. This task has broad relevance: Since entangling gates are associated with the highest error rates, their number is a good indicator of how a circuit will be impacted by hardware noise [38]. Further, linear connectivity is a common target: for example, the connectivity graph for superconducting architectures is typically close to a linear graph, with the average number of connections per qubit (degree of the connectivity graph) very close to 2 [2]. Further, while neutral atom and trapped-ion architectures offer ATA connectivity, in practice these interactions rely on the spatial rearrangement of qubits, such that significant overheads are incurred in shuttling/transport operations [42, 135, 83, 41].

In Sec. 5.2, ATA connectivity was assumed for all circuits. This has been the standard for benchmarking ADAPT-VQE variants [160, 182]. Transpilation to LNN connectivity and circuit optimization can be straightforwardly applied to any ansatz. Fig. 5.5 shows that the relative performance of the pools remains

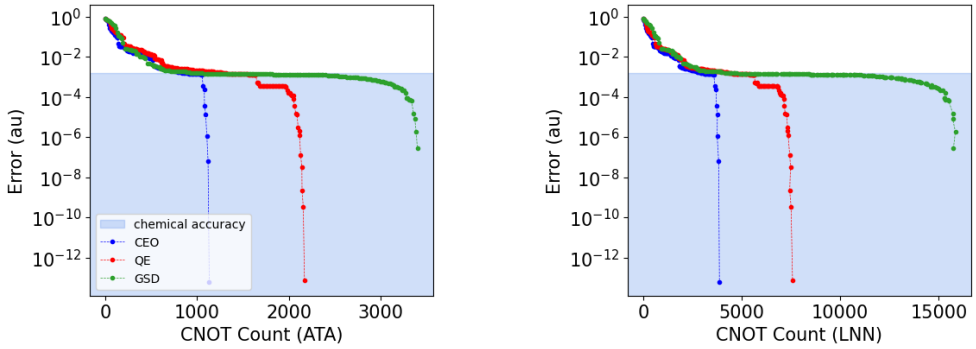


Figure 5.5: Error *versus* CNOT count for the best performing variants of ADAPT-VQE: CEO-, QE- and Qubit-ADAPT-VQE. The molecule considered is H_6 at interatomic distance 3 Å in the STO-3G basis set. The blue shading represents the chemical accuracy region (error below 1 kcal/mol).

the same upon transpilation of the original ATA circuits, with CEO-ADAPT-VQE (see Ch. 4) resulting in the lowest two-qubit gate count, followed by QE-ADAPT-VQE [182] and finally Qubit-ADAPT-VQE [160].

While the relative merits of the algorithmic variants remain similar upon transpilation from ATA to LNN connectivity, Fig. 5.5 tells us that transpilation incurs a roughly 4-fold increase in the CNOT count—an increase which is expected to aggravate with the system size. We will showcase the impact of Co-ADAPT-VQE by using it to mitigate this overhead. We will focus on transpilation to a gate set consisting of single-qubit rotations and CNOTs for LNN connectivity; however, the same techniques may be readily applied to other hardware specifications.

Typical routing protocols are based on networks of SWAP gates, which are two-qubit gates with matrix representation

$$\mathrm{SWAP} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

A SWAP gate acting on qubits i, j exchanges $|0\rangle_i |1\rangle_j \leftrightarrow |1\rangle_i |0\rangle_j$, allowing us to modify the qubit ordering by exchanging the states of qubit pairs. As a specific example, let us consider the qubit single-excitation circuit from Fig. (5.4). If qubits q_{α_1} and q_{α_2} are not adjacent in the connectivity graph, then we cannot apply the CNOT gates directly. However, we can route the qubits together using a ladder of SWAP gates, as depicted in Fig. 5.6. After we apply the desired operations, the adjoint SWAP ladder is applied to restore the original ordering such that the unitaries exactly match.

Note that SWAP gates are not part of the gate set we are considering; however, they can be implemented using CNOT gates as shown in Fig. 5.7.

Qubit double excitations can be transpiled similarly; particularly, for a set of sorted physical qubits $[p, q, r, s]$, we take the central point $k = \mathtt{round}(\frac{p+q+r+s}{2})$ (where $\mathtt{round}(i)$ returns the nearest integer to i) and route them to $[k-1, k, k+1, k+2]$ or $[k-2, k-1, k, k+1]$ if the fraction was rounded up or down, respectively. CNOT gates within the 4-qubit circuit can be straightforwardly transpiled using

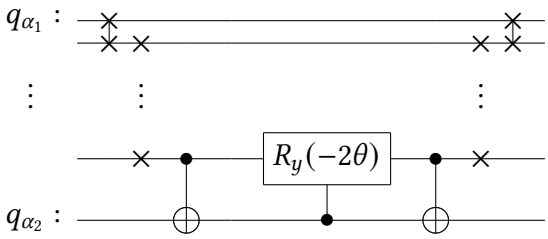


Figure 5.6: Implementation of the qubit single excitation circuit in Fig. 5.1 for LNN connectivity using swap-based routing. Unmarked qubits are irrelevant to the excitation, but they are part of the path from the original to the final physical qubit representing qubit α_1.

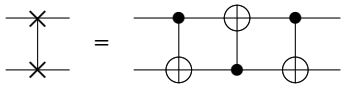


Figure 5.7: Decomposition of a SWAP gate into CNOT gates.

similar methods.

Naively, we could apply the same SWAP-based routing method for fermionic excitations. However, the FSWAP gate proposed in Ref. [90] enables a more efficient transpilation protocol, specifically tailored to fermionic operators. This gate is defined as

$$\texttt{FSWAP} \equiv \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix},$$

and corresponds to the exchange of *fermionic modes* rather than qubits. The additional phase in the last entry of the matrix accounts for fermionic anticommutation effects. If fermionic modes i, j had been mapped to qubits a, b as $i \leftrightarrow a$ and $j \leftrightarrow b$, $\texttt{FSWAP}(a, b)$ reassigns the modes such that $i \leftrightarrow b$ and $j \leftrightarrow a$. Conveniently, this gate can be implemented using only 2 CNOT gates using the circuit in Fig. 5.8. We depict the FSWAP gate using a dashed line so that it can be visually distinguished from the SWAP gate.

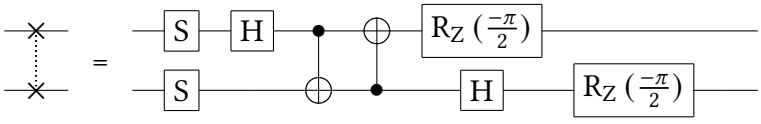


Figure 5.8: Decomposition of an FSWAP gate into CNOT gates.

With this gate, we can implement the fermionic single excitation circuit of Fig. 5.2 using the circuit in Fig. 5.9. Fermionic double excitations can be similarly implemented, as discussed above in the context of qubit double excitations.

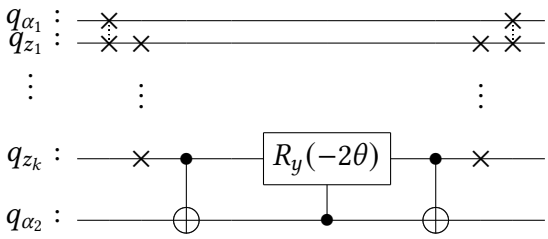


Figure 5.9: Implementation of the single fermionic excitation circuit in Fig. 5.2 for LNN connectivity. It is assumed that the mapping of fermionic modes to physical qubits is consistent with the ordering used in the Jordan-Wigner transform.

While for ATA connectivity fermionic excitations require many more entangling gates ($O(N)$, as opposed to $O(1)$ for qubit excitations), for LNN connectivity they may actually require *fewer* entangling gates than qubit excitations, due to the more efficient decomposition of the FSWAP gate.

Table 5.2 showcases this difference by considering the CNOT counts necessary to implement a fermionic single excitation and its qubit counterpart on a 100-qubit system. Two distinct implementation strategies are considered: One based on multi-controlled Y rotations followed by decomposition into a CNOT + single qubit rotations gate set, as proposed in Ref. [179] and discussed above, and one based on the general implementation of Pauli rotations using ladders of CNOT gates, which can be straightforwardly applied here since all Pauli strings in the generators commute [123].

We observe that for ATA connectivity, qubit excitations are a by far the most gate-efficient option, requiring only 2–4 CNOTs as compared to the hundreds required by the fermionic counterparts. Yet, the number of gates required by qubit excitations increases significantly when the unitaries are transpiled to LNN connectivity. In particular, if we allow the final layout of qubits or fermionic modes to vary—allowing us to measure the same expectation values with shallower circuits by exchanging the final SWAP/FSWAP networks for classically tractable manipulation of the observables—we can implement the fermionic excitation with as few as 198 CNOT gates, while the qubit excitation requires at least 298.

The same strategy can be applied to other fermionic operator pools; in particular, we can create a fermionic version of CEOs by exchanging $Q_i^\dagger \to a_i^\dagger$, $Q_i \to a_i$ in their definition. These fermionic CEO operators can be readily implemented with the same SWAP-based circuits that would implement CEOs in a LNN architecture, by simply exchanging the SWAP gates for FSWAP gates. This technique applies to OVP- as well as MVP-CEOs.

With these ideas in mind, we design 3 variants of Co-ADAPT-VQE:

- SWAP-based: This variant is suitable for local operators (i.e., operators acting on $O(1)$ qubits). Examples of this are the qubit, QE and CEO pools, whose operators are either 2- or 4-local. The transpilation of each operator is based on a SWAP network which brings the relevant qubits together, followed by the implementation of the actual excitation circuit. After the excitation circuit, the physical qubit assignment is classically updated to avoid the application of an order-restoring SWAP network. The penalty applied to each operator according to Eq. (5.6) is the total CNOT count of the circuit when implemented as described. Since the qubit assignment is dynamic, the penalty

Implementation	Product of Pauli Rotations					Multi-Controlled Y Rotations				
Connectivity	All-to-All	Linear Nearest Neighbor				All-to-All	Linear Nearest Neighbor			
Routing	-	Swaps		FSwaps		-	Swaps		FSwaps	
Final Layout	-	Fixed	Variable	Fixed	Variable	-	Fixed	Variable	Fixed	Variable
$a_0 a_{99}^\dagger$ - h.c.	396	396	396	396	200	196	1020	729	394	198
$Q_0 Q_{99}^\dagger$ - h.c.	4	592	298	-	-	2	590	296	-	-

Table 5.2: Number of `CNOT` gates necessary to implement the unitaries generated by fermionic ($a_0 a_{99}^\dagger$ - h.c.) and qubit ($Q_0 Q_{99}^\dagger$ - h.c.) excitations. These unitaries act on a system with 100 spin-orbitals (equivalently, qubits), exchanging fermions from spin-orbitals 0 and 99. The table compares the two types of circuit implementations discussed in the main text under all-to-all and linear nearest neighbor connectivity. For the latter, two routing strategies are considered: one using `SWAP` gates and one using `FSWAP` gates (when applicable). Further, since it is possible to keep track of the qubit layout classically, we consider a setting where the final qubit layout is variable (i.e., equivalent to the initial one up to a permutation). All the circuits are transpiled using the respective techniques with no further optimization. While circuit optimization methods might produce slight improvements, they do not meaningfully change the trends observed in the table.

associated with each operator is recalculated in each iteration. This penalty can be calculated classically without actually constructing the circuits: for a single excitation acting on qubits $a < b$, the total number of SWAP gates will be $b - a - 1$, while for a double excitation acting on qubits $a < b < c < d$ it will be $|a - a'| + |b - b'| + |c - c'| + |d - d'|$, where a', b', c', d' are the target physical qubits which can be obtained from a, b, c, d as described earlier in this section. At the end of the circuit, the final qubit assignment is used to redefine the observable, which is equivalent to applying a final SWAP network while avoiding the associated gate overhead. This reassignment is performed by permuting qubit indices in the Pauli strings that appear in the observable.

- FSWAP-based: This variant is suitable for fermionic operators, such as the generalized singles and doubles pool or the fermionic version of the CEO pool described above. The implementation is similar to the SWAP-based variant, with the difference of using FSWAP networks instead of regular SWAP networks. This allows us to bring fermionic modes together and implement a fermionic excitation—including the anticommutation string—with fewer gates than would be required to implement the corresponding qubit excitation. Accordingly, what we update at the end of each iteration to avoid order-restoring FSWAP ladders is the assignment of fermionic modes to qubits. At the end of the circuit, the final mode-to-qubit assignment is used to redefine the observable. This reassignment is performed by permuting mode indices in the excitations that appear in the fermionic Hamiltonian, which is equivalent to applying a final FSWAP network while avoiding the associated gate overhead. Importantly, this step is done *before* mapping the Hamiltonian under the Jordan-Wigner transform. The method used to calculate the penalty in this FSWAP-based variant is identical to the method used for the SWAP-based one, as described above. Since the mode assignment is dynamic, the penalty associated with each operator is recalculated in each iteration. As before, the penalty can be efficiently calculated classically without building the circuits.

- Qiskit-transpiler-based: This variant is fully generic, and it is suitable for any operator pool. In this case, the penalty associated with an operator is calculated by transpiling the corresponding circuit to the desired connectivity using Qiskit [137] under the maximum optimization level. In this case, the penalty is static: We can create the circuit associated with each pool operator at the start of the algorithm and store it along with the corresponding CNOT count so that it can be used for all iterations.

For all variants listed above, as well as for the regular ADAPT-VQE circuits transpiled to restricted connectivity, we apply the Qiskit [137] transpiler at maximum optimization level to the final circuit. As this transpiler changes the assignment of physical qubits, we update the final observable by exchanging the indices in the Pauli strings according to the permutation. In the case of the SWAP-based Co-ADAPT-VQE variant, this permutation straightforwardly composes with the one arising from the algorithm. In the case of the FSWAP-based Co-ADAPT-VQE variant, we apply the permutation of modes arising from the algorithm to the fermionic Hamiltonian, map the resulting operator to a linear combination of Pauli strings using the Jordan-Wigner transform, and finally permute the indices of the Pauli strings according to the

physical qubit assignment output by the transpiler. In the case of the `Qiskit`-transpiler-based Co-ADAPT-VQE variant, as well as of the regular ADAPT-VQE variants transpiled to restricted connectivity, the only permutation is the one associated with the transpilation, which can be readily corrected for by remapping indices in the observable after the Jordan-Wigner transform.

Finally, we note that the routing strategies we discussed in this section go well beyond LNN connectivity, and connectivity graphs with a higher degree will allow for an even more efficient transpilation. As an example, let us consider the asymptotic scaling of the average distance between qubits in a linear graph (degree 2) as compared to a square lattice (degree 4).

In the case of a linear graph of size N (N linearly connected qubits), we have a total of $\binom{N}{2}$ qubit pairs. For each distance $k \in \{1, ..., N-1\}$ we have exactly $N-k$ pairs (i, j) such that $j - i = k$. As such, the average distance between qubits is

$$\frac{1}{\binom{N}{2}} \sum_{k=1}^{N-1} k(N-k) = \frac{2}{N(N-1)} \frac{N^3 - N}{6} = \frac{N+1}{3}. \tag{5.7}$$

In the case of a square lattice with N qubits, the Manhattan distance $|x_1 - x_2| + |y_1 - y_2|$ gives us the shortest path between two qubits with coordinates (x_1, y_1) and (x_2, y_2). Since the two coordinates are independent, the total distance will be $2\frac{L+1}{3}$, where $L = \sqrt{N}$ is the maximum distance in either direction.

Hence, 2D connectivity offers an asymptotic improvement in the average distance between qubits (or equivalently, the average number of `SWAP` gates required for transpiling an operator) from $\mathcal{O}(N)$ to $\mathcal{O}(\sqrt{N})$, where N is the number of qubits. Similar conclusions can be drawn for higher degrees of connectivity.

5.5 Results

This section presents simulation results confirming that the instances of Co-ADAPT-VQE defined above are effective at decreasing the `CNOT` count of state preparation circuits, as they were designed to do. For all systems, we use the minimal STO-3G basis set. All circuits are transpiled to a gate set consisting of single-qubit gates and `CNOT`s, and all simulation results are noise-free. We note that `CNOT` count was shown to be a good predictor of how much a given circuit will be impacted by noise [38]; hence, equating a lower `CNOT` count to more noise-resilient circuits is a standard assumption. The code used to conduct the simulations is publicly available at `https://github.com/mafaldaramoa/ceo-\gls{ADAPT2}-\gls{VQE}`.

Figure 5.10 compares the regular ADAPT-VQE transpiled to LNN connectivity with Co-ADAPT-VQE for three different pools. All three variants of Co-ADAPT-VQE for `CNOT` minimization defined in the previous section are considered. We consider a H_6 molecule as a toy model for a strongly correlated system. Linear and triangular geometries are both considered to analyze the performance of the algorithm as correlations become stronger. Since the triangular H_6 molecule exhibits spin-frustration, it is particularly difficult to simulate.

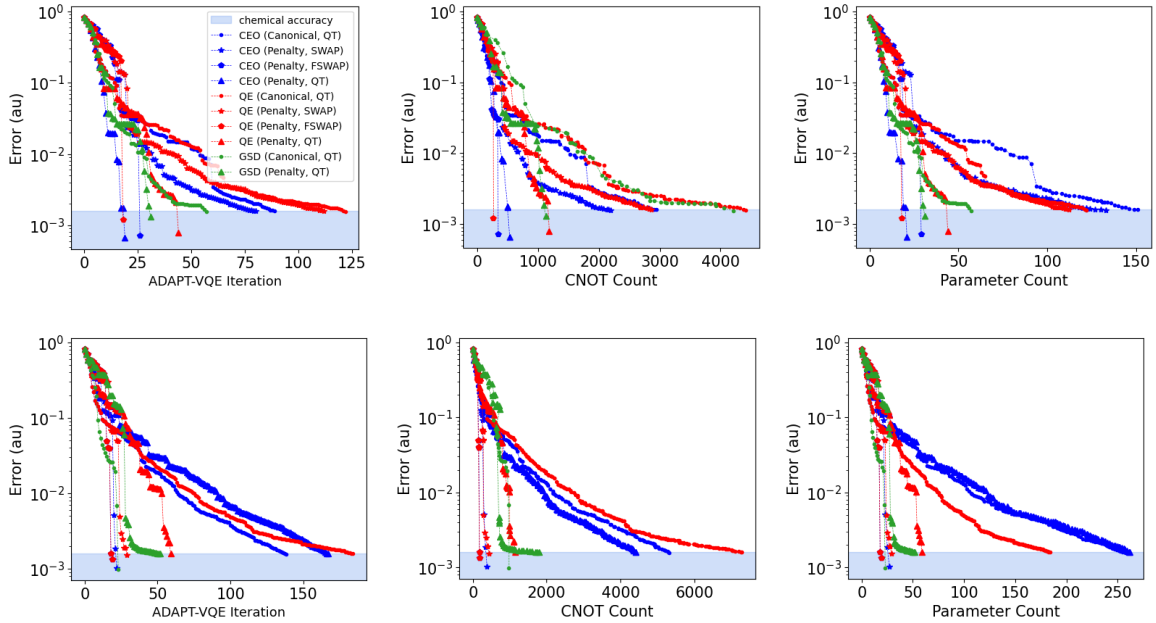
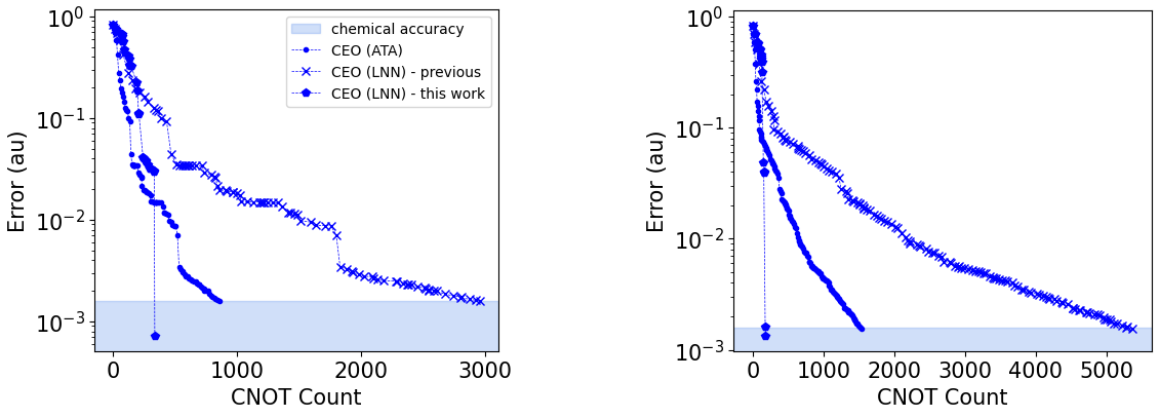


Figure 5.10: ADAPT-VQE error *versus* iteration number, CNOT count for LNN connectivity, and parameter count for linear (top) and triangular (bottom) H_6 (12 qubits) with interatomic distance 3 Å. The pools considered are CEO (see Ch. 4), QE [182] and GSD [72]. For each of these pools, we consider Co-ADAPT-VQE with the SWAP, FSWAP, and Qiskit-transpiler (QT) -based penalties. The canonical version of ADAPT-VQE transpiled to the same (LNN) connectivity is also included for comparison. For the case of CEO and QE pools, the FSWAP-based version of Co-ADAPT-VQE implements the fermionic version of these operators, which is obtained by exchanging SWAP gates for FSWAP. This exchange amounts to adding Jordan-Wigner anticommutation strings to the corresponding generators. All algorithms terminate at chemical accuracy. In the bottom plots, the QE and CEO with FSWAP-based penalties overlap. Note that the GSD pool has no associated curve for SWAP- and FSWAP-based penalties. This is because the QE pool with FSWAP gates already implements GSD operators (with FSWAPs + QE circuit constructions), and implementing a fermionic operator using SWAP rather than FSWAP gates results in unnecessary CNOT gate overheads.

We observe that Co-ADAPT-VQE with the FSWAP-based penalty is the best performer for both molecules. This variant is able to decrease the CNOT count from over 4000 to roughly 300 for linear H_6, and from over 7000 to roughly 200 for triangular H_6. It is interesting to note that the impact is higher for the more strongly correlated system, which we attribute to the improved capability (as compared to qubit excitations) of fermionic excitations, which feature the Jordan-Wigner anticommutation string, to represent strongly correlated ground states. This effect has been noted previously in the literature [182], although previous work did not take advantage of the hardware efficiency of FSWAP networks for implementation for LNN connectivity. The instances of Co-ADAPT-VQE with SWAP- and Qiskit-transpiler-based penalties are also capable of reducing the CNOT counts of the circuits, although the reduction is not as significant as for the FSWAP-based protocol. The iteration and parameter counts required to reach chemical accuracy are actually reduced for the best performing variants of Co-ADAPT-VQE as compared to regular ADAPT-VQE; hence, we do not expect our algorithm to imply an increase in measurement costs from gradient measurement rounds and optimizations, on the contrary.



(a) Linear H_6

(b) Triangular H_6

Figure 5.11: Error *versus* CNOT count for CEO-ADAPT-VQE for ATA connectivity as compared to LNN connectivity, for H_6 at interatomic distance 3 Å. The LNN CNOT counts are obtained in two ways: By transpiling the ATA circuit from the canonical CEO-ADAPT-VQE (the best performing pool in the literature), and by using our Co-ADAPT-VQE protocol with the CEO pool and the FSWAP-based penalty. All algorithms terminate at chemical accuracy.

Considering that most work on ADAPT-VQE has been done considering ATA connectivities [160, 182], it is relevant to compare the ATA and LNN gate counts. We can obtain the LNN ansätze by transpiling the canonical ADAPT-VQE to LNN connectivity — the standard method prior to this work — or by using our co-designed version of the algorithm. Figure 5.11 compares the CNOT count of both approaches with the ATA ansatz for the same molecules as Fig. 5.10.

Surprisingly, our Co-ADAPT-VQE protocol is able to *decrease* the CNOT count as compared to the original ATA ansatz, despite the strong restriction on connectivity (with each qubit in the connectivity graph sharing an edge with only 2 others for LNN , as compared to 14 for ATA). We attribute this to two things: first, the aforementioned fact that the Jordan-Wigner anticommutation improves the convergence of the algorithm for strongly correlated systems; and second, the fact that in the original algorithm, operators with a lower CNOT count are not favored in any way. While the impact of this is more pronounced in restricted connectivity, operators also have varying CNOT counts for ATA connectivity; for example, a single qubit excitation requires only 2 CNOTs, while a double qubit excitation requires 13. This brings the question of whether Co-ADAPT-VQE may be used to reduce CNOT counts for ATA connectivity as well. In App. L, we confirm that this is indeed a possibility. While the impact is not as dramatic as for the LNN case, we still observe a significant (roughly 20%) decrease in the gate count.

To generalize Fig. 5.11 to other pools, we present similar results for the GSD pool in Fig. 5.12. Similarly to the case of the CEO pool, we observe that not only does Co-ADAPT-VQE result in a LNN -connectivity-restricted circuit with an order of magnitude less CNOTs than the canonical ADAPT-VQE ansatz after transpilation, but it also reduces this gate count as compared to the original circuit with no connectivity restrictions.

In Fig. 5.13 we delve into data concerning the evolution of the penalty as well as operator gradients

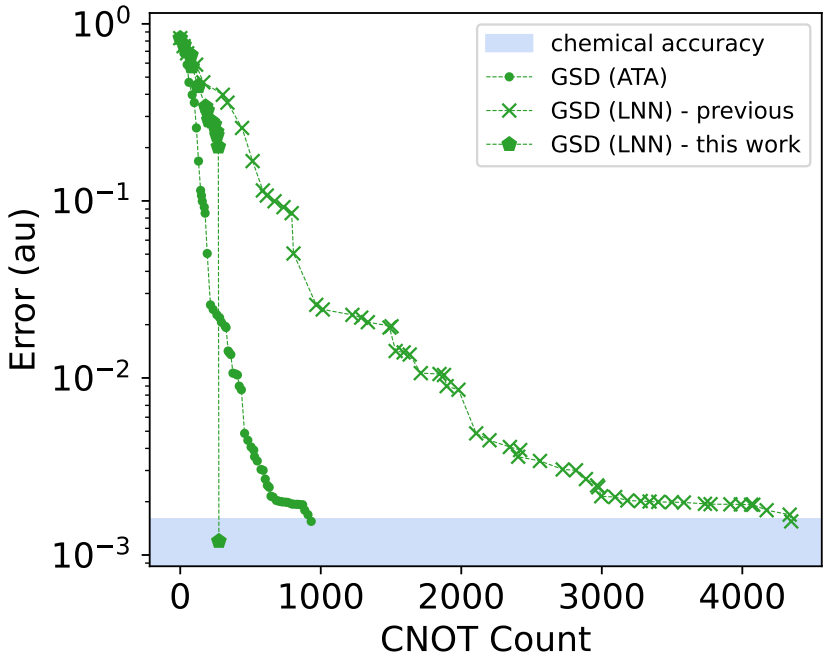


Figure 5.12: Error *versus* CNOT count for GSD -ADAPT-VQE for ATA connectivity as compared to LNN connectivity. The details are the same as for Fig. 5.11(a), with the only difference being the pool (GSD vs CEO). In particular, the curve associated with this work corresponds to Co-ADAPT-VQE with the GSD pool and the FSWAP-based penalty, such that operators are implemented as cascades of FSWAP gates and qubit excitation circuits.

to showcase the inner workings of the algorithm. The first plot compares, for each iteration, the average CNOT count (across all pool operators) with the CNOT count of the selected operator. We observe that, as expected, Co-ADAPT-VQE tends to select operators requiring fewer CNOTs than average. Interestingly, this is not always the case; there are some iterations where the CNOT count of the selected operator is significantly higher than average. This shows that the algorithm is still able to select operators with higher CNOT counts, provided that their gradient magnitude is high enough to offset the penalty.

