UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

INSTITUTO DE FÍSICA

Loading and extraction of information on near-term quantum computers

Renato Mello da Silva Farias

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Física do Instituto de Física da Universidade Federal do Rio de Janeiro - UFRJ, como parte dos requisitos necessários à obtenção do título de Doutor em Ciências (Física).

Orientador: Leandro Aolita

Rio de Janeiro

Março de 2025

CIP - Catalogação na Publicação

M527l

Mello da Silva Farias, Renato
Loading and extraction of information on near term quantum computers / Renato Mello da Silva Farias. -- Rio de Janeiro, 2025.
101 f.

Orientadora: Leandro Aolita.
Tese (doutorado) - Universidade Federal do Rio de Janeiro, Instituto de Física, Programa de Pós Graduação em Física, 2025.

1. praparação de estados. 2. classical shadows. I. Aolita, Leandro, orient. II. Título.

Elaborado pelo Sistema de Geração Automática da UFRJ com os dados fornecidos pelo(a) autor(a), sob a responsabilidade de Miguel Romeu Amorim Neto - CRB-7/6283.

Loading and extraction of information on near-term quantum computers

Renato Mello da Silva Farias

Leandro Aolita

Tese de Doutorado submetida ao Programa de Pós-Graduação em Física, Instituto de Física, da Universidade Federal do Rio de Janeiro – UFRJ, como parte dos requisitos necessários à obtenção do título de Doutor em Ciências (Física).

Aprovada por:

.....
Prof. Dr. Leandro Aolita
(Presidente e Orientador)

.....
Prof. Dr. Fabricio Toscano

.....
Prof. Dr. Daniel Jonathan

.....
Prof. Dr. Daniel Jost Brod

.....
Prof. Dr. Adenilton J. da Silva

Rio de Janeiro - RJ
Março 2025

Resumo

Loading and extraction of information on near-term quantum computers

Renato Mello da Silva Farias

Orientador: Leandro Aolita

Resumo da Tese de Doutorado apresentada ao Programa de Pós-Graduação em Física do Instituto de Física da Universidade Federal do Rio de Janeiro - UFRJ, como parte dos requisitos necessários à obtenção do título de Doutor em Ciências (Física).

A fim de cumprir a promessa de vantagem computacional, a computação quântica necessita interagir eficientemente com computadores clássicos. Portanto, a existência de algoritmos quânticos eficientes é de fundamental importância para o carregamento e extração de dados clássicos em computadores quânticos. Esta necessidade se torna ainda mais relevante no contexto de recursos limitados das tecnologias quânticas de escala ruidosa intermediária (NISQ, em inglês). Nesta tese, apresentamos dois artigos abordando estas questões. Primeiro, apresentamos um esquema de codificação em amplitudes, que carrega dados clássicos em qualquer subespaço de Hilbert desejado. Estes subespaços são caracterizados por estados da base computacional associados a sequências de *bits* que contêm o mesmo peso de Hamming. Este esquema de codificação é utilizado como uma subrotina de outros dois novos algoritmos: um para codificação no modelo de acesso esparso e outro para codificação na base binária completa. Os circuitos quânticos resultantes dos três algoritmos são deterministicos, desprovidos de *qubits* auxiliares, ótimos em número de parâmetros e os estados quânticos são preparados de forma exata. Além disso, demonstramos nosso codificador de amplitudes em um *hardware*

quântico comercial e investigamos seu potencial como um *ansatz* para protocolos variacionais de optimização híbrida quântica-clássica. Em segundo lugar, apresentamos um protocolo de medições aleatórias em estados quânticos de n qubits. Este protocolo implementa as bases de medida usando circuitos com profundidade *ultra* curtas, *i.e.* $\mathcal{O}(1)$ em n, e também é robusto com relação a ruídos. O algoritmo expande a área de *classical shadows* para o território dos computadores NISQ, atualmente mais experimentalmente relevante. Mostramos que o canal de medidas resultante dos nossos circuitos pode ser representado por uma rede de tensores, aliviando o custo de pós-processamento que é típico do protocolo de classical shadows. Numericamente, investigamos a robusteza do nosso protocolo e demonstramos sua performance em uma aplicação. Experimentalmente, implementamos o esquema de medidas em um hardware NISQ comercial.

Palavras-chave: 1. Preparação de estados; 2. Codificação em amplitudes; 3. Circuitos que preservam Hamming weight; 4. Codificação esparsa; 5. Codificação binária; 6. "Classical shadows"; 7. Medições aleatórias; 8. "Classical shadows" robustas; 9. Circuitos de curta profundidade.

Abstract

Loading and extraction of information on near-term quantum computers

Renato Mello da Silva Farias

Orientador: Leandro Aolita

Abstract da Tese de Doutorado apresentada ao Programa de Pós-Graduação em Física do Instituto de Física da Universidade Federal do Rio de Janeiro - UFRJ, como parte dos requisitos necessários à obtenção do título de Doutor em Ciências (Física).

To fulfill its promise of computational advantages, quantum computing requires efficient interfacing with classical computers. Thus, the existence of efficient algorithms for the loading and extraction of classical data on quantum computers is of fundamental importance. This necessity becomes even more relevant in the context of near-term intermediate-scale quantum (NISQ) computers and their limited resources. In this thesis, we present two articles addressing these issues. First, we present an amplitude encoding scheme that loads classical data onto any Hilbert subspace that is defined by computational basis states associated with bitstrings of fixed Hamming weight. We then use this encoding scheme as a subroutine of a novel sparse encoder as well as a new binary encoding protocol. The resulting quantum circuits outputted by the three encoders are deterministic, ancilla-free, and parameter-optimal, and the prepared quantum state is exact. We demonstrate our amplitude encoder on commercial quantum hardware and numerically investigate its capabilities as a variational quantum ansatz. Second, we present a protocol for randomized measurements in n-qubit quantum states that uses *ultra* short-depth circuits, *i.e.* $\mathcal{O}(1)$ in n, and is also robust to noise. This algorithm expands the

classical shadows scheme to the more experimentally relevant scenario for NISQ devices. We show that the measurement channel resulting from our circuits admits a tensor network representation, alleviating the postprocessing costs that are typical of the classical shadows protocol. We numerically investigate the robustness of our protocol and demonstrate its capabilities with numerical experiments as well as deployment on commercially available NISQ hardware.

Keywords: 1. State preparation; 2. Amplitude encoding; 3. Hamming-weight-preserving circuits; 4. Sparse encoding; 5. Binary encoding; 6. Classical shadows; 7. Randomized measurements; 8. Robust classical shadows; 9. Short-depth circuits.

Acknowledgments

To Naihana [1]*, Biriba, Bituca, and Manga.*

I acknowledge and thank the Brazilian National Council for Scientific and Technological Development (CNPq), the Physics Institute (IF) of the Federal University of Rio de Janeiro (UFRJ), and the Technology Innovation Institute (TII) of Abu Dhabi, UAE. These institutions played a major role in supporting me financially during this period.

[1] Não se anima.

List of Publications

Publications are sorted in chronological order.

- Publications included in this thesis:

  1. Renato M. S. Farias, Raghavendra D. Peddinti, Ingo Roth, and Leandro Aolita, *Robust ultra-shallow shadows*, Quantum Sci. Technol. **10**, 025044 (2025) [1].

  2. Renato M. S. Farias, Thiago O. Maciel, Giancarlo Camilo, Ruge Lin, Sergi Ramos-Calderer, and Leandro Aolita, *Quantum encoder for fixed Hamming-weight subspaces*, Phys. Rev. Applied **23**, 044014 (2025) [2].

- Publications not included in this thesis:

  1. Andrea Pasquale, Andrea Papaluca, Renato M. S. Farias, Matteo Robbiati, Edoardo Pedicillo, and Stefano Carrazza, *Beyond full statevector simulation with Qibo*, arXiv preprint arXiv:2408.00384 (2024) [3].

  2. Matteo Robbiati, Edoardo Pedicillo, Andrea Pasquale, Xiaoyue Li, Andrew Wright, Renato M. S. Farias, Khanh Uyen Giang, Jeongrak Son, Johannes Knörzer, Siong Thye Goh, Jun Yong Khoo, Nelly H. Y. Ng, Zoë Holmes, Stefano Carrazza, Marek Gluza, *Double-bracket quantum algorithms for high-fidelity ground state preparation*, arXiv preprint arXiv:2408.03987 (2024) [4].

Table of Contents

List of Figures

List of Algorithms

List of Tables

Chapter 1

Introduction

The field of quantum computing has made significant advancements in the development of *near-term intermediate-scale quantum* (NISQ) processors [5, 6, 7, 8, 9, 10], which are characterized by their limited number of qubits and susceptibility to noise and errors [11]. As we advance in the NISQ era and approach quantum computing's early fault-tolerant era [12, 13], the ability to efficiently insert and extract information from quantum processors is critical to leveraging their full potential despite their current limitations.

On one hand, inefficient ancilla-free uploading of classical or quantum data onto quantum computers can dominate the limited execution runtime of NISQ and early fault-tolerant devices, rendering useless possible quantum advantages provided by subsequent quantum algorithms that are applied to the prepared state [14]. On the other hand, execution runtime can also be dominated by the estimation of expectation values on quantum hardware if these estimations are inefficient in terms of sample complexity [15, Theorems 2 and 3]. Among other reasons, the variance of estimators on NISQ devices is affected by circuit noise and readout errors, rendering the application of measurement-error mitigation techniques a necessity in any NISQ algorithm.

In this thesis, we present two quantum algorithms, one for loading and the other for the extraction of information on near-term quantum computers. These algorithms were published in Refs. [2] and [1], respectively. Reference [2] is presented in Chapter 2 of this thesis, while Chapter 3 contains Ref. [1]. For readability purposes, below we detail the issues addressed, the key contributions, and the broader implications of these works individually.

1.1 Data loading

Amplitude encoding is an important strategy for loading classical data — or a classical description of quantum data — into quantum computers. Given a (ℓ_2-normalized) d-dimensional data vector $\mathbf{x}$, this scheme provides a quantum circuit U that maps each entry of said vector onto the amplitudes of a quantum state $|\psi\rangle$ in the computational basis, *i.e.*,

$$|\psi(\mathbf{x})\rangle \coloneqq U(\mathbf{x})\ |0\rangle^{\otimes n} = \sum_{j=1}^{d} x_j\ |b_j\rangle\ , \tag{1.1}$$

where $|0\rangle^{\otimes n}$ is the n-qubit reference state of the quantum system, and $b_j \in \{0,\,1\}^n$ is the j-th bitstring associated to the j-th amplitude x_j. When all 2^n amplitudes of an n-qubit quantum state are used to encode a data vector of size $d = 2^n$, then the resulting quantum circuit is called a *binary encoder* [16, 17]. This amplitude-encoding scheme provides exponential spatial compression as it uses $n = \mathcal{O}(\log(d))$ qubits to encode a data vector of size d.

Without the use of auxiliary qubits, the best known quantum circuit depth required to encode dense data vectors of size $d = 2^n$ scales as $\mathcal{O}(2^n)$ [18, 19], *i.e.* the resulting circuits are exponentially long in the number of qubits n. Even though Ref. [18] achieves this scaling, it uses parametrized gates that suffer from numerical instability. Meanwhile, Ref. [19] obtains the same $\mathcal{O}(2^n)$ scaling by preparing a quantum state in the Schmidt basis, implicating (i) the preprocessing of the input data via singular value decomposition (SVD) at a $\mathcal{O}\left(2^{3n}\right)$ cost, and (ii) the possible requirement of additional circuit depth to perform a basis change to the computational basis after the state preparation. Thus, even though near-optimal in terms of scalings, the schemes from Refs. [18, 19] are difficult to implement in practice.

On the other end of the space compression "spectrum" lies what is commonly called *unary encoding* [1], where $n = d$ qubits are required to encode a d-dimensional vector [20, 21, 22, 23]. Thus, unary encoders, even though easier to implement due to much smaller circuit depths and parametrization angles that are easy to calculate [20, 22], provide no spatial compression.

[1] Borrowing from the classical machine learning literature, sometimes this encoding scheme is also referred to as *one-hot encoding*. Some authors, however, may distinguish between these two nomenclatures and treat these encodings as different.

In this thesis, we address the region of the "spatial compression spectrum" that lies between the aforementioned encodings. The so-called *Hamming-weight-k encoding* schemes are encodings that populate amplitudes of Hilbert subspaces that are characterized by computational basis states associated with bitstrings of fixed Hamming weight k. Since the cardinality of these subspaces is $\binom{n}{k}$, it is possible to use $n = \mathcal{O}\Big(k\, d^{1/k}\Big)$ qubits to encode a d-dimensional vector. This corresponds to a regime of polynomial compression, which is an interesting balance between circuit width (*i.e.* the number of qubits) and circuit depth, especially for NISQ devices. As referenced in Chapter 2, current proposals for state preparation in fixed Hamming-weight subspaces suffer from a family of drawbacks. Some examples of these drawbacks are: (i) Finding the angles of the parameterized gates in the circuit by either solving big non-linear systems of equations or via variational approaches, leading to approximate encoders [24]; (ii) The necessity of orthogonalization of the input data, leading the a considerable increase in classical preprocessing overhead [22]; (iii) Exponential classical overhead on generating the circuit architecture due to a classical optimization step that requires calculating the Quantum Fisher Information Matrix of the circuit ansatz multiple times at each optimization step [24]. A clean construction of Hamming-weight-k encoders that did not suffer from any of the aforementioned drawbacks remained an open question.

In Chapter 2 of this thesis, we present our work from Ref. [2] that closes this question. In that, we detailed our version of a Hamming-weight-k encoder that is deterministic, exact, auxiliary-qubit-free, and parameter-optimal, thus not suffering from any of the drawbacks above. Since our circuit generation is deterministic, we can provide a detailed resource estimation in terms of CNOT gates. Specifically, the CNOT-gate count of the Hamming-weight-preserving circuits scales as $\mathcal{O}\Big(k\,\binom{n}{k}\Big)$. We also show that the same subroutines used for the Hamming-weight-k encoder can also be used to create an auxiliary-qubit-free, parameter-optimal *sparse encoding* algorithm. The sparse encoder we derive removes the necessity of auxiliary qubits and has a CNOT-gate count of $\mathcal{O}(n\,s)$ for a sparse input data vector $\mathbf{x}$ with s nonzero entries. Moreover, we provided a *binary encoding* algorithm that uses the Hamming-

weight-k encoder as a subroutine by encoding each subspace $k \in [0, n]$ in ascending order. In this case, the circuit depth scales as $\mathcal{O}(n\, 2^n)$. This comes at the price of an $\mathcal{O}(n)$ increase in two-qubit gate count when compared to Refs. [18, 19], but removes the impracticalities of both methods referenced above. Numerically, we verify that the constant factor in the scaling asymptotically converges to 8.

We demonstrated the validity of our encoder by deploying it on commercial quantum hardware as well as using numerical experiments. On the commercial device, we prepared a quantum state encoding a (discretized) q-Gaussian probability distribution and applied an error mitigation technique to alleviate the effect of circuit noise and errors on our results. Based on the hardware results, we numerically studied the fidelity of our encoder under a simple two-qubit depolarizing noise model. Lastly, we numerically showed how the same circuits that are used for deterministic state preparation can be used as ansatze for optimization problems. More precisely, we showed that for the task of finding the ground state of a molecule, our Hamming-weight-preserving circuits can yield better results as variational quantum ansatze than other Hamming-weight-preserving circuits.

1.2 Data extraction.

Randomized measurements are a central tool for the estimation of observables on quantum states [25, 26], with applications in every aspect of quantum information theory. One recently developed algorithm for randomized measurements is the *classical shadows* (CS) protocol from Ref. [15], which derived theoretical guarantees of unbiasedness, bounded variances of estimators, and proved the optimality of linear estimation from basis measurements generated by Clifford rotations. Arguably, the most important result from Ref. [15] is proof that the estimation of expectation values using the n-qubit Clifford group is $\mathcal{O}(1)$ in n, *i.e.* it is optimally efficient (see Ref. [15, Theorem 2]). Another important result from Ref. [15] is the fact that the estimation of expectation values of Pauli words using the n-fold tensor product of 1-local 2-designs scales as $\mathcal{O}\left(3^k\right)$, with $k \in [1,\, n]$ being the locality of a given Pauli word (see Ref.

[15, Theorem 3]). These two results translate to the following practical consequences: (i) even though the global Clifford measurements are optimally efficient, they are also not suitable to NISQ devices since they require the implementation of "long" circuits of depth $\mathcal{O}\Big(n^2 / \log(n)\Big)$ [27], which would consume most, if not all, of the budget of a NISQ device; (ii) The n-fold 1-local Clifford measurements, though easy to implement experimentally, lead to sample complexities that grow exponentially in the locality of the measured observables.

These issues have motivated the study of *shallow shadows*, where shorter circuits are employed to "interpolate" between local and global rotations [28, 29, 30, 31, 32]. Reference [28] expresses the CS of a quantum state as a linear combination of *snapshots* [2] in a set of partitions of the quantum state, with the coefficients of the linear combination being related to the *entanglement features* [33, 34]. The authors use the entanglement features to bound the *shadow norm* [15] of the estimators. However, to find the aforementioned coefficients, it is necessary to solve a linear system of equations. The postprocessing required by the CS protocol is done by expressing the initial state, the (diagonal of) *measurement map* [3], and the observable(s) as tensor networks. Reference [30] expands on the techniques from Ref. [28] by employing finite-depth Clifford circuits that still preserve the diagonality of the measurement map in the Pauli-Liouville basis. Reference [31] provides analytical expressions for the measurement maps of depth-2 brickwork circuits. In Ref. [32], short-depth Clifford circuits are implemented as an interpolation between local and global Cliffords. The authors provide a way to query the diagonal elements of the measurement map directly from the tensor network representation, avoiding having to solve a linear system of equations like in Refs. [28, 30]. They also present bounds on the shadow norm for the aforementioned frames. The inverse frame is estimated by variationally computing the inverse of the tensor network.

On NISQ hardware, however, even measurement circuits in this scaling regime are challenging. For the measurement, one typically wants to work with hardware-efficient circuits of

[2] To be defined in Chap. 3

[3] Also known as *measurement channel*, *frame operator*, or *quantum-classical interface*.

extremely low depth, which we refer to as *ultra-shallow shadows*. Despite the large body of literature on classical shadows, a central question is still open: Can the theoretically established improvement in sample efficiency with entangling gates be turned into a practical advantage in near-term experiments?

We show that advantages are achievable and can be significant, but also limited at current noise levels. We present a robust shadow estimation scheme for ultra-shallow circuits with arbitrary architectures of random local gates, under mild assumptions on the quantum hardware noise and errors. Since we use ultra-shallow circuits, we show that the corresponding measurement map admits a tensor network representation as matrix product states of low rank. This allows us to develop efficient methods for directly estimating the measurement channel using standard tensor-network tools. More importantly, we can obtain the inverse of the frame operator using standard tensor-network tools, though without theoretical guarantees that the inverse will also be of low rank.

We numerically investigate how shadow estimations become increasingly biased as the circuit depth increases in the presence of unmitigated circuit noise and errors. We also show that robust ultra-shallow shadows become more precise as the depth increases up to an *optimal depth* depending on the strength of the incoherent noise. Moreover, we apply our scheme to a recent quantum chemistry experiment [35]. Finally, we perform a proof-of-principle experiment on commercial quantum hardware. We estimate the effective measurement map and find that, indeed, simplified assumptions beyond what our methods exploit are not permissible.

1.3 Thesis structure.

As previously mentioned, this thesis is composed of a collection of two articles. Below, we detail the structure of the remainder of the thesis: In Chapter 2, we present the paper called *Quantum encoder for fixed-Hamming-weight subspaces*, dedicated to providing three protocols for amplitude encoding: (i) on a basis of fixed Hamming-weight computational basis states in Sec. 2.3; (ii) of sparse data in any subset of computational basis states in Sec. 2.4; (iii) in full

binary basis in Sec. 2.5. In Sec. 2.6, we show the aforementioned quantum hardware demonstration, the numerical analysis of noise effects, and the quantum machine learning experiment. In Chapter 3, we present the article called *Robust ultra-shallow shadows*, with a recap of the classical shadows protocol in Sec. 3.2 and the presentation of our results in Sec. 3.3. In Chapter 4, we recap the conclusions of both articles and expand on their outlook. Chapter 5 contains the Appendix sections of Chapters 2 and 3, in that order.

Chapter 2

Quantum encoder for fixed-Hamming-weight subspaces

Renato M. S. Farias[1,2]*, T. O. Maciel*[1]*, G. Camilo*[1]*, R. Lin*[1,3]*, S. Ramos-Calderer*[1,3]*, L. Aolita*[1]

[1]*Quantum Research Center, Technology Innovation Institute, Abu Dhabi, UAE*

[2]*Instituto de Física, Universidade Federal do Rio de Janeiro, P.O. Box 68528, Rio de Janeiro, Rio de Janeiro 21941-972, Brazil*

[3]*Departament de Física Quàntica i Astrofísica and Institut de Ciències del Cosmos (ICCUB), Universitat de Barcelona, Barcelona, Spain.*

We present an exact n-qubit computational-basis amplitude encoder of real- or complex-valued data vectors of $d = \binom{n}{k}$ components into a subspace of fixed Hamming weight k. This represents a polynomial space compression of degree k. The circuit is parameter-optimal, *i.e.* it expresses an arbitrary data vector using only $d-1$ free parameters, which are encoded using $d-1$ (controlled) Reconfigurable Beam Splitter (RBS) gates. The circuit is constructed by an efficient classical algorithm that sequentially generates all bitstrings of weight k and identifies the gates that superpose the corresponding states with the correct amplitudes. An explicit compilation into CNOTs and single-qubit gates is presented, with the total CNOT-gate count of $\mathcal{O}(k\,d)$ provided in analytical form. In addition, we show how to load data in the binary basis by sequentially stacking encoders of different Hamming weights using $\mathcal{O}(d\,\log(d))$ CNOT gates.

Moreover, using generalized RBS gates that mix states of different Hamming weights, we extend the construction to efficiently encode arbitrary sparse vectors. We perform an experimental proof-of-principle demonstration of our scheme on a commercial trapped-ion quantum computer. We successfully uploaded a q-Gaussian probability distribution in the non-log-concave regime with $n = 6$ and $k = 2$. We also showcase how the effect of hardware noise can be alleviated by quantum error mitigation. Numerically, we show how our encoder can improve the performance of variational quantum algorithms for problems that include particle-preserving symmetries. Our results constitute a versatile framework for quantum data compression with various potential applications in fields such as quantum chemistry, quantum machine learning, and constrained combinatorial optimizations.

2.1 Introduction

Amplitude encoding schemes in the basis of Hamming-weight-k (HW-k) states of n qubits correspond to an interesting regime of polynomial space compression $n \in \mathcal{O}\left(k\, d^{1/k}\right)$ for data vectors of size d [36, 37, 24, 38]. They are the middle ground between two distinct encoding scenarios. On one end, there is an amplitude encoding scheme of zero compression that efficiently loads data in the unary basis [20, 21, 22, 23, 39]. On the other end, there are techniques to load amplitudes in binary basis, which provides exponential compression [16, 17, 40, 41, 42, 43, 44, 45, 46], but requires a number of two-qubit gates that is exponential in n [47, 48, 49, 19]. Moreover, HW-k encoders have the additional benefit of being the natural subspace for applications in fields such as quantum chemistry, due to the particle-preserving nature of typical Hamiltonians [36, 50, 37, 51, 52]. They have also been used in the context of quantum machine learning [21, 22, 23, 24, 53, 38] and quantum finance [20, 39, 54, 55]. When used as variational circuits, they take advantage of the reduced subspace to mitigate the barren plateau problem [56, 55, 53].

Current proposals of HW-k encoders present some drawbacks in practice. For instance, Refs. [23, 24] use the aforementioned unary-basis encoders as subroutines. While allowing for

the generalization to HW $k > 1$ using only two-qubit Givens rotations, these encoders require orthogonalization of the input data [22] or solving non-linear systems of $\binom{n}{k} - 1$ equations to compute the rotation angles [24]. Here we address these classical overheads while still performing an exact encoding.

We present an exact, ancilla-free amplitude encoder for real and complex data in HW-k subspaces. We detail a classical algorithm that, given an initial bitstring of Hamming weight k and a data vector of size d, outputs instructions for a sequence of (controlled) parameterized rotations and their respective angles in hyperspherical coordinates of the data vector, hence efficient to compute. An explicit compilation into CNOTs and single-qubit gates is presented, with the total CNOT-gate count of $\mathcal{O}(k\,d)$ provided in analytical form. Moreover, our HW-k encoder can be used as a subroutine to power more complex algorithms. For instance, by combining all Hamming weights $k \leq n$ we obtain a binary encoder using $\mathcal{O}(d\,\log(d))$ CNOT gates. Using a generalized RBS gate designed to superpose states with different Hamming weights, we extend our methods and derive a procedure to load data in the paradigm of the classical sparse-access model. Moreover, we deploy a proof-of-concept instance on the `Aria-1` ion-trap quantum processor from IonQ [57], uploading a q-Gaussian probability distribution. The experimental results are further enhanced using a well-known error mitigation technique called Clifford Data Regression [58, 59].

The remainder of this Chapter is structured as follows. In Sec. 2.2 we introduce the gates used in our algorithm. Then, we present our framework for HW-k encoders in Sec. 2.3. Secs. 2.4 and 2.5, respectively, expand on the sparse and binary encoders using the fixed HW-k encoder as a subroutine. In Sec. 2.6.1, we show the deployment of our HW-k encoder on quantum hardware, and we conclude in Sec. 2.7.

2.2 Preliminaries

Notation. Let $b \in \{0,1\}^{\otimes n}$ be a bitstring of length n and $|b|$ its Hamming weight (HW); given two bitstrings b and b', their Hamming distance is $|b \oplus b'|$, where $\oplus$ is addition mod 2.

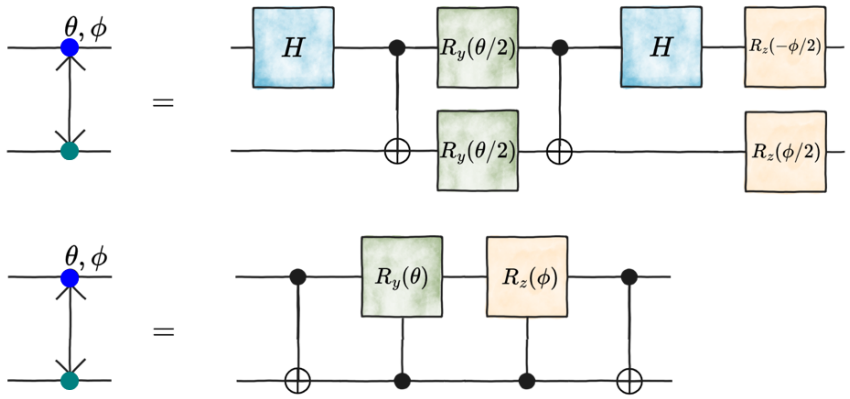

Figure 2.1: **Two possible compilations of the complex** RBS **gate** $\mathrm{R}_{out}^{in}(\theta,\,\phi)$**.** Colors indicate the input (blue) and output (green) qubits. *(Top)* Compilation using two CNOTs and single-qubit gates; *(Bottom)* Compilation using two CNOTs and two controlled-Pauli rotations; these rotations can be merged into a single controlled-$\mathrm{SU}(2)$ gate (see App. 5.1.1.1).

For convenience, we refer to the HW of a computational basis state $|b\rangle$ as being the HW of the bitstring b, and both are used interchangeably. The value of the j-th bit of a bitstring b is denoted by $b(j)$. Given a set $\mathcal{I}$, we call $\mathcal{I}_c$ its complementary set. We also use the shorthand notation $[d] := \{1, 2, \ldots, d\}$.

Here, we introduce gates that superpose two computational basis states with tunable amplitudes. These will be the building blocks for our encoders. We define the single-qubit Pauli-Y rotation as $\mathrm{R_y}(\theta) := e^{-i\,\theta\,Y}$ [1]. This gate superposes the two computational basis states $|0\rangle$ and $|1\rangle$ of a single-qubit subspace by acting as

$$\begin{aligned} \mathrm{R_y}(\theta)\,|0\rangle &= \cos(\theta)\,|0\rangle + \sin(\theta)\,|1\rangle \\ \mathrm{R_y}(\theta)\,|1\rangle &= \cos(\theta)\,|1\rangle - \sin(\theta)\,|0\rangle\ . \end{aligned} \tag{2.1}$$

When acting on a multi-qubit state, (controlled) $\mathrm{R_y}(\theta)$ rotations can be used to superpose any two computational basis states of Hamming distance 1.

Another gate of interest is one creating a superposition between two computational basis states of the same Hamming weight and that have a Hamming distance 2 by acting non-trivially only on a two-qubit subspace. Denoting by *in* (*out*) the bit that has value 1 (0) in the first state

[1] This definition differs from the also common definition $\mathrm{R_y}(\theta) = e^{-i\,\frac{\theta}{2}\,Y}$.

and 0 (1) in the second, we define the two-qubit gate $\mathrm{R}^{in}_{out}(\theta,\,\phi)$ acting on qubits in and out as

$$\begin{aligned}\mathrm{R}^{in}_{out}(\theta,\,\phi)\,|10\rangle &= e^{i\phi}\cos(\theta)\,|10\rangle + e^{-i\phi}\sin(\theta)\,|01\rangle\\ \mathrm{R}^{in}_{out}(\theta,\,\phi)\,|01\rangle &= e^{-i\phi}\cos(\theta)\,|01\rangle - e^{i\phi}\sin(\theta)\,|10\rangle\end{aligned} \tag{2.2}$$

and as the identity elsewhere. We call $\mathrm{R}^{in}_{out}(\theta,\,\phi)$ a *complex Reconfigurable Beam Splitter* (RBS) gate [2], reducing to the usual RBS gate for $\phi = 0$. We also denote $\mathrm{R}^{in}_{out}(\theta,\,0)$ as $\mathrm{R}^{in}_{out}(\theta)$. In Figure 2.1, we present two possible compilations of the complex RBS into CNOTs, $\mathrm{R_y}(\theta)$, and $\mathrm{R_z}(\phi) := e^{-i\,\phi\,Z}$ gates. In Appendix 5.1.1.1, we show how to reduce the number of controlled operations in the second compilation by merging the two controlled-Pauli rotations into one controlled-$\mathrm{SU}(2)$ rotation.

The third gate of interest creates a superposition between two computational basis states of arbitrary Hamming distance. Definition 1 below generalizes Eq. (2.2) for two arbitrary computational basis states of possibly different HWs.

Definition 1 (Generalized complex RBS gate (gRBS))**.** *Let* $m, m' \geq 1$ *be integers,* $\mathbf{in} := \{in_1, \ldots, in_m\} \subset [m+m']$*, and* $\mathbf{out} := \{out_1, \ldots, out_{m'}\} := [m+m'] \setminus \mathbf{in}$*. Let* $|1_{\mathbf{in}}\,0_{\mathbf{out}}\rangle$ *(*$|0_{\mathbf{in}}\,1_{\mathbf{out}}\rangle$*) stand for the computational basis state having* m *ones (zeros) at bit positions* $\mathbf{in}$ *and* m' *zeros (ones) at bit positions* $\mathbf{out}$*. We define the generalized complex* RBS *gate* $\mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\,\phi)$ *relative to the input and output sets* $\mathbf{in}$ *and* $\mathbf{out}$ *as the* $(m+m')$*-qubit gate acting as*

$$\begin{aligned}\mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\,\phi)\;|1_{\mathbf{in}}\,0_{\mathbf{out}}\rangle &= e^{i\phi}\;\cos(\theta)\;|1_{\mathbf{in}}\,0_{\mathbf{out}}\rangle + e^{-i\phi}\;\sin(\theta)\;|0_{\mathbf{in}}\,1_{\mathbf{out}}\rangle\\ \mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\,\phi)\;|0_{\mathbf{in}}\,1_{\mathbf{out}}\rangle &= e^{-i\phi}\;\cos(\theta)\;|0_{\mathbf{in}}\,1_{\mathbf{out}}\rangle - e^{i\phi}\;\sin(\theta)\;|1_{\mathbf{in}}\,0_{\mathbf{out}}\rangle\end{aligned} \tag{2.3}$$

on the 2*-dimensional subspace spanned by* $\{|1_{\mathbf{in}}\,0_{\mathbf{out}}\rangle, |0_{\mathbf{in}}\,1_{\mathbf{out}}\rangle\}$*, and as the identity elsewhere.*

Whenever needed, we will write down the gate addresses explicitly, *i.e.*

$$\mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\,\phi) \equiv \mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}(\theta,\,\phi)\,.$$

For $m' = m$, the gRBS gate in Eq. (2.3) is HW-preserving, reducing to Eq. (2.2) if $m' = m = 1$. The latter will be the basic building block of our fixed-Hamming-weight encoder for

[2] The gate $\mathrm{R}^{in}_{out}(-\theta,\,\phi)$ is preferred by some authors and referred to as a Givens rotation.

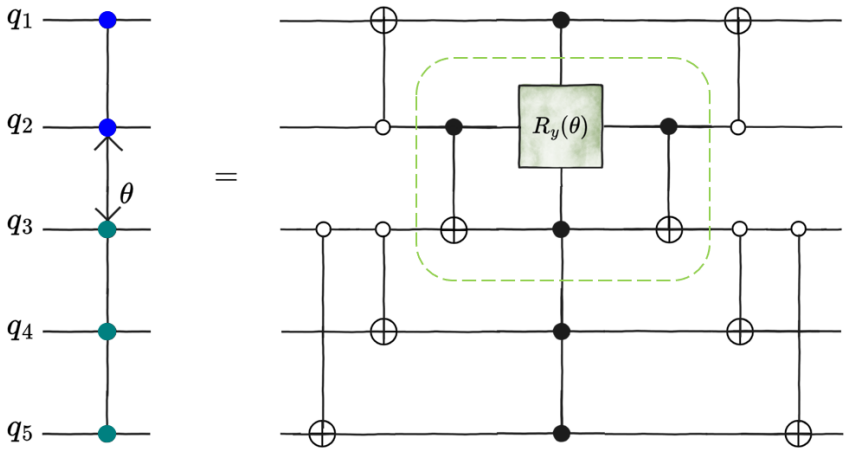

Figure 2.2: **Possible compilation of the** gRBS **gate.** An illustration of the $(m+m')$-qubit gRBS gate for $m=2$ and $m'=3$, namely $\mathrm{R}^{1,2}_{3,4,5}(\theta, \phi=0)$. This gate maps $|00011\rangle$ onto a superposition of $|00011\rangle$ and $|11100\rangle$ with real amplitudes, according to Eq. (2.3). Colors indicate input (blue) and output (green) qubits. We chose the compilation in Fig. 2.1 (*Bottom*) (green dashed box). Controlled-gRBS gates can be built by adding extra controls to the controlled-$\mathrm{R_y}$ rotation. This compilation can be generalized to $\mathrm{R}^{1,2}_{3,4,5}(\theta,\, \phi)$ by replacing the $\mathrm{R_y}$ gate with a generic $\mathrm{SU}(2)$ gate (see App. 5.1.1.1).

dense data in Sec. 2.3. gRBS gates with generic m, m' superpose computational basis states of possibly different HWs and will be used to deal with sparse data in Sec. 2.4 and to build a binary basis encoder in Sec. 2.5.

In Fig. 2.2, we present a possible compilation of the gRBS in Eq. (2.3) when $\phi = 0$ (see App. 5.1.1.1 for the generalization to $\phi \neq 0$). The compilation goes as follows: (i) choose any two qubits, one from the *in* and one from the *out* sets, in which to act with a RBS gate – in this example, $\mathrm{R}^2_3(\theta)$ (green dashed box); (ii) add a network of $m+m'-2$ anti-CNOT gates with the anti-controls on each qubit selected in (i) and the targets on all the other qubits from the respective set (*in* or *out*); (iii) add the RBS gate chosen in (i) controlled by all the remaining qubits from *in*, *out* – in this example, $c_{1,4,5}\ \mathrm{R}^2_3(\theta)$; (iv) undo the anti-CNOT network of step (ii).

As explained in the following sections, our encoders will repeatedly add a new component $|b'\rangle$ to a superposition state $|\psi\rangle$ by acting non-trivially only on the two-dimensional subspace spanned by $|b'\rangle$ and one of the existing components $|b\rangle$ of $|\psi\rangle$. This requires, in general,

controlled-gRBS gates, as detailed in the lemma below.

Lemma 2 (Adding a new amplitude to a state $|\psi\rangle$)**.** *Let $n \geq 2$ be an integer, b, $b' \in \{0,1\}^{\otimes n}$ bitstrings of Hamming weights $|b|$ and $|b'|$, respectively, and $|\psi\rangle$ be an arbitrary n-qubit state having a non-zero amplitude along the computational basis state $|b\rangle$ but not along $|b'\rangle$. Let $\mathcal{I}, \mathcal{I}'$ be the sets of bits where b and b' have a 1, respectively, and define the sets of input, output, control, and anti-control qubits respectively by $\mathbf{in} := \mathcal{I} \setminus \mathcal{I}'$, $\mathbf{out} := \mathcal{I}' \setminus \mathcal{I}$, $\mathbf{ctrl} := \mathcal{I} \cap \mathcal{I}'$, and $\overline{\mathbf{ctrl}} := \mathcal{I}_c \cap \mathcal{I}'_c$. Their respective cardinalities are denoted m, m', ℓ, and ℓ', with $m + m' + \ell + \ell' = n$. Then, the n-qubit multi-controlled gate $\bar{c}_{\overline{\mathbf{ctrl}}}\, c_{\mathbf{ctrl}}\, \mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta, \phi)$ acts on $|\psi\rangle$ by superposing its $|b\rangle$ component with $|b'\rangle$ as in Eq. (2.3) while leaving all the other components untouched. In particular,*

1. *if $m' \geq m$ and $|b\rangle$ is the computational basis state of maximum HW in $|\psi\rangle$, the anti-controls can be removed and the $(n - \ell')$-qubit gate $c_{\mathbf{ctrl}}\, \mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta, \phi)$ superposes $|b\rangle$ with $|b'\rangle$ of HW $|b'| = |b| + (m' - m)$;*

2. *if $m' \leq m$ and $|b\rangle$ is the computational basis state of minimum HW in $|\psi\rangle$, the controls can be removed and the $(n - \ell)$-qubit gate $\bar{c}_{\overline{\mathbf{ctrl}}}\, \mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta, \phi)$ superposes $|b\rangle$ with $|b'\rangle$ of HW $|b'| = |b| - (m - m')$.*

Our HW-k encoders in Sec. 2.3 fall in the scope of the first subcase of the above lemma and, therefore, will be based on multi-controlled gRBS gates. In App. 5.1.1.1 we show that a $c_{ctrl_1,\ldots,ctrl_\ell}\, \mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}(\theta, \phi)$ gate can be compiled using at most $22(m + m') + 20(\ell - 1)$ CNOT gates, with exact compilations provided for $\ell \leq 3$ and $m = m' = 1$, and upper bounds otherwise (see Lemma 9 and Table 5.1 for a summary).

2.3 Hamming-weight-k encoders

We start this section by defining generic amplitude encoders and parameter-optimal encoders. Next, we present our Hamming-weight-k encoder.

2.3.1 Amplitude encoders

Let us define amplitude encoders for an arbitrary set of computational basis states.

Definition 3 (Amplitude encoder)**.** *Let* $\mathbf{x}$ *be any* d*-dimensional data vector,* $\|\mathbf{x}\| = \sqrt{\sum_{j=1}^{d} |x_j|^2}$ *be its* ℓ_2*-norm, and* $B \coloneqq \{|b_j\rangle : b_j \in \{0,1\}^{\otimes n}\}_{j\in[d]}$ *be a set of* d *computational basis states of* n *qubits with* $n \geq \log_2(d)$*. An amplitude encoder in the basis* B *is an* n*-qubit parameterized quantum circuit* $\mathrm{Load}_B(\mathbf{x})$*, with gate parameters depending on* $\mathbf{x}$*, such that*

$$\mathrm{Load}_B(\mathbf{x})\ |0\rangle^{\otimes n} \coloneqq \frac{1}{\|\mathbf{x}\|} \sum_{j=1}^{d} x_j\ |b_j\rangle\,. \tag{2.4}$$

We explicitly distinguish between real-valued data vectors, $\mathbf{x} \in \mathbb{R}^d$, and complex-valued ones, $\mathbf{x} \in \mathbb{C}^d$, since the basic quantum gates will be distinct in each case. Gate parameters in Def. 3 refer to rotation angles such as θ and ϕ introduced in Sec. 2.2. The total number of parameterized gates in a given circuit may vary depending on the specific circuit synthesis. In particular, in the case of variational encoders, usual schemes [24, 38] resort to overparametrization to reach the necessary expressivity to encode an arbitrary vector $\mathbf{x}$. We will present a deterministic protocol to construct circuits using the minimum number of parameters to express any $\mathbf{x}$ exactly. Our construction is based on the observation that, due to the ℓ_2-normalization in Eq. (2.4), we effectively encode the vector $\mathbf{x}/\|\mathbf{x}\|$, which lies in the unit $(d-1)$-sphere for $\mathbf{x} \in \mathbb{R}^d$ or $(2d-1)$-sphere for $\mathbf{x} \in \mathbb{C}^d$. Therefore, expressing an arbitrary data vector requires exactly $d-1$ and $2d-1$ real parameters, respectively. This motivates the following definition of a parameter-optimal encoder.

Definition 4 (Parameter-optimal amplitude encoder)**.** $\mathrm{Load}_B(\mathbf{x})$ *is parameter-optimal if it uses exactly* $d-1$ *gate parameters for* $\mathbf{x} \in \mathbb{R}^d$ *or* $2d-1$ *gate parameters for* $\mathbf{x} \in \mathbb{C}^d$*.*

Efficient quantum data encoders using $\mathcal{O}(d)$ two-qubit gates are known in the unary basis $B_1 = \left\{ |b_j\rangle : b_j \in \{0,1\}^{\otimes n} \text{ and } |b_j| = 1 \right\}_{j\in[d]}$ [20, 21, 22, 23, 39]. However, they require as many qubits as data entries ($n = d$), hence not offering any space compression. The binary basis $B_{\text{binary}} = \left\{ |b_j\rangle : b_j \in \{0,1\}^{\otimes n} \right\}$, with $n = \lceil \log_2(d) \rceil$, gives rise to exponential space

Subroutine 1: Finding the next HW-k bitstring

Input: bitstring b and indices *marked* of marked bits of b
Output: new bitstring b' satisfying $|b'| = |b|$ and $|b \oplus b'| = 2$, and updated list of marked bits

NEXTBS (*b, marked*)
$p \leftarrow \max(marked)$ ▷ index of the pivot bit
if $b(p) = 0$ **then**
  $\ell \leftarrow$ index of nearest 1 to the right of p
else
  $\ell \leftarrow$ index of farthest 0 to the right of p, without passing another 1
$b' \leftarrow b$ with p-th and ℓ-th bits exchanged
$j \leftarrow$ starting index of the last sequence of equal bits in b'
$marked' \leftarrow marked \setminus \{p\} \cup \{p+1, \ldots, j-1\}$
▷ unmark bit p and mark all bits between p and j
return $\{b', marked'\}$

compression, but the corresponding encoders have exponential gate complexity [19]. In the following section, we study amplitude encoding on a Hamming-weight-k subspace, *i.e.*

$$B_k := \Big\{ |b_j\rangle : b_j \in \{0,1\}^{\otimes n} \text{ and } |b_j| = k \Big\}_{j\in[d]}, \tag{2.5}$$

with $k \in [n]$ and $d = \binom{n}{k}$. The amplitude encoder $\mathrm{Load}_{B_k}(\mathbf{x})$ corresponds to the intermediate regime of polynomial space compression, $n \in \mathcal{O}\big(k\, d^{1/k}\big)$. In Sec. 2.5, we construct a binary basis encoder combining HW-k encoders for all k, since $B_{\text{binary}} \equiv \bigcup_{k=0}^{n} B_k$.