In the second and third plots, we observe the evolution of the gradient magnitude and the penalty (CNOT count) for a double and a single excitation, respectively. The single excitation is selected more often than the double excitation, which is typical behavior, considering single excitations can typically be implemented with fewer gates. In both cases, we observe that operators are typically selected after a big drop in the penalty arising from a favorable rearrangement of fermionic modes in the iteration before. In general, we observe that the gradient magnitude stays relatively constant until a sharp drop in the penalty occurs, at which point the operator is selected. This behavior illustrates that the algorithm favors hardware efficiency by waiting until an operator can be conveniently implemented to effectively add it to the ansatz.

Finally, we remark on a few key points concerning our algorithm and results. First, while Co-ADAPT-VQE is ideal for chemically accurate results, we expect its impact to be less significant for extremely low errors. This is because, as we get very close to the ground energy, there are very few operators that can actually rotate the state closer to the target; hence, selecting the top operator becomes more crucial,

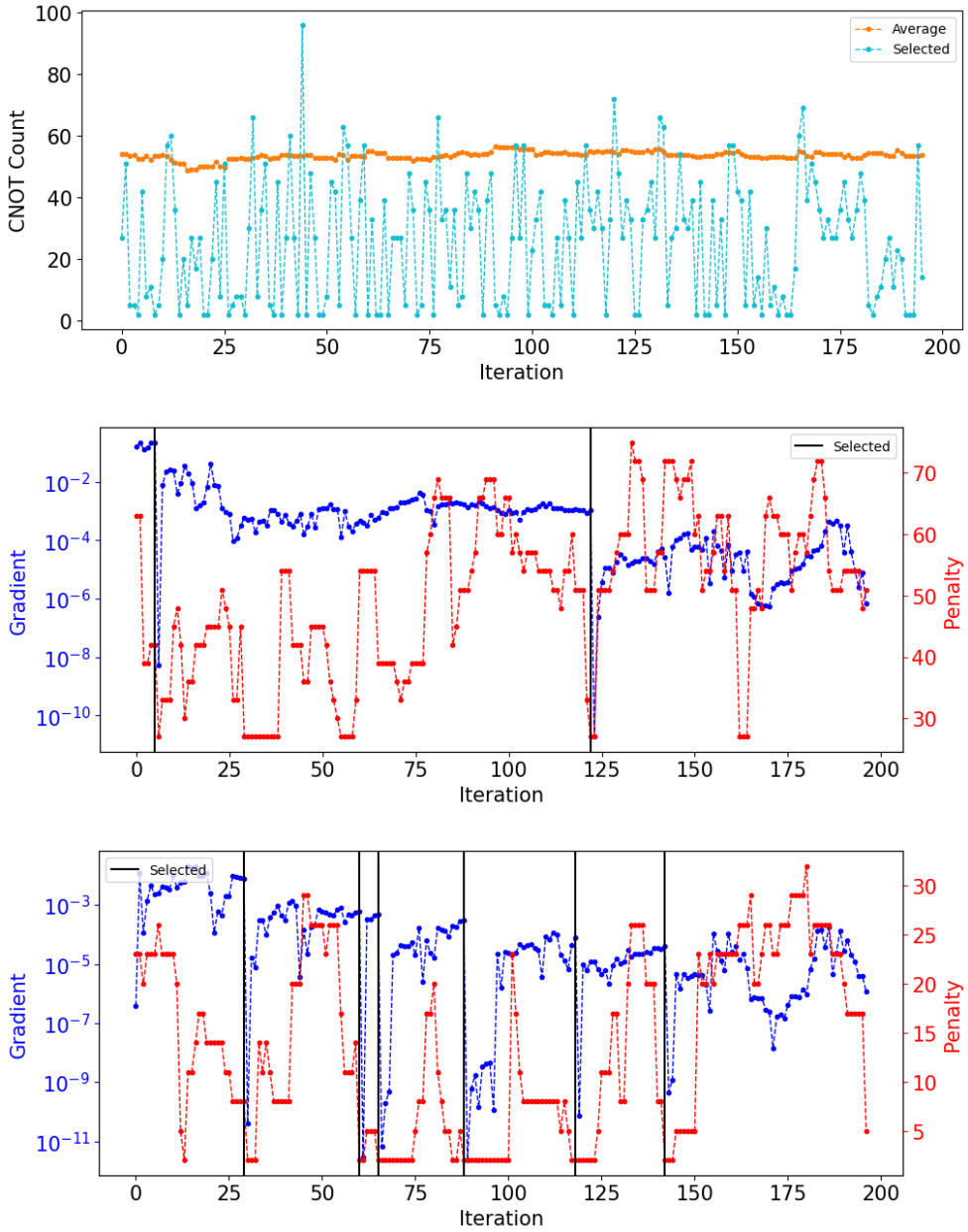


Figure 5.13: Evolution of the penalty and operator gradients for the Co-ADAPT-VQE algorithm. We consider the FSWAP-based variant with the CEO pool, applied to the linear H_6 molecule at interatomic distance 3 Å. The top panel compares the average penalty (equivalently, CNOT count) with the penalty of the selected operator for each iteration. The middle panel shows the evolution of the gradient norm and the penalty for a particular double excitation, and the bottom panel shows the same results for a particular single excitation. Black vertical bars indicate iterations in which the operator was chosen.

which the application of the penalty prevents. We illustrate this in App. M, where we show that while Co-ADAPT-VQE is still able to improve upon the canonical algorithm, the impact is not as remarkable. To counterbalance this, we could modify the penalty such that it is dynamically adapted along the iterations. An interesting possibility is to add a damping term that decreases the impact of the penalty towards later iterations. This term could be a function of the iteration number, the energy changes, the gradients, and/or other available metrics. Second, the choice of Eq. (5.6) for penalizing the gradient is not unique. In fact, we can easily conceive alternative formulas for penalizing the gradient, such as the one we obtain by replacing p (the penalty) with a power of it. We study this possibility in App. N. Third, while we only considered H_6 molecules in different geometries, the algorithm may be straightforwardly applied to other systems. However, for very small or weakly correlated systems requiring very shallow ansätze, standard transpilation techniques (which include reordering qubits at the beginning and end of the circuit) work well and might actually outperform some variants of Co-ADAPT-VQE. We illustrate this by applying the algorithm to LiH in App. O. However, we note that it is expected that larger and more strongly correlated molecules are more important targets when aiming for quantum advantage, and we expect Co-ADAPT-VQE to perform well for such systems. The fact that it leads to such large reductions in gate counts for H_6 in linear geometry, and even greater ones for the more strongly correlated triangular geometry, supports this conjecture.

5.6 Discussion

Due to its ability to avoid trainability issues and its shallow circuits, ADAPT-VQE is increasingly regarded as the leading near-term algorithm for state preparation; regardless, research efforts directed at improving the algorithm have mostly been agnostic to the target hardware. Gate counts in all-to-all connectivity are a standard metric for evaluating the algorithm's performance, but hardware constraints often enforce restrictions on which pairs of qubits can undergo entangling operations. While standard techniques can be used to transpile any ansatz to restricted connectivity, they incur a significant overhead in gate counts, which is counter to the objectives of a near-term algorithm. This begs the question: Can we design a state preparation algorithm that constructs a hardware-specific circuit that improves noise resilience without sacrificing accuracy?

In this chapter, we provided an affirmative answer to this question by introducing Co-ADAPT-VQE, an algorithm that co-designs an ansatz by including hardware-specific constraints and limitations into the criteria used to build the circuit. This was achieved by incorporating factors of choice—such as gate counts in restricted connectivity, operation error rates, etc.—into the selection criterion of ADAPT-VQE. This fertile framework can be adapted to suit any hardware platform and any particular quantum computer by favoring the addition of circuit blocks that can be conveniently implemented in the target device. Importantly, the framework's generality allows one to address whatever one desires to prioritize in a given experiment, whether that is achieving a low CNOT count or depth, using operations with low error rates, or building ansätze with lower parameter counts, among many other options.

To illustrate the effectiveness of Co-ADAPT-VQE, we created several instances of the algorithm to address a common problem: minimizing the CNOT counts of ansätze for preparing molecular ground states in linear nearest-neighbor connectivity, which is directly related to the noise-resilience of the circuits. We showed that Co-ADAPT-VQE achieves a dramatic decrease in this figure, with CNOT counts decreasing from over 7000 to fewer than 200 in the case of the most strongly correlated system (H_6 in triangular geometry). We expect the impact to become even greater for larger systems, where the path between pairs of qubits is on average longer in the LNN connectivity graph. Further, our simulations show that Co-ADAPT-VQE has a greater impact for more strongly correlated systems, suggesting that this algorithm will perform particularly well in simulating precisely those systems for which quantum computers are expected to provide an advantage.

6

Overcoming Gradient Troughs: Exploring Alternative Operator Positions

This chapter is associated with the following authored publication:

Strategies for Overcoming Gradient Troughs in the ADAPT-VQE Algorithm. J. Stadelmann, J. Übelher, **M. Ramôa**, B. Sambasivam, S. E. Economou. arXiv:2512.25004

6.1 Motivation

As discussed in Ch. 2, a yet-unresolved potential roadblock in the implementation of ADAPT-VQE is the presence of gradient troughs [72]. When a gradient trough occurs, the norm of the gradient falls sharply even though the solution has not been reached. This may result in false convergence or in the incorrect operator being selected due to the difficulty in resolving such small gradients. High computational resources (measurements) are required to select the highest-gradient operators during a gradient trough, and even if the correct operators are selected, the energy will typically not reduce significantly during the gradient trough, leading to an increased ansatz size on top of the measurement overhead.

One possible approach to remove the unnecessary circuit depth accrued throughout the gradient troughs is to remove operators. This has been attempted in Refs. [140, 165]. Although these heuristic approaches may help compactify the ansatz in certain cases (such as weakly correlated systems or large basis sets), they only apply *after* a gradient trough, and thereby do not reduce the number of iterations required to escape it, nor do they alleviate the measurement costs of choosing good operators along it. Hence, such strategies do not address the main problems stemming from a gradient trough.

In this chapter, we present methods to diagnose gradient troughs *during* the execution of ADAPT-VQE, allowing them to be properly addressed. We further propose enhanced protocols that are able to escape the gradient troughs. The non-commutativity of the ansatz operators leads to the key insight that inserting the same operator at different points in the ansatz produces distinct gradients and impacts on the energy. To our knowledge, our protocols are the first to use heuristics to select the position in the ansatz to add operators as opposed to always adding them at the end of the ansatz. This introduces new variational paths, leading to a non-vanishing gradient signal throughout the optimization process.

This enables ADAPT-VQE to continue to converge even when gradient troughs emerge and the canonical ADAPT-VQE protocol gets trapped.

This chapter is structured as follows. Section 6.2 briefly summarizes the gradient trough phenomena and illustrates it with an example. In Sec. 6.3, we present new techniques to detect gradient troughs during the execution of ADAPT-VQE. We follow with Sec. 6.4, where we briefly outline our approach and propose protocols aimed at solving the problem of gradient troughs. We compare the cost of each calculation scheme with the original algorithm in Sec. 6.5. In Sec. 6.6, we numerically investigate the effect of our proposed protocols on gradient troughs. Finally, Sec. 6.7 discusses the implications of our new protocols and how they might be combined with other methods to obtain an efficient and robust version of ADAPT-VQE.

6.2 Example of a Gradient Trough

We have seen that, although ADAPT-VQE resolves optimization issues such as barren plateaus and susceptibility to local traps, it encounters a related problem—'gradient troughs'. A gradient trough refers to a regime during the optimization in which the gradient norms decrease rapidly, even though the ground state has not yet been reached, only to increase again after some iterations. In classical simulations with effectively unlimited precision in the energy evaluation, this results in the stalling of the energy and with it error, as shown in Fig. 6.1, which in practice amounts to the allocation of both quantum as well as classical computational resources without a measurable effect on the reduction of the energy. In a realistic setting, the low gradient norms within the troughs might lead to premature and wrongful convergence if the value drops below the threshold that defines termination of the algorithm. Furthermore, gradient troughs increase measurement costs: When gradients become very small, they must be measured to high precision to ensure that the operator with the highest gradient magnitude is added to the ansatz within the ADAPT-VQE selection step.

6.3 Detecting Gradient Troughs

In this section, we propose a new protocol to detect gradient troughs with the primary goal of distinguishing them from true convergence of the algorithm. Our protocol involves the measurement of the gradient norm of the pool operators at various positions p within the ansatz. As discussed earlier, the gradient norm becomes small when p is close to $n+1$ (the end of the ansatz). On the other hand, when p is farther away from the end of the ansatz, the gradient norm can be much larger, having a greater impact on the cost function landscape. This phenomenon can be elucidated by numerically comparing the gradient formulas for the appending position with those for other positions within the ansatz. If the algorithm has indeed truly converged, the gradient norm across all p will be small.

We show a numerical example of this difference between a gradient trough and true convergence in Fig. 6.2. In Fig. 6.2a, the total gradient norm of all operators within the pool is plotted against the

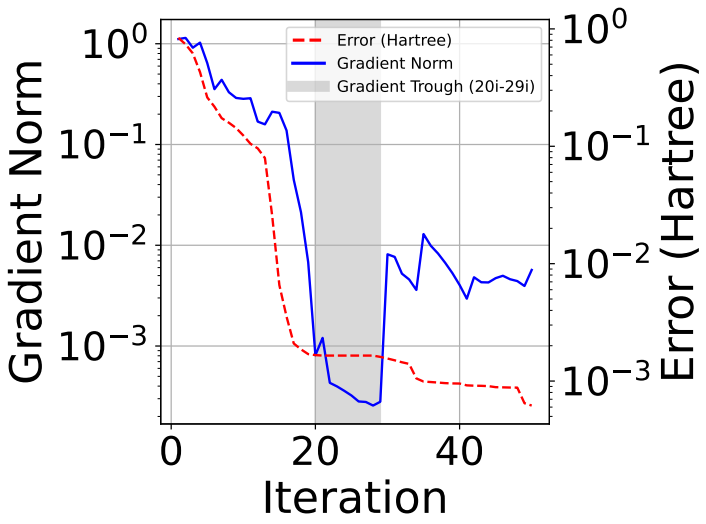
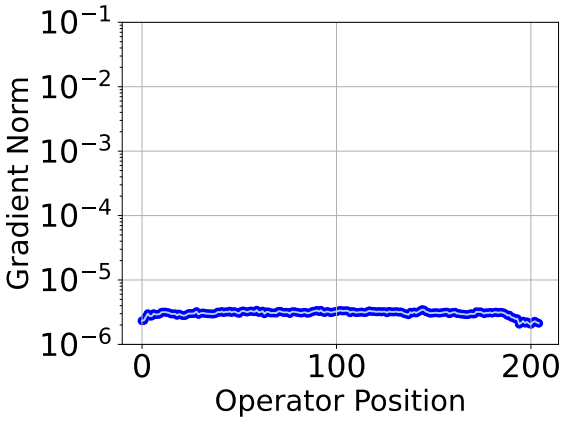


Figure 6.1: Noise-free simulation of ADAPT-VQE for a linear H_6 molecule with 4 Å interatomic distance until iteration 50. Between iterations 20 and 29, a significant gradient trough is evident, accompanied by a substantial decrease in gradient norm and a convergence plateau in energy, as is typical for them. The error is defined with respect to the FCI energy.

ansatz positions for an iteration inside a gradient trough. The gradient norm decreases steadily as the operator insertion point moves from the beginning to the end of the ansatz. When we compare this to true convergence shown in Fig. 6.2b, a major difference can be noted: In the converged case, the gradient norms are much more uniform across the positions, not showing the same decrease we observed within



(a) Gradient norm versus operator position inside a gradient trough (iteration 25).



(b) Gradient versus operator at true convergence (iteration 205) with a convergence threshold of $5 * 10^{-7}$ for the appending position.

Figure 6.2: Gradient norm against operator position for a regular execution of ADAPT-VQE (i.e., appending operators). Panel (a) shows an iteration within a gradient trough, while the right panel considers an iteration at convergence. Within the gradient trough, a clear decrease in the gradient norm from the prepending to the appending position can be observed. For the case at convergence, there is no such consistent decrease visible: Instead, we can observe that the gradient norms are mostly uniform across positions.

the troughs (Fig. 6.2a).

In the context of the appending position, the gradient of a pool operator A_μ is given by Eq. (2.13) and corresponds to the expectation value of the commutator $[\hat{H}, A_\mu]$ in the current state. On the other hand, the gradient of an operator A_μ in a general position p within the ansatz can be defined as:

$$\left.\frac{\partial E}{\partial \theta_\mu}\right|_{\theta_\mu=0} = \langle\psi| \left[\hat{H},\ V(\theta) A_\mu V^\dagger(\theta)\right] |\psi\rangle\,, \tag{6.1}$$

$$V(\theta) = \left(\prod_{i=p}^{n} e^{\theta_{p+n-i} A_{p+n-i}}\right) \tag{6.2}$$

where p must be within the range of position $p = 1$ (prepending) up to position $p = n + 1$ (appending), and the products are defined such that factors corresponding to smaller values of i are on the left, while those with larger values of i are on the right. The derivation of this formula is provided in App. J.

Therefore, the presence of low gradient norms in the appending position, as observed in gradient troughs, does not inherently imply that the gradient norms are small for all ansatz positions. This is due to the fact that the gradients for non-appending positions involve an additional conjugation of A_μ by V(θ), which depends on subsequent variational layers. A concise numerical investigation of this is included in App. K.

This property of gradient troughs can be observed numerically in Fig. 6.3. The plot clearly shows that before the trough, the gradient norm for adding an operator in a specific position of the ansatz has a similar order of magnitude across all positions. In contrast, as the algorithm enters the gradient trough (iteration 20), the gradient norms in the appending position (top) are significantly lower compared to the gradient norms for the prepending position (bottom). As the algorithm approaches iteration 30 and exits the gradient trough, the gradient norms are again within the same order of magnitude across the ansatz. A parallel can be made with Fig. 6.1, which corresponds to the gradient norm in the appending position alone, as both plots exhibit the same gradient trough.

Figures 6.2 and 6.3 provide us with a crucial tool to distinguish between convergence and a gradient trough. To make sure the algorithm has truly converged to the ground state, we must check that all gradient norms are small and of similar magnitude across ansatz positions. If instead we observe that the gradient norms decrease significantly towards the end of the ansatz, we are inside a gradient trough. Using these results, one may redefine the convergence criterion as having all the gradient norms for all ansatz positions below a specific threshold, beyond which it is not practical to resolve the gradients with sufficient accuracy. This guarantees that the algorithm cannot further decrease the energy, as it has either attained (within the available accuracy) a local minimum or, conceivably, the global minimum. Current convergence criteria are based exclusively on the gradient norm in the last ansatz position (appending) falling below a preset threshold, which risks converging too early due to a gradient trough. However, this approach has the benefit of requiring fewer measurements, as measuring the gradient norm over all ansatz positions incurs a significant measurement overhead.

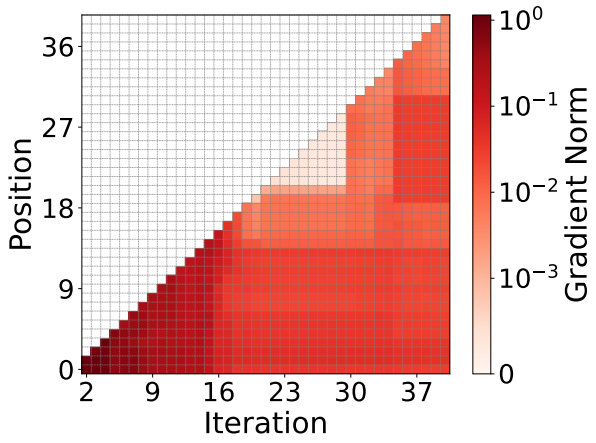


Figure 6.3: Gradient norm across ansatz positions for ADAPT-VQE iterations 1–40. The gradient trough is visible across iterations 20–29 as a decrease in gradient norm as the position moves towards the end of the ansatz (with appending being the extreme case).

To efficiently implement a method that diagnoses convergence and avoids premature termination, a tradeoff between the two convergence criteria may be developed and implemented. This will allow for a realistic incorporation of our findings into the ADAPT-VQE protocol, allowing for the detection of troughs and preventing false convergence while minimizing the measurement overhead. We leave the specific implementation of such a method for future work; however, we outline the basic framework below.

Before a given convergence criterion (based on the appending gradient norm) is met, the regular ADAPT-VQE protocol is executed. Once the gradient norm falls below the convergence threshold, the norm for a select number of other positions within the ansatz is measured. The choice of positions to test can be random or based on heuristics. From Fig. 6.3, we can see that the operator gradients are non-vanishing for any ansatz position that is not within a few positions of the appending position. One possibility is to restrict the test positions by selecting one (or more) at random with a bias towards the beginning of the ansatz (by, e.g., creating a probability distribution over positions that decays with the distance to the beginning of the ansatz). After measuring the gradients in the test positions, we may conclude that a gradient trough is present if the gradient norm is significantly higher in positions other than appending, or (given enough data points) if the measurements show a decline in the gradient norms as the default position for adding new operators (end of ansatz) is approached. If the difference between the norm in the appending position and in other positions is within a preset threshold, we conclude that the algorithm has terminated and the ground state has been found.

6.4 Mitigating Gradient Troughs

While the discussion in Sec. 6.3 proposed tools to diagnose gradient troughs, it did not provide tools to mitigate their effects on the ADAPT-VQE algorithm. In this section, we introduce a set of new protocols designed to escape gradient troughs. These protocols are aimed at ensuring a consistent optimization process even when a gradient trough occurs within the regular ADAPT-VQE calculation. They make use of

the non-commutative properties of the exponential operators, which result in different gradients at each ansatz position for each candidate operator.

We have observed the variation of the gradients for different ansatz positions in Fig. 6.3. The canonical ADAPT-VQE algorithm always appends operators to the ansatz, but as the algorithm enters the gradient trough, the gradient norms for the appending position become smaller, causing the problems we have discussed before. On the other hand, the gradients for inserting operators in other ansatz positions remain high due to the effects of the ansatz position on the commutator within the gradient formula (Eq. (6.1)). The protocols we propose leverage the possibility of adding new operators into ansatz positions other than appending. The ansatz for one of the new protocols can be expressed as:

$$|\psi_{\text{ansatz}}\rangle = \left(\prod_{i=p}^{n} e^{\theta_{p+n-i}A_{p+n-i}}\right) e^{\theta_\mu A_\mu} \left(\prod_{i=1}^{p-1} e^{\theta_{p-i}A_{p-i}}\right) |\psi_0\rangle\,, \tag{6.3}$$

where we introduced the new operator $e^{\theta_\mu A_\mu}$ into position p within the ansatz. In the typical ADAPT-VQE algorithm, $p = n + 1$ always. The energy expectation value on this parametrized variational wavefunction is given by

$$E = \langle\psi_0| \left(\prod_{i=1}^{p-1} e^{-\theta_i A_i}\right) e^{-\theta_\mu A_\mu} \left(\prod_{i=p}^{n} e^{-\theta_i A_i}\right) \hat{H} \left(\prod_{i=p}^{n} e^{\theta_{p+n-i}A_{p+n-i}}\right) e^{\theta_\mu A_\mu} \left(\prod_{i=1}^{p-1} e^{\theta_{p-i}A_{p-i}}\right) |\psi_0\rangle\,. \tag{6.4}$$

In this section, we introduce four protocols: random operator random position (RO/RP), random operator optimized position (RO/OP), optimized operator random position (OO/RP), and optimized operator optimized position (OO/OP). The implementation of these protocols only requires minor changes to the original (appending) algorithm. We depict them in Fig. 6.4 and describe each protocol in detail below.

- **OO/OP (optimized operator optimized position)**: The energy gradient for a fixed number of pool operators (here taken to be 10) is measured for all possible ansatz positions. The 10 operators whose gradients across the ansatz we choose to measure are those with the highest gradient for appending. Out of the 10 operators and all ansatz positions, we select the combination of operator and position associated with the highest gradient.

- **OO/RP (optimized operator random position)**: The ansatz position is selected randomly, and the gradients of all pool operators are measured in this position. We select the operator with the largest energy gradient magnitude for this position.

- **RO/OP (random operator optimized position)**: The energy gradient for a fixed number of pool operators is measured for all possible ansatz positions. Here we choose this number to be 10. The 10 operators whose gradients across the ansatz we choose to measure are those with the highest gradient for appending. The selected position is the one where these gradients are highest. The operator to be added is selected randomly and placed into the identified position. This protocol should be seen as a control for the OO/OP protocol that we use to understand how important it

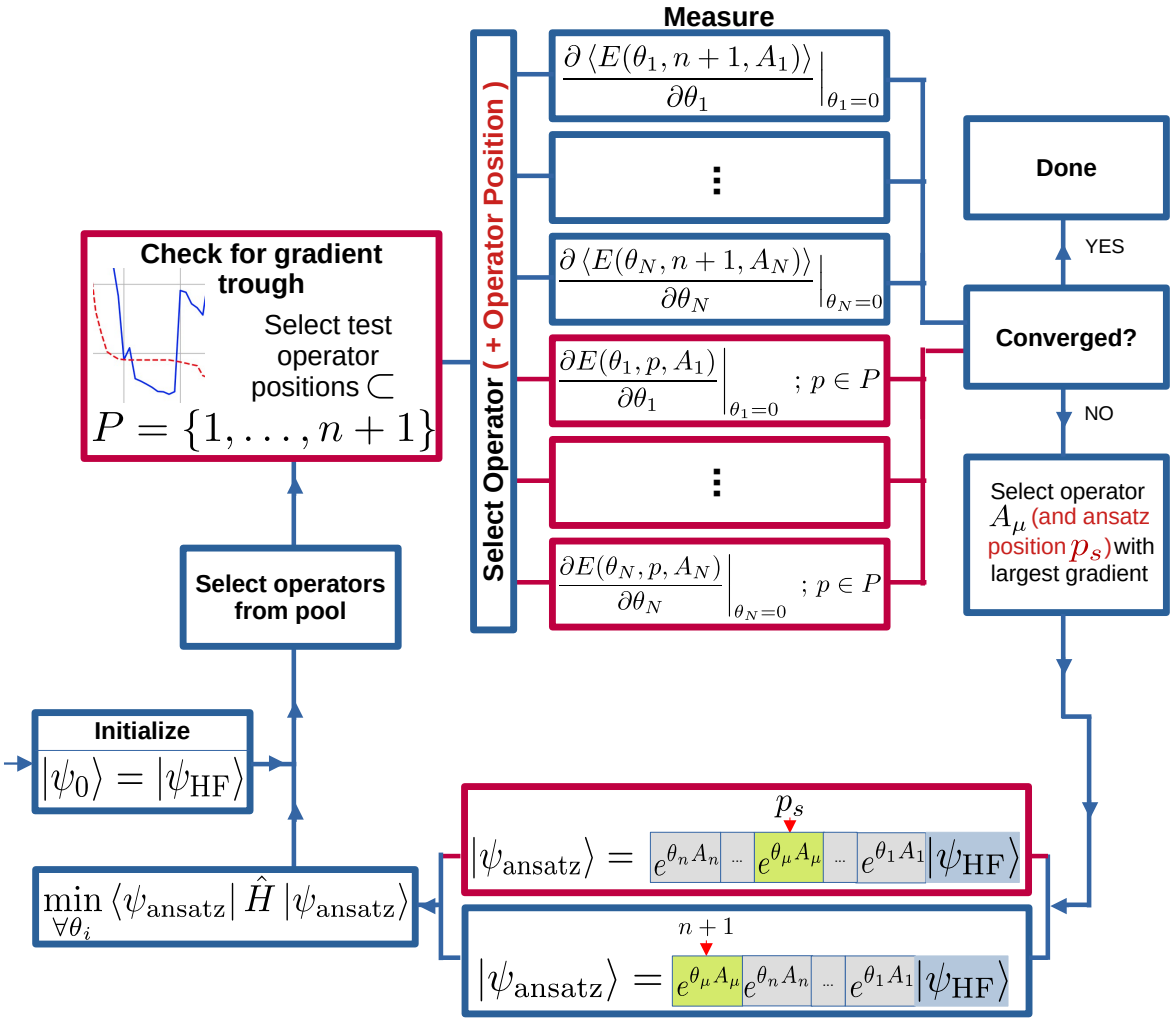


Figure 6.4: Workflow of the enhanced ADAPT-VQE algorithms, where red indicates our changes to the protocols and blue is the original ADAPT-VQE algorithm (see Ch. 2). Unlike the original ADAPT-VQE, our enhanced algorithms check for gradient troughs. If a gradient trough occurs, the protocol measures gradients at positions that are most likely to yield higher gradients than appending. In the next step, the operator and position that yield the largest magnitude gradient are identified, and the operator is inserted accordingly into the ansatz. All three algorithms are based on this principle. For an exact description see Sec. 6.4. The function $E(\theta_\mu, p_s, A_\mu)$ refers to the energy expectation value of the system as defined in Eq. J.2, where in the given case θ_μ is the variational parameter for the operator A_μ, p_s refers to the position of the newly added operator A_μ in the ansatz and A_μ with $\mu \in [1, N_{\mathrm{Pool}}]$ is defined to be an operator within a given pool of size N_{Pool}.

is to select the highest gradient operator, as compared to a random operator, when the position is selected *via* our heuristics. As compared to OO/OP, RO/OP selects the operator from the complete pool, rather than being restricted to the top 10 operators for appending in a specific iteration.

- **RO/RP (random operator random position)**: Both the position and the new ansatz element are chosen randomly.

All these protocols make use of the fact that adding operators in other positions in the ansatz can lead to a higher impact on the cost function landscape, thereby helping escape or avoid gradient troughs. When we encounter such a problem, the gradients of operators will become low for appending (Eq. (2.13)). However, by adding operators in different ansatz positions (Eq. (6.1)), the gradient norm is significantly increased in most cases. Using our protocols to select an operator and its position as a gradient trough occurs, the algorithm is able to escape/avoid it relatively quickly. The apparent price of the application of these algorithms is that, except for the random operator random position (RO/RP) protocol, these algorithms seemingly require a higher computational cost compared to the default ADAPT-VQE protocol. We follow with a comparison of the measurement costs.

6.5 Measurement Cost Analysis

In the context of a realistic implementation on a quantum computer, it is essential to incorporate the statistical deviation coming from measuring expectation values with a finite number of shots. To obtain sufficient accuracy, we expect it to be necessary to measure each gradient approximately $\mathcal{O}\left(\frac{1}{\|g\|_\infty^2}\right)$ times, where $\|g\|_\infty$ is the infinity norm of the gradient vector, since this is the scaling of the number of shots required to distinguish the highest pool operator gradient from zero. For qubit or fermionic excitations acting on real states (as is our case), it has been shown that the cost of measuring each gradient is roughly equivalent to the cost of measuring the energy (requiring only a constant number of additional gates), just like the case of a single Pauli string [91]. Thus, we can estimate the number of shots required to measure all pool operator gradients in any given ansatz position, $N_{\text{g_meas}}$, as

$$N_{\text{g_meas}} = \mathcal{O}\left(\frac{N_{\text{H}} N_{\text{Pool}}}{\|g\|_\infty^2}\right) \tag{6.5}$$

where N_{H} is the number of Pauli strings in the Hamiltonian and N_{Pool} is the number of operators in the pool. For molecular Hamiltonians and the pool we are considering, we have that $N_{\text{H}} = N_{\text{Pool}} = \mathcal{O}\left(N_{\text{orbs}}^4\right)$, where N_{orbs} is the number of orbitals, leading to a scaling of $\mathcal{O}\left(\frac{N_{\text{orbs}}^8}{\|g\|_\infty^2}\right)$ for the number of shots required.

We may already note that, despite seemingly incurring higher measurement costs, our protocols are actually capable of saving quantum resources on the ADAPT-VQE operator selection step by decreasing the precision requirements via an increase in the gradient norm $\|g\|_\infty$, which typically drops drastically (thereby requiring higher measurement costs) in a gradient trough. In our simulations, we observed it to be typical for our protocols to increase the gradient norm by two orders of magnitude with respect to the

original protocol within a gradient trough, leading to savings on the order of 10^4 in the measurement costs required to evaluate the gradients in each iteration. This difference will be exacerbated by the parameter optimization, since, during a gradient trough, more measurements will be required to repeatedly resolve the search direction.

While measuring the appending operator gradients might come at a lower cost in our protocols, most of them require measuring gradients in additional ansatz positions, which results in additional quantum computer calls. More precisely, the cost of selecting an operator and position for the four protocols we propose is given by

$$\begin{aligned} N_{\text{m-oo/op}} &= N_{\text{g_meas}} + 10 N_{\text{Positions}} N_{\text{H}} = \mathcal{O}\left(\frac{N_{\text{orbs}}^8}{\|g\|_\infty^2}\right) + \mathcal{O}\left(\frac{N_{\text{orbs}}^4 N_{\text{Positions}}}{\|g\|_\infty^2}\right), \\ N_{\text{m-oo/rp}} &= N_{\text{g_meas}} = \mathcal{O}\left(\frac{N_{\text{orbs}}^8}{\|g\|_\infty^2}\right), \\ N_{\text{m-ro/op}} &= N_{\text{g_meas}} + 10 N_{\text{Positions}} N_{\text{H}} = \mathcal{O}\left(\frac{N_{\text{orbs}}^8}{\|g\|_\infty^2}\right) + \mathcal{O}\left(\frac{N_{\text{orbs}}^4 N_{\text{Positions}}}{\|g\|_\infty^2}\right), \\ N_{\text{m-ro/rp}} &= 0, \end{aligned} \tag{6.6}$$

where the subscripts indicate the protocol and $N_{\text{Positions}}$ corresponds to the number of positions we are testing within the ansatz.

We observe that only the protocols OO/OP and RO/OP incur a measurement overhead with respect to the regular ADAPT-VQE operator selection cost given by Eq. (6.5). This cost is dependent on the protocol-specific hyperparameter $N_{\text{Positions}}$. In our simulations, we chose to test all possible positions within the ansatz. This is typically significantly lower than the number of operators in the pool, meaning that the additional costs are eclipsed by the typical cost of operator selection in ADAPT-VQE, $\mathcal{O}\left(\frac{N_{\text{orbs}}^8}{\|g\|_\infty^2}\right)$. Further, the value of $\|g\|_\infty^2$ will be much higher for our protocols during a gradient trough, leading to an effective overall decrease in measurement costs.

To further lower the cost, we could only measure gradients for a select subset of ansatz positions, thereby decreasing $N_{\text{Positions}}$. As we discussed in Sec. 6.3, one interesting possibility is to select positions randomly with a bias towards the beginning of the ansatz. We leave the exploration of such ideas for future work.

Finally, we note that when we append operators, we can use Eq. (2.13) to obtain the gradients, which has the form of an expectation value for all pool operators. This opens the door to strategies that group the observables into commuting sets, which may bring the cost down from $\mathcal{O}\left(\frac{N_{\text{orb}}^8}{\|g\|_\infty^2}\right)$ to $\mathcal{O}\left(\frac{N_{\text{orb}}^5}{\|g\|_\infty^2}\right)$ for molecular Hamiltonians [9]. In contrast, adding operators at other ansatz positions necessarily implies using parameter-shift rules, leading us to the scaling of Eq. (6.5). However, from our cost estimates in Eq. (6.6) and the discussion above, we can see that the asymptotic costs of our protocols would still match those of the original algorithm, or improve upon them by increasing $\|g\|_\infty^2$. Further, our protocols and cost analysis are not restricted to molecular Hamiltonians and can be readily applied to any instance of ADAPT-VQE.

6.6 Results

In this section, we compare the four protocols for inserting operators we proposed in Sec. 6.4. The protocols differ on how the operator and its position within the ansatz are chosen. This will allow us to obtain a more comprehensive picture of gradient trough detection and mitigation.

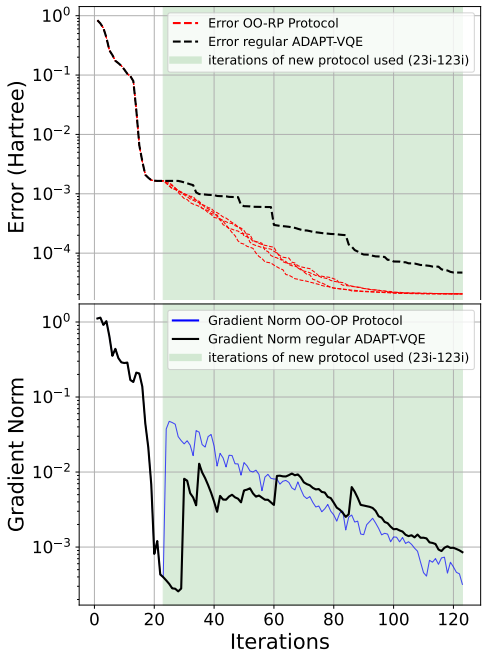
In order to quantitatively address our proposed protocols, we investigate their impact on the gradient trough we observed in Sec. 6.3. First, we will apply the new protocols for a fixed number of iterations, three iterations after the start of the gradient trough. Subsequently, an investigation will be conducted to ascertain whether the algorithm exits the trough or re-enters it.

Our numerical calculations focus exclusively on the linear H_6 molecule with an interatomic distance of 4 Å. This is a typical toy problem that serves as a proxy for strongly correlated systems and suffers from gradient troughs. The basis set is STO-3G, such that the test system has 12 qubits. In all cases, we take the error with respect to the FCI energy, which is exact within the chosen basis set. Regarding the operator pool, we used the problem-specific QE pool in all cases [182]. We allow the algorithm to run for as many iterations as allows us to observe the relevant features of a gradient trough.

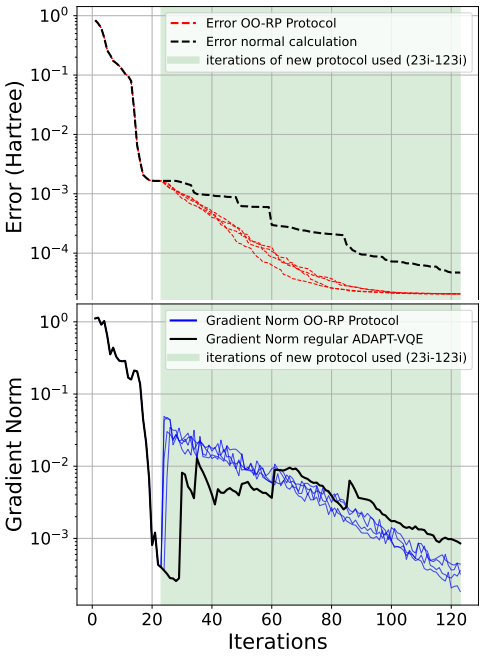
Fig. 6.5 shows the energy error as a function of the adaptive iterations for the various protocols (red lines) in comparison with standard ADAPT-VQE (black lines), where the operators are appended to the ansatz. The green regions in the plots indicate the iterations in which the new protocols have been deployed. The reason for this choice is due to the presence of a gradient trough starting at iteration 22 for standard ADAPT-VQE, which is applied outside the green regions (causing the black and red lines to coincide prior to the green region). In practice, the detection of a gradient trough at iteration 22 can be done using the ideas discussed in Sec. 6.3 to differentiate between gradient troughs and true convergence.

We clearly see the ability of our protocols to actively escape a gradient trough. As is evident in the figure, the OO/OP and the OO/RP protocols successfully escape gradient troughs, constantly pushing the optimization further towards the ground energy. RO/RP and RO/OP, on the other hand, are not as effective, resulting in a worse energy convergence than the canonical ADAPT-VQE.

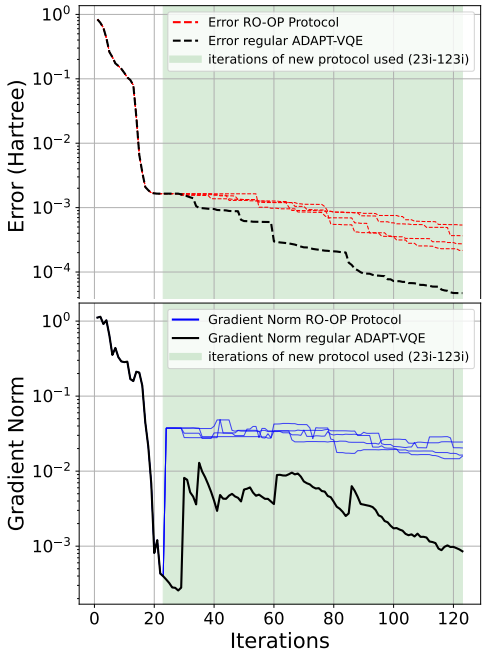
Although the best among the new protocols seem to inherently improve energy convergence, it is conceivable to only apply them in the vicinity of a gradient trough instead of throughout the entire optimization process to avoid the additional steps required. Figure 6.6 shows results for a simulation in which the new protocols are only applied for 30 iterations after the detection of a gradient trough. We plot both the energy error (top plots, red) and gradient norms (bottom plots, blue) as a function of the adaptive iterations. We can observe that the OO/OP and OO/RP schemes are able to escape troughs within the first iterations of being applied, as the norm increases rapidly with their application. After applying them, we observe a steady decrease in the gradient norm that does not present the defining features of a gradient trough. For the RO/RP and RO/OP, we observe a similar increase in the gradient norm, but the energy remains stalled. Generally, applying the proposed protocols throughout may allow us to decrease measurement costs by increasing the gradient norm (as discussed in Sec . 6.5). However, to determine if a consistent application of the protocols yields a significant advantage, a careful analysis of all costs is necessary. Such an analysis must consider all gradient measurements across all iterations, as well as all



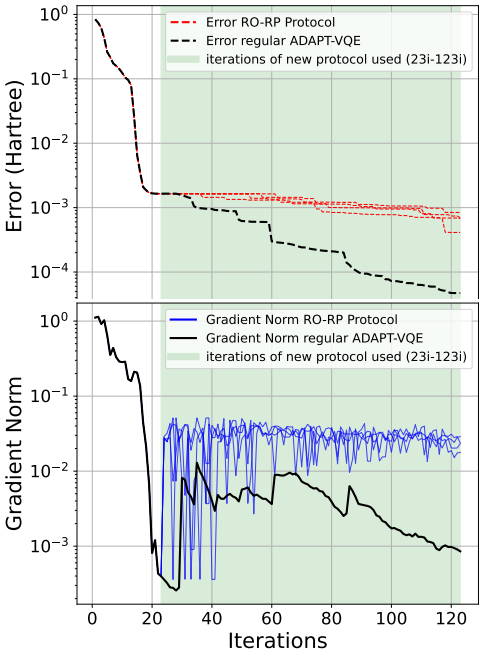
(a) OO/OP Protocol (optimized operator, optimized position)



(b) OO/RP Protocol (optimized operator, random position)



(c) RO/OP Protocol (random operator, optimized position)



(d) RO/RP Protocol (random operator, random position)

Figure 6.5: Results of applying the four new protocols to ADAPT-VQE. The protocols start being applied three iterations after the first occurrence of a gradient trough, ensuring the algorithm is within the trough. The simulation uses the typical ADAPT-VQE (appending protocol) until iteration 22. From iteration 23 on, the enhanced protocols are activated until the algorithm converges.

the measurements required by the optimization process. We leave this analysis for future work.

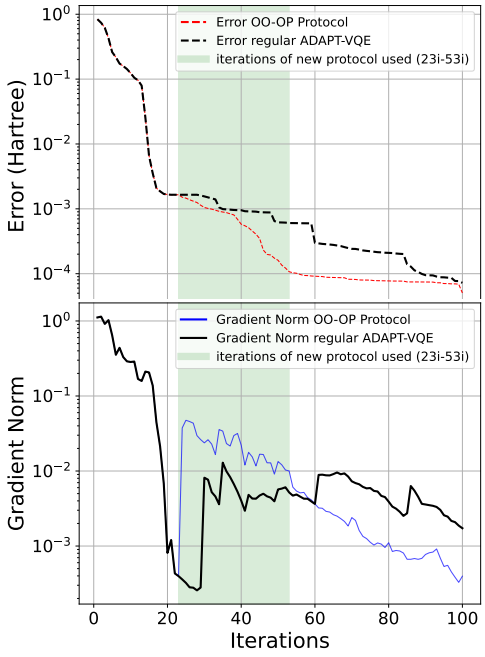
While Fig. 6.6 considers the application of our protocols for 30 iterations, we also carried out simulations applying them for 5 and 10 iterations after the detection of a trough. In these cases, ADAPT-VQE generally gets trapped again after switching back to appending for all new protocols. Only when the new protocols are implemented for 30 iterations or more is the gradient trough avoided in a sustainable manner. This is likely specific to the system we are studying here, and more investigation is required using other (molecular) systems to make a more general statement on the required iterations. That being said, the protocols we have developed in Sec. 6.3 can always be deployed to check if the number of iterations of the new protocols is insufficient to escape the gradient trough. If it is still in the trough, it continues with the new protocol; otherwise, it reverts to the appending/regular ADAPT-VQE protocol.

We expect the OO/OP and OO/RP protocols to converge more quickly than the ADAPT-VQE protocol. This has the potential for an even greater impact on larger molecular systems, because the calculation cost required to escape a trough increases rapidly as it becomes more difficult for a strict appending procedure to affect the total ansatz when only a few operators are added and the ansatz is comparably large. On the other hand, the RO/RP and RO/OP protocols converge slower than ADAPT-VQE because they eliminate the mathematically driven selection criterion.

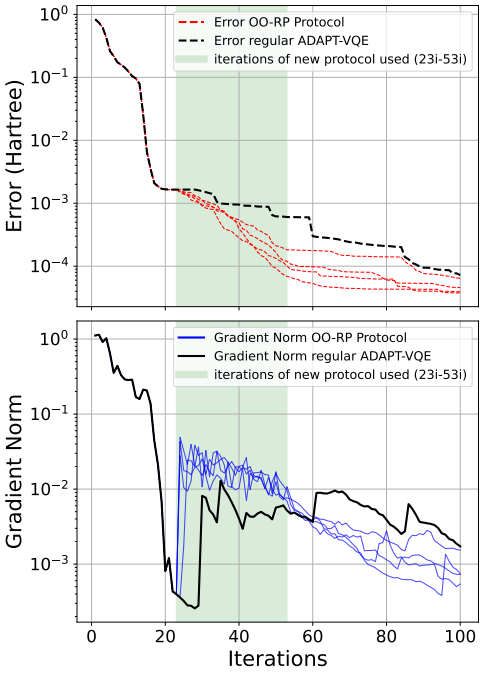
6.7 Discussion

In this chapter, we introduced and analyzed several modified operator selection/addition protocols for ADAPT-VQE aimed at escaping gradient troughs, a problem where the gradients with respect to all operators become simultaneously small and thus stall the ansatz improvement. Our results demonstrate that these troughs are limited to the particular ansatz position where operators are added in the original algorithm (i.e., end of the ansatz). The proposed protocols, which make use of the non-commutative properties of the exponentiated operators by introducing the possibility of adding operators in different positions where gradients are higher, significantly minimize the likelihood of getting trapped within these regions by allowing the protocol to escape the troughs. As an additional contribution, we introduce new criteria that allow ADAPT-VQE to systematically identify the occurrence of a gradient trough and distinguish it from converging, allowing one to avoid the problem of false convergence within a realistic setting. Overall, our findings show that exploring alternative positions for operator addition in ADAPT-VQE can restore its progress in situations where the original version of ADAPT-VQE stalls. This improves both robustness and practicality of ADAPT-VQE.

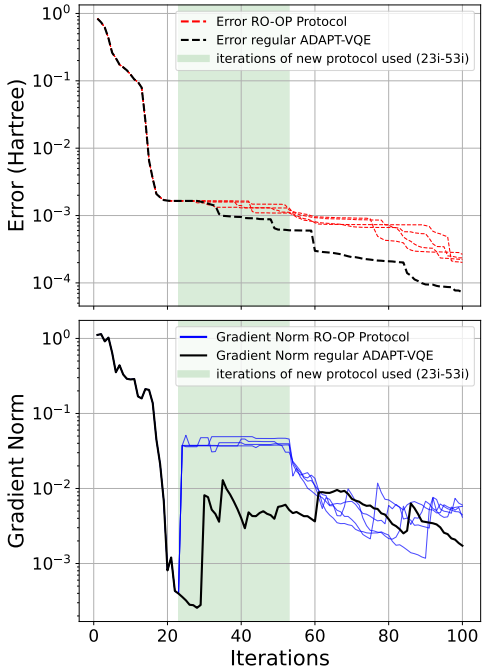
In addition to these advantages, our proposed protocols are able to decrease the measurement costs of ADAPT-VQE. Even though identifying and monitoring the existence of a gradient trough requires additional gradient measurements, we design our protocols such that this overhead is negligible with respect to the costs of the original algorithm. Further, by increasing the gradient norm, our protocols reduce the accuracy required in the gradient evaluation during the operator selection step, which effectively results in a reduction in measurement costs by a factor of 10^4 even for small (12-qubit) molecules. Additional



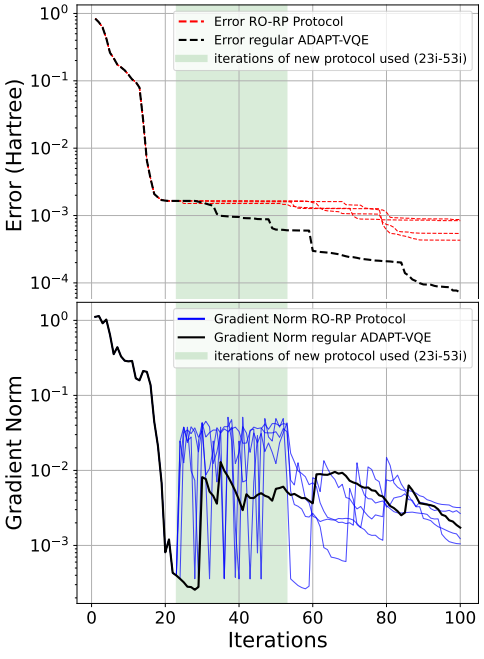
(a) OO/OP Protocol (optimized operator, optimized position) used from iteration 23 to iteration 53 for a linear H_6 molecule with 4Å interatomic distance.



(b) OO/RP Protocol (optimized operator, random position) used from iteration 23 to iteration 53.



(c) RO/OP Protocol (random operator, optimized position) used from iteration 23 to iteration 53.



(d) RO/RP Protocol (random operator, random position) used from iteration 23 to iteration 53 for a linear H_6 molecule with 4Å interatomic distance.

Figure 6.6: Results of applying each enhanced protocol for 30 iterations of ADAPT-VQE, from iteration 23 to iteration 53 (within the occurrence of a gradient trough). After these iterations, the algorithm continues using the normal appending protocol.

heuristics may help further reduce costs.

One potential explanation for the behavior of the algorithm with respect to the new protocols can be found in the Dynamical Lie Algebra (DLA) of the ansatz. The insertion of an operator into an ansatz position that differs from appending likely establishes a new, advantageous, optimization direction within a single iteration, as stated in Sec . 6.3. The regular ADAPT-VQE algorithm may require multiple iterations to achieve the same outcome through the BCH expansion of the ansatz with appending pool operators. Further numerical investigation is required to better understand the behavior of our protocols. We leave this for future work.

The simulations presented in this paper were carried out under idealized conditions. The performance of the algorithms may differ on real quantum computers with the presence of sampling noise, decoherence, and gate errors. We leave the study of measurement-efficient protocols and of the impact of noise for future work. Additionally, an interesting prospect for future research is developing a deeper structural understanding of when gradient troughs emerge and categorizing structural indicators for gradient troughs. This knowledge should allow us to predict when a trough emerges or enable the development of alternative ways of detecting the presence of a trough. Further, it seems promising to investigate an algorithm that combines our operator-position-based methodology with strategies to remove of irrelevant operators that have been proposed in the literature [140, 165]. While these methods are applied outside of gradient troughs, they might mitigate the gate overhead stemming from low-impact operators selected due to the stochastic criteria in the new algorithms. Finally, analyzing our protocols with different operator pools beyond the one based on QE may reveal deeper connections between pool operators, ansatz construction, and gradient geometry. This may tell us more about the fundamental workings of adaptive quantum variational algorithms.

7

Conclusions and Future Work

Quantum simulation is more than a promising application of quantum computing—it is the reason why quantum computing came to be. Given the limitations of classical memory and processing in simulating quantum mechanical systems, which in general require a number of bits and a runtime that grow exponentially with the size of the system, using quantum mechanical systems *as computational resources* is a promising option.

While many quantum algorithms have been proposed for the task of quantum simulation, most of them are beyond the capabilities of current quantum devices: Quantum information is fragile, and protecting it from noise and decoherence to build a fault-tolerant quantum computer is a research field in itself. The initial proposals for quantum algorithms assumed they were to be implemented on ideal machines, but despite significant progress, this is far from the case today. Hence, the qubit counts and circuit depths of these early proposals belong to a future regime.

While the development of quantum hardware and error correction protocols leading up to such a regime is underway, we can seek practical quantum advantage by designing algorithms that can utilize noisy quantum devices. In this context, ADAPT-VQE was proposed: An algorithm that builds an ansatz from scratch by adding blocks using a system-driven selection criterion. This strategy has been shown to lead to improved trainability and reduced circuit depths as compared to static alternatives, improving scalability and bringing us a step closer to practical demonstrations.

However, ADAPT-VQE itself faces some challenges. The measurement cost overhead incurred by the operator selection step has been pointed out as a major shortcoming. Furthermore, the circuit depths remain beyond what can be reliably executed in current quantum devices. Finally, the algorithm has its own convergence problem: the phenomenon of 'gradient troughs', which can induce early and wrongful convergence and exacerbate measurement costs.

In this thesis, we propose various improvements to ADAPT-VQE that address all of these challenges: (1) We develop a quasi-Newton optimizer that takes the dynamic circuit construction and parameter warm-starts into account by recycling second derivatives from one adaptive iteration to the next, reducing measurement costs; (2) We create a novel operator pool comprised of CEOs that allows for particularly efficient circuit implementations, reducing gate counts and depths; (3) We propose Co-ADAPT-VQE, a co-designed variant of the algorithm that embeds hardware details into the selection criterion, resulting in shallower

and more noise-resilient circuits; and (4) We outline strategies to detect and mitigate gradient troughs, reducing measurement costs and preventing wrongful convergence.