2.3.2 HW-k amplitude encoder

In this section, we introduce an amplitude encoder $\mathrm{Load}_{B_k}(\mathbf{x})$ on a Hamming-weight-k subspace B_k given by Eq. (2.5). Hereafter, without loss of generality, the vector $\mathbf{x}$ is assumed to have dimension $d = \binom{n}{k}$. The resulting quantum circuit uses as basic ingredients the HW-preserving gates $\mathrm{R}_{out}^{in}(\theta, \phi)$ introduced in Sec. 2.2. The circuit architecture is decided by a classical algorithm that generates a sequence $[b_j]_{j\in[d]}$ of all HW-k bitstrings ordered such that the Hamming distance $|b_j \oplus b_{j+1}|$ between subsequent bitstrings equals two. To generate these sequences, we use a *Gray code* called Ehrlich's algorithm [60]. The code receives an initial

bitstring b with $|b| = k$ as input and outputs the sequence of all bitstrings of HW k by flipping only a pair of bits, *i.e.* permuting a 0 and a 1, at each step of an iterative procedure. The total number of iterations is exactly $\binom{n}{k} - 1$. For completeness, the algorithm for one such iterative step is presented in Subroutine 1. The fact that $|b_j \oplus b_{j+1}| = 2$ for every j allows one to superpose the corresponding states $|b_j\rangle$ and $|b_{j+1}\rangle$ using (controlled) RBS gates, as explained in Sec. 2.2 (see Eq. (2.2)). In Subroutine 2, we present a routine that, given b_j and b_{j+1}, computes the parameters $\{in, out, ctrl\}$ of the gate needed to superpose $|b_j\rangle$ and $|b_{j+1}\rangle$ as follows: the 1's that remain unchanged between bitstrings b_j and b_{j+1} correspond to control qubits, while the bits that were 1 (0) in b_j and became 0 (1) in b_{j+1} are associated with the input (output) of the RBS gate. Since all bitstrings involved have the same HW, they are uniquely characterized by the position of the 1s in each string. Therefore, we are able to drop all anti-controls that would be necessary otherwise. The angles (θ_j, ϕ_j) are coordinates of the vector $\mathbf{x}/\|\mathbf{x}\|$ on the hypersphere and depend on whether $\mathbf{x} \in \mathbb{R}^d$ or $\mathbf{x} \in \mathbb{C}^d$. They will be presented in Secs. 2.3.2.1 and 2.3.2.2 below.

The general procedures to construct circuits for $\mathbf{x} \in \mathbb{R}^d$ and $\mathbf{x} \in \mathbb{C}^d$ are given, respectively, by Alg. 3 and Alg. 4. These algorithms are classical and have four independent routines that can be summarized as follows:

Summary of Algs. 3 and 4

1. Use the Ehrlich algorithm (Subroutine 1) to generate a sequence $[b_j]_{j\in[d]}$ of HW-k bitstrings with Hamming distances $|b_j \oplus b_{j+1}| = 2$ for all j;
2. Given b_j and b_{j+1}, use Subroutine 2 to compute gate addresses $ctrl$, in, out needed to superpose states $|b_j\rangle$ and $|b_{j+1}\rangle$;
3. If $\mathbf{x} \in \mathbb{R}^d$, compute gate parameter θ_j using Eq. (2.7); if $\mathbf{x} \in \mathbb{C}^d$, calculate (θ_j, ϕ_j) using Eqs. (2.9) and (2.10);
4. Add gate $c_{ctrl}\, \mathrm{R}^{in}_{out}(\theta_j, \phi_j)$ (or $c_{ctrl}\, \mathrm{R}^{in}_{out}(\theta_j)$) to the circuit;
5. Repeat steps 2-4, $\forall\, j$.

We refer to the particular bitstring sequence generated in step 1 of the Summary as the *Ehrlich*

Subroutine 2: Getting gate addresses from b, b'

Input: two bitstrings b and b', and set *untouched* of indices of bits with value 1 in the initial bitstring
Output: Gate parameters $\{in,\ out,\ ctrl\}$ needed to superpose $|b\rangle, |b'\rangle$, and updated list of untouched bits

GATEADDRESSES $(b, b', untouched)$
$\mathcal{I} \leftarrow$ list of indices where b has an 1
$\mathcal{I}' \leftarrow$ list of indices where b' has an 1
$ctrl \leftarrow \mathcal{I} \cap \mathcal{I}'$
$in \leftarrow \mathcal{I} \setminus ctrl$
$out \leftarrow \mathcal{I}' \setminus ctrl$
$untouched' \leftarrow untouched \setminus \{in, out\}$
▷ remove in, out since now they have been touched
$ctrl \leftarrow ctrl \setminus untouched'$ ▷ remove unnecessary controls
return $\{in,\ out,\ ctrl, untouched'\}$

ordering (see Tab. 5.2 in App. 5.1.2 for an example with $n = 6$ and $k = 2$). It is worth noting that if one is required to encode amplitudes into specific pre-established states $|b_j\rangle$ (*e.g.*, preserving standard binary ordering), then one must sort the input data vector $\mathbf{x}$ beforehand to match the Ehrlich ordering. Hereafter, we assume that either $\mathbf{x}$ is sorted in Ehrlich ordering or that no specific ordering order is required by the data. When that is not the case, more complicated gates may be needed to superpose $|b_j\rangle$ and $|b_{j+1}\rangle$ — this is the scope of the sparse-access model encoder discussed in Sec. 2.4. Also, for $k = 1$, the circuit recovers the unary encoding architecture [61].

A priori, the procedure above adds a controlled RBS gate with exactly $k-1$ controls to the circuit at each iteration. This would result in the naive count of $\binom{n}{k} - 1$ controlled RBS gates in total, with $k-1$ controls each. However, several controls can be removed at the initial iterations since the corresponding control qubits are in the state $|1\rangle$ and have not been entangled with any other qubit yet. The variable *untouched* in Subroutine 2 keeps track of these disentangled qubits and is used to remove unnecessary controls from the list of gates returned by the naive algorithm. As a result of this optimization, the circuit resulting from Alg. 3 consists of $\binom{n-(k-\ell)}{\ell+1}$ controlled RBS gates having ℓ controls each, with $\ell + 1 \in [k]$ ($\ell = 0$ corresponds to removing all the

controls and $\ell = k-1$ to no removal). This leads to a significant reduction in the total CNOT-gate count of the circuit. The total number of parameterized gates, $\sum_{\ell=0}^{k-1} \binom{n-(k-\ell)}{\ell+1} = \binom{n}{k} - 1$, remains the same, while the total CNOT-gate count of the optimized HW-k encoder $\mathrm{Load}_{B_k}(\mathbf{x})$ generated by Alg. 3 is

$$\#\,\mathrm{CNOTs} = \sum_{\ell=0}^{k-1} \binom{n-(k-\ell)}{\ell+1} C_\ell \;\leq\; \left[\binom{n}{k} - 1\right] C_{k-1}\,, \tag{2.6}$$

where C_ℓ is the CNOT-gate cost of a single ℓ-controlled $\mathrm{R}_{out}^{in}(\theta,\,\phi)$ gate. The upper bound is the naive count without simplifying initial controls. The latter will be the exact CNOT count whenever the simplification is not possible because the initial state already contains a superposition, such as for the binary encoder in Sec. 2.5. The precise values of C_ℓ depend on the compilation (see App. 5.1.1.1) and also differ for $\phi = 0$ and $\phi \neq 0$. We present explicit expressions for each of these cases in Lemmas 5 and 6. We observed heuristically that choosing $1^k 0^{n-k}$ as the initial bitstring in Subroutine 1 maximizes the removal of redundant controls in the initial iterations. Furthermore, it has the extra benefit of grouping the controls of some of the subsequent controlled RBS gates. This opens the possibility of further reducing the total CNOT cost in the circuit using one clean ancillary qubit to control these groups of gates all at once (see App. 5.1.3). For simplicity, we only present CNOT-gate counts for the ancilla-free implementation.

It is important to mention that, for HW $k > n/2$, one can avoid the costly implementation of $(k-1)$-controlled-RBS gates that would arise from the above procedure by exploring the fact that HW-k bitstrings are the negation of bitstrings of HW-$(n-k)$. Therefore, a more efficient circuit can be built by copying the architecture of an HW-$(n-k)$ encoder circuit with the negated initial bitstring and controlled RBS gates replaced by anti-controlled RBSs. However, this procedure only works if the initial state is not in a superposition.

Finally, the way we construct the circuits leads to obtaining the angles of the RBS gates using the hyperspherical coordinate system. The next two subsections cover the case when $\mathbf{x}$ is real- and complex-valued, respectively.

Algorithm 3: HW-k encoder for dense $\mathbf{x} \in \mathbb{R}^d$

Input: number of qubits n, Hamming weight k, and data vector $\mathbf{x}$ of dimension $d \leq \binom{n}{k}$
Output: Quantum circuit $\mathcal{C} = \mathrm{Load}_{B_k}(\mathbf{x})$

HWEncoder (n, k, $\mathbf{x}$)
$\mathcal{C} \leftarrow \mathrm{Circuit}(n)$
for $i \leftarrow n-k+1$ **to** n **do**
  Add gate X on qubit i to $\mathcal{C}$
$b \leftarrow 1^k 0^{n-k}$ ▷ initial bitstring
$marked \leftarrow \{1, \ldots, k\}$ ▷ indices of marked bits; at first, all qubits initialized at 1 with an X gate
$untouched \leftarrow \{n-k+1, \ldots, n\}$ ▷ indexes of qubits that have never been acted on with a gate yet
for $j \leftarrow 1$ **to** $d-1$ **do**
  $\{b', marked'\} \leftarrow \textsc{NextBS}(b, marked)$
  $\{in, out, ctrl, untouched'\} \leftarrow \textsc{GateAddresses}(b, b', untouched)$
  Compute θ_j from Eq. (2.7)
  Add gate $c_{ctrl}\,\mathrm{R}^{in}_{out}(\theta_j)$ to $\mathcal{C}$
  $\{b, marked, untouched\} \leftarrow \{b', marked', untouched'\}$
return $\mathcal{C}$

2.3.2.1 Dense real-valued data

For $\mathbf{x} \in \mathbb{R}^d$, the $d-1$ angles of the $c_{ctrl}\,\mathrm{R}^{in}_{out}(\theta_j)$ gates used to build the circuit in Alg. 3 are

$$\begin{aligned} \theta_j &:= \mathrm{atan2}\left(\sqrt{\sum_{j'=j+1}^{d} x_{j'}^2}\,,\; x_j\right),\; j \in [d-2] \\ \theta_{d-1} &:= \mathrm{atan2}\left(x_d\,,\; x_{d-1}\right), \end{aligned} \tag{2.7}$$

where the two-argument arctangent function $\mathrm{atan2}(y,\, x) \in [-\pi,\, \pi]$ is the principal value of $\arg(x + i\,y)$. These are the hyperspherical coordinates of the normalized vector $\mathbf{x}/\|\mathbf{x}\| \in \mathbb{S}^{d-1}$,

i.e.

$$
\begin{aligned}
x_1 &= \|\mathbf{x}\| \cos(\theta_1) \\
x_2 &= \|\mathbf{x}\| \sin(\theta_1) \cos(\theta_2) \\
&\vdots \\
x_{d-1} &= \|\mathbf{x}\| \sin(\theta_1) \cdots \sin(\theta_{d-2}) \cos(\theta_{d-1}) \\
x_d &= \|\mathbf{x}\| \sin(\theta_1) \cdots \sin(\theta_{d-2}) \sin(\theta_{d-1}) \,.
\end{aligned}
$$

In Fig. 2.3, we illustrate an example of the resulting quantum circuit for $k = 2$ and $n = 6$ qubits (hence $d = 15$). An explicit step-by-step sketch of the execution of Alg. 3 for the same example is presented in Tab. 5.2 in App. 5.1.2. The following lemma provides the total CNOT-gate count to implement $\mathrm{Load}_{B_k}(\mathbf{x})$ using Alg. 3.

Lemma 5 (Total CNOT cost of $\mathrm{Load}_{B_k}(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^d$)**.** *Let $n \geq 2$ and $k \in [1, n/2]$ be integers, $d = \binom{n}{k}$, and $\mathbf{x} \in \mathbb{R}^d$. The HW-$k$ encoder $\mathrm{Load}_{B_k}(\mathbf{x})$ generated by Alg. 3 can be implemented using a number $\mathcal{O}(k\,d)$ of* CNOT *gates, namely*

$$
\begin{cases}
2(n-1), & k = 1 \\
(n-2)(3n-1), & k = 2 \\
\frac{1}{3}(n-3)(5n^2-6n-2), & k = 3 \\
\frac{1}{12}(n-4)(13n^3-58n^2+79n-42), & k = 4 \\
\frac{1}{12}\sum_{\ell=1}^{4} a_\ell (n-k)^\ell + b_\ell & k \geq 5
\end{cases}
\tag{2.8}
$$

with $a_1 = 178$, $a_2 = 239$, $a_3 = 98$, $a_4 = 13$, and $b_\ell \leq \sum_{\ell=5}^{k-1} \binom{n-(k-\ell)}{\ell+1}\big(16\ell - 6\big)$.

Proof. The total CNOT-gate count is given by the general expression in Eq. (2.6), where C_ℓ is the cost of compiling a single ℓ-controlled $\mathrm{R}_{out}^{in}(\theta)$ gate. We calculate this cost in Lemma 9, and the results are summarized in Tab. 5.1 in App. 5.1.1.1. Plugging the results in Lemma 9 in Eq. (2.6) immediately leads to Eq. (2.8). □

2.3.2.2 Dense complex-valued data

For $\mathbf{x} \in \mathbb{C}^d$, we need two sets of angles, $\{\theta_j\}_{j\in[d-1]}$ and $\{\phi_j\}_{j\in[d]}$, to encode the absolute values and phases of each amplitude $\{x_j = |x_j| \exp(i \arg(x_j))\}_{j\in[d]}$. The $d-1$ angles θ_j are

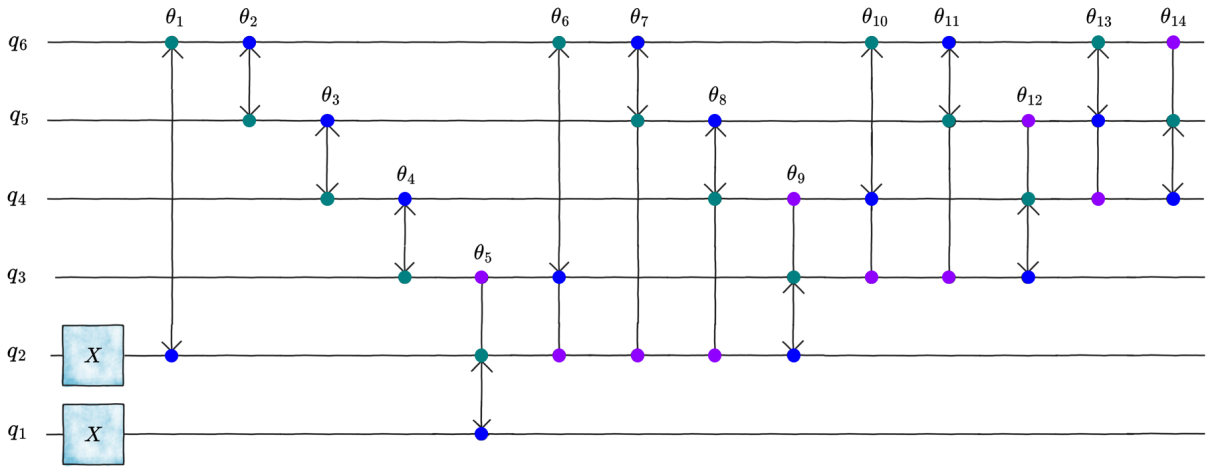

Figure 2.3: **Fixed-Hamming-weight amplitude encoder.** Circuit generated by Alg. 3 with $n = 6$ and $k = 2$ (see Tab. 5.2 for details). Colors are used to indicate input (blue), output (green), and control (violet) qubits of each controlled RBS gate. Angles $\{\theta_j\}_{j\in[14]}$ are calculated using Eq. (2.7).

responsible for the encoding of the absolute values $|x_j|$, and are calculated as the hyperspherical coordinates of the vector $\{|x_1|, \ldots, |x_d|\}$, similarly to Eq. (2.7). Explicitly, the θ_j angles are given by

$$\begin{aligned} \theta_j &:= \operatorname{atan2}\left(\sqrt{\sum_{j'=j+1}^{d} \left|x_{j'}\right|^2},\ |x_j|\right),\ j \in [d-2]\,; \\ \theta_{d-1} &:= \operatorname{atan2}\left(|x_d|,\ |x_{d-1}|\right). \end{aligned} \tag{2.9}$$

The first $d-1$ angles ϕ_j, on the other hand, are responsible for the encoding of the complex phases of each entry $\{x_j\}_{j\in[d-1]}$. The first angle, ϕ_1, is directly related to the complex phase of the first data entry, x_1. Inspecting Eq. (2.2), one can see that, while the complex RBS gate encodes a desired complex phase onto the amplitude of an "initial" computational basis state, it also encodes the complex conjugate of the same phase onto the amplitude of the "new" computational basis state created in the superposition. This leads to the accumulation of undesired phases every time a complex RBS gate is applied. To remove these undesired phases, we add

Algorithm 4: HW-k encoder for dense $\mathbf{x} \in \mathbb{C}^d$

Input: number of qubits n, Hamming weight k, and data vector $\mathbf{x}$ of dimension $d \leq \binom{n}{k}$
Output: Quantum circuit $\mathcal{C} = \mathrm{Load}_{B_k}(\mathbf{x})$

HWENCODER (n, k, $\mathbf{x}$)
$\mathcal{C} \leftarrow \mathrm{Circuit}(n)$
for $i \leftarrow n-k+1$ **to** n **do**
    Add gate X on qubit i to $\mathcal{C}$
$b \leftarrow 1^k 0^{n-k}$ ▷ initial bitstring
$marked \leftarrow \{1, \ldots, k\}$ ▷ indices of marked bits
$untouched \leftarrow \{n-k+1, \ldots, n\}$ ▷ indices of ones that have never been flipped
for $j \leftarrow 1$ **to** $d-1$ **do**
    $\{b', marked'\} \leftarrow \textsc{NextBS}(b, marked)$
    $\{in, out, ctrl, untouched'\} \leftarrow \textsc{GateAddresses}(b, b', untouched)$
    Calculate θ_j and ϕ_j via Eqs. (2.9) and (2.10)
    Add gate $c_{ctrl}\,\mathrm{R}^{in}_{out}(\theta_j, \phi_j)$ to $\mathcal{C}$
    $\{b, marked, untouched\} \leftarrow \{b', marked', untouched'\}$
Calculate ϕ_d via Eq. (2.10)
$ctrl \leftarrow$ set of k indices where the bits of b' are 1
$in \leftarrow$ any index where b' has a 0
Add gate $c_{ctrl}\overline{\mathrm{Ph}}(\phi_d)$ acting on qubit in to $\mathcal{C}$
return $\mathcal{C}$

correction terms to the subsequent angles, $\phi_{j>1}$ as follows:

$$\begin{aligned} \phi_1 &:= -\arg\left(x_1\right), \\ \phi_j &:= -\arg\left(x_j\right) + \sum_{j'=1}^{j-1} \phi_{j'}, \quad j \in [2,\, d]\,. \end{aligned} \tag{2.10}$$

To guarantee numerical stability, all the angles are taken $\mod 2\pi$. In addition to the complex RBS gate defined in Eq. (2.2), we also need to add an anti-phase gate $\overline{\mathrm{Ph}}(\phi) := e^{-i\phi}\,\mathrm{R_z}(\phi/2)$ at the end of the circuit to cancel out the excess complex phase created by the sequence of complex RBS gates, where $\overline{\mathrm{Ph}}(\phi)$ is controlled by all the qubits corresponding to a 1 value in b_{d-1} and acts on any qubit corresponding to a 0 value.

Thus, the resulting quantum circuit has the same architecture of controlled-RBS gates as in the real case (see Fig. 2.3), plus an extra (controlled) $\overline{\mathrm{Ph}}$ gate at the end of the circuit, as well as

the choice of gate synthesis (see Fig. 2.1). The following lemma provides the total CNOT-gate count to implement $\mathrm{Load}_{B_k}(\mathbf{x})$ using Alg. 4.

Lemma 6 (Total CNOT-gate cost of $\mathrm{Load}_{B_k}(\mathbf{x})$ for $\mathbf{x} \in \mathbb{C}^d$)**.** *Let $n \geq 2$ and $k \in [1, n/2]$ be integers, $d = \binom{n}{k}$, and $\mathbf{x} \in \mathbb{C}^d$. The HW-$k$ encoder $\mathrm{Load}_{B_k}(\mathbf{x})$ generated by Alg. 3 can be implemented using a number $\mathcal{O}(k\,d)$ of* CNOT *gates, namely*

$$\begin{cases} 2(n-1), & k=1 \\ (n-2)(3n-1), & k=2 \\ \frac{1}{3}(n-3)(7n^2-12n+2), & k=3 \\ \frac{1}{12}(n-4)(19n^3-86n^2+105n-30), & k=4 \\ \frac{1}{12}\sum_{\ell=1}^{4}\tilde{a}_\ell(n-k)^\ell+\tilde{b}_\ell & k\geq 5 \end{cases} \tag{2.11}$$

with $\tilde{a}_1 = 230, \tilde{a}_2 = 329, \tilde{a}_3 = 142, \tilde{a}_4 = 19$, and $\tilde{b}_\ell \leq \sum_{\ell=5}^{k-1}\binom{n-(k-\ell)}{\ell+1}\big(20\ell+4\big)$.

Proof. The total CNOT-gate count is given by the general expression (2.6), where C_ℓ here is the cost of compiling a single ℓ-controlled $\mathrm{R}_{out}^{in}(\theta,\, \phi \neq 0)$ gate. This is computed in Lemma 9 in App. 5.1.1.1, and plugging in the expression immediately leads to Eq. (2.11). □

2.4 Sparse encoder

Here, we extend the construction of Sec. 2.3 to the case of sparse data. For a d-dimensional data vector $\mathbf{x}$ having $s \ll d$ non-zero entries, the HW-k encoder of Sec. 2.3 still uses $d-1$ parameterized gates. However, in this case, a parameter-optimal amplitude encoder, as expressed in Def. 4, would require only $s-1$ gate parameters.