Our improvements to the algorithm result in the ADAPT-VQE variant with the lowest measurement costs and shallowest circuits. Further, they motivate the development of ADAPT-VQE-tailored optimization methods and the option of exploring the addition of operators to the ansatz in alternative positions (as opposed to always appending them), both strategies being novel in the literature.

Additionally, our results benchmark the state-of-the-art of ADAPT-VQE against the early version of the algorithms and static ansätze, allowing us to conclude that not only has the algorithm improved dramatically since its inception, it has also become competitive with the best static ansätze in all metrics—including measurement costs, in spite of the common belief that the adaptive ansatz construction inevitably incurs a measurement overhead.

While this thesis highlights the progress of the research community in bringing ADAPT-VQE to a state of experimental viability, additional research efforts will be necessary to reach that point. Interesting venues for future work include exploring other types of ADAPT-VQE-tailored optimizers, developing efficient operator pools tailored to other areas of application and/or suited for different quantum computing architectures, working towards experimental demonstrations, and benchmarking algorithm variants under noise, among others.

Currently, the main challenge faced by not only ADAPT-VQE but VQEs in general resides in the impractically high measurement costs necessary to obtain chemically accurate energy evaluations. Albeit polynomially scaling, these remain too high in light of the time and cost of executing the target circuits in today's quantum computers. Hence, we believe that strategies to reduce measurement costs will be particularly impactful going forward.

A

Evaluating Gradients on Quantum Hardware

The gradient function ∇f that must be supplied to gradient-based optimization methods in Ch. 3 merits a discussion, since it is not evident that this information is available when the cost function is evaluated on a quantum computer.

We note that some implementations of BFGS (including SciPy's [167]) allow the user to bypass the construction of a gradient function by implementing it internally via Finite Difference (FD) methods. These numerical methods approximate derivatives from evaluations of the function at points which differ by small shifts, and can be applied to any function in a black-box fashion [74]. However, when the cost function is the expectation value of a (generic) quantum mechanical observable, it must be obtained by averaging over a finite number of samples, and thus it is inevitably noisy. Additionally, we have hardware limitations inherent to NISQ computers, including miscalibrated rotation gates and other sources of noise. All of this hampers the task of gauging minute shifts in the function value, and the fact that the finite difference quotient must have a small denominator (because its magnitude is related to the magnitude of the approximation error) exacerbates these problems. As would be expected, BFGS with FD methods has been found to perform poorly in the presence of noise [94].

A realistic alternative to FD methods is Parameter-Shift Rules (PSRs), proposed in Ref. [113] and extended in Ref. [150]. PSRs use a clever manipulation of analytical expressions to express the derivatives as linear combinations of measurable expectation values. Unlike FD formulas, which are generic numerical approximations, PSRs are analytical derivatives with a circuit-specific structure. When the qubit pool is used, the generators of the ansatz elements consist of a single Pauli string, whose gradient can be obtained from two energy measurements using the simplest PSRs. When the QE pool is used, more sophisticated techniques are required because each of the generators has three eigenvalues. In this case, the gradients can be measured using the fermionic PSRs proposed in Ref. [91]. For real wave functions (as we are concerned with), this will similarly imply the measurement of two expectation values, with the corresponding circuits having a negligible increase in circuit depth with respect to the energy measurement circuits.

B

BFGS Optimizer

B.1 Line Search Subroutine

Algorithm 2 in the main text requires a `line_search` subroutine to choose the next iterate. The task of this algorithm is to find the step size α which minimizes the function f along the search direction p_k,

$$\min_{\alpha>0} f(x_k + \alpha p_k). \tag{B.1}$$

We now briefly describe this subroutine.

Because finding the minimum with high accuracy might be unnecessarily costly, this minimization is often approximate and terminates when some reasonable conditions are met. A common choice is the Wolfe conditions,

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f(x_k)^T p_k, \tag{B.2}$$

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f(x_k)^T p_k, \tag{B.3}$$

where $0 < c_1 < c_2 < 1$. Example values for quasi-Newton methods are $c_1 = 10^{-4}$, $c_2 = 0.9$ [125]. Note that if p_k is a descent direction, as should be, then $\nabla f(x_k)^T p_k$ is negative.

Condition (B.2), known as the *sufficient decrease* or *Armijo* condition, asserts that the decrease in the value of f is lower (and so higher in magnitude) than f's instantaneous rate of change along p_k at the initial point, weighed by c_1. This condition is evidently satisfied for small enough α_k, even if the decrease in the function value is negligible.

Condition (B.3), known as the *curvature* condition, asserts that the derivative (along p_k) of the function at the new point is higher (and so lower in magnitude) than the one at the initial point, weighed by c_2. This bound is placed because the higher the magnitude of the rate of change of f, the higher the decrease in f we expect from further refining the step α_k.

Another important choice in a line search algorithm is how to initialize and vary α. One option is to start with a guess and increase it until we find a point that either satisfies the desired conditions or brackets points that do. In the latter case, we can backtrack, decreasing α until a valid point is found.

The cost of this subroutine depends on how many attempts it takes us to find an α that satisfies Wolfe's conditions ((B.2), (B.3)), which depends on the shape of the cost function and the proximity to a

minimum. Each iteration of the line search requires measuring the cost function and the gradient vector, which imply respectively 1 and $2n$ energy evaluations for an n-dimensional ADAPT-VQE optimization.

B.2 BFGS Update Rule

Another subroutine required by algorithm 2 is `update_h`, which updates the inverse Hessian H_k at the end of a line search. This update depends on the vectors s_k and y_k, obtained from the difference between the coefficient and gradient vectors, respectively, at the kth and $(k-1)$th iterations (see steps 8 and 9 of Alg. 2).

The BFGS update rule is obtained by enforcing three conditions:

1. The gradient of the new quadratic model for f (defined by a second-order Trotter expansion) matches the true gradient at the last two iterates.
2. H_{k+1} is symmetric and positive definite.
3. Among all matrices satisfying the above, H_{k+1} is the one which minimizes the distance to H_k, with respect to some norm.

These three conditions give rise to the BFGS update rule

$$H_{k+1} = (I - \rho_k s_k y_k^T) H_k (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T, \tag{B.4}$$

where $\rho_k = \frac{1}{y_k^T s_k}$. Conveniently, the BFGS method deals directly with the *inverse* Hessian, required to decide the next search direction via Eq. 3.5 of the main text. Other quasi-Newton formulas, such as DFP (proposed by Davidon in 1959 [40] and further analyzed by Fletcher and Powell [57]), handle the inversion posteriorly.

We note that the approximate inverse Hessian H (and the induced Hessian H^{-1}) will be positive definite even if the same is not true for the real Hessian (and its inverse).

B.3 Properties of BFGS

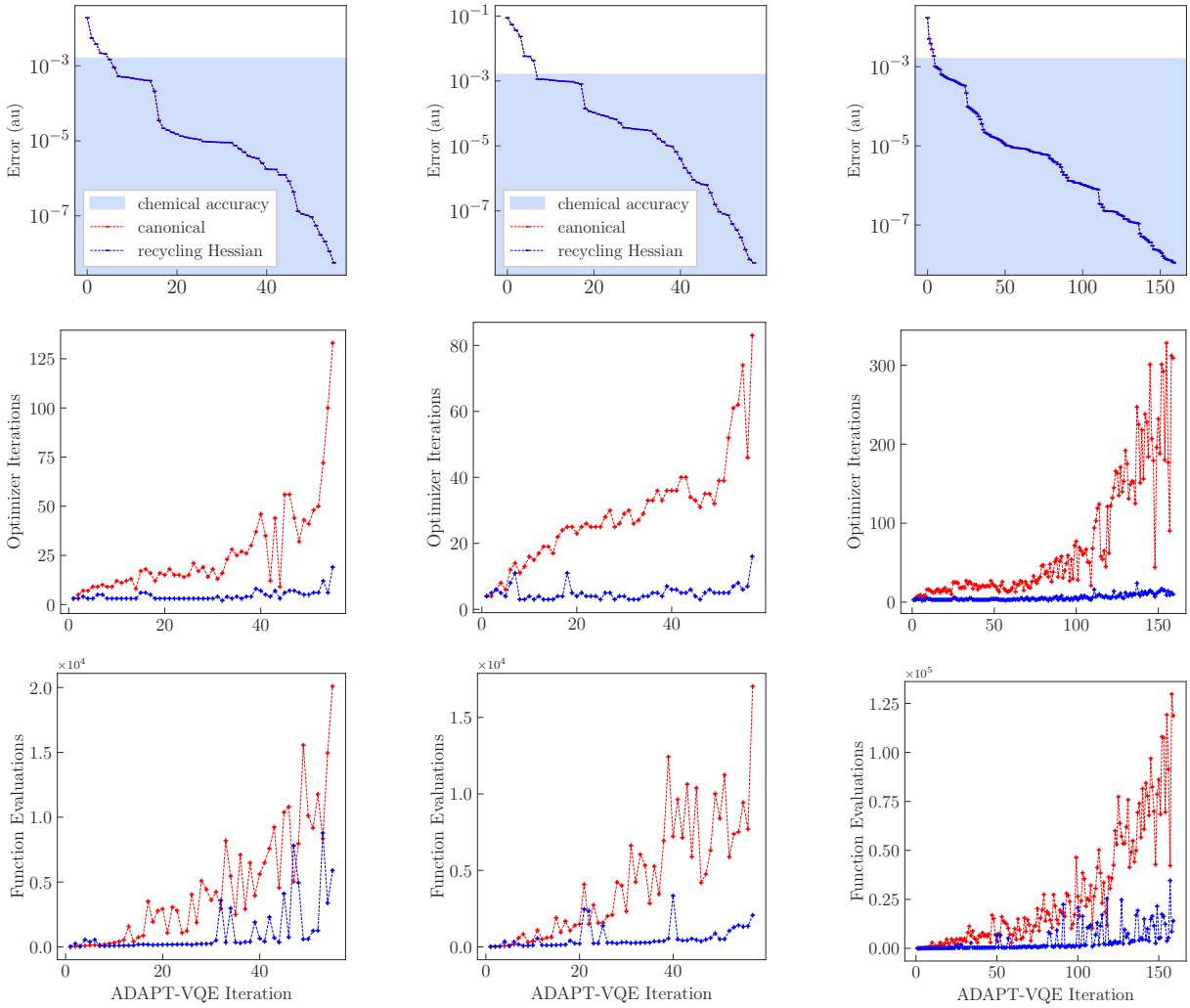
BFGS is a robust optimizer with good convergence [136] and self-correcting properties [124]. Despite never directly evaluating second-order derivatives, it enjoys a super-linear convergence rate and performs well in practice, reaching a minimum sufficiently fast for most practical purposes [85, 125]. Due to these features, this optimizer has become a popular choice for VQAs and is often employed in practical implementations [77, 103, 72, 182, 108].

It has been proved that when implemented with Wolfe line searches (see App.B.1), this optimizer is globally convergent for convex functions, i.e., iterates will converge to a minimum regardless of the initial point [136]. In the context of numerical optimization, *global* is typically used to stress the independence of the convergence on the initial point [158]. Accordingly, local convergence properties are those that only

hold when the initial point is close enough to a minimum. We note that in this context, the words ‘local’ and ‘global’ do not refer to the type of minimum. Globally convergent methods may converge to a local minimum and vice versa.

C

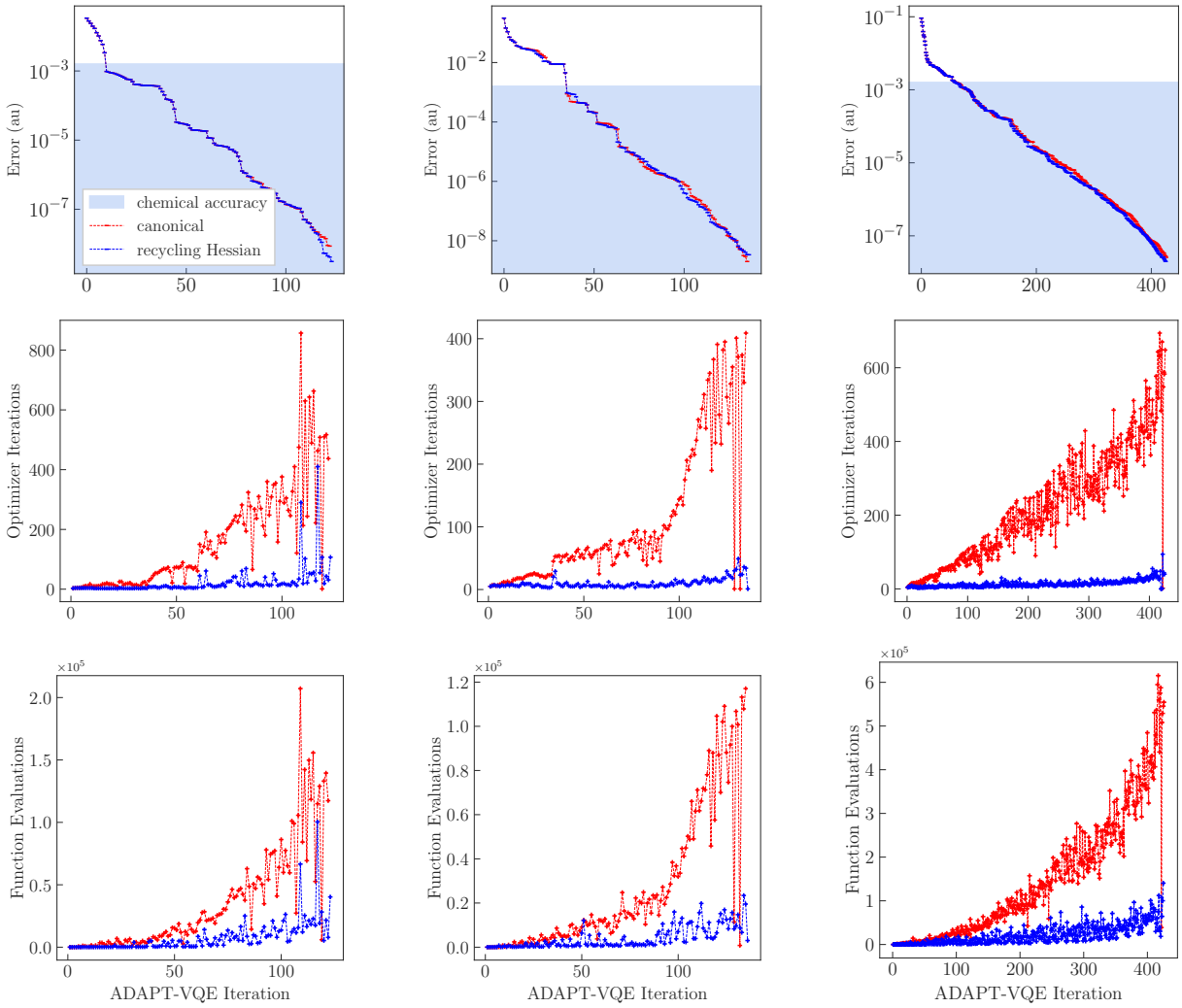
Hessian Recycling for Other Molecules



(a) QE pool, 1.5Å (b) QE pool, 3Å (c) Qubit pool, 1Å

Figure C.1: Equivalent plots to Fig. 3.1 for LiH.

In Ch. 3, we showed that our Hessian recycling protocol results in a significant decrease in the measurement costs of ADAPT-VQE for H_6, and that this decrease tends to become more relevant as the system



(a) QE pool, 1.3Å (b) QE pool, 3Å (c) Qubit pool, 2Å

Figure C.2: Equivalent plots to Fig. 3.1 for BeH_2.

size increases. In this appendix, we show that these conclusions generalize to other molecules.

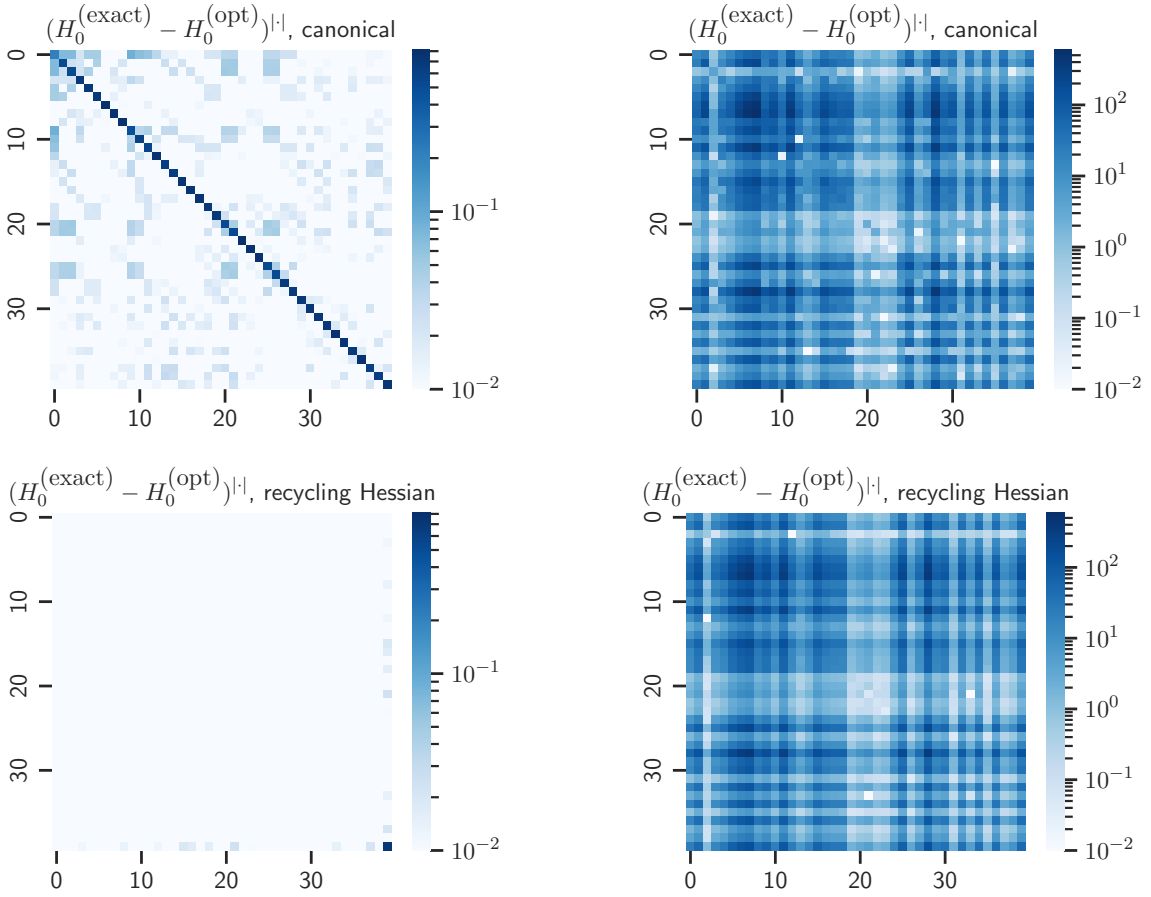
Figures C.1 and C.2 show the evolution of error, line searches, and measurement costs as the dimension of the ADAPT-VQE optimization grows. The molecules considered are respectively LiH and BeH_2. Once again, we consider both equilibrium and stretched geometries, as well as two different pools.

We confirm that recycling the Hessian significantly reduces total measurement costs for all systems and pools.

D

Hessian Heatmaps for Other Iterations

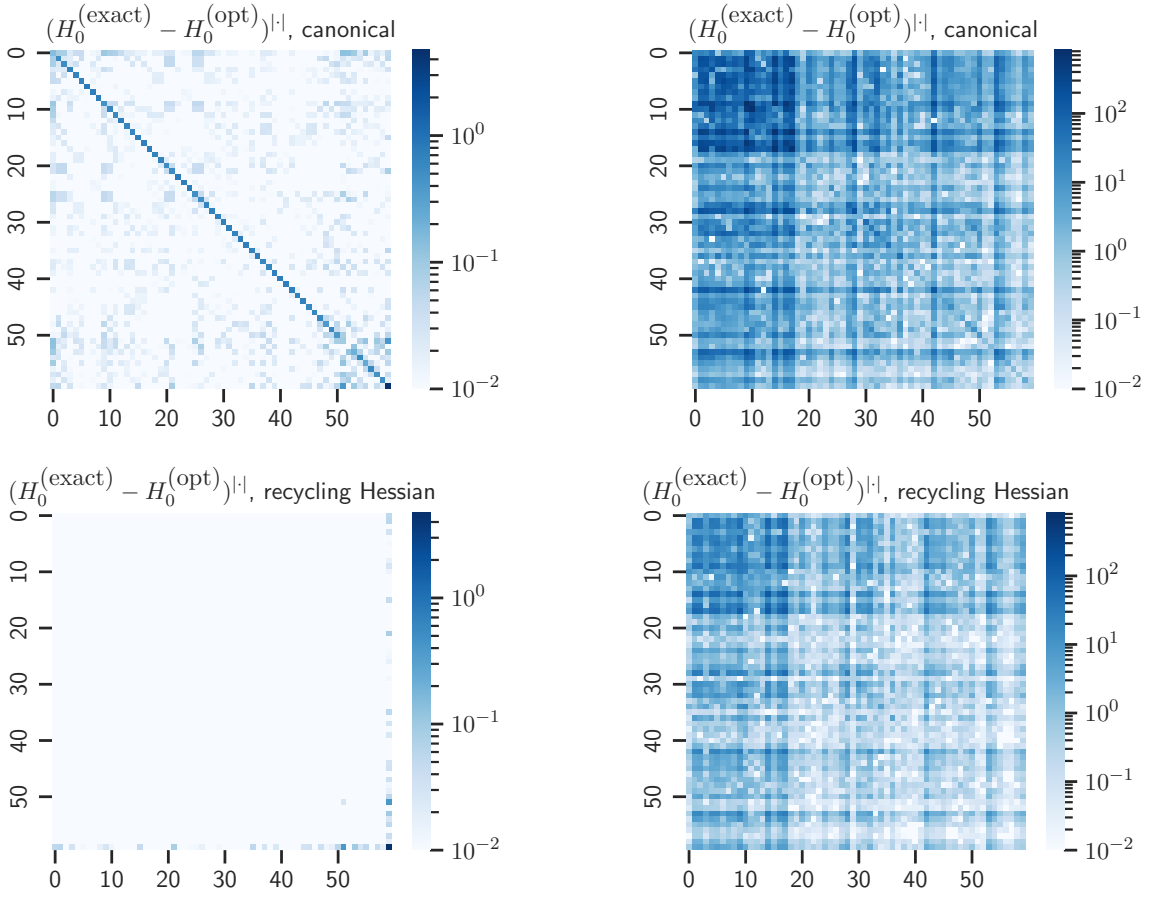
In order to confirm that the behavior showcased in Fig. 3.2 of Ch. 3 generalizes to other iterations, we consider equivalent heatmaps for other iterations. Figures D.1 and D.2 consider iterations 40 and 60, respectively.



(a) H_6 at 1Å

(b) H_6 at 3Å

Figure D.1: Heatmaps showing the difference between the initial approximate inverse Hessian in the optimization process and the initial exact inverse Hessian, with (bottom row) and without (top row) Hessian recycling, for the 40th iteration of QEB-ADAPT-VQE. The plots show the element-wise difference between these two matrices. We consider H_6 at equilibrium and stretched geometries.



(a) H_6 at 1Å (b) H_6 at 3Å

Figure D.2: The equivalent of Fig. D.1 for the 60th optimization.

As happened in the examples provided in the main text, we observe that the entries of the approximate Hessian are closer to the exact ones when we recycle the Hessian. While this happens for all test cases, once again we see that the impact of our protocol is more significant at the equilibrium geometry. Moreover, as happened in iteration 50, we can observe that the entries farther away from the true value are the diagonal ones for such a configuration, but that is no longer the case for the larger bond distance. The reasons behind these results were discussed in Sec. 3.5.2.

E

Evolution of the 50th QEB-ADAPT-VQE Iteration for H6

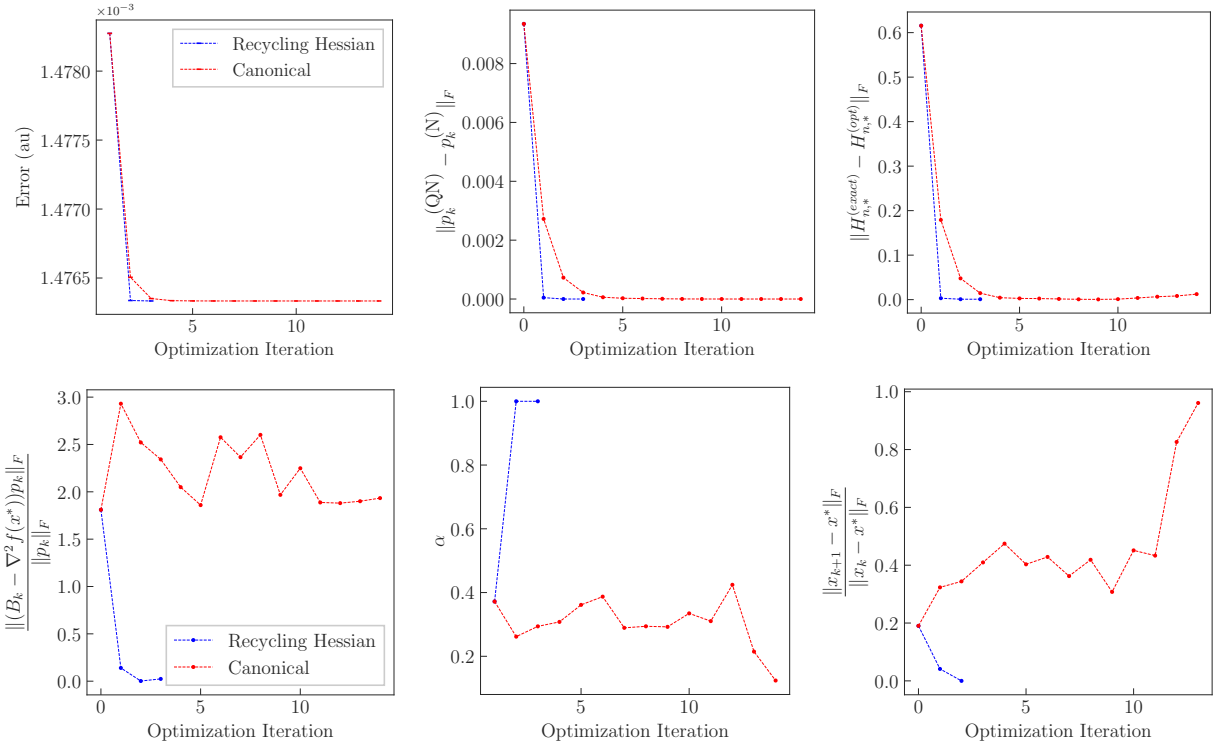


Figure E.1: Relevant quantities for the study of the convergence rate of the 50th QEB-ADAPT-VQE optimization for H_6 at 1 Å.

In this appendix, we analyze a different optimization of QEB-ADAPT-VQE for H_6. Our purpose is to confirm that the results observed and discussed concerning the 75th optimization in Sec. 3.5.3 of Ch. 3 were not fortuitous.

Figure E.1 contains the same data depicted in Figs. 3.5 and 3.6, concerning H_6 at 1 Å; however, we now focus on the 50th iteration as opposed to the 75th one.

We confirm that the evolution of the optimization is similar. Recycling the Hessian speeds up the convergence of the energy, improves the alignment with Newton's direction, and brings the approximate

Hessian closer to the exact one. Additionally, recycling the Hessian allows us to gather all conditions necessary for superlinear convergence, which we verify occurs in the last plot.

F

Variants of CEO-ADAPT-VQE

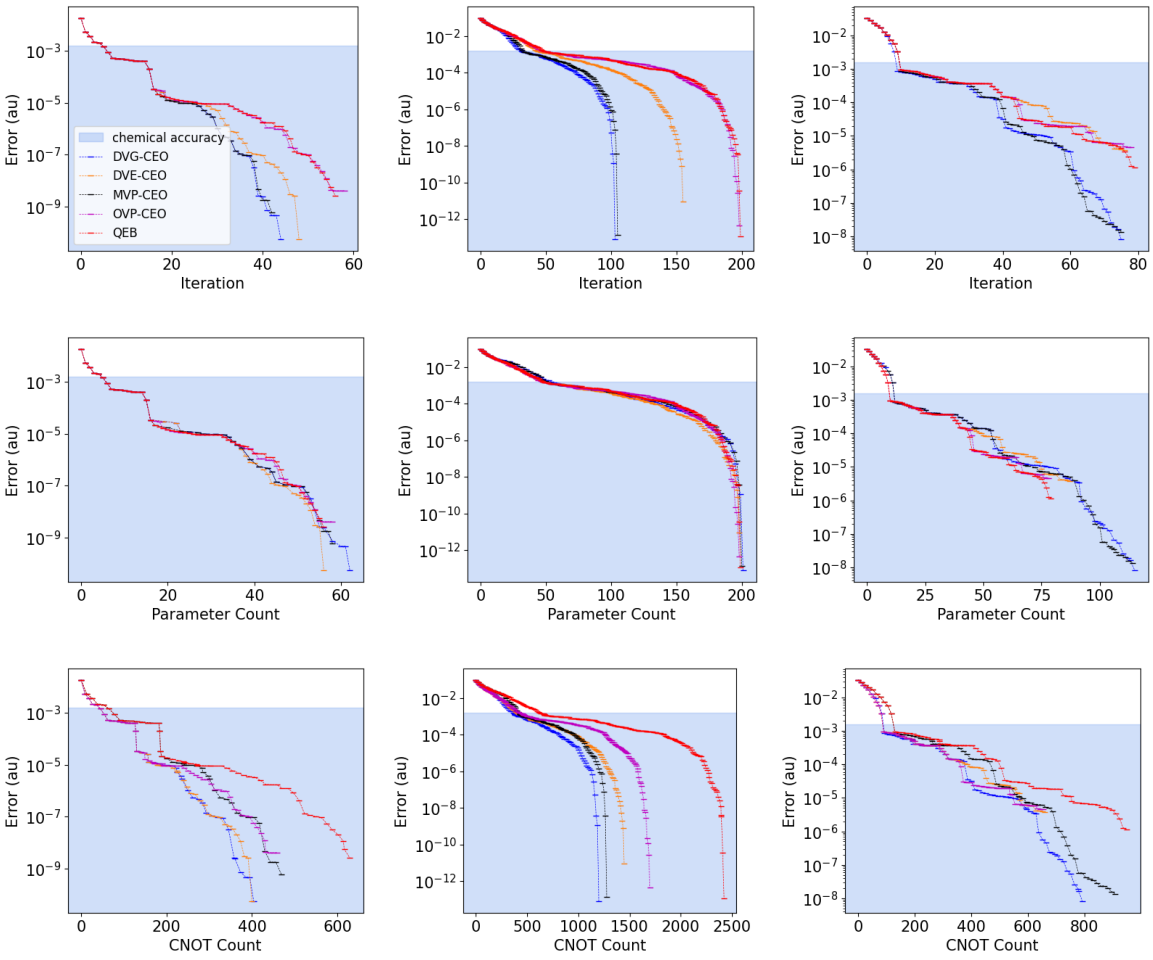
CEO-ADAPT-VQE can be defined as a variant of ADAPT-VQE that uses a pool comprised of CEOs. In Ch. 4, we defined two types of such operators: OVP- and MVP-CEOs, with one or up to three variational parameters respectively. One key decision in the algorithm is how to choose between these subsets in each iteration. In this section, we compare four different decision criteria.

As explained in the main text, in each iteration of our algorithm we use the gradients of OVP-CEOs to select two candidates: $T_n^{(OVP-CEO)}$, the OVP-CEO with the highest gradient, and $T_n^{(MVP-CEO)}$, the MVP-CEO formed from all QEs with nonzero gradients acting on the same spin-orbitals as $T_n^{(OVP-CEO)}$. One of these operators must then be chosen to generate the new ansatz unitary.

We note that often, the gradient of $T_n^{(OVP-CEO)}$ is the same as the sum of the gradient magnitudes of the excitations in $T_n^{(MVP-CEO)}$. This is only not the case when the spin-orbitals are all of the same type, in which case the gradient of $T_n^{(MVP-CEO)}$ may be higher by virtue of including one more operator. Hence, we cannot, in general, use the gradient to decide between the two operators. One might expect that OVP-CEOs favor hardware-efficiency, since the corresponding evolutions are implemented with 9 `CNOT`s while MVP-CEOs are implemented with 13. However, the latter offer more variational freedom by implementing up to three independent qubit excitations each.

We use the notation of the main text to define four variants of CEO-ADAPT-VQE that serve the purpose of assessing different criteria for this decision. The gradient measurement step is left unchanged with respect to the one defined in the main text. In what concerns the operator selection step, the definitions of $T_n^{(MVP-CEO)}$ and $T_n^{(OVP-CEO)}$ remain the same, but the criteria of choice between the two vary across variants as indicated below.

- OVP-CEO-ADAPT-VQE: Add $e^{T_n^{(OVP-CEO)}}$ to the ansatz.
- MVP-CEO-ADAPT-VQE: Add $e^{T_n^{(MVP-CEO)}}$ to the ansatz.
- Decision via Gradient (DVG)-CEO-ADAPT-VQE: Add $e^{T_n^{(OVP-CEO)}}$ to the ansatz if $\#M_{\neq 0}^{(QE)}(T_n^{(OVP-CEO)}) = 1$. Otherwise, add $e^{T_n^{(MVP-CEO)}}$.
- Decision via Energy (DVE)-CEO-ADAPT-VQE: Add $e^{T_n^{(OVP-CEO)}}$ to the ansatz if $\#M_{\neq 0}^{(QE)}(T_n^{(OVP-CEO)}) = 1$. Otherwise, obtain ΔE_{OVP} and ΔE_{MVP}, the energy changes produced by adding to the ansatz



(a) LiH at 1.5Å (b) Linear H_6 at 1Å (c) BeH_2 at 1.3Å

Figure F.1: Comparison of CEO-ADAPT-VQE variants at equilibrium bond distances. The subfigures depict the convergence of different variants of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at bond distances close to equilibrium. The QEB-ADAPT-VQE algorithm is also included for reference. The error is plotted against the iteration number (top), parameter count (middle), and CNOT count (bottom). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively.

$e^{T_n^{(OVP-CEO)}}$ and $e^{T_n^{(MVP-CEO)}}$ (respectively) and performing a full optimization. Add the former unitary if $\frac{\Delta E_{MVP}}{13} > \frac{\Delta E_{OVP}}{9}$, and the latter otherwise [1].

In summary, we have two algorithms that use OVP-CEOs or MVP-CEOs exclusively, and two algorithms that combine the two operator types. DVG-CEO-ADAPT-VQE is the algorithm defined in the main text, where the decision is gradient-based: $T_n^{(MVP-CEO)}$ is chosen unless there is only one operator with a nonzero gradient. In DVE-CEO-ADAPT-VQE, the decision is energy-based: We perform two independent optimizations of the ansatz with each of the two candidate unitaries, and effectively select the one which leads to the highest absolute value of the energy change per unit CNOT. The aim is to maximize the impact on the energy with respect to the added CNOT count in each iteration. Naturally, we could consider other

[1] Note that this energy change is expected to be negative.

hardware-related criteria, such as the CNOT depth. Note that this decision criterion roughly doubles the number of optimizations required per ADAPT-VQE iteration.

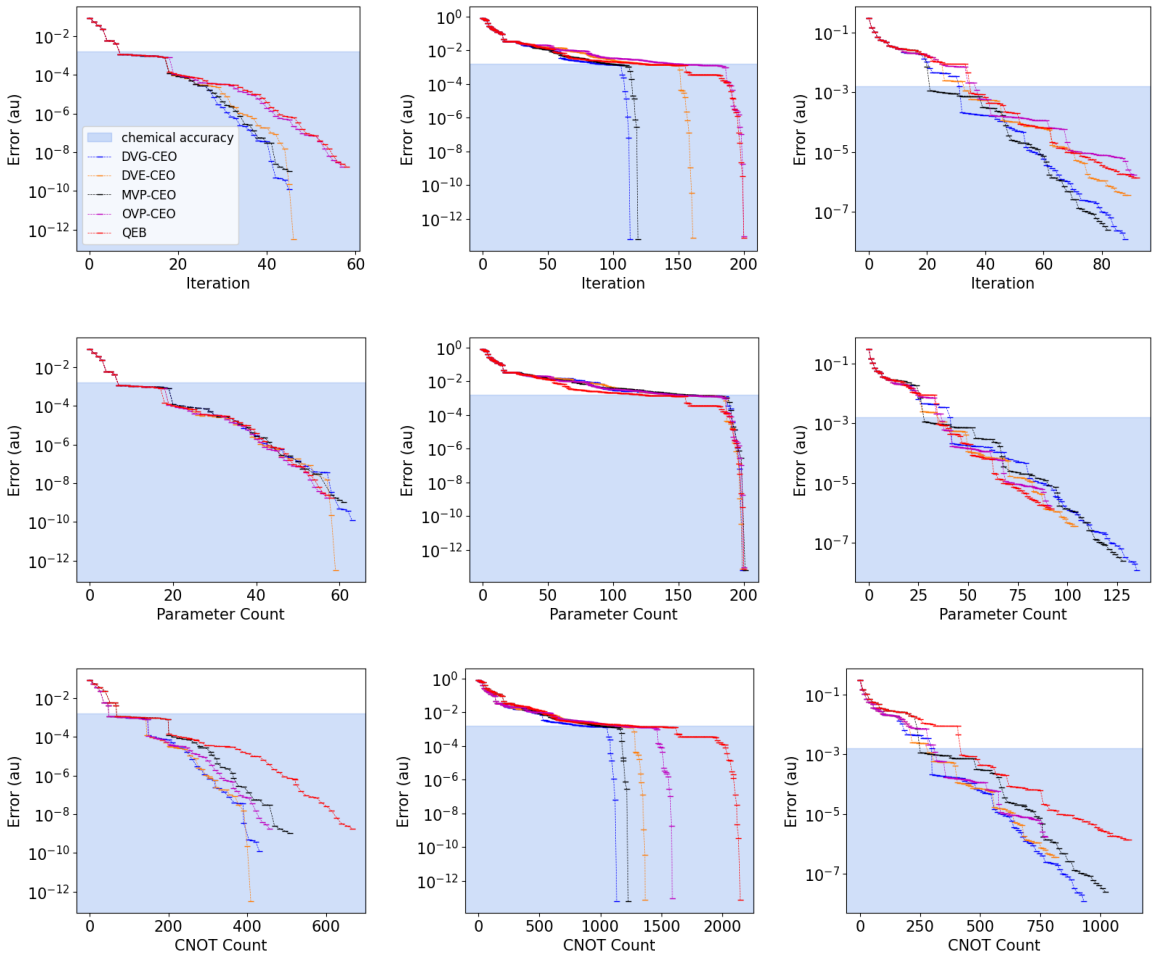
Figure F.1 compares these four algorithms. QEB-ADAPT-VQE is also plotted for reference. We verify that all variants of CEO-ADAPT-VQE improve upon QEB-ADAPT-VQE in terms of the number of CNOTs required for a given error, while being roughly matched in terms of the parameter count.

The decision via gradient performs the best among all variants of CEO-ADAPT-VQE, outperforming all others for all systems. This is remarkable, given the fact that the decision via energy is locally optimal (with respect to the CNOT count). In fact, in the first iteration where DVG- and DVE-CEO-ADAPT-VQE diverge, the latter is sure to have a lower error, because the optimization of the former is essentially a restricted version of the optimization of the latter. Yet, as the iterations proceed, the DVE variant lags behind. This can only be explained as a nonlocal effect. We hypothesize that the fact that DVG-CEO-ADAPT-VQE always includes all independently parameterized QEs with nonzero gradients leads to a higher variational flexibility that becomes more advantageous as the iterations proceed. Certain operators may have a low impact on the energy when added, but prove beneficial later on — perhaps they introduce important Slater determinants into the superposition state, or they interact favorably with operators added after them.

In general, the dynamic strategies (DVG and DVE), which decide between CEO types on a case-by-case basis, seem to outperform the predetermined ones. This is expected, since the latter make no effort to optimize the choice of operator type. However, there is a notable exception: MVP- beats DVE-CEO-ADAPT-VQE for H_6. Once again, this suggests that privileging extra variational freedom can be rewarding in the long term, particularly for highly correlated systems, which strengthens the conjecture we made in the preceding paragraph.

Figure F.2 shows similar plots for stretched bond distances. We observe the same trends: QEB-ADAPT-VQE is outperformed by all variants of CEO-ADAPT-VQE, with the leading one being DVG. The parameter count is roughly equivalent for all five algorithms.

It has become evident that the decision via gradient is the optimal choice in terms of the CNOT count and number of iterations, despite requiring no extra optimizations as compared to the canonical ADAPT-VQE (unlike the decision via energy). Therefore, we take DVG-CEO-ADAPT-VQE as the standard ADAPT-VQE algorithm with CEOs and refer to it simply as CEO-ADAPT-VQE in the main text.



(a) LiH at 3Å (b) Linear H_6 at 3Å (c) BeH_2 at 3Å

Figure F.2: Comparison of CEO-ADAPT-VQE variants at stretched bond distances. The subfigures depict the convergence of different variants of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at stretched bond distances. The QEB-ADAPT-VQE algorithm is also included for reference. The error is plotted against the iteration number (top), parameter count (middle), and CNOT count (bottom). The convergence criterion is the same as in Fig. F.1.

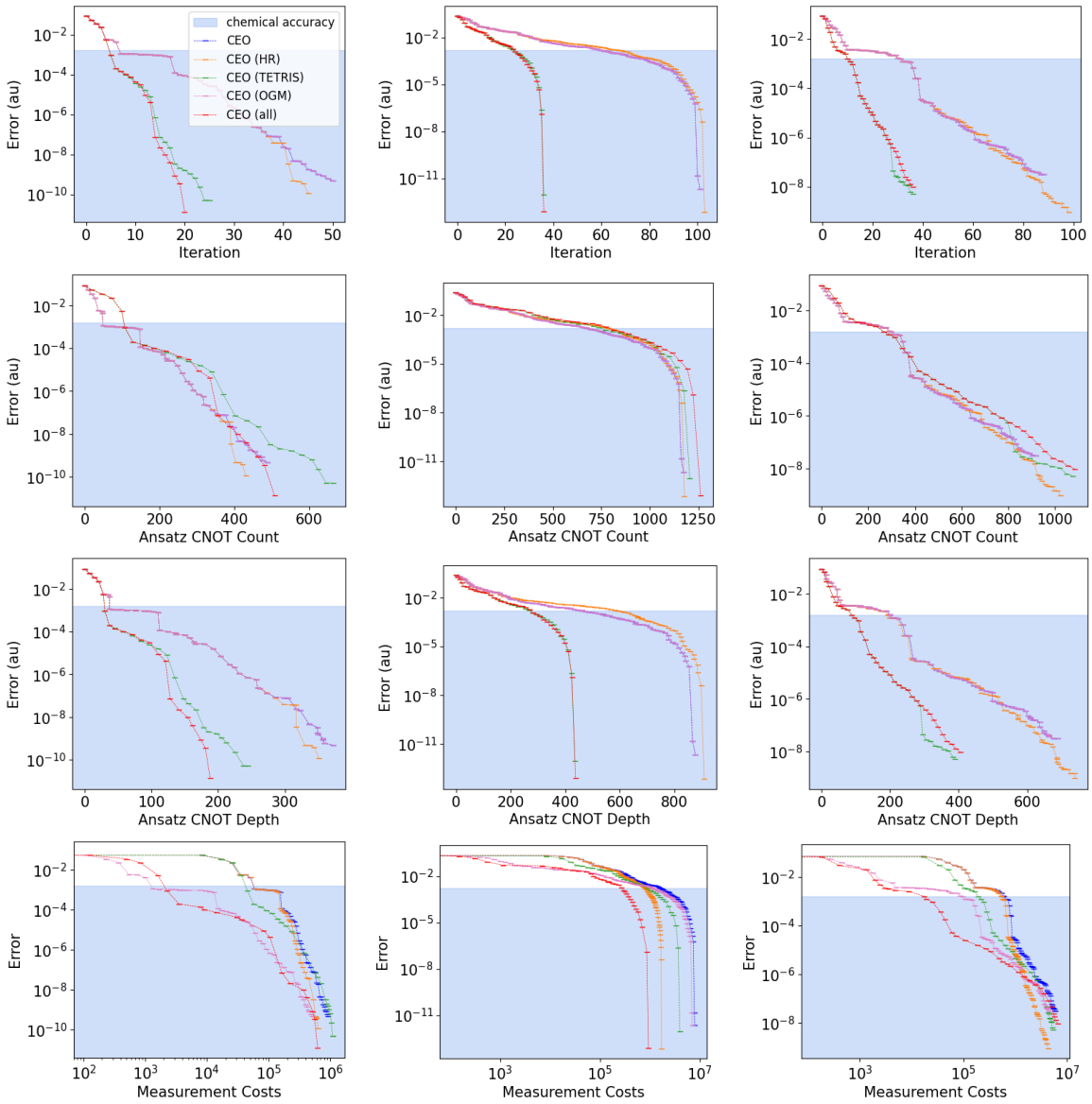
G

Combining CEO-ADAPT-VQE with Transversal Proposals

In this section, we investigate the individual impact of each of the strategies consolidated in Ch. 4 to create the CEO-ADAPT-VQE* algorithm: TETRIS [10], OGM [9] and HR (Ch. 3. We refer to the main text for details about the methods.

Figure G.1 compares the convergence of CEO-ADAPT-VQE with no improvements, with all improvements, and with each improvement individually. The uppermost panels show that, as would be expected, only TETRIS relevantly affects the iteration count. The remaining strategies affect the total number of measurements per iteration, but leave the iteration count unchanged. The CNOT count (second row of panels) is similar for all curves, while the CNOT depth (third row) is predictably lowered by TETRIS while being roughly unchanged by the the other proposals. Finally, the last row of panels shows that OGM and HR contribute to decreasing the measurement costs of the algorithm. Interestingly, the relative impact of the two is system-dependent. While OGM has a greater impact than HR for LiH and, in earlier iterations, for BeH_2, the reverse happens for H_6 and BeH_2 in later iterations. We attribute this to the complexity of the optimizations. As an example, in the case of H_6, not only are the optimizations higher-dimensional on average, but they also tend to require more cost function evaluations than optimizations for other molecules (even for matched parameter counts). As such, in this case, the cost of the energy measurements throughout the optimizations prevails over the cost of the gradient measurements, such that a strategy which tackles the cost of the optimization (HR) is more beneficial than one which tackles the cost of the gradient measurement round (OGM). We note that while its focus is decreasing circuit depth, TETRIS also offers a slight reduction in measurement costs by virtue of requiring a lower number of iterations (and thus fewer gradient measurement rounds and fewer optimizations in total).

Finally, we remark that the benefits of HR are expected to take more iterations to be harvested when we employ strategies that increase the number of new variational parameters per iteration, such as MVP-CEOs and TETRIS, due to the higher number of cold-started entries in the inverse Hessian. In the case of the 12-qubit molecules, TETRIS-CEO-ADAPT-VQE can add up to three MVPs-CEOs per iteration. Up to two of them may act on spatial orbitals of the same type, and thus have 3 variational parameters. In total, this may lead to up to 8 variational parameters, and the count will be higher for larger molecules. In



(a) LiH at 3Å (b) Linear H_6 at 1.5Å (c) BeH_2 at 2Å

Figure G.1: Combining CEO-ADAPT-VQE with transversal improvements. The subfigures depict the convergence of the CEO-ADAPT-VQE algorithm for **a** LiH, **b** H_6, **c** BeH_2, at various bond distances. The algorithm is implemented *in tandem* with other recent proposals: HR (Ch 3), TETRIS [10], OGM [9], and all three (all). The baseline case, which uses the CEO pool while following the original ADAPT-VQE protocol [72] with a vanilla measurement strategy, is also included for reference. The error is plotted against the iteration number, CNOT count, CNOT depth, and measurement costs. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively. The curves for CEO and CEO (OGM) overlap on all plots except those pertaining to measurement costs.

early iterations, a large number of new parameters represents a significant number of second derivatives about which the recycled Hessian contains no information. Therefore, we expect the impact of HR to be more significant for larger and more strongly correlated systems, where the number of old parameters far outweighs the number of freshly added parameters in later iterations.

H

Orbital Optimization

In Ch. 4, we combined CEO-ADAPT-VQE with compatible improvements in the literature (Sec. 4.4.3). In this appendix, we consider an additional improvement proposed by Ref. [54]: ADAPT-VQE-Self-Consistent Field (SCF), an algorithm that merges orbital optimization techniques with ADAPT-VQE.

Given a variational state $|\psi(\boldsymbol{\theta})\rangle$, we can obtain an orbital-optimized version $\left|\tilde{\psi}(\boldsymbol{\kappa},\boldsymbol{\theta})\right\rangle$ as

$$\left|\tilde{\psi}(\boldsymbol{\kappa},\boldsymbol{\theta})\right\rangle = e^{\boldsymbol{\kappa}} |\psi(\boldsymbol{\theta})\rangle . \tag{H.1}$$

Here, $\boldsymbol{\kappa}$ is an anti-hermitian operator defined by

$$\boldsymbol{\kappa} = \sum_{p>q} \kappa_{pq}(\hat{E}_{pq} - \hat{E}_{qp}), \tag{H.2}$$

where the $\hat{E}_{pq}$ are spin-adapted one-body operators,

$$\hat{E}_{pq} = \hat{a}^{\dagger}_{p\alpha}\hat{a}_{q\alpha} + \hat{a}^{\dagger}_{p\beta}\hat{a}_{q\beta}, \tag{H.3}$$

and the κ_{pq} are variational parameters that can be optimized to decrease the energy via orbital rotations. These rotations can be implemented efficiently in a classical computer by producing a new Hamiltonian with updated molecular integrals; therefore, they do not imply additional circuit costs.

ADAPT-VQE-SCF optimizes the molecular orbital basis along with the ansatz parameters (i.e., it carries out a simultaneous optimization of $\boldsymbol{\kappa}$ and $\boldsymbol{\theta}$ in each iteration). The gradients of the orbital rotation coefficients κ_{pq} can be obtained from the two-particle reduced density matrices, which are measured when evaluating the energy itself; therefore, these gradients are provided to us 'for free' during the optimization. However, $\boldsymbol{\kappa}$ comes with an additional $\frac{N_s(N_s-1)}{2}$ variational parameters (with N_s the number of spatial orbitals). This results in a higher-dimensional optimization, which may increase the total number of energy and ansatz gradient measurements required for the optimizer to converge, and therefore indirectly lead to additional measurement costs.