In this setting, we consider a sparse-access model, where the data vector $\mathbf{y}$ to be encoded is of the form

$$\mathbf{y} \coloneqq \{(x_1,\, b_1),\, (x_2,\, b_2),\, \ldots,\, (x_s,\, b_s)\}\,, \tag{2.12}$$

with $\{x_j\}_{j\in[s]}$ being the non-zero components of $\mathbf{x}$, and $\{b_j\}_{j\in[s]}$ being the addresses (in bits-tring format) associated with these values. The goal is to prepare the state

$$s\text{-}\mathrm{Load}(\mathbf{y})\ |0\rangle^{\otimes n} \coloneqq \frac{1}{\|\mathbf{x}\|}\sum_{j\in[s]} x_j\ |b_j\rangle\ , \tag{2.13}$$

Algorithm 5: Amplitude encoder for sparse data

Input: number of qubits n and tuple $y \coloneqq \{(x_i, b_i)\}_{i\in[s]}$ of non-zero data values x_i and their addresses b_i
Output: Quantum circuit $\mathcal{C} = s\text{-Load}(y)$

SPARSEHWENCODER (n, y)
$bs \leftarrow b_1$ ▷ address b of the 1st element of the y tuple
$\mathcal{I} \leftarrow$ set of indices where the bits of bs have value 1
$\mathcal{C} \leftarrow \text{Circuit}(n)$
for $i \leftarrow 1$ **to** $|\mathcal{I}|$ **do**
  Add gate X on qubit $\mathcal{I}_i$ to $\mathcal{C}$
$untouched \leftarrow [n] \setminus \mathcal{I}$
for $j \leftarrow 2$ **to** s **do**
  $bs' \leftarrow b_j$
  $\{in, out, ctrl, untouched'\} \leftarrow$ GATEADDRESSES$(bs, bs', untouched)$
  Calculate θ_j and ϕ_j via Eqs. (2.9) and (2.10)
  ▷ using x values from y
  Add gate $c_{ctrl}\,\mathrm{R}^{in}_{out}(\theta_j, \phi_j)$ to $\mathcal{C}$
  $\{b, untouched\} \leftarrow \{b', untouched'\}$
Calculate ϕ_d via Eq. (2.10)
$ctrl \leftarrow$ set of k indices where the bits of b' are 1
$in \leftarrow$ any index where b' has a 0
Add gate $c_{ctrl}\overline{\mathrm{Ph}}(\phi_d)$ acting on qubit in to $\mathcal{C}$
return $\mathcal{C}$

similarly to the fixed HW encoders discussed in Sec. 2.3. However, in the sparse-access model, b_j and b_{j+1} do not need to satisfy any constraints, *e.g.* equal HWs or fixed Hamming distance. Consequently, the complex RBS gate used in the previous sections is not enough to cover all possible sparsity configurations. To this end, we need the generalized RBS gate, $\mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\, \phi)$ in Eq. (2.3). With this $(m + m')$-qubit gate, it is possible to create superpositions of computational basis states of different Hamming weights.

The procedure to build the circuit is given in Alg. 5 for complex data and goes as follows. Based on the first bitstring address b_1, we apply Pauli-X gates to generate the initial state $|b_1\rangle$. Then, we use Subroutine 2 to extract the generalized RBS gate parameters (input, output, and control qubits) needed to generate the superposition between $|b_1\rangle$ and $|b_2\rangle$, and add the gate

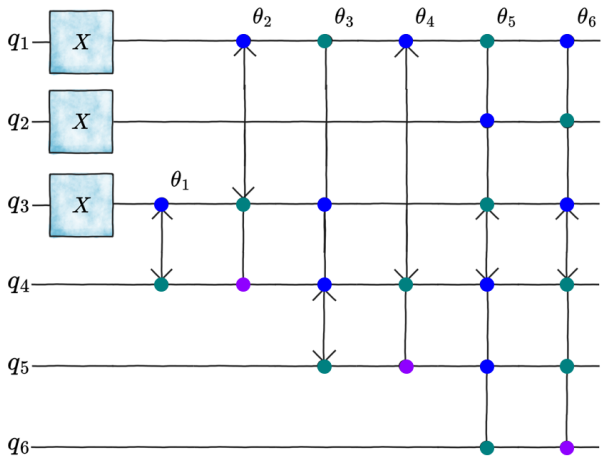

Figure 2.4: **Sparse quantum data loader.** An example with $n = 6$ qubits, and sparsity $s = 7$ with $x_j \neq 0$ at positions $b_j \in \{7, 11, 14, 19, 26, 37, 58\}$ for real data. Colors are used to indicate the input (blue), output (green), and control (violet) qubits of each gRBS gate. Angles are calculated using Eq. (2.7) for the non-zero entries. The execution of Alg. 5 to construct this circuit is shown in Tab. 5.3 in App. 5.1.2. This circuit can be compiled using at most 174 CNOTs (see Tab. 5.1 in App. 5.1.1.1).

$\mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}(\theta_1,\,\phi_1)$ to the circuit; The angles are computed from Eqs. (2.9) and (2.10). The same procedure is repeated for all elements in the tuple in Eq. (2.12), adding $s-1$ (possibly controlled and generalized) RBS gates to the resulting circuit. Subroutine 2 can output lists of inputs and outputs with different lengths (*i.e.* $m \neq m'$), depending on whether an increase/-decrease of Hamming-weight is needed to superpose $|b_j\rangle$ and $|b_{j+1}\rangle$. The ancilla-free circuit architecture output by Alg. 5 will depend on the particular sparsity structure of the data vector $\mathbf{y}$. In the case of real-data encoding, we compute the angles θ_j from Eq. (2.7) instead of calculating θ_j and ϕ_j via Eqs. (2.9) and (2.10) in addition to not executing lines $18-21$ of Alg. 5. The worst-case bound on the total number of CNOTs can be calculated using the compilation of gRBS gates presented in Tab. 5.1 in App. 5.1.1.1. An explicit example for $n = 6$ qubits and sparsity $s = 7$ is illustrated in Fig. 2.4, and in Tab. 5.3 in App. 5.1.2.

In Fig. 5.1(a) in App. 5.1.1.2, we show a numerical comparison between the encoders from Alg. 5, Ref. [62] (as implemented in the `qclib` package [63]), and Ref. [64] (as implemented in the `Qiskit` package [65]). We compare, as a function of the number of qubits n, the

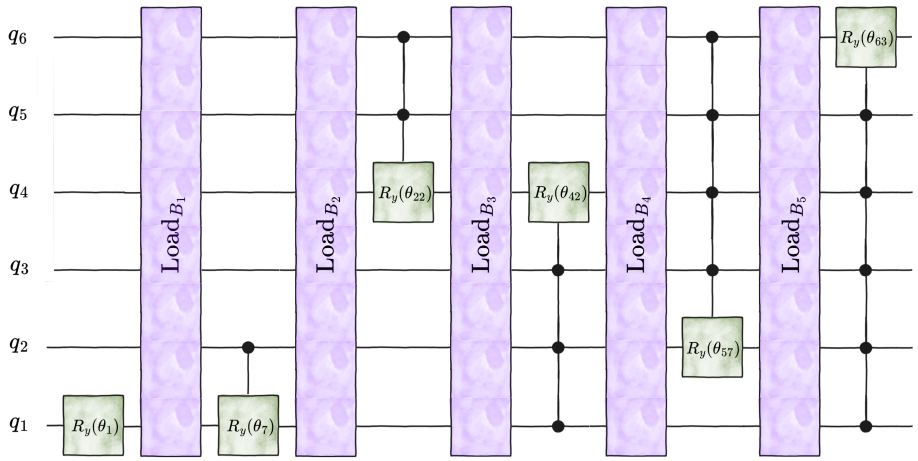

Figure 2.5: **Binary-basis amplitude encoder based on HW-k encoders.** Encoding of a real, 64-dimensional vector $\mathbf{x}$ on a 6-qubit quantum state. The initial state is the zero state, $|0^6\rangle$, and the circuit is constructed by stacking HW-k encoders, Load_{B_k}, with $k \in [0,\, 6]$, and (controlled) $\mathrm{R_y}$ rotations. An initial $\mathrm{R_y}(\theta_1)$ gate creates a superposition between $|0^6\rangle$ and $|0^5 1\rangle$. Subsequently, the bitstring $0^5 1$ serves as the input of the Ehrlich Algorithm (Subroutine 1) that generates the circuit Load_{B_1}, responsible for the encoding of amplitudes associated with all 6 computational basis states of HW-1. Next, based on the last computational basis state added to the superposition by the previous Load_{B_1}, the controlled $\mathrm{R_y}(\theta_7)$ rotation adds the first computational basis state of HW-2. The bitstring associated with this state is then used as input for the next Ehrlich Algorithm. The procedure is repeated until the last amplitude is encoded in the $|1^6\rangle$ state. The gate parameters $\{\theta_j\}_{j\in[63]}$ are calculated using Eq. (2.7). See Tab. 5.4 in App. 5.1.2 for more details. This circuit can be compiled using at most 1048 CNOTs (see Tab. 5.1 in App. 5.1.1.1).

average CNOT-gate count to encode 100 random 2^n-dimensional data vectors $\mathbf{y}$ of sparsity $s \in \{n,\, n^2\}$. For each instance, we used a Haar-random vector of non-vanishing amplitudes $\mathbf{x}$ and uniformly sampled computational basis state addresses $|b_j\rangle$. The results indicate that the average CNOT-gate count of Alg. 5 and of Ref. [62] scales as $\mathcal{O}(\mathrm{poly}(n))$ for both sparsity levels s. Meanwhile, the algorithm from Ref. [64] scales as $\mathcal{O}(2^n)$ regardless of the sparsity s. Moreover, Ref. [62]'s encoder displays an improvement of $\mathcal{O}(n)$ when compared to the results for Alg. 5. However, while our sparse encoder is ancilla-free, the sparse encoder from Refs. [62, 63] uses $\mathcal{O}(n)$ ancillas.

2.5 Binary encoder

In this section we show how to stack together the dense HW-k encoders introduced in Sec. 2.3 to encode data on multiple constant-HW subspaces (e.g., $\mathrm{Load}_{B_{k_1}\cup B_{k_2}}$ with $k_1 < k_2$). The strategy is to populate the different fixed HW subspaces in ascending order while using a generalized RBS gate to connect the two subsequent subspaces. For instance, $\mathrm{Load}_{B_{k_1}\cup B_{k_2}}$ can be built as $\mathrm{Load}_{B_{k_1}}$ followed by a gRBS gate that increases k_1 to k_2 and $\mathrm{Load}_{B_{k_2}}$; in the special case of successive HWs, *i.e.* $k_2 = k_1 + 1$, the gRBS can be replaced by a k_1-controlled R_y gate (see Fig. 2.5).

In particular, this strategy can be used to build a full binary-basis amplitude encoder by stacking dense HW-k encoders of all possible $0 \leq k \leq n$ in ascending order. We will first explain the general ideal for real-valued data and will later generalize it to complex data. The procedure goes as follows: (i) first, initialize the circuit in the state $|0^n\rangle$; (ii) apply a R_y gate in the last qubit to generate the superposition $\cos\theta_1 |0^n\rangle + \sin\theta_1 |10^{n-1}\rangle$ according to Eq. (2.1); (iii) use 10^{n-1} as the initial bitstring of the Ehrlich algorithm (Subroutine 1), creating the circuit Load_{B_1}, which is responsible for adding all HW-1 computational basis states to the superposition; (iv) apply a controlled R_y rotation to add the first computational basis state of HW-2 to the superposition; (v) use the state encoded in the previous step as input of the algorithm that will generate Load_{B_2}; (vi) iterate over steps (iv) and (v) until all HWs are populated, and the last multi-controlled R_y generates the superposition with $|1^n\rangle$. The (multi-)controlled R_y gates applied between Load_{B_k} and $\mathrm{Load}_{B_{k+1}}$ should be chosen such that the last computational basis state added to the superposition is associated to a bitstring that is a viable input for the Ehrlich algorithm [60, Theorem 2.4]. Alternatively, we can use the Hamming-weight-increasing step (iv) using a HW-mixing gRBS gate $R^{\mathbf{in}}_{\mathbf{out}}(\theta, \phi)$ introduced in Eq. (2.3). However, since the operations necessary for the binary encoder are local, we chose to use the cheaper multi-controlled R_y gates (see Tab. 5.1 in App. 5.1.1.1).

In case of complex amplitudes, the (controlled) $R_y(\theta_j)$ rotations must be replaced by (con-

trolled) $\mathrm{Ph}(\phi_j)\, R_y(\theta_j)$ to generate the superposition with the correct phases. The angles $\{\theta_j\}_{j\in[d-1]}$ and $\{\phi_j\}_{j\in[d-2]}$ are calculated using Eqs. (2.9) and (2.10). The last multi-controlled R_y gate, however, must be replace by a multi-controlled $\mathrm{SU}(2)$ rotation that can be decomposed as $R_z(\phi_{d-1})\, R_y(\theta_{d-1})\, R_z(\lambda)$. The angles ϕ_{d-1} and λ are calculated as

$$\begin{aligned} \phi_{d-1} &= \frac{1}{2}\left(\arg(x_d) - \arg(x_{d-1})\right) \,; \\ \lambda &= -\frac{1}{2}\left(\arg(x_d) + \arg(x_{d-1})\right) + \sum_{j=1}^{d-3} \phi_j \,. \end{aligned} \tag{2.14}$$

An example of the binary encoder for $n = 6$ qubits is shown in Fig. 2.5. The total CNOT-gate count is given in the following lemma.

Lemma 7 (CNOT count of the binary encoder)**.** *Our binary encoder for* $\mathbf{x} \in \mathbb{R}^{2^n}$ *uses*

$$\#\,\mathrm{CNOTs} \leq \frac{1}{12}(13\,n^4 - 58\,n^3 + 119\,n^2 - 50\,n + 120) + \sum_{k=5}^{n-1} \left[\binom{n}{k} (16\,k - 22) - 2 \right] . \tag{2.15}$$

Proof. First, it is important to notice that the two "tricks" mentioned in Sec. 2.3 to reduce the number of controls (*i.e.* eliminating initial controls and constructing an HW-$(n-k)$ encoder from the corresponding HW-k encoder) cannot be used here, since the initial state for each Load_{B_k} in Fig. 2.5 contains a superposition of states. Therefore, the cost of each Load_{B_k} is given by the upper bound $\left[\binom{n}{k} - 1\right] C_{k-1}$ in the general expression in Eq. (2.6), where C_{k-1} is the cost of compiling a single $(k-1)$-controlled-$\mathrm{R}_{out}^{in}(\theta)$ gate. After each Load_{B_k} is applied, one needs a k-controlled $\mathrm{R_y}$ gate to increase the HW to $k+1$. The total CNOT count of the circuit is therefore $\sum_{k=0}^{n-1} \left(\left[\binom{n}{k} - 1\right] C_{k-1} + \chi_k\right)$, where χ_k is the CNOT cost of a k-controlled $\mathrm{R_y}$. Substituting the values of C_{k-1} and χ_k calculated in App. 5.1.1.1 immediately leads to Eq. (2.15) See Tab. 5.1 for a summary, and Lemmas 8 and 9 for the derivation. The first line contains exact CNOT counts coming from $k = 1$ to 4, while the second line contains upper bounds coming from $k \geq 5$. □

Numerically, we observe the CNOT-gate count of Eq. (2.15) to be $\leq 8\,n\,2^n = n\,2^{n+3}$ (see Fig. 5.1(b) in App. 5.1.1.2). This scaling is close to the minimum scaling in practice for

ancilla-free implementations. Asymptotically, this is the same scaling obtained by the sparse encoder from Ref. [62]. While the CNOT-gate count of Ref. [19] is $\mathcal{O}(2^n)$, they perform their amplitude encoding in the Schmidt basis instead of the computational basis, with at most $2^{n/2}$ coefficients. For generic states, this requires classically preprocessing the state via a singular value decomposition, a process that costs $\mathcal{O}(2^{3n})$. In contrast, we explore all 2^n computational basis states without the aforementioned classical preprocessing overhead. Reference [18] has a CNOT count of exactly $4\,(2^n - n - 1)$ gates. However, their implementation uses quantum gates that suffer from numerical instabilities and therefore are not scalable in practice.

2.6 Experimental and numerical results

In this Section, we show a proof-of-principle deployment of the dense HW-k encoder on the IonQ 's `Aria-1` quantum processor. We also numerically demonstrate that the encoder circuits can be used as an ansatz for a variational quantum algorithm. Then, we analyze the fidelity of the encoder under circuit noise.

2.6.1 Quantum hardware demonstration

To demonstrate our encoding protocol, we encode a q-Gaussian probability distribution on the `Aria-1` quantum processor from IonQ [57]. The q-Gaussian probability density function $p_{q,\beta}(x)$, where $q \in (-\infty, 3)$ and $\beta \in (0, \infty)$, is proportional to the *q-exponential* function $e_q\left(-\beta\, x^2\right)$, defined as e^x if $q = 1$; $[1 + (1-q)x]^{1/(1-q)}$ if $q \neq 1$ and $1 + (1-q)x > 0$; and $0^{1/(1-q)}$ otherwise [66, 67, 68]. This family of distributions has a wide range of applications, *e.g.* nonextensive statistical mechanics [69], finance [70, 71], metrology [72], and biology [73]. It is worth noting that in general the q-Gaussian is a non-log-concave distribution, falling outside of the scope of other encoding strategies [40]. Here, we chose the parameters $q = 3/2$ and $\beta = 2$.

We use the circuit in Fig. 2.3 for 6 qubits and Hamming weight $k = 2$, which allows us to encode a data vector of size $d = \binom{6}{2} = 15$. This vector corresponds to a discretization of

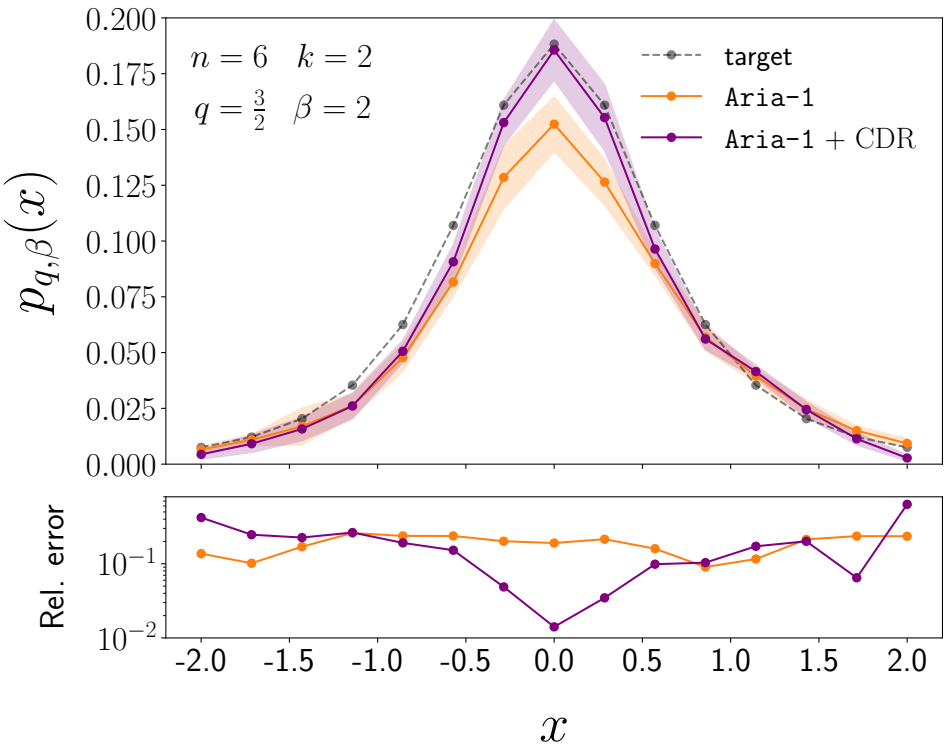

Figure 2.6: **Experimental deployment on IonQ .** *(Top)* Experimental implementation of the encoding of a q-Gaussian probability distribution $p_{q,\beta}(x)$ on IonQ 's `Aria-1` processor. The domain of $p_{q,\beta}(x)$ was truncated to $x \in [-2, 2]$ and discretized into $d = 15$ points (dashed black line). The circuit deployed used $n = 6$ and $k = 2$ and is shown in Fig. 2.3. The solid purple (orange) line represents experimental results with (without) error mitigation using Clifford Data Regression (CDR). Shaded areas represent the uncertainty regions estimated via bootstrapping. *(Bottom)* Relative error of experimental implementation of $p_{q,\beta}(x)$ w.r.t. the target function, in log scale.

the distribution truncated to the interval $x \in [-2, 2]$. For this circuit, the first 4 parameterized rotations are uncontrolled RBS gates, while the last 10 RBSs require 1 control each. Each RBS gate is compiled using 2 CNOT gates, while each controlled RBS requires 6 CNOTs, leading to a circuit with 68 CNOTs in total (see Tab. 5.1 in App. 5.1.1.1). However, the `Aria-1` processor uses the Mølmer-Sørensen (MS) gate as the two-qubit native gate [57]. In this case, it is possible to implement one-controlled $\mathrm{R_y}$ rotations using only one MS gate. This allows us to implement the aforementioned circuit using 48 MS gates.

In Fig. 2.6, we show the results of the experimental implementation. In the top panel, we

plot the probability distribution estimated from the experiment by running the circuit 10^4 times and measuring on the computational basis on each qubit to recover the encoded data. The solid orange line shows the empirical probability distribution estimated from the raw experimental data from the `Aria-1` quantum processor. The solid purple line shows the estimated probability distribution after error mitigation using *Clifford Data Regression* (CDR) [58, 59]. CDR models hardware noise by simulating near-Clifford circuits similar to the one at hand and then applying the trained response function to raw experimental data. The training of the response functions was performed using IonQ's capabilities for classical simulation of noisy circuits (see App. 5.1.4 for details). The dashed black line shows the ideal target distribution. All relative errors concerning the target distribution are plotted in Fig. 2.6 *(Bottom)* in $\log$ scale. We see that the data recovered from the experiment differs from the target distribution by 10-$30\,\%$ in relative error. After CDR is applied to the experimental data, relative errors can be improved. For instance, the CDR protocol implemented improved the relative error at the peak of the distribution by one order of magnitude. The improvement can also be seen in the overall state infidelity $1 - \mathrm{Tr}\left(\rho\,\sigma\right)$ (σ being the state prepared by the noisy circuit), which decreased from 0.17 to 0.08 after error mitigation. We attribute these discrepancies in the experimental results to the fact that `Aria-1` is a NISQ device. In general, noisy implementations of Hamming-weight-preserving gates will result in an effective channel that is not Hamming-weight-preserving. Consequently, amplitudes are created in the Hilbert space "outside" of the subspace of interest, while there is a reduction in the probability density "inside" said subspace. We showed that this effect can be reduced by the application of error mitigation techniques such as CDR .

2.6.2 HW-k encoder under local depolarizing noise

Motivated by the results in the Sec. 2.6.1, in this section we numerically investigate the robustness of the HW-k encoder in the presence of circuit noise as a function of both k and the noise strength. For the noise model, given a pure quantum state ρ and any pair of qubits labeled

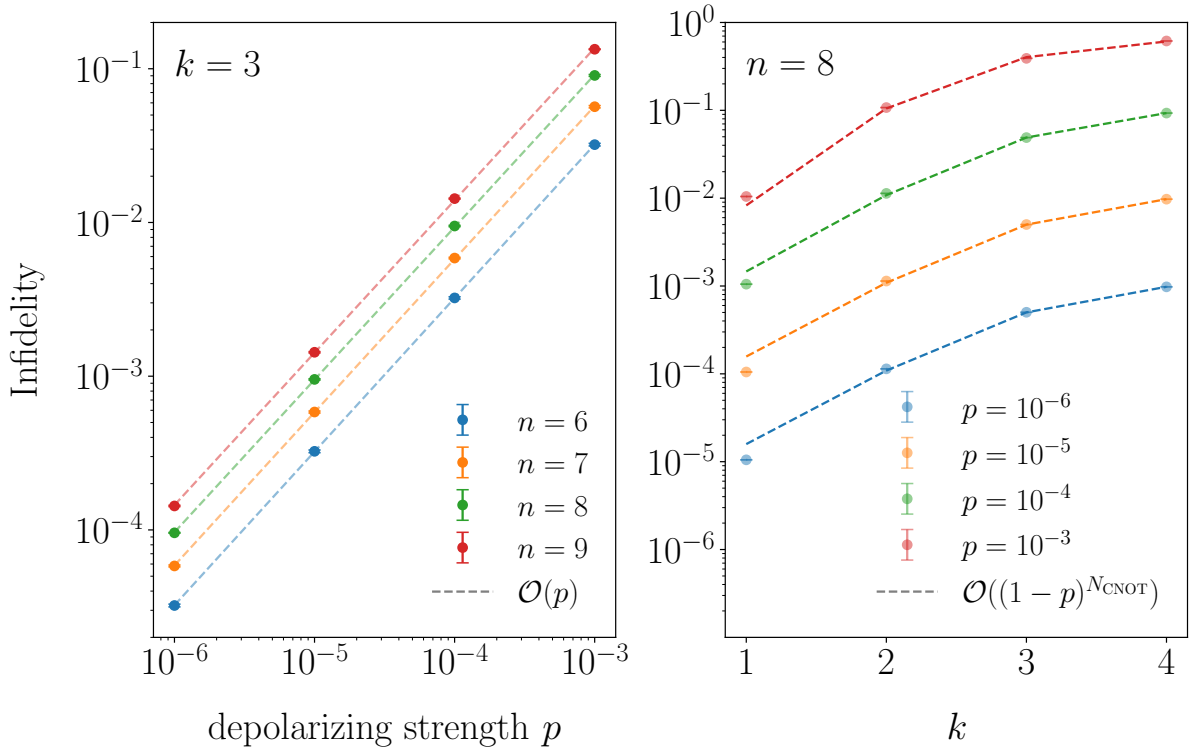

Figure 2.7: **Performance of HW-k encoder under local depolarizing noise model.** Circuits for real-valued data are compiled into CNOTs and single-qubit gates (see Secs. 2.2 and 2.3.2.1). Each CNOT is followed by a two-qubit depolarizing channel of strength p. (*Left*) State infidelity as a function of depolarizing strength p, averaged over 10 Haar-random instances, for fixed $k = 3$ and $n \in [5, 9]$. Dashed lines represent a linear fit $\mathcal{O}(p)$. (*Right*) State infidelity as a function of k, average over 10 Haar-random instances for fixed $p \in \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$ and $n = 8$. Dashed lines represent curve fit of the function $\mathcal{O}\left((1-p)^{N_{\mathrm{CNOT}}}\right)$ to the numerical data, where $N_{\mathrm{CNOT}} \equiv N_{\mathrm{CNOT}}(n, k) = \mathcal{O}\left(k \binom{n}{k}\right)$ is the number of CNOTs in each circuit for a given n and k.

by A, we assume every CNOT gate acting on A is followed by a depolarizing channel $\mathcal{D}_p$,

$$\mathcal{D}_p(\rho) = (1-p)\,\rho + p\ \mathrm{Tr}_A(\rho) \otimes \frac{\mathbb{1}}{4}\,, \tag{2.16}$$

where p is the depolarizing strength, $\mathbb{1}$ is the two-qubit identity matrix, and $\mathrm{Tr}_A(\cdot)$ is the partial trace over A. This noise model is compatible with reported noise models derived from randomized benchmarking techniques [74]. Since two-qubit gate errors are the main source of circuit noise in NISQ devices, we assume that all single-qubit gates are noiseless. We compiled the circuits resulting from Alg. 3 into CNOTs, single-qubit gates, and multi-controlled R_y rotations as described in Secs. 2.2 and 2.3.2.1, and App. 5.1.1.1. While the circuits were simulated using

the `Qibo` package [75], the multi-controlled R_y gates were compiled into CNOTs and single-qubit gates using the integration between the `qclib` library [63] and the `Qiskit` package [65].

In Fig. 2.7 (left panel), we show a log-log plot of the resulting infidelity for fixed $k = 3$ as a function of the depolarizing strength p for $n \in [5, 9]$. We observed that for fixed n and k, the average infidelity scales polynomially with p, namely $\mathcal{O}(p^\gamma)$ with $\gamma \approx 1$. For $p \sim 10^{-3}$, which is a per-gate noise level compatible with several of the currently available quantum platforms [76, 77, 78, 9], the infidelities in the range of n studied are in between 10^{-2} and 10^{-1}. In Fig. 2.7 (right panel), we plot the infidelity (in log scale) as a function of k for fixed $n = 8$, and $p \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\}$. We observed that for fixed n and p, the average infidelity scales exponentially with k as $\mathcal{O}\Big((1-p)^{N_{\mathrm{CNOT}}(n,k)}\Big)$, where $N_{\mathrm{CNOT}}(n, k)$ is the total CNOT-gate count given by Theorem 5. This scaling can be intuitively explained by the fact that the depolarizing channel is isotropic, namely, the composition of N_{CNOT} local depolarizing channels is equivalent to a global depolarizing channel with strength $1 - (1 - p)^{N_{\mathrm{CNOT}}}$. As expected from NISQ devices, the accumulation of errors on deep circuits leads to the preparation of states with high infidelity concerning the target states. However, as reflected by the linear scaling in p, we see that each order-of-magnitude reduction in p yields also an order-of-magnitude reduction in infidelity.

2.6.3 HW-k encoder as a variational quantum ansatz

Here, we demonstrate the use of the encoders presented in Secs. 2.3-2.5 as a variational quantum ansatz. We focus on the HW-k encoder of dense real-valued data, though results immediately extend to complex-valued or sparse data as well as binary encoders.

We first recall that, given n and k, the quantum circuits generated by Alg. 3 parameterize the subspace spanned by the basis B_k of size $d = \binom{n}{k}$ defined in Eq. (2.5). Therefore, this parametrization is ideally suited for solving optimization tasks that have some type of Hamming weight constraint. For instance, this is the case for ground-state energy estimation of

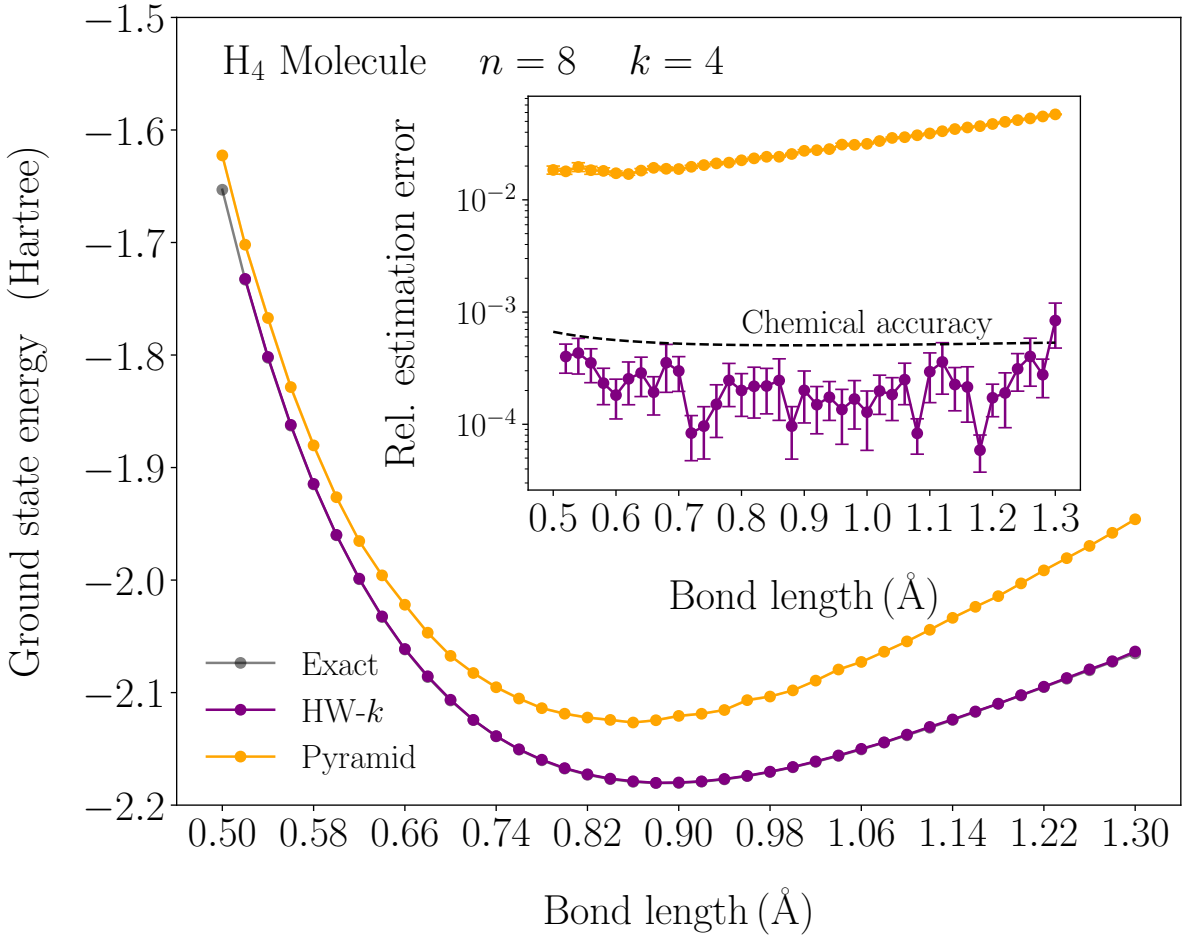

Figure 2.8: **Hamming-weight-k encoder as a variational quantum ansatz.** (*main*) Ground state energy (in Hartrees) for the H_4 molecule as a function of the bond length (in Angstroms). The purple curve represents the energies obtained using Alg. 3 with $n = 8$ and $k = 4$, while the orange curve was obtained using the "pyramid" circuit architecture from Ref. [38]. The number of gradient descent epochs is fixed in both cases to 1000, results are averaged over 50 random initializations and error bars are smaller than the markers. The *exact* energy curve calculated from brute-force diagonalization is plotted in gray and is nearly indistinguishable from the purple curve. (*inset*) Energy estimation error relative to the exact ground state energy as a function of the bond length. The color scheme is the same as in the main plot. Relative chemical accuracy is plotted as a dashed black line.

molecular Hamiltonians [36, 37], which have particle number-preserving symmetry, as well as portfolio optimization constrained by a fixed number of assets [54]. Without loss of generality, here we illustrate this feature using a quantum chemistry example. Given a molecule having k electrons and a basis set describing its orbitals, the Jordan-Wigner transform maps its fermionic Hamiltonian into a qubit Hamiltonian H where each qubit corresponds to one of the

n molecular spin-orbitals and the state $|1\rangle$ ($|0\rangle$) is associated with an occupied (unoccupied) spin-orbital. The ground state of the qubitized molecular Hamiltonian is therefore constrained to a subspace B_k of the form (2.5), and one can use the corresponding HW-k circuit Load_{B_k} as a variational ansatz to approximate the ground state by classically optimizing the RBS gate angles $\boldsymbol{\theta} \coloneqq \{\theta_j\}_{j\in[d-1]}$ to minimize the energy function

$$E(\boldsymbol{\theta}) \coloneqq \langle 0^n|\ \mathrm{Load}^\dagger_{B_k}(\boldsymbol{\theta})\, H\ \mathrm{Load}_{B_k}(\boldsymbol{\theta})\ |0^n\rangle\ . \tag{2.17}$$

Below we numerically demonstrate this by finding the ground state energy of the H_4 molecule, with the qubitized Hamiltonian obtained via the quantum dataset from `Pennylane` [79, 80]. Using the $\mathrm{STO\text{-}3G}$ basis set, this molecule can be described by a $n = 8$ qubit Hamiltonian (corresponding to 8 spin-orbitals) with $k = 4$ electrons in the ground state. The results are presented in Fig. 2.8. In the main panel, we plot the ground state energy (in Hartrees) of the H_4 molecule as a function of the bond length (in Angstroms) between the Hydrogen atoms. We estimate the energies using our amplitude encoder Load_{B_4} as an ansatz (solid purple line) and compare our results with the "pyramid" HW-preserving ansatz from Ref. [55] (solid orange line), composed solely of uncontrolled RBS gates. For a fair comparison, we allow this ansatz to have the same number of parameters as ours, *i.e.* $\binom{n}{k} - 1 = 69$ parameters. We did not observe any improvement in the expressivity of the pyramid ansatz by adding more parameters beyond this number, hence we did not compare the two ansatze by equating their number of two-qubit gates. We leveraged the integration between `Qibo` and `PyTorch` [81] and used the ADAM optimizer [82] with fixed learning rate $\eta = 10^{-3}$ as the gradient descent method of choice for both ansatze. We plot the best energy results after 1000 epochs, averaged over 50 Haar-random initializations. The "exact" ground state energies, calculated via brute-force diagonalization of the corresponding Hamiltonians, are plotted in a solid black line. We see that our encoder approximates very well the ground state energy of the H_4 molecule throughout the entire energy curve, unlike the pyramid ansatz, whose results worsen with increased bond length. The *inset* of Fig. 2.8 shows the energy estimation errors relative to the exact ground

state energies. We see that, after 1000 epochs, the HW-k ansatz on average reached chemical accuracy while the pyramid ansatz remained at least an order of magnitude away throughout the entire bond length domain.

2.7 Conclusions

We provide an efficient and explicit classical algorithm to construct, gate by gate, a quantum circuit that uploads an arbitrary data vector into a subspace of fixed Hamming-weight quantum states. The quantum circuit uses the minimum number of parameterized gates needed to express generic data of a given dimension. The construction uses (generalized) RBS gates and allows us to precisely state the quantum resources needed for its execution, as well as deploy a proof-of-concept instance on real quantum hardware. We also provide the tools needed to further explore quantum encoders for this subspace and beyond.

As shown for the sparse and binary encoder, our HW-k encoder can be used as a subroutine to power more complex algorithms. These two examples are but a small sample, and future work will explore more avenues. Still, we remark on the importance of binary encoding, as most algorithms would benefit from robust state preparation work on this basis. Other encoding schemes, such as the unary-basis encoder, can deploy basis change circuits [83] to allow for further post-processing. A generalization of this circuit to the HW-k basis would open new possibilities for more efficient general state preparation.

Furthermore, our algorithm brings interesting implications from the lens of quantum machine learning. Variational ansatze that explore a constrained space are popular due to their ability to mitigate barren plateaus [24]. As shown by the numerical experiment with the H_4 molecule, our HW-k encoder, due to its expressivity, has the potential to improve the performance of optimizations that involve particle-preserving symmetry. Further investigation of this potential is needed, and we leave it as an open question for future work.

Algorithms with space compressions that go beyond linear, as is the case with our HW-k encoder, are also a promising direction for other machine learning schemes [84], and show how

exploring these subspaces can be successful.

Lastly, we believe that explicit constructive algorithms for state preparation, with precise analysis of the quantum resources required, are essential to reach a useful quantum advantage.

2.8 Article acknowledgments

We thank Ariel Bendersky and André J. Ferreira-Martins for insightful discussions. We thank Jadwiga Wilkens for the availability of the *Quantum Circuit Library* [85].

Chapter 3

Robust ultra-shallow shadows

R. M. S. Farias[1,2]*, R. D. Peddinti*[1]*, I. Roth*[1]*, L. Aolita*[1]

[1]*Quantum Research Center, Technology Innovation Institute, Abu Dhabi, UAE*

[2]*Instituto de Física, Universidade Federal do Rio de Janeiro, P.O. Box 68528, Rio de Janeiro, Rio de Janeiro 21941-972, Brazil*

We present a robust shadow estimation protocol for wide classes of low-depth measurement circuits that mitigates noise as long as the effective measurement map, including noise, is locally unitarily invariant. This is, in practice, an excellent approximation, encompassing, for instance, the case of ideal single-qubit Clifford gates composing the first circuit layer of an otherwise arbitrary circuit architecture and even non-Markovian, gate-dependent noise in the rest of the circuit. We argue that for weakly-correlated local noise, the measurement channel has an efficient matrix-product representation, and show how to estimate this directly from experimental data using tensor-network tools, eliminating the need for analytical or numeric calculations. We illustrate the relevance of our method with both numerics and proof-of-principle experiments on an IBM Quantum device. Numerically, we show that unmitigated shallow shadows with noisy circuits become more biased as the depth increases. In contrast, using the same number of samples, robust ultra-shallow shadows become more precise with increasing depth for relevant parameter regimes. The gain in sample efficiency is still limited by the noise per gate, resulting

in an optimal circuit depth per noise level. Experimentally, we observe improved precision in two simple fidelity estimation tasks using five-qubit circuits with up to two layers of entangling gates, by about an order of magnitude. Under the practical constraints of current and near-term noisy quantum devices, our method maximally realizes the potential of shadow estimation with global rotations and identifies its fundamental limitations in the presence of noise.

3.1 Introduction

Randomized measurements are a central tool for the estimation of observables on repeatedly prepared quantum states [25, 26, 86]. Applications abound [87, 35, 6, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98]. Ref. [15] derived guarantees and proved the optimality of linear estimation from basis measurements generated by global or local Clifford rotations and coined the term "classical shadows". Besides optimality, an important contribution of Ref. [15] is the efficiency of estimation from global Clifford rotations. For example, with the same classical shadow from constantly, *i.e.* $\mathcal{O}(1)$, many state copies in the system size n, one can estimate many different pure state fidelities. In practice, however, harnessing this efficiency is typically not possible, since global Clifford rotations require the accurate implementation of "long" circuits of $\mathcal{O}\Big(n^2/\log(n)\Big)$ many local gates [27]. This obstacle has motivated the study of "shallow shadows", where shorter circuits are employed to "interpolate" between local and global rotations [28, 29, 30, 31, 32]. There is evidence that for certain observables, classical shadows with global rotations can be approximately de-randomized to $\mathcal{O}(n\log(n))$ gates [32]. On noisy intermediate-scale quantum (NISQ) hardware, however, even measurement circuits in this scaling regime are challenging. For the measurement, one typically wants to work with hardware-efficient circuits of very low depth. For clarity, we refer to such construction as *ultra-shallow shadows*.

The two-design property of uniformly random Clifford rotations allows for straightforward analytical inversion of the effective measurement map [99], a prerequisite to constructing the classical shadow. However, the works on shallow shadows resort to numerics for the evaluation

of the effective measurement map for depths larger than two [31], exploiting the local invariance of the circuit ensembles [28, 29, 30, 32]. For measurement circuits involving entangling gates, noise and (calibration) errors will typically cause sizable biases in the derived estimators on current NISQ hardware. For this reason, using ultra-shallow circuits for precise estimations in practice unavoidably requires a variant of classical shadows that mitigates circuit noise. That such mitigation is possible for classical shadows with single-qubit or global random Clifford rotations if the noise is well-characterized was shown for simple depolarizing noise models [100], local gate-dependent noise models [101], and via randomized benchmarking experiments [102] [1]. Going beyond these works, ideally, one wishes for a robust shadow scheme that replaces analytical or numerical calculations of the measurement map with a direct empirical estimation of it in the presence of noise. Such a tool was developed in Ref. [109], again for the cases of single-qubit or global random Clifford rotations under the assumptions of gate-independent Markovian noise acting after the rotation. However, a careful extension to the most crucial, experimentally relevant regime of ultra-shallow circuits, where no analytical calculation of the measurement frame is, in general, available, is missing. Despite the large body of literature on classical shadows, a central question is still open: Can the theoretically established improvement in sample efficiency of randomizing measurements with entangling gates be turned into a practical advantage in near-term experiments?

Our work answers this question in the affirmative. We show that achievable advantages are significant but also limited at current noise levels. To this end, we present a robust shadow estimation scheme for ultra-shallow circuits with arbitrary architectures of random local gates, under mild and practically plausible assumptions on the noise. We assume only that the average noisy measurement channel is invariant under local rotations and the accurate implementation of a single initial state. A simple but experimentally crucial observation is that, for this to happen, it is sufficient, for instanc,e that the first circuit layer consists of accurately im-

[1] Note that related but differently motivated research lines are mitigating noise in the state preparation using non-linear shadow estimation [50, 103, 90, 104, 105, 106, 107], or existing error-mitigation techniques [91], and the optimization of dual frames [108], e.g.

plemented single-qubit random Cliffords, regardless of the subsequent circuit, which can even be subject to time-stationary, gate-dependent, non-Markovian noise. As observed in previous works, single-qubit Pauli invariance guarantees that the measurement map is Pauli diagonal, and the circuit shallowness guarantees that the corresponding diagonal elements admit a tensor train (TT) description of low rank. For us, this structure must be maintained if the noise is approximately local. This allows us to develop efficient methods for directly estimating the measurement channel using standard tensor-network tools such as the modified alternating least squares (MALS) [110].

We numerically quantify how unmitigated shadow estimations (of both fidelities and Pauli observables) in the presence of noise become increasingly biased as the circuit depth increases. In contrast, we show that, for experimentally relevant regimes, robust ultra-shallow shadows become more precise as the depth increases up to an *optimal depth* depending on the strength of the incoherent noise. For instance, for depth $D = 2$ and $n = 7$, we observe that with our method, average errors of fidelity estimation with random target states improve by a factor of two over noise-free local shadows. Without mitigation, errors of shallow shadows are about an order of magnitude larger than with our method. We find that, even for these small depths and system sizes, infidelities of the entangling gates of at most 10^{-3} are required to realize improvements in precision at all. Moreover, we apply our scheme to a recent experiment [35] where ideal, four-qubit global Clifford rotations were approximated by noisy circuits of depth five, in an unmitigated fidelity estimation. In numerical simulations, we observe a four-fold reduction of error with our method. Finally, we perform a proof-of-principle experiment on the `ibm_nairobi` device. We estimate the effective measurement map and find that, indeed, simplified assumptions beyond what our methods exploit are not permissible.

Our framework offers a versatile and practical solution to classical shadows in the most relevant regime for near-term quantum devices, where randomized measurements are implemented by noisy shallow circuits.

3.2 Robust classical shadows

We start by reviewing the classical shadow protocol on an ideal noiseless device. Consider a quantum device that can be repeatedly prepared in an unknown n-qubit quantum state ϱ. Furthermore, we will work with a quantum device that is capable of implementing circuits composed from a unitary gate set $\mathcal{G}$ and of measuring in a fixed (computational) basis, specified by the effects $\mathcal{B} = \{\Pi_z = |z\rangle\langle z|\}_{z\in[d]}$, with $d = 2^n$ and $[d] = \{0, 1, \ldots, d-1\}$.