Figure H.1 shows the result of merging OO with the CEO-ADAPT-VQE* algorithm. We can see that the impact on the relevant markers is negligible: The `CNOT` count and depth required to reach a given error

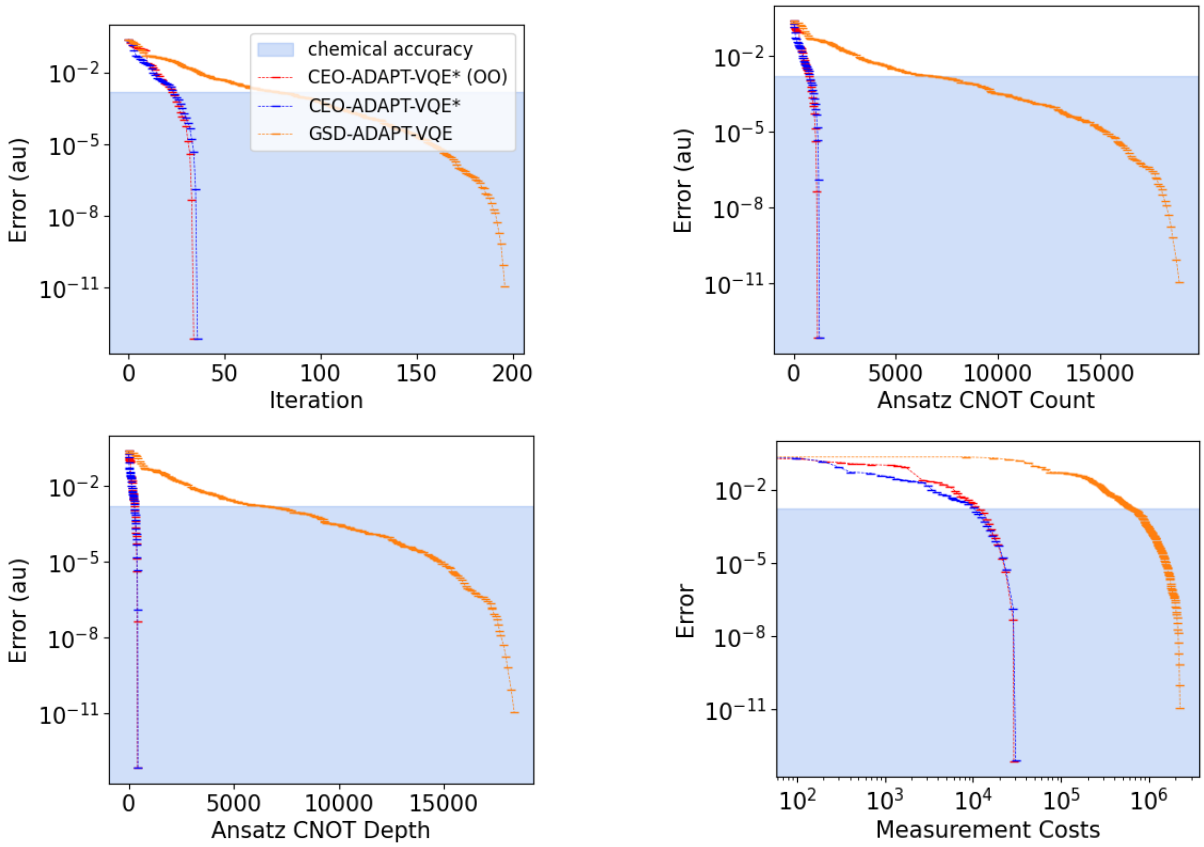


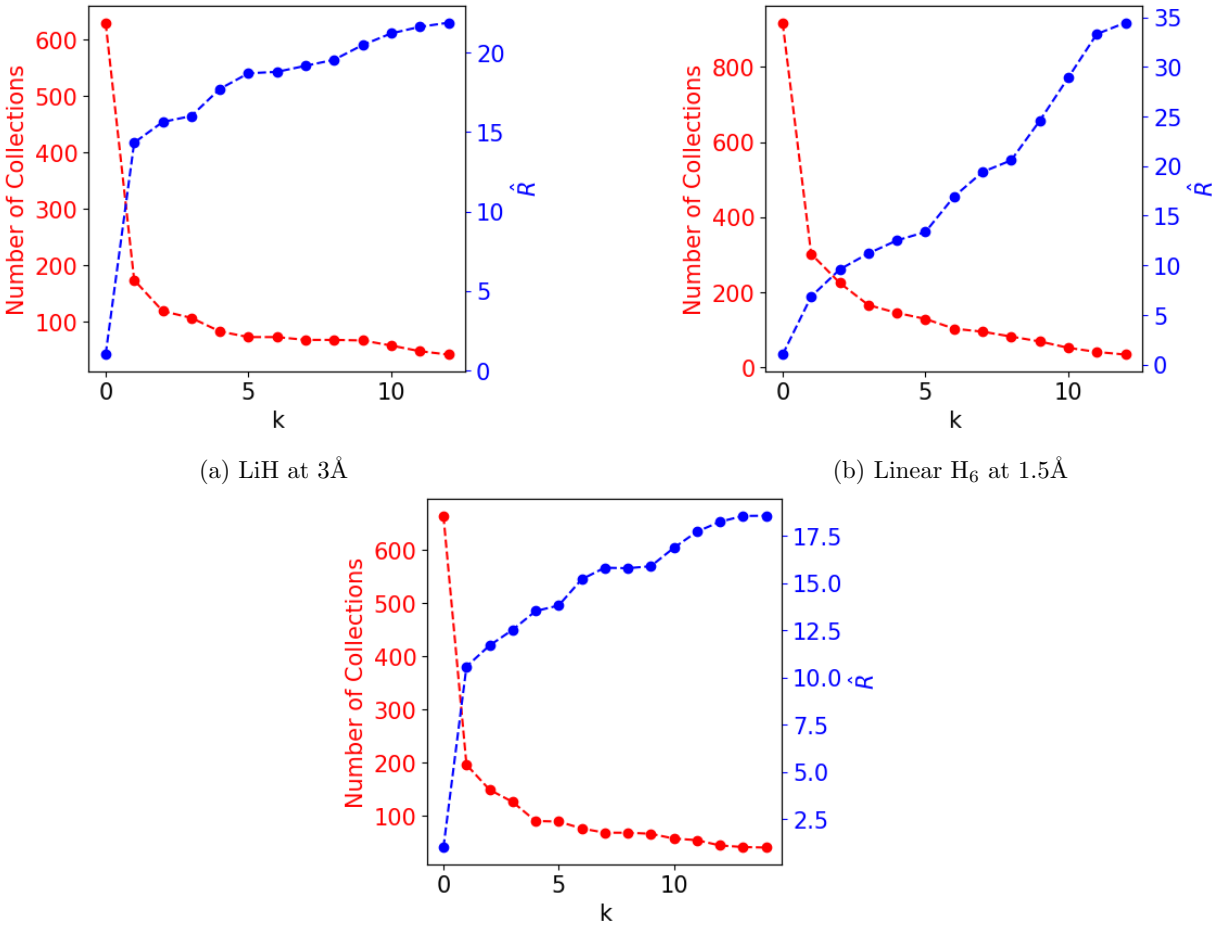
Figure H.1: Combining CEO-ADAPT-VQE with orbital optimization. The subfigures depict the convergence of CEO-ADAPT-VQE*, as defined in the Ch. 4, with and without orbital optimization. GSD-ADAPT-VQE is also included for reference. We consider H_6 at an interatomic distance of 1.5 Å. CEO-ADAPT-VQE*, GSD-ADAPT-VQE and UCCSD-VQE are defined in the main text (Chapters 2 and 4). The error is plotted against the iteration number, `CNOT` count, `CNOT` depth, and measurement costs. The measurement costs are given as multipliers for the cost of one energy measurement. The region shaded blue is the region of chemical accuracy (error below 1 kcal/mol). The convergence criterion is a gradient threshold of 10^{-6} and 10^{-5} on the 12-qubit and 14-qubit molecules, respectively.

are only very slightly decreased, while the measurement costs are slightly increased. We note that ADAPT-VQE-SCF was developed to enable simulations with large atomic orbital basis sets; it is not surprising that it has a subpar performance in the case of minimal basis sets.

While we recognize that orbital optimization may be an important enhancement to ADAPT-VQE in the case of larger molecules and/or basis sets, we exclude it from the algorithms in the main text due to its minor impact on the energy under the circumstances we consider.

I

Hamiltonian Grouping



(a) LiH at 3Å

(b) Linear H_6 at 1.5Å

(c) BeH_2 at 2Å

Figure I.1: Impact of k on the k-commutativity grouping. The plots show the evolution of the number of Pauli string collections and $\hat{R}$ against k, which defines the granularity of the commutativity considered in the observable grouping [37], for **a** LiH, **b** H_6, **c** BeH_2. While it is expected that fewer collections will be associated with lower measurement costs, covariances and varying coefficient magnitudes result in a nonlinear relation between the two. The metric $\hat{R}$ [35] approximates the ratio of measurement costs with and without grouping. The point $k = 0$ represents no grouping, such that $\hat{R}$ is one and the number of collections is the total number of Pauli strings in the Hamiltonian.

In Ch. 4, we applied the k-commutativity grouping [37] to molecular Hamiltonians and took the $\hat{R}$ metric as an approximation to the ratio M_u/M_g between measurement costs without (M_u) and with (M_g) grouping [35]. As discussed in the main text, we chose $k = n$ (general commutativity) in light of the expectation that the depth of the measurement circuit will not be significant relative to the depth of the state preparation circuit. In this case, it is convenient to consider full commutativity to maximize the reduction in measurement costs at the expense of negligible additional circuit depth. However, in general, choosing the optimal k involves a compromise between circuit depth and measurement costs. As such, it is interesting to analyze the evolution of the savings in measurement costs against k.

Figure I.1 showcases this evolution for the systems we consider in this paper. We observe that for LiH_2 and BeH_2, the case $k = 1$ (qubit-wise commutativity) offers the greatest improvement in measurement costs as compared to other unit increments of k. However, in the case of H_6, the molecule where the grouping achieves the greatest savings, the improvement resulting from incrementing k is roughly consistent throughout the whole plot.

Unlike Ref. [37], we do not observe $\hat{R}$ to stall after a given value k^* - in fact, $\hat{R}$ steadily increases until reaching the maximum value for the highest value of k. We conjecture that this difference may stem from the following facts: (i) the systems we consider are distinct from those where such behavior was encountered (Fermi-Hubbard vs molecular Hamiltonians), and (ii) the number of qubits of our systems is significantly lower. The stabilization of $\hat{R}$ at a maximum value for $k < n$ could happen for larger n. In that case, the measurement cost reduction of general commutativity can be achieved with shallower measurement circuits.

J

Generalized Gradient Formula

In this appendix, we derive the formula to obtain the gradient in any position within the ansatz that we use in Ch. 6.

For a typical ADAPT-VQE ansatz, the energy expectation value is:

$$E = \langle\psi_0| \left(\prod_{i=1}^{n} e^{-\theta_i A_i} \right) \hat{H} \left(\prod_{i=1}^{n} e^{\theta_{n+1-i} A_{n+1-i}} \right) |\psi_0\rangle \,, \tag{J.1}$$

where $|\psi_0\rangle$ is the reference state. A new operator $e^{\theta_\mu A_\mu}$ can be introduced into the ansatz at any position p constrained between 1 (prepending) and $n+1$ (appending):

$$E = \langle\psi_0| \left(\prod_{i=1}^{p_s-1} e^{-\theta_i A_i} \right) e^{-\theta_\mu A_\mu} \hat{M} e^{\theta_\mu A_\mu} \left(\prod_{i=1}^{p_s-1} e^{\theta_{p_s-i} A_{p_s-i}} \right) |\psi_0\rangle \,, \tag{J.2}$$

where $\hat{M}$ is:

$$\hat{M} := \left(\prod_{i=p_s}^{n} e^{-\theta_i A_i} \right) \hat{H} \left(\prod_{i=p_s}^{n} e^{\theta_{p_s+n-i} A_{p_s+n-i}} \right) . \tag{J.3}$$

Then, the energy derivative with respect to the variational parameter θ_μ can be written as:

$$\frac{\partial E}{\partial \theta_\mu} = \langle\psi_0| \left(\prod_{i=1}^{p_s-1} e^{-\theta_i A_i} \right) \left(-A_\mu e^{-\theta_\mu A_\mu} \hat{M} e^{\theta_\mu A_\mu} + e^{-\theta_\mu A_\mu} \hat{M} A_\mu e^{\theta_\mu A_\mu} \right) \left(\prod_{i=1}^{p_s-1} e^{\theta_{p_s-i} A_{p_s-i}} \right) |\psi_0\rangle \,. \tag{J.4}$$

Setting $\theta_\mu = 0$ (typical ADAPT-VQE initialization) and using the commutator $[A_\mu, \hat{M}] = A_\mu \hat{M} - \hat{M} A_\mu$ this can be simplified to

$$\left. \frac{\partial E}{\partial \theta_\mu} \right|_{\theta_\mu = 0} = \langle\psi_0| \left(\prod_{i=1}^{p_s-1} e^{-\theta_i A_i} \right) \left[\hat{M}, A_\mu \right] \left(\prod_{i=1}^{p_s-1} e^{\theta_{p_s-i} A_{p_s-i}} \right) |\psi_0\rangle \,. \tag{J.5}$$

By expanding each ansatz side, we obtain:

$$\left.\frac{\partial E}{\partial \theta_\mu}\right|_{\theta_\mu=0} = \langle\psi_n| \left[\hat{H},\ \left(\prod_{i=p_s}^{n} e^{\theta_{ps+n-i}A_{ps+n-i}}\right) A_\mu \left(\prod_{i=p_s}^{n} e^{-\theta_i A_i}\right)\right] |\psi_n\rangle \tag{J.6}$$

with $|\psi\rangle$ being defined as the wavefunction of the ansatz prior to the insertion of the new operator. In this notation, this parametrized wavefunction would be defined as:

$$|\psi_n\rangle = \left(\prod_{i=1}^{n} e^{\theta_{1+n-i}A_{1+n-i}}\right) |\psi_0\rangle\ . \tag{J.7}$$

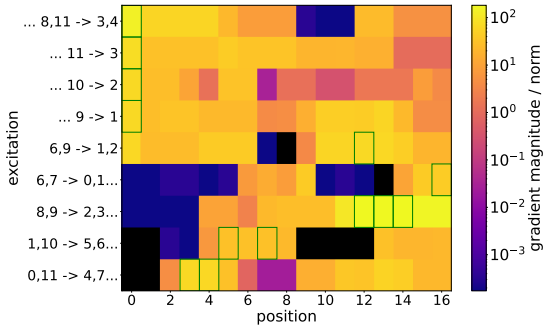
In order to implement this gradient measurement circuit for arbitrary positions other than appending in a quantum computer, the parameter-shift rule [113] needs to be applied on Eq. J.5. In the context of qubit or fermionic excitations operating on real states, as is the case in this study, it has been demonstrated that the expense associated with measuring each gradient is approximately equal to the cost of measuring the energy. This is accomplished by implementing a constant amount of supplementary gates, a phenomenon comparable to that observed in the context of a single Pauli string [91]. This results in total cost in the order of $\mathcal{O}(N^8)$ for measuring the gradient in any arbitrary ansatz position.

K

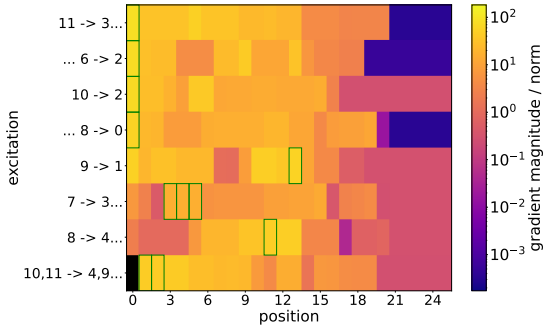
Relation Between Gradient Troughs and Gradients Across the Ansatz

In this appendix, we further analyze the behavior of the gradients over the positions as discussed in Ch. 6, Sec. 6.3. Figure K.1 shows a comparison of normalized gradient magnitudes before (iteration 16), within (iteration 25) and after (iteration 32) a gradient trough for the specified operators over all ansatz positions within a regular ADAPT-VQE execution (operators are always appended to the end of the ansatz). The gradients magnitudes are normalized by the minimal gradient magnitude across positions for iteration 28. We consider the 5 operators with largest gradients when appending and when prepending. When prepending, we write '...[source orbitals] → [target orbitals]'; when appending, we write '[source orbitals] → [target orbitals] ...'. Operators represented by '[source orbitals] → [target orbitals]' ranked among the top 5 both for prepending and for appending. The green boxes highlight the highest gradient magnitude position for the operator under scrutiny. The specific iterations which are represented within Fig.K.1 are highlighted in Fig.K.2, which showcases the conventional appending ADAPT-VQE calculation for the initial 50 iterations, accompanied by a gradient trough spanning from iteration 20 to iteration 29.

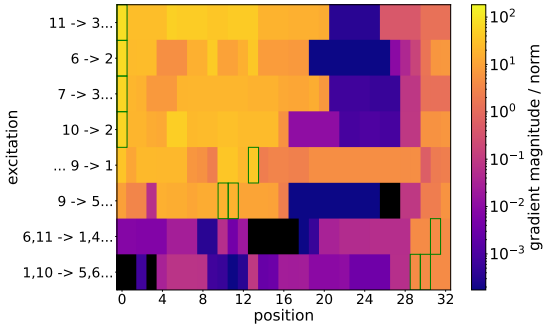
At iteration 16 (Fig. K.1a), before the gradient trough, the operators with the highest gradients for prepending/appending have near-maximum gradients at that same position. In what concerns the top 5 operators for appending, the gradients typically vanish when we consider prepending them (or more generally, placing them very early on the ansatz). This behavior is expected: when we select an operator due to it having a high gradient when appending/prepending, this will tend to be the position where it performs the best. However, this is not the case during a gradient trough, as we can observe in Fig. K.1b. In this case, even the top 5 operators selected for appending have significantly higher gradients when they are prepended. This can be explained by the commutator argumentation in Sec. 6.3. Finally, after leaving the gradient trough at iteration 32 (Fig. K.1c), the trend reverts back to what we observed before the gradient trough was encountered: The operators with highest gradients for appending have near-maximum gradients in the appending position.



(a) iteration 16 - ahead of gradient trough



(b) iteration 25 - within the gradient trough



(c) iteration 32 - after the gradient trough

Figure K.1: We compare the normalized gradient magnitudes before (iteration 16), during (iteration 25), and after (iteration 32) a gradient trough for the specified operators at all ansatz positions during a regular ADAPT-VQE execution (appending) for the top 5 operators for the prepending and appending positions. Top operators for prepending are labeled with '...[source orbitals] ⟶ [target orbitals]' and the ones for appending with '[source orbitals] ⟶ [target orbitals] ...'. Operators denoted as '[source orbitals] ⟶ [target orbitals]' are within the top for appending and prepending. The green boxes highlight the highest gradient magnitude position for the operator investigated.

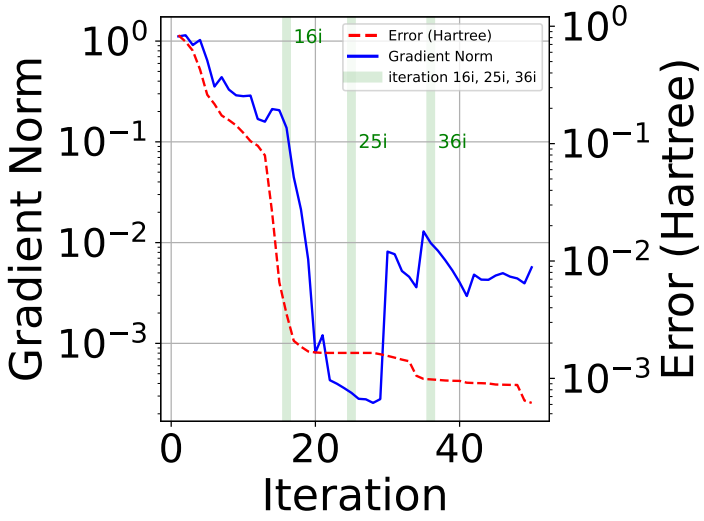


Figure K.2: Iterations considered in Fig. K.1 are highlighted. The gradient trough occurs in a regular ADAPT-VQE execution (appending operators).

L

Co-ADAPT-VQE for ATA connectivity

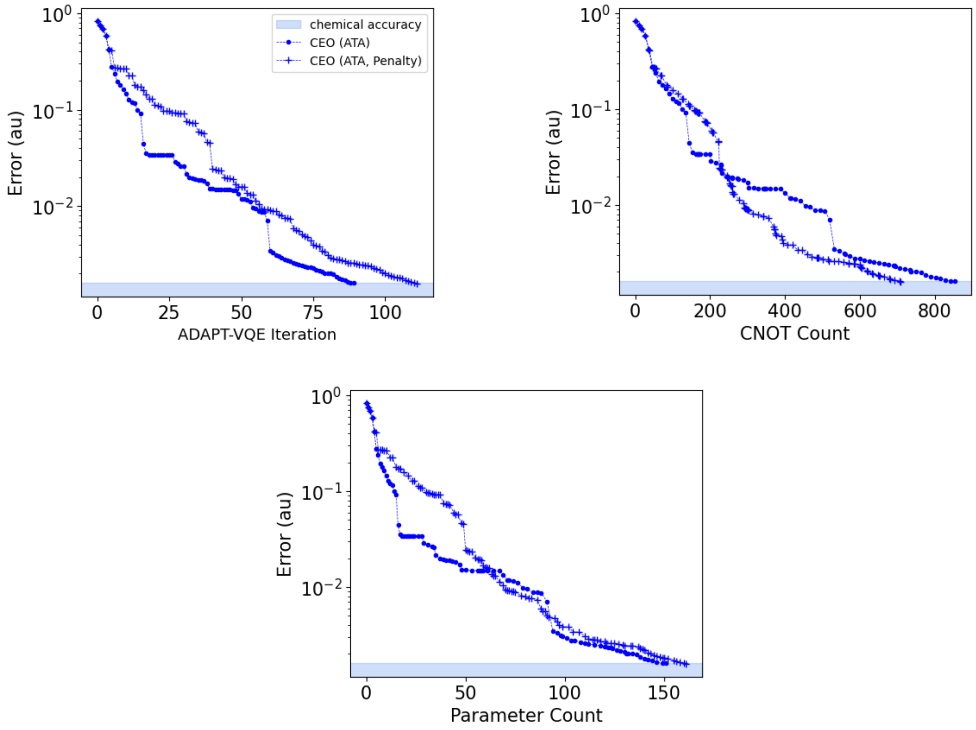


Figure L.1: Error *versus* ADAPT-VQE iteration, CNOT count, and parameter count for linear H_6 at interatomic distance 3 Å. Two variants are considered: CEO-ADAPT-VQE and the co-designed version, where the gradients are divided by the corresponding CNOT count for ATA connectivity. Both algorithms terminate when chemical accuracy is reached.

In the main text, we analyzed the impact of Co-ADAPT-VQE in producing state preparation circuits for LNN connectivity with low CNOT counts. We remark that while the impact will be greater in such a setting, due to the overhead incurred by the transpilation to restricted connectivity, all operator pools proposed in the literature include operators with varying CNOT counts. Hence, Co-ADAPT-VQE can also be applied to reduce gate counts for ATA connectivity.

We illustrate this in Fig. L.1 for the linear H_6 molecule. We observe that while Co-ADAPT-VQE requires

slightly more iterations to converge to chemical accuracy and terminates with slightly more parameters, it does succeed in decreasing the CNOT count by roughly 20%. This tradeoff between gate counts and parameters is usually favorable, since the size of the circuit determines whether or not it can be viably executed to reasonable accuracy.

We note that the purpose of this appendix is to analyze the impact of the penalty-based selection criterion on ADAPT-VQE for ATA connectivity. Parallel to this, we observe that the leading versions of Co-ADAPT-VQE improve the energy convergence by using FSWAPs in the operator routing. While this choice may at first be motivated by the lower number of CNOTs necessary to implement an FSWAP gate as compared to a SWAP gate, this additionally contributes to convergence due to the favorable action of the Jordan-Wigner Z strings for strongly correlated systems [182].

M

Co-ADAPT-VQE for Higher Accuracy

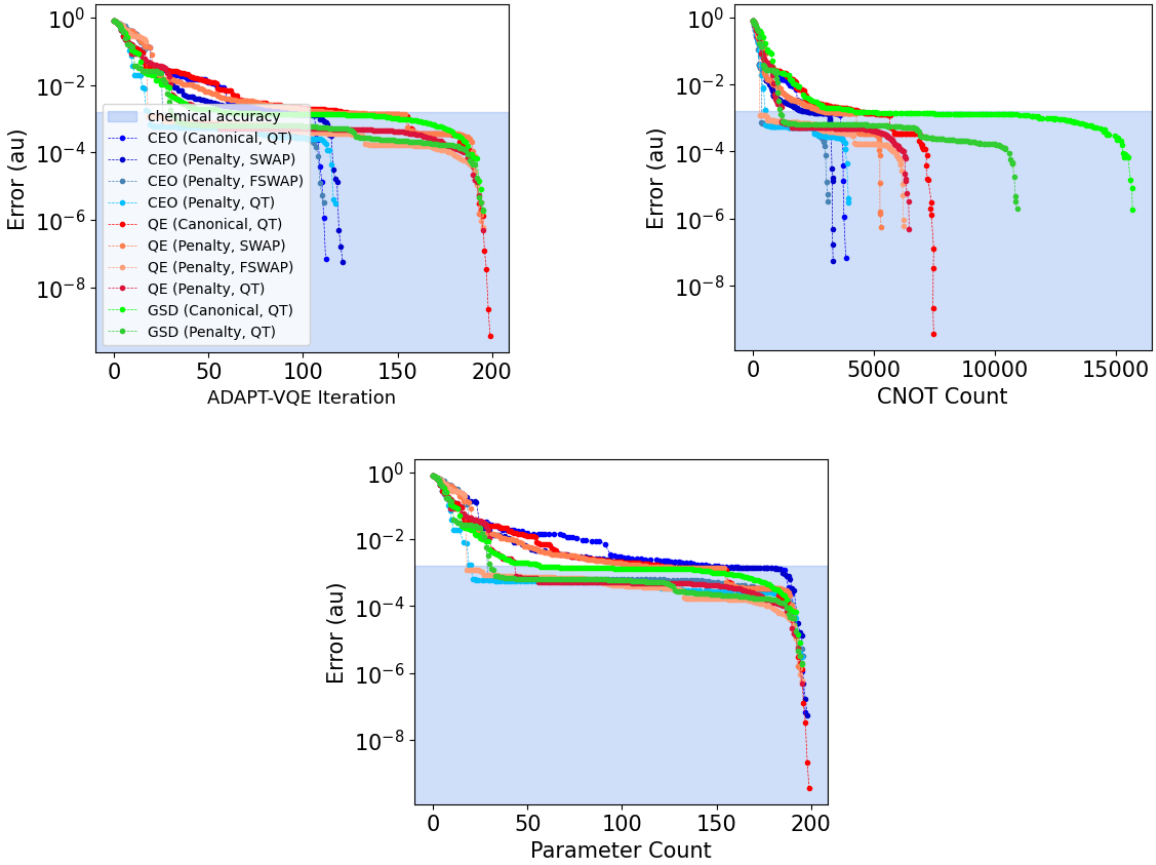


Figure M.1: ADAPT-VQE error *versus* iteration number, CNOT count for LNN connectivity, and parameter count for linear H_6 (12 qubits) with interatomic distance 3 Å. All algorithms terminate when the gradient norm falls below 10^{-4}.

Throughout this work, we focused on the gate and parameter counts required to reach chemically-accurate energies. This is the most important regime, since it provides experimentally useful results without exacerbating the high measurement costs of molecular ground-state preparation [67]. However, for completeness, we study a higher accuracy regime in Fig. M.1, with energy errors going as low as 10^{-10}

Hartree. We consider the same variants of ADAPT-VQE as in Fig. 5.10 of the main text.

We observe that while Co-ADAPT-VQE is still able to offer a decreased gate count with respect to the canonical version of the algorithm, the improvement is not as significant as in the chemically-accurate regime with errors around 10^{-3} Hartree. We attribute this to the fact that once we reach very low errors, we are very limited in the range of operators we can choose to approach the ground state, as very specific excitations will be required to fine-tune the final wavefunction.

N

Penalty Scaling

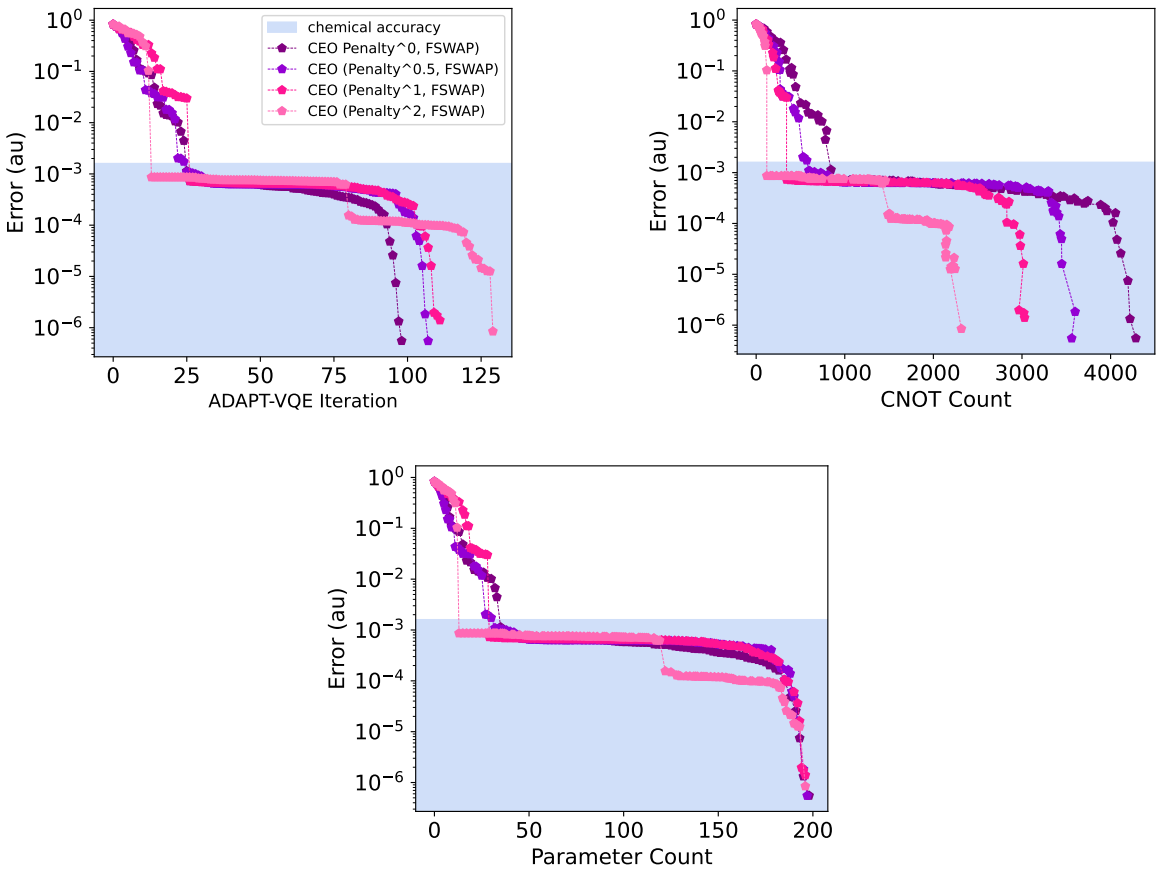


Figure N.1: Impact of scaling the penalty via varying exponentials on the performance of Co-CEO-ADAPT-VQE for the linear H_6 molecule. A darker curve corresponds to a lower exponent.