A probability measure ν on the unitary group $\mathrm{U}(d)$ with support in the implementable circuits defines an ensemble of random basis measurements. The corresponding *measurement protocol on an input state* ϱ repeats the following primitive: (i) draw $g \sim \nu$; (ii) prepare ϱ, apply g, and perform the measurement $\mathcal{B}$; (iii) observing measurement outcome z, record the tuple (z, g). The output of the data acquisition is a sequence $\mathcal{S} = \{(z_k, g_k)\}_{k=1,\ldots|\mathcal{S}|}$, with $|\mathcal{S}|$ the cardinality of $\mathcal{S}$.

Every iteration of the protocol is described by the same positive operator-valued measure (POVM) taking values $\Pi_{z,g} = g^\dagger \Pi_z \, g$, known as *snapshots*, with g in the support of ν and $z \in [d]$. If the support of ν is sufficiently large, the POVM is informationally complete (IC), and asymptotically any information about ϱ can be reconstructed from the measurement statistic $\mathcal{S}$.

To be instructive, in the following, we restrict ourselves to the simplest task of estimating a linear functional $(O|\varrho) = \mathrm{Tr}\big[O^\dagger \varrho\big]$, here defined as the Hilbert-Schmidt inner product with an endomorphism O of $\mathbb{C}^d$. Examples include the estimation of the fidelity of ϱ with respect to (w.r.t.) a pure target state $O = |\psi\rangle\langle\psi|$ or multi-qubit Pauli observables. The robust shallow shadows explored in our work can also be readily employed in non-linear estimation (see *e.g.* Ref. [111]).

Given $\mathcal{S}$, an estimator for $(O \,|\, \varrho)$ can be constructed via standard linear inversion [112]: Consider the frame operator of the measurement POVM, also known as the effective measure-

ment map [15],

$$S = \mathbb{E}_{g\sim\nu}\left[\sum_{z\in[d]} |\Pi_{z,g})(\Pi_{z,g}|\right], \tag{3.1}$$

where we have used Dirac notation w.r.t. the Hilbert-Schmidt inner product. Since the POVM is informationally complete (IC), S is invertible and the elements of the canonical dual frame are given by $\bar{\Pi}_{z,g} = S^{-1}(\Pi_{z,g})$, fulfilling $\mathbb{E}_{g\sim\nu}\left[\sum_{z\in[d]} |\bar{\Pi}_{z,g})(\Pi_{z,g}|\right] = S^{-1}S = \mathbb{1}_{d^2}$, with $\mathbb{1}_{d^2}$ being the $d^2 \times d^2$ identity operator. In the classical post-processing, we evaluate the scalar function $o(z,g) = (O|\bar{\Pi}_{z,g})$ on each entry of $\mathcal{S}$ and take the empirical mean $\hat{o} = \sum_{(z,g)\in\mathcal{S}} o(z,g)$. For the ideal device, we observe (z,g) with probability density $p(z,g) = (\Pi_{z,g}|\varrho)\,\mathrm{d}\nu(g)$ and, thus, $\hat{o}$ is indeed an unbiased estimator of

$$\mathbb{E}_{z,g}(\hat{o}) = \mathbb{E}_{g\sim\nu}\left[\sum_{z\in[d]} (O|\bar{\Pi}_{z,g})(\Pi_{z,g}|\varrho)\right] = (O|\varrho)\,. \tag{3.2}$$

In practice, however, experimental imperfections in the implementation of the basis measurement $\mathcal{B}$ and the circuits in the support of ν induce a bias in the estimator. We denote by $\tilde{\Pi}_{z,g}$ the elements of the POVM that are implemented on the device and define the "one-sided" [2] *noisy frame operator* as

$$\tilde{S} = \mathbb{E}_{g\sim\nu}\left[\sum_{z\in[d]} |\Pi_{z,g})(\tilde{\Pi}_{z,g}|\right]. \tag{3.3}$$

The same argument above now yields a non-vanishing estimator bias such that $\mathrm{bias}(O,\varrho) = |\mathbb{E}(\hat{o}) - (O|\varrho)| = |(O|S^{-1}\tilde{S} - \mathbb{1}_{d^2}|\varrho)|$.

The form of the bias already hints at a conceptually simple solution to mitigate the noise. Assume we have an empirical estimate of the noisy frame operator $\tilde{S}$ and it is still invertible, which happens if $\{\tilde{\Pi}_{z,g}\}_{z,g}$ is still informationally complete. We can make use of this estimate in the construction of a scalar function $\tilde{o}(z,g) = (O|\tilde{S}^{-1}|\Pi_{z,g})$ that we evaluate instead of

[2] While the noiseless left-hand-side $|\Pi_{z,g})$ of $\tilde{S}$ represent the postprocessing stage performed with noiseless representations of the POVM elements, the noisy right-hand-side $(\tilde{\Pi}_{z,g}|$ represent the noisy implementations of the POVM elements realized in practice.

$o(z, g)$ in the classical post-processing. The resulting mean estimator $\hat{\hat{o}}$ is again unbiased with $\mathbb{E}(\hat{\hat{o}}) = (O|\tilde{S}^{-1}\,\tilde{S}|\varrho)$.

When ν is the uniform measure over the multi-qubit Clifford group, the ideal frame operator S takes the particular simple form of a global depolarizing channel with effective depolarization strength $1/(d+1)$ [99]. The simple structure becomes apparent when realizing that S is a channel twirl of the dephasing channel $\mathcal{M} = \sum_{z\in[d]} |\Pi_z)(\Pi_z|$ that describes the computational basis measurement. For the Clifford group, a unitary 2-design [113, 114, 115], S commutes with arbitrary unitary channels [116]. If ν is the uniform measure on the n-fold tensor product of single-qubit Clifford unitaries, S is only invariant under local one-qubit unitary gates. Since $\mathcal{M}$ is the tensor product of local dephasing channels, the projection of $\mathcal{M}$ onto the locally unitarily invariant subspace is the product of local depolarizing channels.

If the noise does not break the measures' invariance property, $\tilde{S}$ has the same structure as S, a global (linear span of local) depolarizing channel(s) for ν the uniform measure over the global (local) Clifford group. This insight is at the heart of the robust shadows introduced in Ref. [109]. There, the invariance assumption is justified by assuming a noise channel Λ acting *directly before the measurement* and *independent* of the unitary rotation g that is implemented. Under this assumption, $\tilde{S}$ simply is the ideal channel twirl of the concatenation of the dephasing channel and the noise channel.

More concretely, recall that conjugation with the Clifford group (as with the unitary group) acts irreducibly on the space of traceless matrices, *e.g.* [26]. Correspondingly, tensor-products of single-qubit Clifford channels decompose into d irreducible representations (*irreps*) without multiplicities, labeled by $k \in \{0,1\}^n$. The projectors onto the irreps are $\Phi_k = \bigotimes_{l\in[n]} \Phi_{k_l}$ with $\Phi_0 = d^{-2}\,|\mathbb{1}_2)(\mathbb{1}_2|$ and $\Phi_1 = \mathbb{1}_4 - \Phi_0$. Assuming single-qubit Clifford-invariance, the noisy frame operator can, thus, be written as

$$\tilde{S} = \sum_{k\in\{0,1\}^n} f_k\,\Phi_k \tag{3.4}$$

with coefficients $f_k \in \mathbb{R}$.

Ref. [109] proposed the following *calibration procedure* to learn the coefficients f_k: Perform a randomized basis measurement on the input state $|0\rangle\langle 0|$. The post-processing is analogous to that of linear shadow estimation. But here we evaluate the scalar function

$$\phi_k(z, g) = (Z_k|\Pi_{z,g}) \tag{3.5}$$

with $Z_k = \bigotimes_{l\in[n]} Z^{k_l}$ and Z the Pauli-Z operator. In contrast to the scalar function o, ϕ does not use the dual frame. Assuming the initial state $|0\rangle\langle 0|$ can be accurately prepared, the mean of ϕ_k evaluated on $\mathcal{S}$ is an unbiased estimator for f_k. From the form of $\tilde{o}(z, g)$ and Eq. (3.5), we see that the spectrum of $\tilde{S}$ must be characterized to relative precision to guarantee controlled additive precision in estimator $\hat{\tilde{o}}$.

3.3 Results

The starting observation for our work is the following: Let ν be an arbitrary random ensemble of quantum circuits whose implementation suffers time-stationary, gate-dependent noise and errors. The noise can be non-Markovian. If the first gate layer of the circuits, that is applied to the state under scrutiny, accurately implements local channel twirls, then $\tilde{S}$ in Eq. (3.3) still has the structure of Eq. (3.4) even in the presence of noise. Hence, $\tilde{S}$ can be estimated via the above calibration procedure introduced for local Clifford rotations in Ref. [109]. A practically motivated assumption is that the major sources of noise in the implemented circuits are entangling gates and readout noise. If the first layer then consists of accurately implemented Haar random unitary or Clifford single qubit gates, the invariance assumption holds.

3.3.1 Circuits and tensor network methods

If the quantum circuit implementing g has low depth, we can make use of tensor network methods at multiple places of the protocol to reduce the runtime and storage complexity of the calibration protocol, as well as partially for the estimation protocol. To be concrete, we specify a circuit ensemble that is directly motivated by existing hardware architectures and applicati-

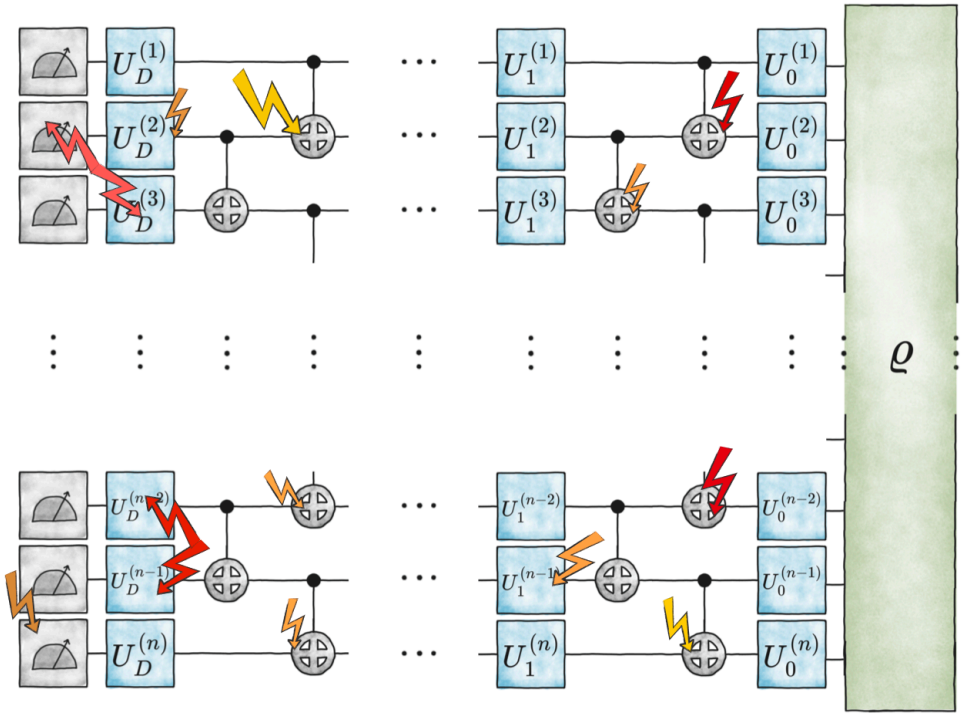

Figure 3.1: **Hardware-efficient circuits for robust shallow shadows.** Given an initial state ϱ, the circuit implementing the measurement consists of $(D+1)$ random layers. For $D=0$, only a layer of n single-qubit rotations is present. Each extra layer is composed of a sub-layer of entangling gates followed by n single-qubit rotations. At the end, a computational basis measurement is performed. We assume that the implementation of the circuit can suffer from noise that does not break the local-invariance and low-rank representation of the measurement map. For instance, approximately local gate-dependent non-Markovian noise (indicated with the lightning symbols) affecting the gates after the first gate-layer and the read-out is admissible.

ons. The circuit architecture we use is based on the so-called *hardware-efficient ansatz* [117], depicted in Fig. 3.1. The circuit starts with a layer of single-qubit rotations, i.i.d. sampled from the Haar measure on single-qubit unitaries or Clifford unitaries. This layer is labeled as depth $D=0$. Every subsequent layer is composed of a "sub-layer" of entangling gates [3] followed by another sub-layer of single-qubit rotations sampled from the same measure as before. In principle, for the subsequent single qubit layers also other measures are also admissible.

The entangling layer of the hardware-efficient ansatz has a matrix-product operator (MPO) representation of bond dimension $\chi_C = 2$, see App. 5.2.1. Thus, $\Pi_{z,g}$ for g of depth D is

[3] Even though we made the sufficient choice to use the brickwork architecture with open boundary conditions as our entangling layers, this choice is neither necessary nor unique. The only condition that needs to be satisfied is that the implemented POVM converges to a 2-design when $D \to \infty$.

exactly described by a matrix-product state (MPS)—also known as tensor train (TT)—of bond dimension (TT-rank) χ_C^D. As a consequence, also the vector of scalar functions $\phi(z,g) = (\phi_k(z,g))_{k\in[n]}$ is described by a TT of rank at most χ_C^{2D} and can be efficiently computed and stored for every observed tuple (z,g).

Furthermore, if the noise does not introduce sizable non-local correlation, also $\tilde{S}$ has an efficient tensor representation, and the coefficient vector $f = (f_k)_{k\in[n]} \in \mathbb{R}$ is given by a low-rank TT. We illustrate this argument with a specific noise model in App. 5.2.1. In particular, for single-qubit Pauli noise, the TT-rank of f is at most $\chi = 4^D$. This justifies formulating the post-processing of the calibration procedure as the TT recovery problem

$$f = \underset{f' \text{ rank-}\chi \text{ TT}}{\arg\min} \left\| \frac{1}{|\mathcal{S}|} \sum_{(z,g)\in\mathcal{S}} \phi(z,g) - f' \right\|_{\ell_2}, \tag{3.6}$$

i.e. the problem of fitting an average of low-rank TTs with a TT of low-rank.

Standard approaches for tensor recovery, such as modified alternating least-squares optimization (MALS) [110] can be readily adopted. In particular, for MALS, we can perform the local update step for every core using contractions and pseudo-inverses with storage and time complexity scaling polynomially in the system size, TT ranks, and number of samples. This suggests that for sufficiently local noise and shallow circuits, $\tilde{S}$ can be efficiently learned. Below, we numerically investigate the performance of the scheme using MALS. Note, however, that analytically establishing the overall efficiency would additionally require proof of a guarantee for the TT recovery algorithms from $\mathcal{S}$ with polynomial cardinality.

The construction of the dual frame for the estimation protocol uses the inverse of $\tilde{S}$. Since Eq. (3.4) is an orthogonal decomposition, $\tilde{S}^{-1} = \sum_{k\in\{0,1\}^n} (f_k)^{-1}\Phi_k$. From f in TT format, we can efficiently evaluate entries of its element-wise inverse f^{-1} and, thus, the eigenvalues of $\tilde{S}^{-1}$. For certain observables that themselves have an efficient tensor-network representation, having f^{-1} in low-rank TT format would even allow for efficiently computing the estimators $\hat{\hat{o}}$. Even though f is a low-rank TT, f^{-1} might have a higher rank. Finding a low-rank MPS representation for the inverse of a low-rank MPS is, in general, an open question. Nonetheless,

one can attempt to find a low-rank TT description either from the estimate for f or directly from the data using the TT recovery algorithms.

3.3.2 Numerical results

Here, we numerically study the performance of robust estimation with shallow circuits using architectures that are motivated by applications. We use two noise models: (i) two-qubit *incoherent* noise model, where a different unitary noise is applied to a circuit at each execution (see App. 5.2.2 for details); (ii) composition of a two-qubit Pauli noise with *gate-dependent* single-qubit Pauli noise, where different single-qubit gates suffer from different Pauli noise channels.

We begin by assessing the need to mitigate noise in shallow circuits. The top panel of Fig. 3.2 shows the worst-case bias, $\max_{O,\varrho} \mathrm{bias}(O, \varrho)$, of standard non-mitigated shadow estimation for Pauli observables (see App. 5.2.3 for details) for different system sizes, circuit depths, and incoherent noise levels. The worst-case bias is monotonically increasing with all three parameters and the results are consistent with a linear increase of the bias with both the circuit depth and system size. For two-layer shadows, we observe Pauli worst-case biases that are already for three to four qubits about one order of magnitude larger than the average infidelity of the CNOTs used in the circuit. For example, targeting an estimation precision of 10^{-2}, even fairly accurate CNOTs gates with infidelity 10^{-3} can introduce sizable biases in the linear estimation. Pauli observables with large non-trivial support tend to have larger biases, see App. 5.2.4. We conclude that without mitigation using even very shallow shadows is typically not possible in practice already for small system sizes. This underpins the motivation to study our noise-robust variant.

A key application of shadows with global Clifford rotations is fidelity estimation. Here, the sample complexity is constant in the system size, while randomized local basis measurements have exponential scaling [15]. This leads to the question of whether an ultra-shallow hardware-efficient ansatz can improve the sample efficiency even in the presence of experimental imperfections. In other words: *Can the theoretically established sample-efficiency of classical*

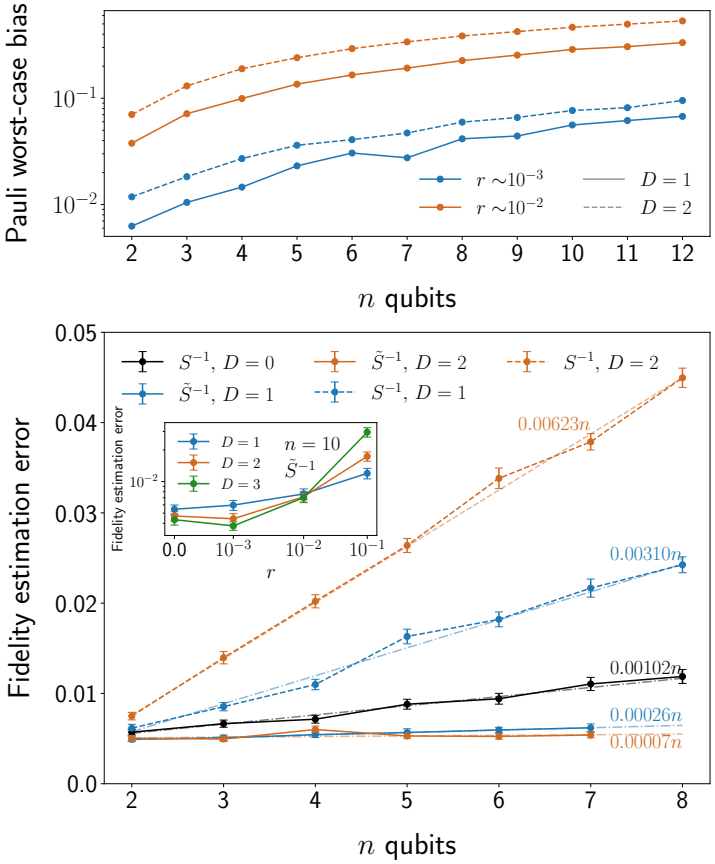

Figure 3.2: **Numerical experiments under two-qubit incoherent noise model. (top)** Worst-case bias for the estimation of Pauli-observables without mitigation for different system sizes, depth, and noise levels. Single-qubit layers sampled from the Haar measure. Noise level indicated by r, the average gate infidelity per CNOT, and samples $|\mathcal{S}| = 10^6$. We observe a sizable bias about an order of magnitude larger than the two-qubit infidelity and moderately increasing with the system size. **(bottom)** Error in fidelity estimation due to finite statistics $|\mathcal{S}| = 10^5$ for different system sizes, using both $\tilde{S}^{-1}$ and S^{-1}. Single-qubit layers sampled from the Haar measure. The error is averaged over 50 perfectly prepared uniformly random pure states; errors bars denote the standard error. Noise only affects CNOT gates with $r = 10^{-3}$. We observe a reduction of the system-size dependence and, in consequence, smaller estimation errors for shallow circuits compared to (noiseless) local basis measurements ($D = 0$). The error in unmitigated fidelity estimation grows linearly with n, with a higher slope for increasing depth. In contrast, for mitigated fidelity estimation the slope decreases with increasing depth. **(inset)** Fidelity estimation error using $\tilde{S}^{-1}$ as a function of r for $n = 10$ qubits and $|\mathcal{S}| = 10^5$. We observe a cross-over point at $r = 10^{-2}$; for smaller (larger) infidelities the error decreases (increases) with circuit depth.

shadows with entangling bases actually be translated into a practical advantage? We simulate the protocol of robust shadow fidelity estimation assuming perfectly prepared pure input states. The fidelity is estimated w.r.t. the input state itself. Only the two-qubit gates implementing the

basis rotations for the shadow suffer from incoherent noise with average infidelity 10^{-3}, and the number of samples is $|\mathcal{S}| = 10^5$. The noisy frame operator is estimated with $|\mathcal{S}| = 10^6$, which we found sufficient for ensuring that the estimation error of $\tilde{S}$ does not limit the accuracy of the fidelity estimation.

Fig. 3.2 (bottom) shows the estimation error averaged over multiple uniformly random pure states for different system sizes and circuit depths. We find that the system-size dependence is significantly reduced already for shadows with $D = 1, 2$ when compared to local circuits. In effect, the estimation error is reduced by a factor of about 2 already for these small systems of size $n = 7, 8$ when using short-depth circuits. Importantly, we observe that to harness these effects, the empirical estimation of $\tilde{S}$ put forward here is essential. Without noise mitigation, the noise-induced bias, depicted in the figure with dashed lines, deteriorates fidelity estimation by factors of 3 to 4 for the same small system sizes and $D = 1$. For $D = 2$ the estimation error using the noiseless frame operator S is already almost one order of magnitude worse than the one that uses $\tilde{S}$.

3.3.3 Limitations of classical shadows in practice

In general, we expect that even assuming perfect bias mitigation, the improvements of the estimation error by going to deeper circuits will be limited by incoherent noise. Intuitively, every CNOT with incoherent errors will increase the probability that one only learns compromised information about the state. Even if the noise is perfectly mitigated by accurately estimating $\tilde{S}$, with some probability we learn less about the state. Thus, there will be an increased sample complexity of the fidelity estimation itself. At the same time, longer circuits (without noise) will improve the variance of fidelity estimation eventually reaching the efficiency of 3-designs. We therefore have two competing effects on the sample complexity. We expect that for many estimation tasks, there exists a cross-over point depending on the infidelity of the entangling gates. Below this cross-over infidelity, increasing the depth improves estimation accuracy (at a fixed number of samples). Above it, accuracy is deteriorated by deeper circuits.

Using the same fidelity estimation setting as above, we numerically determine the estimation error for different infidelities of the two-qubit gates and circuit depth, averaged over random stabilizer states at fixed system size $n = 10$ and $|\mathcal{S}| = 10^5$. We display such a cross-over in Fig. 3.2 (inset). We observe that, for infidelities smaller than 10^{-2}, the estimation error using $\tilde{S}^{-1}$ is decreased with circuit depth. For larger infidelities, deeper circuits suffer from large estimation errors. We conclude that indeed even with the robust estimation scheme, deeper shadows being experimentally beneficial is dependent on the noise strength.

We now take a more detailed look at the TT structure of the frame operator and its inverse. We begin with using sequential SVDs to solve the recovery problem in Eq. (3.6) for different ranks χ using the same noise setting as above. Fig. 3.3 panel (a) shows the additional (relative) worst-case bias in estimating Pauli observables when using a TT approximation to f instead of the empirically learned dense vector. We indeed find that for $D = 1$ a rank $\chi = 4$ approximation yields a small relative error that is constant with the system size for $n \geq 6$ and higher ranks do not yield an improvement. Furthermore, we find that for smaller system sizes the MPS is also fitting the statistical fluctuations. For $D = 2$ we find that $\chi = 8$ accurately approximates $\tilde{S}$ up to statistical error.

We then apply the computationally and storage-efficient MALS algorithm to directly learn a TT representation of $\tilde{S}$ from the shadow data and calculate the induced bias in estimating Pauli observables. We initialize the algorithm with the analytical result for S for the local Clifford frame. We observe an identical performance to the SVD. MALS is rank-adaptive but does not increase the rank beyond $\chi = 8$ for $D = 2$ even if permitted.

3.3.4 Application.

As an application, we simulate the quantum experiment of the optimization protocol that was performed on the Sycamore quantum processor in Ref. [35]. The authors prepared an 8-qubit state $|\Psi_T\rangle$ and used classical shadows to estimate the wavefunction overlap $\langle\phi|\Psi_T\rangle$, where $|\phi\rangle$ belongs to a suitable quantum chemistry basis. For the specific choices of $|\Psi_T\rangle$ and

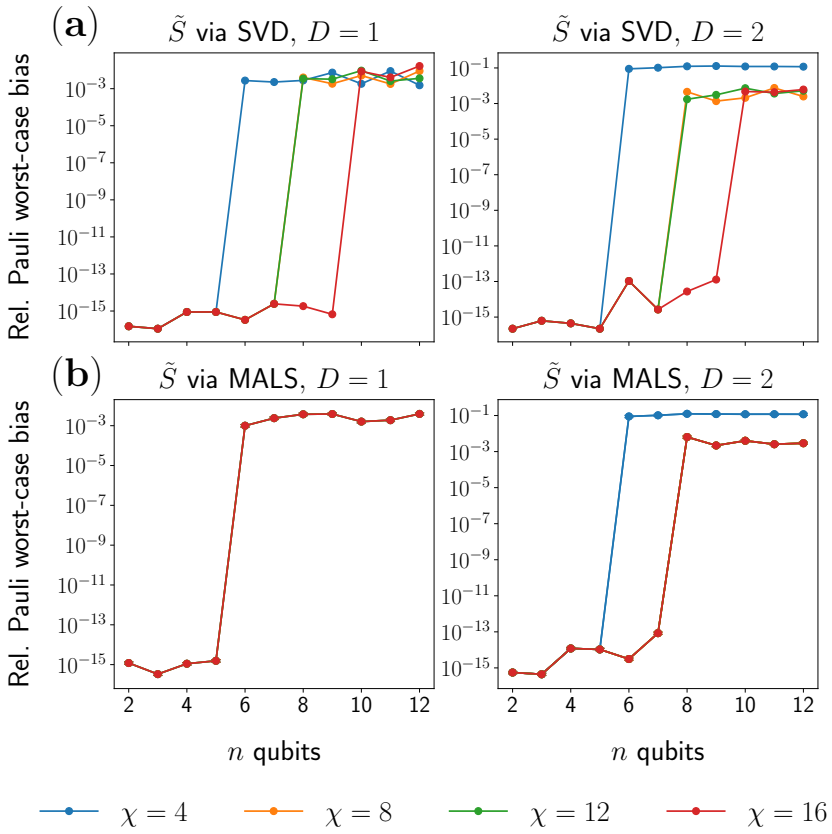

Figure 3.3: **Tensor network representation of robust measurement frames of shallow circuits. (a)** Worst-case bias for the estimation of Pauli observables introduced by converting $\tilde{S}$ to a matrix product state (TT) representation using sequential SVDs, relative to the size of the corresponding eigenvalue, as a function of n, for different bond dimensions χ, and depths $D = 1, 2$. We observe that TT ranks 4 and 8 suffice to describe $\tilde{S}$ up to statistical error. **(b)** Worst-case bias for the estimation of Pauli observables when learning $\tilde{S}$ directly from the calibration data using a modified ALS method. We find a similar performance as for the sequential SVD. Colors indicate the permitted maximal ranks. Actual rank of the result of the rank-adaptive MALS is $\chi = 4$ for $D = 1$ ($\chi = 8$ for $D = 2$).

$|\phi\rangle$ in our simulations see App. 5.2.5. The measurement circuit architecture used in Ref. [35] is the 2-fold tensor product of 4-qubit circuits of the same form as Fig. 3.1. We call the resulting (noisy) frame operator $S_{n=4}^{\otimes 2}$ ($\tilde{S}_{n=4}^{\otimes 2}$). Here, the single-qubit rotations are uniformly sampled from the one-local Clifford group. We simulate the experiment using the *gate-dependent* Pauli noise model. More specifically, different single-qubit Clifford gates suffer from different Pauli noise channels. We use the readout error model for the Sycamore chip described in Ref. [6].

The left panel of Fig. 3.4 shows, for different circuit depths D, the worst-case bias using

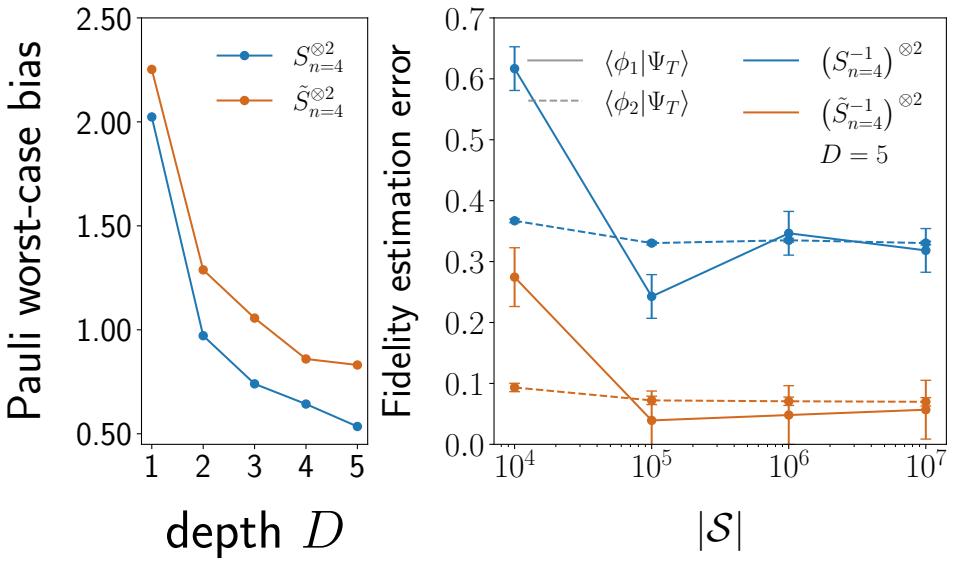

Figure 3.4: **Simulation of overlap estimation from Huggins *et al.* (2022) under a gate-dependent noise model. (left)** Worst-case bias for the estimation of Pauli observables introduced by implementing short-depth circuits with and without noise in an experiment, but using the ideal global Clifford frame in the post-processing stage, as a function of D, for two frames: $S^{\otimes 2}_{n=4}$ (noiseless, blue line) and $\tilde{S}^{\otimes 2}_{n=4}$ (noisy, brown line). Frames were estimated with $|\mathcal{S}| = 10^6$. The noisy frame was generated under a gate-dependent circuit noise as well as readout noise. **(right)** Fidelity estimation error using aforementioned frames. States $|\Psi_T\rangle$, $|\phi_1\rangle$, and $|\phi_2\rangle$ are defined in Appendix 5.2.5.

shallow shadows as an approximation of shadow estimation using the 2-fold tensor product of the 4-qubit global Clifford group, $S^{\otimes 2}_{\mathrm{gc},\,n=4}$. This is motivated by the strategy of Ref. [35]. We see that $\tilde{S}^{\otimes 2}_{n=4}$ displays slower convergence to $S^{\otimes 2}_{\mathrm{gc},\,n=4}$ compared to $S^{\otimes 2}_{n=4}$. This suggests that experimentally performing shallow shadows but analytically inverting $S^{\otimes 2}_{\mathrm{gc},\,n=4}$ for the linear estimation introduces a bias in the estimation. This bias is more pronounced in the presence of (gate-dependent) noise, stressing the necessity to correctly characterize the noisy measurement frame and use its inverse for the estimations. This is presented in the right panel of Fig. 3.4.

3.3.5 Quantum hardware demonstration.

In Fig. 3.5, we show the results of an experimental implementation of robust shallow shadows on the `ibm_nairobi` quantum processor from IBM Quantum . We implemented the measurement circuit of Fig. 3.1 in 5 of the 7 qubits available in the device, for depths $D \in \{0, 1, 2\}$.

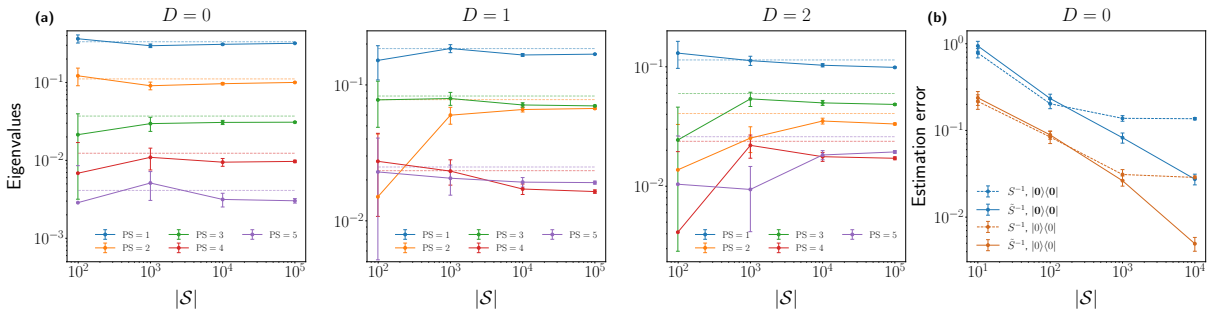

Figure 3.5: **Experimental realization on IBM Quantum . (a)** Eigenvalues of the noisy frame operator $\tilde{S}$ in the Pauli-Liouville basis as a function of $|\mathcal{S}|$, for $n = 5$ qubits, and $D \in \{0, 1, 2\}$. Error bars correspond to standard errors, *i.e.* to standard deviations divided by $\sqrt{|\mathcal{S}|}$. Horizontal dashes represent error bars that were masked to avoid displaying negative values in log scale. We plot one eigenvalue per Pauli support (PS). Dashed lines show the corresponding eigenvalues if the frames were noiseless. **(b)** Cross-validation for $D = 0$ by using shadow tomography to estimate the fidelity of the state preparation of the single-qubit zero state $|0\rangle\langle 0|$ and the n-qubit zero state $|\mathbf{0}\rangle\langle\mathbf{0}| \equiv |0\rangle\langle 0|^{\otimes n}$.

In Fig. 3.5(a), we plot the estimated eigenvalues of $\tilde{S}$ as a function of the cardinality of the shadow $|\mathcal{S}|$ up to $|\mathcal{S}| = 10^5$. (We also provide histograms for the distribution of samples in Fig. 3.4 (a) in App. 5.2.6.) For simplicity, we show one eigenvalue per Pauli support. The dashed lines show the eigenvalues of the corresponding noiseless frame. In all three cases, the eigenvalues differ from their noiseless counterparts by a statistically significant margin. This confirms that the measurement frame to be inverted in practice should indeed be $\tilde{S}$ instead of S. We further observe that, as expected, with increasing depth, the eigenvalues get closer. At $D = 2$, the noiseless eigenvalues significantly deviate from each other, and also the noise-induced deviation differs for different eigenvalues. Thus, an approximate construction with the (robust) global shadows is not permissible.

To cross-validate our results, we separate our 10^5 samples in two batches: with a first batch of 90% of the snapshots, we estimate $\tilde{S}$; with the remaining 10%, in turn, we perform a shadow tomography protocol on the 5-qubit state $|\mathbf{0}\rangle\langle\mathbf{0}|$ as well as on its marginal state $|0\rangle\langle 0|$ on one of the qubits. For each of the two states, we invert the classical shadows using both $\tilde{S}^{-1}$ and S^{-1}. In Fig. 3.5 (b), we plot the fidelity biases introduced by the mitigated and unmitigated estimations, respectively. For the unmitigated estimation, one can observe that at $|\mathcal{S}| \approx 10^3$

the statistical error becomes smaller than the bias (*i.e.* the systematic error due to the 'wrong' inverted frame). Increasing the number of samples does not improve the accuracy. In contrast, for the robust shallow shadows, the accuracy continues to increase steadily as $|\mathcal{S}|$ goes beyond 10^3.

3.4 Conclusions.

We presented a robust classical shadow scheme for a broad family of noisy, ultra-shallow measurement circuit ensembles. Under plausible noise assumptions, the measurement calibration stage admits an efficient tensor-network representation and can be performed using standard tensor-network tools. We illustrated the relevance of our method with both numerics and proof-of-principle experiments on an IBM Quantum device. Numerically, for relevant parameter regimes, we showed that while unmitigated shallow shadows with noisy circuits become more biased as the depth increases, robust ones become significantly more precise up to a crossing point. We thereby established that classical shadows can significantly improve sample efficiency in practice. At the same time, we established that the detrimental effect of incoherent noise on the sample efficiency fundamentally limits capitalizing on the efficiency of classical shadows in practice. Experimentally, we observed major error reductions in fidelity estimations. Our framework does not require analytical calculations of the effective measurement map but replaces it with an efficient direct estimation protocol. This offers versatile, practical estimation protocols that realize the efficacy of classical shadow as much as it is possible on actually existing and near-term quantum devices by using the available hardware-efficient noisy shallow circuits.

3.5 Article acknowledgments

We thank Stefano Carrazza and Stavros Efthymiou for their insights on numerical simulations. We further thank Michał Oszmaniec for discussions on noise mitigation. We thank

Jadwiga Wilkens for the availability of the *Quantum Circuit Library* [85].

Chapter 4

Conclusions and Outlook

In the two articles that are part of this thesis, we showed how to *"upload"* and *"download"* information on near-term quantum computers. Here, we recap the main results of both articles while expanding on the possible lines of future work. For better clarity, below we split their expanded conclusions and outlooks into two subsections.

Data loading. In the first paper, presented in Chapter 2, we showed how to construct parameter-optimal, ancilla-free encoders to upload real- and complex-valued data into fixed Hamming-weight subspaces, *i.e.* subspaces that are characterized by computational basis states that are associated to bitstrings of the same Hamming weight. These encoders unlock the possibility of state preparation schemes of polynomial compression, which are more suitable for near-term quantum computers than binary encoders due to requiring shallower circuits. Hamming-weight-k amplitude encoders are also more advantageous than the previously studied unary encoders due to their reduced qubit requirements. This is because unary encoders require $n = d$ qubits to encode a 1-dimensional data vector of size d while HW-k encoders require $n = \mathcal{O}\Big(k\, d^{1/k}\Big)$ qubits. Therefore, we believe that Hamming-weight-k amplitude encoders are suited to extract the most potential from NISQ devices.

We also showed how the same subroutines implemented in the HW-k algorithm can be used to create parameter-optimal, ancilla-free *sparse* and *binary* encoders. The sparse encoding scheme follows the same Subroutines 1 and 2 as the HW-k encoder for calculating gate para-

meters and gate placements, respectively. Meanwhile, the binary encoder follows all the same routines as the HW-k encoder by populating each fixed HW subspace from $k = 0$ to $k = n$ in ascending order. This allows both encoders to be implemented without requiring any ancillary qubits. However, the sparse encoder requires the introduction of a new gate (gRBS) that does not conserve Hamming weight — consequently, requiring more CNOTs than the regular RBS gate —, and the binary encoder requires controlled operations with up to $n - 1$ controls. These two downsides lead to circuits that are deeper than amplitude encoding schemes that use ancillas. Nevertheless, when compared with the state-of-the-art [45], we believe that our protocols provide users of quantum hardware the freedom of choice between the addition of ancillary qubits or circuit depth and the ease of the classical preprocessing stage in their state preparation methods.

Lastly, in Chapter 2, we demonstrated the viability of our encoder when used on a currently available commercial quantum hardware from IonQ [57, 76]. We observed that even though theoretically exact, the fidelity of our encoder was degraded by noise and errors of the NISQ device. However, we showed that the fidelity of the information extracted from the hardware could be improved by using error-mitigation techniques. Particularly in this case, we used the technique called Clifford Data Regression. The experimental results led us to numerically study the fidelity of our HW-k encoding scheme when under a two-local depolarizing noise model. Under this experimentally-relevant noise assumption, given fixed k, we could see that the infidelity in the state preparation scales linearly with the depolarizing strength of the noise channel. Therefore, we expect that each order-of-magnitude improvement in the hardware noise will yield an order-of-magnitude improvement in the fidelity of our encoder. We also studied the viability of our encoder as a variational quantum ansatz candidate. Our encoder performed very well as an ansatz for a ground-state optimization task from quantum chemistry, beating other Hamming-weight-preserving ansatze by an order of magnitude in terms of chemical accuracy.

Several possible directions can be explored as follow-ups of the work presented in Chapter 2. One immediate direction is the extension of the state preparation scheme to the decomposi-

tion of generic unitary operators [118, 119]. This has implications in several areas of quantum information theory. A non-comprehensive list of areas includes conversions between encoding schemes (*e.g.* from HW-k encoding to binary encoding and vice-versa, or from amplitude encodings to phase encodings and vice-versa), explicit circuits for the block-encoding of generic matrices [120, 121, 122], end-to-end Hamming-weight-preserving quantum algorithms [37, 53], and explicit circuit decompositions for the algebraic and coordinate Bethe ansatze [123, 124]. Another future line of work involves an in-depth exploration of the state preparation circuits as ansatze for variational quantum algorithms. In particular, we would like to investigate how the parametrization of quantum states in hyperspherical coordinates affects optimizers such as the quantum natural gradient [125, 126]. We also envision exploring the extension of these encoders to other families of coordinate systems on a d-dimensional (complex) hypersphere [127, 128].

Data extraction. We dealt with the downloading of data from NISQ devices in the second article, which is part of this thesis, presented in Chapter 3. We showed that the recently developed randomized measurement scheme called *shallow shadows* can be made robust to measurement circuit noise under experimentally relevant noise assumptions that are milder than previously stated in the literature. Our scheme uses ultra-shallow circuits to implement POVMs, increasing the precision of estimators when compared to randomized measurements based on 1-local 2-designs even in the presence of noise. This significantly improves the sample efficiency of the state-agnostic estimation of expectation values in NISQ devices. We also showed that the precision increases with increased depth as long as the noise in the circuit is not too strong, which leads to diminishing returns in terms of estimation errors. We also showed that under the same noise assumptions, the measurement channel admits a matrix product state representation, significantly reducing the classical cost of one of the postprocessing steps required by the classical shadow protocol. The inverse measurement channel, required in the postprocessing state, can be obtained either using standard tensor network tools available or by querying individual

eigenvalues of the channel and performing the element-wise inverse.

We demonstrated the applicability of the robust shallow shadows in two scenarios. First, we numerically simulated the quantum experiment stage of the optimization protocol from Ref. [35]. We showed that, for a fixed number of samples, the robust ultra-shallow shadows significantly diminish the errors in the overlap estimations required by the optimization protocol under a gate-dependent noise model. This demonstrates that our robust classical shadow protocol has practical implications even under more realistic noise assumptions. Second, we performed an experimental implementation of the protocol in one of the IBM Quantum processors. We showed that, regardless of circuit depth, the eigenvalues of the measurement channel estimated experimentally differ significantly from the eigenvalues of the corresponding frames if the experiment was noiseless. This shows that not mitigating measurement circuit noise introduces a bias in the expectation values experimentally estimated using classical shadows on NISQ devices. Lastly, we showed how this bias affects expectation values by measuring the fidelity of the $|\mathbf{0}\rangle\langle\mathbf{0}|$ state and one of its single-qubit marginals.

A possible follow-up of this work is the integration of the calibration procedure with gate set tomography. In this situation, instead of estimating the measurement frame by sampling the hardware as detailed in Chapter 3, one would first perform gate set tomography of the native gates of the quantum hardware and then classically build the tensor train that corresponds to the POVM implemented. It is an open question about which method would yield better precision in terms of eigenvalue and expectation value estimations, as well as which protocol would require the least amount of samples collected from the quantum hardware.

References

[1] Renato M S Farias, Raghavendra D Peddinti, Ingo Roth, and Leandro Aolita, *Robust ultra-shallow shadows*, Quantum Science and Technology **10**, 025044 (2025).

[2] Renato M.S. Farias, Thiago O. Maciel, Giancarlo Camilo, Ruge Lin, Sergi Ramos-Calderer, and Leandro Aolita, *Quantum encoder for fixed-Hamming-weight subspaces*, Phys. Rev. Appl. **23**, 044014 (2025).

[3] Andrea Pasquale, Andrea Papaluca, Renato M. S. Farias, Matteo Robbiati, Edoardo Pedicillo, and Stefano Carrazza, *Beyond full statevector simulation with Qibo* (2024), arXiv:2408.00384 [quant-ph] .

[4] Matteo Robbiati *et al.*, *Double-bracket quantum algorithms for high-fidelity ground state preparation* (2024), arXiv:2408.03987 [quant-ph] .

[5] F. Arute *et al.*, *Quantum supremacy using a programmable superconducting processor*, Nature **574**, 505–510 (2019).

[6] H.-Y. Huang *et al.*, *Quantum advantage in learning from experiments*, Science **376**, 1182–1186 (2022).