In the proposal of the Co-ADAPT-VQE algorithm introduced in the main text, we modify the selection criterion by dividing the gradient by a penalty of choice (Eq. (5.6)). However, this is not the only way of incorporating hardware characteristics into the selection criterion.

To exemplify alternative protocols, we make the penalty weaker or stronger by modifying the equation

defining the selection criterion to

$$\frac{\left.\frac{\partial E^{(n)}}{\partial \theta_i}\right|_{\theta_i=0}}{p^k} = \frac{\left\langle \psi^{(n-1)} \right| \left[\hat{H}, \hat{A}_i\right] \left| \psi^{(n-1)} \right\rangle}{p^k}, \tag{N.1}$$

for various values of the hyperparameter k. In particular, we take $k = 0$ (no penalty), $k = 0.5$ (penalty weaker than in the main text), $k = 1$ (penalty as in the main text), and $k = 2$ (penalty stronger than in the main text). Figure N.1 shows the results of the penalty scaling. As expected, we observe that stronger penalties lead to lower `CNOT` counts while requiring more iterations.

O

Results for LiH

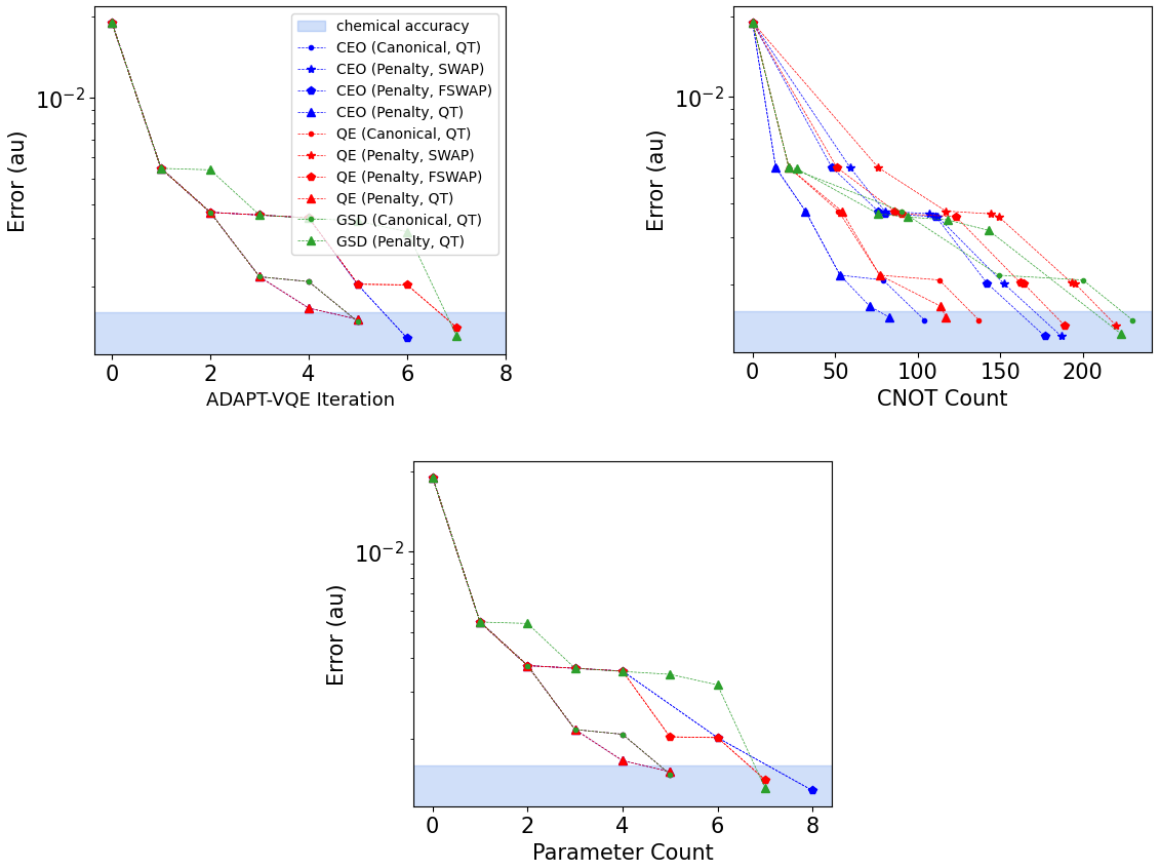


Figure O.1: Error *versus* the iteration number, LNN `CNOT` count, and parameter count for multiple variants of ADAPT-VQE applied to the LiH molecule (12 qubits) at interatomic distance 1.5 Å. We consider three different pools (CEO, QE, and GSD), three different variants of Co-ADAPT-VQE (`SWAP`-based, `FSWAP`-based, and Qiskit-transpiler-based), and the canonical ADAPT-VQE algorithm.

In the main text, we focused on two strongly correlated molecules: linear and triangular H_6. For the sake of completeness, we include results for the LiH molecule, an easier to simulate (less correlated) 12-qubit system.

The resulting CNOT and parameter counts are plotted in Fig. O.1. We observe that while some variants of Co-ADAPT-VQE decrease the CNOT counts with respect to the canonical ADAPT-VQE algorithm transpiled to LNN connectivity, that is not always the case, and the improvement is less significant than for the systems studied in the main text.

We attribute this difference to the fact that for very shallow ansätze, the transpiler achieves a significant improvement of the circuit by reordering the initial and final qubit assignments. For example, regardless of the first excitation that is chosen, we can always assume that it acts on adjacent qubits at the start of the algorithm, since we can redefine the initial qubit order for that to be the case. Recall that we apply Qiskit's transpilation with maximum optimization to the original ADAPT-VQE circuit; hence, these transpilation techniques that work well for shallow circuits will yield particularly good results for molecules such as LiH, where the ansatz needs no more than a few operators.

However, we note that such shallow ansätze are within the realm of classical simulability, and that larger / more strongly correlated systems are bound to be the relevant use cases of quantum computers. For those cases, as we show in the main text, we have deeper ansätze which can benefit the most from Co-ADAPT-VQE .

Bibliography

[1] S. Aaronson and D. Gottesman. “Improved simulation of stabilizer circuits”. In: *Physical Review A* 70.5 (2004-11). issn: 1094-1622. doi: 10.1103/physreva.70.052328 (cit. on p. 76).

[2] M. AbuGhanem. “IBM Quantum Computers: Evolution, Performance, and Future Directions”. In: *The Journal of Supercomputing* 81.5 (2025-04), p. 687. issn: 1573-0484. doi: 10.1007/s11227-025-07047-7 (cit. on p. 87).

[3] D. Aharonov et al. “Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation”. In: (2004). doi: 10.48550/ARXIV.QUANT-PH/0405098 (cit. on p. 7).

[4] V. Akshay et al. “Parameter concentrations in quantum approximate optimization”. In: *Physical Review A* 104.1 (2021-07). doi: 10.1103/physreva.104.l010401 (cit. on p. 33).

[5] J. R. M. et al. “OpenFermion: the electronic structure package for quantum computers”. In: *Quantum Science and Technology* 5.3 (2020-06), p. 034014. doi: 10.1088/2058-9565/ab8ebc (cit. on pp. 44, 67).

[6] Q. S. et al. “PySCF: the Python-based simulations of chemistry framework”. In: *WIREs Computational Molecular Science* 8.1 (2018), e1340. doi: https://doi.org/10.1002/wcms.1340 (cit. on pp. 44, 67).

[7] A. Ambainis. “Quantum search algorithms”. In: *ACM SIGACT News* 35.2 (2004-06), pp. 22–35. doi: 10.1145/992287.992296 (cit. on p. 10).

[8] A. Ambainis et al. “Quantum Speedups for Exponential-Time Dynamic Programming Algorithms”. 2018. doi: 10.48550/ARXIV.1807.05209 (cit. on p. 10).

[9] P. G. Anastasiou et al. “How to Really Measure Operator Gradients in ADAPT-VQE”. 2023-09. doi: 10.48550/arXiv.2306.03227 (cit. on pp. 18, 37, 38, 47, 65, 73–75, 80, 108, 130, 131).

[10] P. G. Anastasiou et al. “TETRIS-ADAPT-VQE: An Adaptive Algorithm That Yields Shallower, Denser Circuit *Ansätze*”. In: *Physical Review Research* 6.1 (2024-03), p. 013254. issn: 2643-1564. doi: 10.1103/PhysRevResearch.6.013254 (cit. on pp. 2, 19, 28, 52, 73, 74, 80, 130, 131).

[11] G.-L. R. Anselmetti et al. “Local, expressive, quantum-number-preserving VQE ansätze for fermionic systems”. In: *New Journal of Physics* 23.11 (2021-11), p. 113010. issn: 1367-2630. doi: 10.1088/1367-2630/ac2cb3 (cit. on pp. 30, 78, 79).

[12] S. Apers and R. de Wolf. “Quantum Speedup for Graph Sparsification, Cut Approximation and Laplacian Solving”. 2019. doi: 10.48550/ARXIV.1911.07306 (cit. on p. 9).

[13] A. Arrasmith et al. “Effect of barren plateaus on gradient-free optimization”. 2020. arXiv: 2011.12245 [quant-ph] (cit. on p. 12).

[14] A. Aspuru-Guzik et al. “Simulated Quantum Computation of Molecular Energies”. In: *Science* 309.5741 (2005-09), pp. 1704–1707. doi: 10.1126/science.1113479 (cit. on p. 9).

[15] A. Asthana et al. “Quantum Self-Consistent Equation-of-Motion Method for Computing Molecular Excitation Energies, Ionization Potentials, and Electron Affinities on a Quantum Computer”. In: *Chemical Science* 14.9 (2023), pp. 2405–2418. doi: 10.1039/D2SC05371C (cit. on p. 21).

[16] A. Asthana et al. “Quantum self-consistent equation-of-motion method for computing molecular excitation energies, ionization potentials, and electron affinities on a quantum computer”. In: *Chemical Science* 14.9 (2023), pp. 2405–2418. issn: 2041-6539. doi: 10.1039/d2sc05371c (cit. on p. 21).

[17] P. K. Barkoutsos et al. “Improving Variational Quantum Optimization using CVaR”. In: *Quantum* 4 (2020-04), p. 256. doi: 10.22331/q-2020-04-20-256 (cit. on p. 33).

[18] M. A. Barroca, R. Neumann Barros Ferreira, and M. Steiner. “Exploring Qubit-ADAPT-VQE for Materials Discovery in Direct Air Capture”. In: *APL Quantum* 1.4 (2024-10), p. 046103. issn: 2835-0103. doi: 10.1063/5.0219500 (cit. on p. 21).

[19] R. J. Bartlett and M. Musiał. “Coupled-cluster theory in quantum chemistry”. In: *Rev. Mod. Phys.* 79 (1 2007-02), pp. 291–352. doi: 10.1103/RevModPhys.79.291 (cit. on p. 13).

[20] R. Bellman. “Dynamic Programming Treatment of the Travelling Salesman Problem”. In: *Journal of the ACM* 9.1 (1962-01), pp. 61–63. doi: 10.1145/321105.321111 (cit. on p. 10).

[21] L. W. Bertels et al. “Symmetry Breaking Slows Convergence of the ADAPT Variational Quantum Eigensolver”. In: *Journal of Chemical Theory and Computation* 18.11 (2022-11), pp. 6656–6669. issn: 1549-9618. doi: 10.1021/acs.jctc.2c00709 (cit. on pp. 20, 26, 69).

[22] M. Bilkis et al. “A semi-agnostic ansatz with variable structure for quantum machine learning”. 2021. doi: 10.48550/ARXIV.2103.06712 (cit. on pp. 25, 26).

[23] M. Born and V. Fock. “Beweis des Adiabatensatzes”. In: *Zeitschrift f□r Physik* 51.3-4 (1928-03), pp. 165–180. doi: 10.1007/bf01343193 (cit. on p. 6).

[24] F. G. S. L. Brandao et al. “For Fixed Control Parameters the Quantum Approximate Optimization Algorithm’s Objective Function Value Concentrates for Typical Instances”. 2018. doi: 10.48550/ARXIV.1812.04170 (cit. on p. 33).

[25] G. Brassard et al. “Quantum amplitude amplification and estimation”. 2002. doi: 10.1090/conm/305/05215 (cit. on p. 9).

[26] C. G. BROYDEN. “The Convergence of a Class of Double-rank Minimization Algorithms”. In: *IMA Journal of Applied Mathematics* 6.3 (1970), pp. 222–231. doi: 10.1093/imamat/6.3.222 (cit. on p. 40).

[27] H. G. A. Burton. “Accurate and gate-efficient quantum Ansätze for electronic states without adaptive optimization”. In: *Physical Review Research* (2024). doi: 10.1103/PhysRevResearch.6.023300 (cit. on pp. 30, 78, 79).

[28] H. G. A. Burton et al. “Exact electronic states with shallow quantum circuits through global optimisation”. 2022. doi: 10.48550/ARXIV.2207.00085 (cit. on pp. 27, 29, 30).

[29] M. Cerezo et al. “Does Provable Absence of Barren Plateaus Imply Classical Simulability?” In: *Nature Communications* 16.1 (2025-08), p. 7907. issn: 2041-1723. doi: 10.1038/s41467-025-63099-6 (cit. on p. 2).

[30] M. Cerezo et al. “Variational Quantum Algorithms”. 2020. arXiv: 2012.09265 [quant-ph] (cit. on p. 14).

[31] H. H. S. Chan et al. “Molecular Excited State Calculations with Adaptive Wavefunctions on a Quantum Eigensolver Emulation: Reducing Circuit Depth and Separating Spin States”. In: *Physical Chemistry Chemical Physics* 23.46 (2021), pp. 26438–26450. issn: 1463-9076, 1463-9084. doi: 10.1039/D1CP02227J (cit. on p. 21).

[32] Y. Chen et al. “How Much Entanglement Do Quantum Optimization Algorithms Require?” In: *Quantum 2.0 Conference and Exhibition*. Optica Publishing Group, 2022, QM4A.2. doi: 10.1364/QUANTUM.2022.QM4A.2 (cit. on pp. 21, 35).

[33] A. Childs. “The quantum adiabatic theorem”. 2008 (cit. on p. 6).

[34] S. Choi, T.-C. Yen, and A. F. Izmaylov. “Improving Quantum Measurements by Introducing “Ghost” Pauli Products”. In: *Journal of Chemical Theory and Computation* 18.12 (2022-11), pp. 7394–7402. issn: 1549-9626. doi: 10.1021/acs.jctc.2c00837 (cit. on p. 75).

[35] O. Crawford et al. “Efficient quantum measurement of Pauli operators in the presence of finite sampling error”. In: *Quantum* 5 (2021-01), p. 385. doi: 10.22331/q-2021-01-20-385 (cit. on pp. 76, 135, 136).

[36] F. M. Creevey, C. D. Hill, and L. C. L. Hollenberg. “GASP – A Genetic Algorithm for State Preparation”. 2023. doi: 10.48550/ARXIV.2302.11141 (cit. on p. 23).

[37] B. DalFavero et al. “k-commutativity and measurement reduction for expectation values”. 2024. arXiv: 2312.11840 [quant-ph] (cit. on pp. 75, 135, 136).

[38] K. Dalton et al. “Quantifying the Effect of Gate Errors on Variational Quantum Eigensolvers for Quantum Chemistry”. In: *npj Quantum Information* 10.1 (2024-01), p. 18. issn: 2056-6387. doi: 10.1038/s41534-024-00808-x (cit. on pp. 2, 52, 67, 87, 93).

[39] E. Dantsin, V. Kreinovich, and A. Wolpert. “On quantum versions of record-breaking algorithms for SAT”. In: *ACM SIGACT News* 36.4 (2005-12), pp. 103–108. doi: 10.1145/1107523.1107524 (cit. on p. 10).

[40] W. Davidon. *Variable Metric Method for Minimization*. Office of Scientific and Technical Information (OSTI), 1959-11. doi: 10.2172/4222000 (cit. on p. 118).

[41] M. DeCross et al. “The Computational Power of Random Quantum Circuits in Arbitrary Geometries”. In: *Physical Review X* 15.2 (2025-05), p. 021052. issn: 2160-3308. doi: 10.1103/PhysRevX.15.021052 (cit. on p. 87).

[42] R. D. Delaney et al. “Scalable Multispecies Ion Transport in a Grid-Based Surface-Electrode Trap”. In: *Physical Review X* 14.4 (2024-11), p. 041028. issn: 2160-3308. doi: 10.1103/PhysRevX.14.041028 (cit. on p. 87).

[43] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial and Applied Mathematics, 1996-01. doi: 10.1137/1.9781611971200 (cit. on p. 36).

[44] N. L. Diaz et al. “Showcasing a Barren Plateau Theory Beyond the Dynamical Lie Algebra”. 2023-10. doi: 10.48550/arXiv.2310.11505 (cit. on p. 1).

[45] S. Ding, Z. Jin, and Q. Yang. “Evolving quantum circuits at the gate level with a hybrid quantum-inspired evolutionary algorithm”. In: *Soft Computing* 12.11 (2008-01), pp. 1059–1072. doi: 10.1007/s00500-007-0273-9 (cit. on p. 23).

[46] C. Dürr et al. “Quantum Query Complexity of Some Graph Problems”. In: *SIAM Journal on Computing* 35.6 (2006-01), pp. 1310–1328. doi: 10.1137/050644719 (cit. on p. 9).

[47] E. Farhi, J. Goldstone, and S. Gutmann. “A Quantum Approximate Optimization Algorithm”. 2014. arXiv: 1411.4028 [quant-ph] (cit. on pp. 31–33).

[48] E. Farhi and A. W. Harrow. “Quantum Supremacy through the Quantum Approximate Optimization Algorithm”. 2016. doi: 10.48550/ARXIV.1602.07674 (cit. on p. 34).

[49] E. Farhi et al. “Quantum Computation by Adiabatic Evolution”. 2000. doi: 10.48550/ARXIV.QUANT-PH/0001106 (cit. on p. 6).

[50] R. C. Farrell et al. “Scalable Circuits for Preparing Ground States on Digital Quantum Computers: The Schwinger Model Vacuum on 100 Qubits”. In: *PRX Quantum* 5 (2 2024-04), p. 020315. doi: 10.1103/PRXQuantum.5.020315 (cit. on p. 2).

[51] D. A. Fedorov et al. “Unitary Selective Coupled-Cluster Method”. In: *Quantum* 6 (2022-05), p. 703. doi: 10.22331/q-2022-05-02-703 (cit. on p. 15).

[52] C. Feniou et al. “Overlap-ADAPT-VQE: practical quantum chemistry on quantum computers via overlap-guided compact Ansätze”. In: *Communications Physics* 6.1 (2023-07). issn: 2399-3650. doi: 10.1038/s42005-023-01312-y (cit. on p. 2).

[53] R. P. Feynman. “Simulating Physics with Computers”. In: () (cit. on p. 1).

[54] A. Fitzpatrick et al. “Self-Consistent Field Approach for the Variational Quantum Eigensolver: Orbital Optimization Goes Adaptive”. In: *The Journal of Physical Chemistry A* 128.14 (2024-03), pp. 2843–2856. issn: 1520-5215. doi: 10.1021/acs.jpca.3c05882 (cit. on pp. 2, 73, 133).

[55] R. Fletcher. “A new approach to variable metric algorithms”. In: *The Computer Journal* 13.3 (1970-03), pp. 317–322. doi: 10.1093/comjnl/13.3.317 (cit. on p. 40).

[56] R. Fletcher. “Second order corrections for non-differentiable optimization”. In: *Lecture Notes in Mathematics*. Springer Berlin Heidelberg, 1982, pp. 85–114. doi: 10.1007/bfb0093151 (cit. on pp. 39, 40).

[57] R. Fletcher and M. J. D. Powell. “A Rapidly Convergent Descent Method for Minimization”. In: *The Computer Journal* 6.2 (1963-08), pp. 163–168. doi: 10.1093/comjnl/6.2.163 (cit. on p. 118).

[58] R. Fletcher. *Practical Methods of Optimization*. Second. New York, NY, USA: John Wiley & Sons, 1987 (cit. on pp. 36, 39, 40).

[59] E. Fontana et al. “Characterizing Barren Plateaus in Quantum Ansätze with the Adjoint Representation”. In: *Nature Communications* 15.1 (2024-08), p. 7171. issn: 2041-1723. doi: 10.1038/s41467-024-49910-w (cit. on p. 1).

[60] J. C. Getelina et al. “Adaptive variational quantum minimally entangled typical thermal states for finite temperature simulations”. In: *SciPost Phys.* 15 (2023), p. 102. doi: 10.21468/SciPostPhys.15.3.102 (cit. on p. 21).

[61] P. Gokhale and F. T. Chong. “$O(N^3)$ Measurement Cost for Variational Quantum Eigensolver on Molecular Hamiltonians”. 2019. arXiv: 1908.11857 [quant-ph] (cit. on p. 75).

[62] P. Gokhale et al. “$O(N^3)$ Measurement Cost for Variational Quantum Eigensolver on Molecular Hamiltonians”. In: *IEEE Transactions on Quantum Engineering* 1 (2020), pp. 1–24. doi: 10.1109/TQE.2020.3035814 (cit. on p. 76).

[63] P. Gokhale et al. “Optimization of Simultaneous Measurement for Variational Quantum Eigensolver Applications”. In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2020-10. doi: 10.1109/qce49297.2020.00054 (cit. on pp. 37, 75).

[64] D. Goldfarb. “A family of variable-metric methods derived by variational means”. In: *Mathematics of Computation* 24.109 (1970), pp. 23–26. doi: 10.1090/s0025-5718-1970-0258249-6 (cit. on p. 40).

[65] N. Gomes, D. B. Williams-Young, and W. A. de Jong. “Computing the Many-Body Green’s Function with Adaptive Variational Quantum Dynamics”. In: *Journal of Chemical Theory and Computation* 19.11 (2023-06), pp. 3313–3323. issn: 1549-9618. doi: 10.1021/acs.jctc.3c00150 (cit. on p. 21).

[66] N. Gomes et al. “Adaptive Variational Quantum Imaginary Time Evolution Approach for Ground State Preparation”. In: *Advanced Quantum Technologies* 4.12 (2021-10). issn: 2511-9044. doi: 10.1002/qute.202100114 (cit. on pp. 2, 21).

[67] J. F. Gonthier et al. “Measurements as a Roadblock to Near-Term Practical Quantum Advantage in Chemistry: Resource Analysis”. In: *Physical Review Research* 4.3 (2022-08), p. 033154. doi: 10.1103/PhysRevResearch.4.033154 (cit. on pp. 2, 11, 143).

[68] E. Grant et al. “An initialization strategy for addressing barren plateaus in parametrized quantum circuits”. In: *Quantum* 3 (2019-12), p. 214. issn: 2521-327X. doi: 10.22331/q-2019-12-09-214 (cit. on pp. 1, 22).

[69] E. K. Grant and T. S. Humble. “Adiabatic Quantum Computing and Quantum Annealing”. 2020-07. doi: 10.1093/acrefore/9780190871994.013.32 (cit. on p. 7).

[70] H. R. Grimsley and F. A. Evangelista. “Challenging Excited States from Adaptive Quantum Eigensolvers: Subspace Expansions vs. State-Averaged Strategies”. 2025-01. doi: 10.48550/arXiv.2409.11210 (cit. on p. 21).

[71] H. R. Grimsley et al. “Adaptive, Problem-Tailored Variational Quantum Eigensolver Mitigates Rough Parameter Landscapes and Barren Plateaus”. In: *npj Quantum Information* 9.1 (2023-03), p. 19. issn: 2056-6387. doi: 10.1038/s41534-023-00681-0 (cit. on pp. 2, 15, 20, 21, 28, 77).

[72] H. R. Grimsley et al. “An Adaptive Variational Algorithm for Exact Molecular Simulations on a Quantum Computer”. In: *Nature Communications* 10.1 (2019-12), p. 3007. issn: 2041-1723. doi: 10.1038/s41467-019-10988-2 (cit. on pp. 2, 15, 16, 36, 59, 67, 68, 73, 74, 82, 94, 100, 118, 131).

[73] H. R. Grimsley et al. “Is the Trotterized UCCSD Ansatz Chemically Well-Defined?” In: *Journal of Chemical Theory and Computation* 16.1 (2020-01), pp. 1–6. issn: 1549-9618, 1549-9626. doi: 10.1021/acs.jctc.9b01083 (cit. on p. 14).

[74] C. Grossmann, H.-G. Roos, and M. Stynes. *Numerical Treatment of Partial Differential Equations*. Springer Berlin Heidelberg, 2007. isbn: 9783540715849. doi: 10.1007/978-3-540-71584-9 (cit. on p. 116).

[75] L. K. Grover. “A fast quantum mechanical algorithm for database search”. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*. ACM Press, 1996. doi: 10.1145/237814.237866 (cit. on p. 9).

[76] C. Gu et al. “Towards Scalable Quantum Computations of Atomic Nuclei”. 2025-07. doi: 10.48550/arXiv.2507.14690 (cit. on p. 21).

[77] G. G. Guerreschi and M. Smelyanskiy. “Practical optimization for hybrid quantum-classical algorithms”. 2017. arXiv: 1701.01450 [quant-ph] (cit. on pp. 36, 118).

[78] E. Gustafson et al. “Surrogate Constructed Scalable Circuits ADAPT-VQE in the Schwinger Model”. https://arxiv.org/abs/2408.12641v1. 2024-08. (Visited on 2026-01-07) (cit. on p. 2).

[79] G. Gyawali and M. J. Lawler. “Adaptive Variational Preparation of the Fermi-Hubbard Eigenstates”. In: *Physical Review A* 105.1 (2022-01), p. 012413. doi: 10.1103/PhysRevA.105.012413 (cit. on p. 21).

[80] S. Hadfield et al. “From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz”. In: *Algorithms* 12.2 (2019-02), p. 34. doi: 10.3390/a12020034 (cit. on p. 32).

[81] N. Hatano and M. Suzuki. “Finding Exponential Product Formulas of Higher Orders”. In: *Quantum Annealing and Other Optimization Methods*. Springer Berlin Heidelberg, 2005-11, pp. 37–68. doi: 10.1007/11526216_2 (cit. on p. 14).

[82] M. Held and R. M. Karp. “A dynamic programming approach to sequencing problems”. In: *Proceedings of the 1961 16th ACM national meeting on -*. ACM Press, 1961. doi: 10.1145/800029.808532 (cit. on p. 10).

[83] W. K. Hensinger. “Quantum Computer Based on Shuttling Trapped Ions”. In: *Nature* 592.7853 (2021-04), pp. 190–191. doi: 10.1038/d41586-021-00844-z (cit. on p. 87).

[84] “IBM Quantum, 2021”. https://quantum-computing.ibm.com/. 2021 (cit. on p. 24).

[85] J. J. E. Dennis and J. J. Moré. “Quasi-Newton Methods, Motivation and Theory”. In: *SIAM Review* 19.1 (1977-01), pp. 46–89. doi: 10.1137/1019005 (cit. on pp. 36, 40, 118).

[86] P. Jordan and E. Wigner. “Über das Paulische Äquivalenzverbot”. In: *Zeitschrift für Physik* 47.9 (1928-09), pp. 631–651. issn: 0044-3328. doi: 10.1007/BF01331938 (cit. on pp. 11, 59).

[87] T. Kadowaki and H. Nishimori. “Quantum annealing in the transverse Ising model”. In: *Physical Review E* 58.5 (1998-11), pp. 5355–5363. doi: 10.1103/physreve.58.5355 (cit. on p. 7).

[88] A. Kandala et al. “Hardware-Efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets”. In: *Nature* 549.7671 (2017-09), pp. 242–246. issn: 1476-4687. doi: 10.1038/nature23879 (cit. on pp. 1, 12, 82).

[89] A. Y. Kitaev. “Quantum measurements and the Abelian Stabilizer Problem”. 1995. doi: 10.48550/ARXIV.QUANT-PH/9511026 (cit. on p. 8).

[90] I. D. Kivlichan et al. “Quantum Simulation of Electronic Structure with Linear Depth and Connectivity”. In: *Physical Review Letters* 120.11 (2018-03), p. 110501. issn: 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.120.110501 (cit. on p. 89).

[91] J. S. Kottmann, A. Anand, and A. Aspuru-Guzik. “A Feasible Approach for Automatically Differentiable Unitary Coupled-Cluster on Quantum Computers”. In: *Chemical Science* 12.10 (2021-03), pp. 3497–3508. issn: 2041-6539. doi: 10.1039/D0SC06627C (cit. on pp. 21, 44, 76, 107, 116, 138).

[92] A. Kumar et al. “Floquet-ADAPT-VQE: A Quantum Algorithm to Simulate Non-Equilibrium Physics in Periodically Driven Systems”. 2025-06. doi: 10.48550/arXiv.2503.11613 (cit. on pp. 2, 21).

[93] M. Larocca et al. “Barren Plateaus in Variational Quantum Computing”. In: *Nature Reviews Physics* 7.4 (2025-03), pp. 174–189. issn: 2522-5820. doi: 10.1038/s42254-025-00813-9 (cit. on p. 2).

[94] W. Lavrijsen et al. “Classical Optimizers for Noisy Intermediate-Scale Quantum Devices”. In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2020-10. doi: 10.1109/qce49297.2020.00041 (cit. on p. 116).

[95] J. Lee et al. “Generalized Unitary Coupled Cluster Wave functions for Quantum Computation”. In: *Journal of Chemical Theory and Computation* 15.1 (2018-11), pp. 311–324. issn: 1549-9626. doi: 10.1021/acs.jctc.8b01004 (cit. on pp. 13, 78, 79).

[96] S. Lee et al. “Is there evidence for exponential quantum advantage in quantum chemistry?” 2022. doi: 10.48550/ARXIV.2208.02199 (cit. on pp. 5, 9).

[97] X. Lee et al. “Parameters Fixing Strategy for Quantum Approximate Optimization Algorithm”. In: *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2021-10. doi: 10.1109/qce52317.2021.00016 (cit. on pp. 22, 28, 34).

[98] C.-T. Li and H.-C. Cheng. “Adaptive Circuit Learning of Born Machine: Towards Realization of Amplitude Embedding and Quantum Data Loading”. In: *Quantum Science and Technology* 10.2 (2025-04), p. 025019. issn: 2058-9565. doi: 10.1088/2058-9565/adaede (cit. on p. 21).

[99] D. Linteau et al. “Adaptive Projected Variational Quantum Dynamics”. In: *Physical Review Research* 6.2 (2024-05), p. 023130. issn: 2643-1564. doi: 10.1103/PhysRevResearch.6.023130 (cit. on pp. 2, 21).

[100] J. Liu, Z. Li, and J. Yang. “An efficient adaptive variational quantum solver of the Schrödinger equation based on reduced density matrices”. In: *The Journal of Chemical Physics* 154.24 (2021-06), p. 244112. issn: 1089-7690. doi: 10.1063/5.0054822 (cit. on pp. 2, 75).

[101] X. Liu et al. “Layer VQE: A Variational Approach for Combinatorial Optimization on Noisy Quantum Computers”. In: *IEEE Transactions on Quantum Engineering* 3 (2022), pp. 1–20. doi: 10.1109/tqe.2021.3140190 (cit. on p. 21).

[102] C. K. Long et al. “Layering and Subpool Exploration for Adaptive Variational Quantum Eigensolvers: Reducing Circuit Depth, Runtime, and Susceptibility to Noise”. https://arxiv.org/abs/2308.11708v1. 2023-08. doi: 10.1103/PhysRevA.109.042413 (cit. on pp. 2, 52).

[103] P. C. Lotshaw et al. “Empirical performance bounds for quantum approximate optimization”. In: *Quantum Information Processing* 20.12 (2021-11). doi: 10.1007/s11128-021-03342-3 (cit. on pp. 36, 118).

[104] M. Lukac et al. “Evolutionary Approach to Quantum and Reversible Circuits Synthesis”. In: *Artificial Intelligence in Logic Design*. Springer Netherlands, 2004, pp. 201–257. doi: 10.1007/978-1-4020-2075-9_7 (cit. on p. 23).

[105] M. Majland, P. Ettenhuber, and N. T. Zinner. "Fermionic adaptive sampling theory for variational quantum eigensolvers". In: *Phys. Rev. A* 108 (5 2023-11), p. 052422. doi: 10.1103/PhysRevA.108.052422 (cit. on p. 2).

[106] M. Majland et al. "Vibrational ADAPT-VQE: Critical Points Lead to Problematic Convergence". In: *The Journal of Chemical Physics* 160.15 (2024-04), p. 154109. issn: 0021-9606. doi: 10.1063/5.0191074 (cit. on p. 21).

[107] M. Majland et al. "Vibrational ADAPT-VQE: Critical points lead to problematic convergence". In: *The Journal of Chemical Physics* 160.15 (2024-04), p. 154109. issn: 0021-9606. doi: 10.1063/5.0191074 (cit. on p. 2).

[108] G. B. Mbeng, R. Fazio, and G. Santoro. "Quantum Annealing: a journey through Digitalization, Control, and hybrid Quantum Variational schemes". 2019. arXiv: 1906.08948 [quant-ph] (cit. on pp. 36, 118).

[109] J. R. McClean et al. "The theory of variational hybrid quantum-classical algorithms". In: *New Journal of Physics* 18.2 (2016-02), p. 023023. issn: 1367-2630. doi: 10.1088/1367-2630/18/2/023023 (cit. on p. 75).

[110] J. R. McClean et al. "Barren Plateaus in Quantum Neural Network Training Landscapes". In: *Nature Communications* 9.1 (2018-11), p. 4812. issn: 2041-1723. doi: 10.1038/s41467-018-07090-4 (cit. on pp. 1, 12, 20, 82).

[111] S. G. Mehendale et al. "Parameter Redundancy in the Unitary Coupled-Cluster Ansatze for Hybrid Variational Quantum Computing". 2023. doi: 10.48550/ARXIV.2301.09825 (cit. on p. 15).

[112] J. J. Meyer et al. "Exploiting Symmetry in Variational Quantum Machine Learning". In: *PRX Quantum* 4.1 (2023-03). issn: 2691-3399. doi: 10.1103/prxquantum.4.010328 (cit. on p. 2).

[113] K. Mitarai et al. "Quantum circuit learning". In: *Phys. Rev. A* 98 (3 2018-09), p. 032309. doi: 10.1103/PhysRevA.98.032309 (cit. on pp. 76, 116, 138).

[114] M. Mitchell. *An Introduction to Genetic Algorithms*. The MIT Press, 1998. doi: 10.7551/mitpress/3927.001.0001 (cit. on p. 22).

[115] R. Molteni, C. Gyurik, and V. Dunjko. "Exponential Quantum Advantages in Learning Quantum Observables from Classical Data". In: *npj Quantum Information* (2026-01). issn: 2056-6387. doi: 10.1038/s41534-025-01162-2 (cit. on p. 1).

[116] M. Mootz and Y.-X. Yao. "Efficient Berry Phase Calculation via Adaptive Variational Quantum Computing Approach". 2025. arXiv: 2506.19150 [quant-ph] (cit. on p. 21).

[117] J. J. More and D. C. Sorensen. *Newton's method*. 1982-02. doi: 10.2172/5326201 (cit. on p. 40).

[118] A. Mukherjee et al. "Comparative Study of Adaptive Variational Quantum Eigensolvers for Multi-Orbital Impurity Models". In: *Communications Physics* 6.1 (2023-01), p. 4. issn: 2399-3650. doi: 10.1038/s42005-022-01089-6 (cit. on p. 21).

[119] Y. O. Nakagawa et al. “ADAPT-QSCI: Adaptive Construction of an Input State for Quantum-Selected Configuration Interaction”. https://arxiv.org/abs/2311.01105v2. 2023-11. doi: 10.1021/acs.jctc.4c00846 (cit. on p. 21).

[120] Y. O. Nakagawa et al. “ADAPT-QSCI: Adaptive Construction of an Input State for Quantum-Selected Configuration Interaction”. In: *Journal of Chemical Theory and Computation* 20.24 (2024-12), pp. 10817–10825. issn: 1549-9618. doi: 10.1021/acs.jctc.4c00846 (cit. on p. 2).

[121] Y. Nam et al. “Ground-state energy estimation of the water molecule on a trapped-ion quantum computer”. In: *npj Quantum Information* 6.1 (2020-04), p. 33. issn: 2056-6387. doi: 10.1038/s41534-020-0259-3 (cit. on pp. 57, 58, 61, 62).

[122] J. Napp. “Variational quantum algorithms and geometry”. In: *Quantum Views* 4 (2020-06), p. 37. doi: 10.22331/qv-2020-06-04-37 (cit. on p. 39).

[123] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2012-06. isbn: 9780511976667. doi: 10.1017/cbo9780511976667 (cit. on pp. 1, 8, 14, 17, 18, 57, 90).

[124] J. Nocedal. “Theory of algorithms for unconstrained optimization”. In: *Acta Numerica* 1 (1992-01), pp. 199–242. doi: 10.1017/s0962492900002270 (cit. on p. 118).

[125] J. Nocedal and S. J. Wright, eds. *Numerical Optimization*. Springer-Verlag, 1999. doi: 10.1007/b98874 (cit. on pp. 36, 39–42, 117, 118).

[126] A. Nykänen et al. “Mitigating the Measurement Overhead of ADAPT-VQE with Optimised Informationally Complete Generalised Measurements”. https://arxiv.org/abs/2212.09719v2. 2022-12. (Visited on 2025-10-12) (cit. on p. 2).

[127] A. Nykänen et al. “Toward Accurate Calculation of Excitation Energies on Quantum Computers with ΔADAPT-VQE: A Case Study of BODIPY Derivatives”. In: *The Journal of Physical Chemistry Letters* 15.28 (2024-07), pp. 7111–7117. doi: 10.1021/acs.jpclett.4c01301 (cit. on pp. 2, 21).

[128] A. Nykänen et al. “Toward Accurate Post-Born–Oppenheimer Molecular Simulations on Quantum Computers: An Adaptive Variational Eigensolver with Nuclear-Electronic Frozen Natural Orbitals”. In: *Journal of Chemical Theory and Computation* 19.24 (2023-12), pp. 9269–9277. issn: 1549-9618. doi: 10.1021/acs.jctc.3c01091 (cit. on pp. 2, 21).

[129] B. O’Gorman et al. “Generalized swap networks for near-term quantum computing”. 2019. arXiv: 1905.05118 [quant-ph] (cit. on p. 79).

[130] P. O’Malley et al. “Scalable Quantum Simulation of Molecular Energies”. In: *Physical Review X* 6.3 (2016-07). doi: 10.1103/physrevx.6.031007 (cit. on p. 14).

[131] H. Patil, Y. Wang, and P. S. Krstić. “Variational Quantum Linear Solver with a Dynamic Ansatz”. In: *Physical Review A* 105.1 (2022-01), p. 012423. doi: 10.1103/PhysRevA.105.012423 (cit. on p. 21).

[132] A. Pellow-Jarman et al. “A comparison of various classical optimizers for a variational quantum linear solver”. In: *Quantum Information Processing* 20.6 (2021-06). issn: 1573-1332. doi: 10.1007/s11128-021-03140-x (cit. on p. 53).

[133] A. Peruzzo et al. “A Variational Eigenvalue Solver on a Quantum Processor”. In: *Nature Communications* 5.1 (2014-09), p. 4213. issn: 2041-1723. doi: 10.1038/ncomms5213 (cit. on p. 1).

[134] A. Pesah et al. “Absence of Barren Plateaus in Quantum Convolutional Neural Networks”. In: *Physical Review X* 11.4 (2021-10). issn: 2160-3308. doi: 10.1103/physrevx.11.041011 (cit. on p. 1).

[135] J. M. Pino et al. “Demonstration of the Trapped-Ion Quantum CCD Computer Architecture”. In: *Nature* 592.7853 (2021-04), pp. 209–213. issn: 1476-4687. doi: 10.1038/s41586-021-03318-4 (cit. on p. 87).

[136] M. Powell et al. *Some global convergence properties of a variable metric algorithm for minimization without exact line searches*. CSS (Series). UKAEA, Harwell Atomic Energy Research Establishment, 1975. url: https://books.google.com/books?id=JDLgPQAACAAJ (cit. on p. 118).

[137] Qiskit Community. “Qiskit: An Open-Source Framework for Quantum Computing”. 2017-03. doi: 10.5281/zenodo.2562110 (cit. on pp. 83, 92).

[138] F. A. Quinton et al. “Quantum Annealing Applications, Challenges and Limitations for Optimisation Problems Compared to Classical Solvers”. In: *Scientific Reports* 15.1 (2025-04), p. 12733. issn: 2045-2322. doi: 10.1038/s41598-025-96220-2 (cit. on p. 7).

[139] M. Ragone et al. “A Lie Algebraic Theory of Barren Plateaus for Deep Parameterized Quantum Circuits”. https://arxiv.org/abs/2309.09342v3. 2023-09. doi: 10.1038/s41467-024-49909-3 (cit. on p. 1).

[140] M. Ramôa. “Ans\”atze for Noisy Variational Quantum Eigensolvers”. 2022-12. doi: 10.48550/arXiv.2212.04323 (cit. on pp. 100, 113).

[141] M. Ramôa. “Ansätze for Noisy Variational Quantum Eigensolvers”. 2022. doi: 10.48550/ARXIV.2212.04323 (cit. on p. 77).

[142] A. G. Rattew et al. “A Domain-agnostic, Noise-resistant, Hardware-efficient Evolutionary Variational Quantum Eigensolver”. 2019. doi: 10.48550/ARXIV.1910.09694 (cit. on pp. 22, 24).

[143] D. Rolf. “3–SAT ϵ RTIME(1.32971^n)”. In: (2003-01) (cit. on p. 10).

[144] A. M. Romero et al. “Solving Nuclear Structure Problems with the Adaptive Variational Quantum Algorithm”. In: *Physical Review C* 105.6 (2022-06), p. 064317. doi: 10.1103/PhysRevC.105.064317 (cit. on p. 21).

[145] A. M. Romero et al. “Solving nuclear structure problems with the adaptive variational quantum algorithm”. In: *Physical Review C* 105.6 (2022-06). issn: 2469-9993. doi: 10.1103/physrevc.105.064317 (cit. on p. 2).

[146] J. Romero et al. “Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz”. In: *Quantum Science and Technology* 4.1 (2018-10), p. 014008. doi: 10.1088/2058-9565/aad3e4 (cit. on p. 13).

[147] B. Rubinstein. “Evolving quantum circuits using genetic programming”. In: *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*. Vol. 1. 2001, 144–151 vol. 1. doi: 10.1109/CEC.2001.934383 (cit. on p. 23).

[148] B. Sambasivam et al. “TEPID-ADAPT: Adaptive Variational Method for Simultaneous Preparation of Low-Temperature Gibbs and Low-Lying Eigenstates”. 2025-03. doi: 10.48550/arXiv.2503.14490 (cit. on pp. 2, 21).

[149] T. Schoning. “A probabilistic algorithm for k-SAT and constraint satisfaction problems”. In: *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*. IEEE Comput. Soc. doi: 10.1109/sffcs.1999.814612 (cit. on p. 10).

[150] M. Schuld et al. “Evaluating analytic gradients on quantum hardware”. In: *Physical Review A* 99.3 (2019-03). issn: 2469-9934. doi: 10.1103/physreva.99.032331 (cit. on pp. 76, 116).

[151] D. F. Shanno. “Conditioning of quasi-Newton methods for function minimization”. In: *Mathematics of Computation* 24.111 (1970), pp. 647–656. doi: 10.1090/s0025-5718-1970-0274029-x (cit. on p. 40).

[152] K. Sherbert et al. “Adaptive Quantum Generative Training using an Unbounded Loss Function”. In: *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*. Vol. 01. 2024, pp. 1731–1738. doi: 10.1109/QCE60285.2024.00202 (cit. on p. 21).

[153] V. O. Shkolnikov et al. “Avoiding Symmetry Roadblocks and Minimizing the Measurement Overhead of Adaptive Variational Quantum Eigensolvers”. In: *Quantum* 7 (2023-06), p. 1040. issn: 2521-327X. doi: 10.22331/q-2023-06-12-1040 (cit. on pp. 2, 19, 37).

[154] A. Skolik et al. “Layerwise learning for quantum neural networks”. 2020. arXiv: 2006.14904 [quant-ph] (cit. on pp. 28, 52).

[155] I. O. Sokolov et al. “Quantum orbital-optimized unitary coupled cluster methods in the strongly correlated regime: Can quantum algorithms outperform their classical equivalents?” In: *The Journal of Chemical Physics* 152.12 (2020-03), p. 124107. issn: 1089-7690. doi: 10.1063/1.5141835 (cit. on p. 13).

[156] J. Spall. “Multivariate stochastic approximation using a simultaneous perturbation gradient approximation”. In: *IEEE Transactions on Automatic Control* 37.3 (1992-03), pp. 332–341. issn: 0018-9286. doi: 10.1109/9.119632 (cit. on p. 53).

[157] V. K. Sridhar et al. “ADAPT-QAOA with a classically inspired initial state”. 2023. arXiv: 2310.09694 [quant-ph] (cit. on p. 21).

[158] B. K. Sriperumbudur and G. R. G. Lanckriet. “A Proof of Convergence of the Concave-Convex Procedure Using Zangwill’s Theory”. In: *Neural Computation* 24.6 (2012-06), pp. 1391–1407. doi: 10.1162/neco_a_00283 (cit. on p. 118).

[159] H. L. Tang et al. “Nonvariational ADAPT Algorithm for Quantum Simulations”. In: *Physical Review Research* 7.2 (2025-06), p. 023275. issn: 2643-1564. doi: 10.1103/x8g1-7h1k (cit. on p. 2).

[160] H. L. Tang et al. “Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansatze on a Quantum Processor”. In: *PRX Quantum* 2.2 (2021-04), p. 020310. issn: 2691-3399. doi: 10.1103/PRXQuantum.2.020310 (cit. on pp. 2, 17, 19, 44, 59, 63, 67, 87, 88, 95).

[161] J. Tilly et al. “The Variational Quantum Eigensolver: A review of methods and best practices”. In: *Physics Reports* 986 (2022-11), pp. 1–128. doi: 10.1016/j.physrep.2022.08.003 (cit. on p. 39).

[162] I. Tyagin et al. “QAOA-GPT: Efficient Generation of Adaptive and Regular Quantum Approximate Optimization Algorithm Circuits”. 2025. arXiv: 2504.16350 [quant-ph] (cit. on p. 21).

[163] A. V. Uvarov and J. D. Biamonte. “On barren plateaus and cost function locality in variational quantum algorithms”. In: *Journal of Physics A: Mathematical and Theoretical* 54.24 (2021-05), p. 245301. issn: 1751-8121. doi: 10.1088/1751-8121/abfac7 (cit. on p. 1).

[164] J. S. Van Dyke et al. “Scaling Adaptive Quantum Simulation Algorithms via Operator Pool Tiling”. In: *Physical Review Research* 6.1 (2024-02), p. L012030. doi: 10.1103/PhysRevResearch.6.L012030 (cit. on p. 19).

[165] N. Vaquero-Sabater, A. Carreras, and D. Casanova. 2025. arXiv: 2504.04652 [physics.chem-ph] (cit. on pp. 100, 113).

[166] V. Verteletskyi, T.-C. Yen, and A. F. Izmaylov. “Measurement optimization in the variational quantum eigensolver using a minimum clique cover”. In: *The Journal of Chemical Physics* 152.12 (2020-03), p. 124114. issn: 1089-7690. doi: 10.1063/1.5141458 (cit. on p. 75).

[167] P. Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. doi: 10.1038/s41592-019-0686-2 (cit. on pp. 40, 41, 68, 116).

[168] H. Wakaura, R. Mulyawan, and A. B. Suksmono. “Enhanced Variational Quantum Kolmogorov-Arnold Network”. 2025-06. doi: 10.48550/arXiv.2503.22604 (cit. on p. 21).

[169] Q. Wang et al. “Resource-Optimized Fermionic Local-Hamiltonian Simulation on a Quantum Computer for Quantum Chemistry”. In: *Quantum* 5 (2021-07), p. 509. doi: 10.22331/q-2021-07-26-509 (cit. on pp. 57, 58, 61, 62).

[170] S. Wang et al. “Noise-Induced Barren Plateaus in Variational Quantum Algorithms”. 2021. arXiv: 2007.14384 [quant-ph] (cit. on p. 12).

[171] A. Warren et al. “Adaptive Variational Algorithms for Quantum Gibbs State Preparation”. 2022-03. doi: 10.48550/arXiv.2203.12757 (cit. on pp. 2, 21).

[172] D. Wecker, M. B. Hastings, and M. Troyer. “Progress towards practical quantum variational algorithms”. In: *Phys. Rev. A* 92 (4 2015-10), p. 042303. doi: 10.1103/PhysRevA.92.042303 (cit. on p. 45).

[173] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik. “Simulation of electronic structure Hamiltonians using quantum computers”. In: *Molecular Physics* 109.5 (2011-03), pp. 735–750. issn: 1362-3028. doi: 10.1080/00268976.2011.552441 (cit. on p. 8).

[174] R. de Wolf. “Quantum Algorithms for Optimization”. url: https://www.youtube.com/watch?v=1-2LIopvNIk&list=PLgKuh-lKre11rEOFaO62MJCzBXfjZSDqA&index=28. Quantum Colloquium (Simons Institute) (cit. on p. 6).

[175] N. Yanakiev et al. “Dynamic Adaptive Quantum Approximate Optimization Algorithm for Shallow, Noise-Resilient Circuits”. In: *Physical Review A* 109.3 (2024-03), p. 032420. doi: 10.1103/PhysRevA.109.032420 (cit. on p. 21).

[176] Y.-X. Yao et al. “Adaptive Variational Quantum Dynamics Simulations”. In: *PRX Quantum* 2 (3 2021-07), p. 030307. doi: 10.1103/PRXQuantum.2.030307 (cit. on p. 2).

[177] Y.-X. Yao et al. “Adaptive Variational Quantum Dynamics Simulations”. In: *PRX Quantum* 2 (3 2021-07), p. 030307. doi: 10.1103/PRXQuantum.2.030307 (cit. on p. 21).

[178] T.-C. Yen, V. Verteletskyi, and A. F. Izmaylov. “Measuring All Compatible Operators in One Series of Single-Qubit Measurements Using Unitary Transformations”. In: *Journal of Chemical Theory and Computation* 16.4 (2020-03), pp. 2400–2409. doi: 10.1021/acs.jctc.0c00008 (cit. on p. 37).

[179] Y. S. Yordanov, D. R. M. Arvidsson-Shukur, and C. H. W. Barnes. “Efficient Quantum Circuits for Quantum Computational Chemistry”. In: *Physical Review A* 102.6 (2020-12), p. 062612. issn: 2469-9926, 2469-9934. doi: 10.1103/PhysRevA.102.062612 (cit. on pp. 18, 57, 59, 61, 63, 75, 84, 90).

[180] Y. S. Yordanov, C. H. W. Barnes, and D. R. M. Arvidsson-Shukur. “Molecular Excited State Calculations with the QEB-ADAPT-VQE”. 2021-10. doi: 10.48550/arXiv.2106.06296 (cit. on p. 2).

[181] Y. S. Yordanov, C. H. W. Barnes, and D. R. M. Arvidsson-Shukur. “Molecular-Excited-State Calculations with the Qubit-Excitation-Based Adaptive Variational Quantum Eigensolver Protocol”. In: *Physical Review A* 106.3 (2022-09), p. 032434. doi: 10.1103/PhysRevA.106.032434 (cit. on p. 21).

[182] Y. S. Yordanov et al. “Qubit-Excitation-Based Adaptive Variational Quantum Eigensolver”. In: *Communications Physics* 4.1 (2021-12), p. 228. issn: 2399-3650. doi: 10.1038/s42005-021-00730-0 (cit. on pp. 2, 36, 44, 55, 58, 59, 67, 84, 87, 88, 94, 95, 109, 118, 142).

[183] T. Yoshikura, S. L. Ten-no, and T. Tsuchimochi. “Quantum Inverse Algorithm via Adaptive Variational Quantum Linear Solver: Applications to General Eigenstates”. In: *The Journal of Physical Chemistry A* 127.31 (2023). PMID: 37515571, pp. 6577–6592. doi: 10.1021/acs.jpca.3c02800 (cit. on p. 2).

[184] T. Yoshikura, S. L. Ten-no, and T. Tsuchimochi. “Quantum Inverse Algorithm via Adaptive Variational Quantum Linear Solver: Applications to General Eigenstates”. In: *The Journal of Physical Chemistry A* 127.31 (2023-08), pp. 6577–6592. issn: 1089-5639. doi: 10.1021/acs.jpca.3c02800 (cit. on p. 21).

[185] F. Zhang et al. “Adaptive variational quantum dynamics simulations with compressed circuits and fewer measurements”. In: *Physical Review B* 111.9 (2025-03). issn: 2469-9969. doi: 10.1103/physrevb.111.094310 (cit. on p. 21).

[186] F. Zhang et al. “Adaptive variational quantum eigensolvers for highly excited states”. In: *Phys. Rev. B* 104 (7 2021-08), p. 075159. doi: 10.1103/PhysRevB.104.075159 (cit. on p. 2).

[187] J. Zhang and D. Lacroix. “Excited States from ADAPT-VQE Convergence Path in Many-Body Problems: Application to Nuclear Pairing Problem and H_4 Molecule Dissociation”. 2025-06. doi: 10.48550/arXiv.2506.22275 (cit. on p. 21).

[188] L. Zhu et al. “Adaptive Quantum Approximate Optimization Algorithm for Solving Combinatorial Problems on a Quantum Computer”. In: *Physical Review Research* 4.3 (2022-07), p. 033029. doi: 10.1103/PhysRevResearch.4.033029 (cit. on pp. 2, 21, 52).