[7] S. A. Moses *et al.*, *A race-track trapped-ion quantum processor*, Physical Review X **13** (2023).

[8] L. Caune *et al.*, *Demonstrating real-time and low-latency quantum error correction with superconducting qubits* (2024), arXiv:2410.05202 [quant-ph] .

[9] Google Quantum AI and Collaborators, *Quantum error correction below the surface code threshold*, Nature **638**, 920–926 (2025).

[10] H. Aghaee Rad *et al.*, *Scaling and networking a modular photonic quantum computer*, Nature **638**, 912–919 (2025).

[11] Z. Zimborás *et al.*, *Myths around quantum computation before full fault tolerance: What no-go theorems rule out and what they don't* (2025), arXiv:2501.05694 [quant-ph] .

[12] A. Katabarwa *et al.*, *Early fault-tolerant quantum computing*, PRX Quantum **5**, 020101 (2024).

[13] Oriel Kiss, Utkarsh Azad, Borja Requena, Alessandro Roggero, David Wakeham, and Juan Miguel Arrazola, *Early fault-tolerant quantum algorithms in practice: Application to ground-state energy estimation*, Quantum **9**, 1682 (2025).

[14] S. Aaronson, *Read the fine print*, Nature Physics **11**, 291 (2015).

[15] H.-Y. Huang, R. Kueng, and J. Preskill, *Predicting many properties of a quantum system from very few measurements*, Nature Physics **16**, 1050–1057 (2020).

[16] D. Ventura and T. Martinez, *Initializing the amplitude distribution of a quantum state*, Foundations of Physics Letters **12**, 547–559 (1999).

[17] Gui-Lu Long and Yang Sun, *Efficient scheme for initializing a quantum register with an arbitrary superposed state*, Phys. Rev. A **64**, 014303 (2001).

[18] Mikko Möttönen, Juha J. Vartiainen, Ville Bergholm, and Martti M. Salomaa, *Transformation of quantum states using uniformly controlled rotations*, Quantum Info. Comput. **5**, 467–473 (2005).

[19] M. Plesch and Č. Brukner, *Quantum-state preparation with universal gate decompositions*, Physical Review A **83**, 032302 (2011).

[20] S. Ramos-Calderer, A. Pérez-Salinas, D. García-Martín, C. Bravo-Prieto, J. Cortada, J. Planagumà, and J. I. Latorre, *Quantum unary approach to option pricing*, Phys. Rev. A **103**, 032414 (2021).

[21] S. Johri *et al.*, *Nearest centroid classification on a trapped ion quantum computer*, npj Quantum Information **7**, 122 (2021).

[22] I. Kerenidis, J. Landman, and N. Mathur, *Classical and quantum algorithms for orthogonal neural networks* (2022), arXiv:2106.07198 [quant-ph] .

[23] I. Kerenidis and A. Prakash, *Quantum machine learning with subspace states* (2022), arXiv:2202.00054 [quant-ph] .

[24] L. Monbroussou, E. Z. Mamon, J. Landman, A. B. Grilo, R. Kukla, and E. Kashefi, *Trainability and expressivity of Hamming-weight preserving quantum circuits for machine learning*, Quantum **9**, 1745 (2025).

[25] J. Eisert *et al.*, *Quantum certification and benchmarking*, Nature Reviews Physics **2**, 382–390 (2020).

[26] M. Kliesch and I. Roth, Theory of quantum system certification, PRX Quantum **2**, 010201 (2021).

[27] Scott Aaronson and Daniel Gottesman, *Improved simulation of stabilizer circuits*, Phys. Rev. A **70**, 052328 (2004).

[28] H.-Y. Hu, S. Choi, and Y.-Z. You, *Classical shadow tomography with locally scrambled quantum dynamics*, Phys. Rev. Res. **5**, 023027 (2023).

[29] K. Bu, D. E. Koh, R. J. Garcia, and A. Jaffe, *Classical shadows with Pauli-invariant unitary ensembles*, npj Quantum Information **10**, 6 (2024).

[30] A. A. Akhtar, H.-Y. Hu, and Y.-Z. You, *Scalable and flexible classical shadow tomography with tensor networks*, Quantum **7**, 1026 (2023).

[31] M. Arienzo, M. Heinrich, I. Roth, and M. Kliesch, *Closed-form analytic expressions for shadow estimation with brickwork circuits*, Quantum Information and Computation **23**, 961–993 (2023).

[32] C. Bertoni *et al.*, *Shallow Shadows: Expectation estimation using low-depth random Clifford circuits*, Phys. Rev. Lett. **133**, 020602 (2024).

[33] Yi-Zhuang You, Zhao Yang, and Xiao-Liang Qi, *Machine learning spatial geometry from entanglement features*, Phys. Rev. B **97**, 045153 (2018).

[34] Yi-Zhuang You and Yingfei Gu, *Entanglement features of random Hamiltonian dynamics*, Phys. Rev. B **98**, 014309 (2018).

[35] W. J. Huggins *et al.*, *Unbiasing fermionic quantum Monte Carlo with a quantum computer*, Nature **603**, 416–420 (2022).

[36] G.-L. R. Anselmetti *et al.*, *Local, expressive, quantum-number-preserving VQE ansätze for fermionic systems*, New Journal of Physics **23**, 113010 (2021).

[37] J. M. Arrazola *et al.*, *Universal quantum circuits for quantum chemistry*, Quantum **6**, 742 (2022).

[38] E. A. Cherrat *et al.*, *Quantum vision transformers*, Quantum **8**, 1265 (2024).

[39] H. Zhang *et al.*, *Efficient option pricing with a unary-based photonic computing chip and generative adversarial learning*, Photon. Res. **11**, 1703–1712 (2023).

[40] L. Grover and T. Rudolph, *Creating superpositions that correspond to efficiently integrable probability distributions*, arXiv preprint quant-ph/0208112 (2002).

[41] S. Lloyd and C. Weedbrook, *Quantum generative adversarial learning*, Phys. Rev. Lett. **121**, 040502 (2018).

[42] P.-L. Dallaire-Demers and N. Killoran, *Quantum generative adversarial networks*, Phys. Rev. A **98**, 012324 (2018).

[43] K. Mitarai, M. Kitagawa, and K. Fujii, *Quantum analog-digital conversion*, Phys. Rev. A **99**, 012301 (2019).

[44] A. Holmes and A. Y. Matsuura, *Efficient quantum circuits for accurate state preparation of smooth, differentiable functions*, in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2020) pp. 169–179.

[45] I. F. Araujo, D. K. Park, F. Petruccione, and A. J. da Silva, *A divide-and-conquer algorithm for quantum state preparation*, Scientific Reports **11**, 6329 (2021).

[46] S. Jaques and A. G. Rattew, *QRAM: A survey and critique* (2023), arXiv:2305.10310 [quant-ph] .

[47] A. Barenco *et al.*, *Elementary gates for quantum computation*, Physical Review A **52**, 3457–3467 (1995).

[48] A. Y. Kitaev, *Quantum computations: Algorithms and error correction*, Russian Mathematical Surveys **52**, 1191 (1997).

[49] P. O. Boykin *et al.*, *On Universal and Fault-Tolerant Quantum Computing* (1999), arXiv:quant-ph/9906054 [quant-ph] .

[50] W. J. Huggins *et al.*, *Virtual distillation for quantum error mitigation*, Phys. Rev. X **11**, 041036 (2021).

[51] J. Gibbs *et al.*, *Long-time simulations with high fidelity on quantum hardware*, npj Quantum Information **8**, 135 (2022).

[52] L. Zhao, J. Goings, K. Shin, W. Kyoung, J. I. Fuks, J.-K. Kevin Rhee, Y. M. Rhee, K. Wright, J. Nguyen, J. Kim, and S. Johri, *Orbital-optimized pair-correlated electron simulations on trapped-ion quantum computers*, npj Quantum Information **9**, 60 (2023).

[53] N. Jain, J. Landman, N. Mathur, and I. Kerenidis, *Quantum Fourier networks for solving parametric PDEs*, Quantum Science and Technology **9**, 035026 (2024).

[54] Z. He *et al.*, *Alignment between initial state and mixer improves QAOA performance for constrained optimization*, npj Quantum Information **9** (2023).

[55] E. A. Cherrat *et al.*, *Quantum deep hedging*, Quantum **7**, 1191 (2023).

[56] M. Ragone *et al.*, *A Lie algebraic theory of barren plateaus for deep parametrized quantum circuits*, Nature Communications **15** (2024).

[57] IonQ Inc., *IonQ Aria* (2024).

[58] P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, *Error mitigation with Clifford quantum-circuit data*, Quantum **5**, 592 (2021).

[59] A. Lowe, M. H. Gordon, P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, *Unified approach to data-driven quantum error mitigation*, Phys. Rev. Res. **3**, 033098 (2021).

[60] S. Even, *Algorithmic Combinatorics* (The Macmillan Company, 1973).

[61] J. Landman *et al.*, *Quantum methods for neural networks and application to medical image classification*, Quantum **6**, 881 (2022).

[62] T. M. L. de Veras, L. D. da Silva, and A. J. da Silva, *Double sparse quantum state preparation*, Quantum Information Processing **21**, 204 (2022).

[63] I. F. Araujo *et al.*, *Quantum computing library* (2023).

[64] V. V. Shende, S. S. Bullock, and I. L. Markov, *Synthesis of quantum-logic circuits*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **25**, 1000–1010 (2006).

[65] A. Javadi-Abhari *et al.*, *Quantum computing with Qiskit* (2024), arXiv:2405.08810 [quant-ph] .

[66] Constantino Tsallis, *Possible generalization of Boltzmann-Gibbs statistics*, Journal of Statistical Physics **52**, 479–487 (1988).

[67] C. Tsallis, S. V. F. Levy, A. M. C. Souza, and R. Maynard, *Statistical-mechanical foundation of the ubiquity of Lévy distributions in nature*, Phys. Rev. Lett. **75**, 3589–3593 (1995).

[68] D. Prato and C. Tsallis, *Nonextensive foundation of Lévy distributions*, Phys. Rev. E **60**, 2398–2401 (1999).

[69] C. Tsallis, *Nonadditive entropy and nonextensive statistical mechanics - An overview after 20 years*, Brazilian Journal of Physics **39**, 337–356 (2009).

[70] L. Borland, *Option pricing formulas based on a non-Gaussian stock price model*, Physical Review Letters **89** (2002).

[71] L. Borland, The pricing of stock options, in *Nonextensive Entropy: Interdisciplinary Applications* (Oxford University Press, 2004).

[72] V. Witkovský, *Characteristic function of the Tsallis q-Gaussian and its applications in measurement and metrology*, Metrology **3**, 222–236 (2023).

[73] F. Fernández-Navarro *et al.*, *Evolutionary q-Gaussian Radial Basis Function Neural Network to determine the microbial growth/no growth interface of* Staphylococcus aureus, Applied Soft Computing **11**, 3012–3020 (2011).

[74] J. Helsen *et al.*, General framework for randomized benchmarking, PRX Quantum **3**, 020357 (2022).

[75] S. Efthymiou *et al.*, *Qibo: a framework for quantum simulation with hardware acceleration*, Quantum Science and Technology **7**, 015018 (2021).

[76] IonQ Inc., *IonQ Quantum Cloud* (2024).

[77] IBM Inc., IBM Quantum (2025).

[78] Quantinuum, Quantinuum H Series (2025).

[79] V. Bergholm *et al.*, *PennyLane: Automatic differentiation of hybrid quantum-classical computations* (2022), arXiv:1811.04968 [quant-ph] .

[80] U. Azad, *PennyLane Quantum Chemistry Datasets* (2023).

[81] Adam Paszke *et al.*, *PyTorch: An imperative style, high-performance deep learning library* (2019), arXiv:1912.01703 [cs.LG] .

[82] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization* (2017), arXiv:1412.6980 [cs.LG] .

[83] Sergi Ramos-Calderer, *Efficient quantum interpolation of natural data*, Phys. Rev. A **106**, 062427 (2022).

[84] M. Sciorilli, L. Borges, T. L. Patti, D. García-Martín, G. Camilo, A. Anandkumar, and L. Aolita, *Towards large-scale quantum optimization solvers with few qubits*, Nature Communications **16**, 476 (2025).

[85] J. Wilkens, *Quantum Circuit Library* (2023).

[86] A. Elben *et al.*, *The randomized measurement toolbox*, Nat. Rev. Phys. **5**, 9–24 (2022).

[87] T. Brydges *et al.*, *Probing Rényi entanglement entropy via randomized measurements*, Science **364**, 260–263 (2019).

[88] S. H. Sack *et al.*, *Avoiding barren plateaus using classical shadows*, PRX Quantum **3**, 020365 (2022).

[89] Ewout van den Berg, Zlatko K. Minev, and Kristan Temme, *Model-free readout-error mitigation for quantum expectation values*, Phys. Rev. A **105**, 032620 (2022).

[90] A. Seif *et al.*, *Shadow distillation: Quantum error mitigation with classical shadows for near-term quantum processors*, PRX Quantum **4**, 010303 (2023).

[91] H. Jnane *et al.*, *Quantum error mitigated classical shadows*, PRX Quantum **5**, 010324 (2024).

[92] A. Abbas *et al.*, *On quantum backpropagation, information reuse, and cheating measurement collapse*, in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23 (Curran Associates Inc., Red Hook, NY, USA, 2024).

[93] A. Rath *et al.*, *Entanglement barrier and its symmetry resolution: Theory and experimental observation*, PRX Quantum **4**, 010318 (2023).

[94] B. Vermersch *et al.*, *Enhanced estimation of quantum properties with common randomized measurements*, PRX Quantum **5**, 010352 (2024).

[95] V. Vitale *et al.*, *Robust estimation of the Quantum Fisher Information on a quantum processor*, PRX Quantum **5** (2024).

[96] J. Denzler, A. A. Mele, E. Derbyshire, T. Guaita, and J. Eisert, *Learning fermionic correlations by evolving with random translationally invariant Hamiltonians*, Phys. Rev. Lett. **133**, 240604 (2024).

[97] Y. Zhou, P. Zeng, and Z. Liu, *Single-copies estimation of entanglement negativity*, Phys. Rev. Lett. **125**, 200502 (2020).

[98] Z. Liu, P. Zeng, Y. Zhou, and M. Gu, *Characterizing correlation within multipartite quantum systems via local randomized measurements*, Physical Review A **105** (2022).

[99] A. J. Scott, *Tight informationally complete quantum measurements*, Journal of Physics A: Mathematical and General **39**, 13507–13530 (2006).

[100] D. E. Koh and S. Grewal, *Classical shadows with noise*, Quantum **6**, 776 (2022).

[101] R. Brieger, I. Roth, and M. Kliesch, Compressive gate set tomography, PRX Quantum **4**, 010325 (2023).

[102] E. Onorati *et al.*, *Noise-mitigated randomized measurements and self-calibrating shadow estimation* (2024), arXiv:2403.04751 .

[103] H.-Y. Hu *et al.*, *Logical shadow tomography: Efficient estimation of error-mitigated observables* (2022), arXiv:2203.07263 .

[104] D. McNulty, F. B. Maciejewski, and M. Oszmaniec, *Estimating Quantum Hamiltonians via Joint Measurements of Noisy Noncommuting Observables*, Phys. Rev. Lett. **130**, 100801 (2023).

[105] Y. Zhou and Z. Liu, *A hybrid framework for estimating nonlinear functions of quantum states*, npj Quantum Information **10**, 62 (2024).

[106] Q. Liu *et al.*, *Auxiliary-free replica shadow estimation* (2024), arXiv:2407.20865 [quant-ph] .

[107] D. Grier, H. Pashayan, and L. Schaeffer, *Principal eigenstate classical shadows*, in *Proceedings of Thirty Seventh Conference on Learning Theory*, Proceedings of Machine Learning Research, Vol. 247 (PMLR, 2024) pp. 2122–2165.

[108] L. E. Fischer, T. Dao, I. Tavernelli, and F. Tacchino, *Dual-frame optimization for informationally complete quantum measurements*, Phys. Rev. A **109**, 062415 (2024).

[109] S. Chen, W. Yu, P. Zeng, and S. T. Flammia, *Robust shadow estimation*, PRX Quantum **2**, 030348 (2021).

[110] I. V. Oseledets and S. V. Dolgov, *Solution of Linear Systems and Matrix Inversion in the* TT-*Format*, SIAM Journal on Scientific Computing **34**, A2718–A2739 (2012).

[111] R. J. Garcia, Y. Zhou, and A. Jaffe, *Quantum scrambling with classical shadows*, Phys. Rev. Res. **3**, 033155 (2021).

[112] S. F. D. Waldron, *An introduction to finite tight frames* (Birkhäuser New York, NY, 2018).

[113] D. P. DiVincenzo, D. W. Leung, and B. M. Terhal, *Quantum data hiding*, IEEE Transactions on Information Theory **48**, 580–598 (2002).

[114] D. Gross, K. Audenaert, and J. Eisert, *Evenly distributed unitaries: On the structure of unitary designs*, Journal of Mathematical Physics **48** (2007).

[115] H. Zhu, R. Kueng, M. Grassl, and D. Gross, *The Clifford group fails gracefully to be a unitary 4-design* (2016), arXiv:1609.08172 [quant-ph] .

[116] A. W. Harrow, *The Church of the Symmetric Subspace* (2013), arXiv:1308.6595 [quant-ph] .

[117] A. Kandala *et al.*, *Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets*, Nature **549**, 242–246 (2017).

[118] Dmytro Fedoriaka, *Decomposition of unitary matrix into quantum gates* (2025), arXiv:2501.07786 [quant-ph] .

[119] T. O. Maciel *et al.*, *in preparation* (2025).

[120] G. H. Low and I. L. Chuang, *Hamiltonian simulation by qubitization*, Quantum **3**, 163 (2019).

[121] S. Chakraborty, A. Gilyén, and S. Jeffery, *The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation* (Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019).

[122] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, *Grand unification of quantum algorithms*, PRX Quantum **2**, 040203 (2021).

[123] A. Sopena *et al.*, *Algebraic Bethe circuits*, Quantum **6**, 796 (2022).

[124] R. Ruiz *et al.*, *The Bethe ansatz as a quantum circuit*, Quantum **8**, 1356 (2024).

[125] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, *Quantum natural gradient*, Quantum **4**, 269 (2020).

[126] A. J. Ferreira-Martins, R. M. S. Farias, G. Camilo, T. O. Maciel, A. Tosta, R. Lin, A. Alhajri, T. Haug, and L. Aolita, *Variational quantum algorithms with exact geodesic transport* (2025), arXiv:2506.17395 [quant-ph] .

[127] H. S. Cohl, *Fourier, Gegenbauer and Jacobi expansions for a power-law fundamental solution of the polyharmonic equation and polyspherical addition theorems*, Symmetry, Integrability and Geometry: Methods and Applications (2013).

[128] A. Tosta *et al.*, *in preparation* (2025).

[129] R. Iten, R. Colbeck, I. Kukuljan, J. Home, and M. Christandl, *Quantum circuits for isometries*, Phys. Rev. A **93**, 032318 (2016).

[130] R. Vale *et al.*, *Circuit Decomposition of Multicontrolled Special Unitary Single-Qubit Gates*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **43**, 802–811 (2023).

[131] Yan V. Fyodorov, *Introduction to the random matrix theory: Gaussian Unitary Ensemble and beyond*, in *Recent Perspectives in Random Matrix Theory and Number Theory*, London Mathematical Society Lecture Note Series (Cambridge University Press, 2005) p. 31–78.

Chapter 5

Appendices

5.1 Appendices of Chapter 2

5.1.1 Gate compilations and CNOT-gate counts

5.1.1.1 Compilation of multi-controlled gates

Here, we calculate the CNOT cost of the RBS and gRBS gates, as well as their multi-controlled versions. The following lemma will be useful.

Lemma 8 (CNOT cost of a multi-controlled $\mathrm{U}(2)$)**.** *Let $\chi_\ell(U)$ be the number of CNOT gates to implement a ℓ-controlled single-qubit gate $U \in \mathrm{U}(2)$, where $\ell = 0$ corresponds to no controls. Then $\chi_0(U) = 0$, $\chi_1(U) = 2$, $\chi_2(U) = 4$, $\chi_3(U) = 12$, $\chi_4(U) = 36$, and, for $\ell \geq 5$, $\chi_\ell(U) \leq 20\ell - 18$. Moreover, if $U = \mathrm{R_y}(\theta)$ or $U = \mathrm{R_z}(\theta)$,the bound for $\ell \geq 5$ can be improved to $\chi_\ell(U) \leq 16\ell - 24$.*

Proof. Exact compilations for $\ell \leq 4$ follow from Ref. [47, 129]. The upper bounds for $\ell \geq 5$ are given by Ref. [130]. □

Depending on the chosen decomposition for the RBS gate, the gRBS gate admits different decompositions into controlled-$\mathrm{R_y}$ and $\mathrm{R_z}$ gates (*e.g.* see Fig. 2.1). Using Lemma 8, we can now calculate the CNOT cost of a multi-controlled gRBS gate as follows.

# controls	X	$\mathrm{R_y}(\theta)$	$\mathrm{R}^{in}_{out}(\theta)$	$\mathrm{R}^{in}_{out}(\theta,\,\phi)$	$\mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}(\theta)$	$\mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}(\theta,\,\phi)$
$\ell = 0$	0	0	2	2	$\leq 18(m+m') - 42$	$\leq 22(m+m') - 20$
$\ell = 1$	1	2	6	6	$\leq 18(m+m') - 26$	$\leq 22(m+m')$
$\ell = 2$	6	4	10	14	$\leq 18(m+m') - 10$	$\leq 22(m+m') + 20$
$\ell = 3$	≤ 24	12	26	38	$\leq 18(m+m') + 6$	$\leq 22(m+m') + 40$
$\ell = 4$	≤ 40	36	≤ 58	≤ 84	$\leq 18(m+m') + 22$	$\leq 22(m+m') + 60$
$\ell \geq 5$	$\leq 16\ell - 24$	$\leq 16\ell - 24$	$\leq 16\ell - 6$	$\leq 20\ell + 4$	$\leq 18(m+m') + 16\ell - 42$	$\leq 22(m+m') + 20\ell - 20$

Table 5.1: CNOT **count of multi-controlled** X**,** $\mathrm{R_y}$**,** RBS **and** gRBS **gates used throughout this work.** Details are provided in Sec. 5.1.1.1. For each multi-controlled RBS and gRBS gate the CNOT count corresponds to the best between the two compilations in Figs. 2.1 and 2.2. For multi-controlled-$\mathrm{R_y}$ gates, we use the CNOT counts provided in [47, 130]. The numbers in the second column are the χ_ℓ in Lemma 8, while the second and third columns correspond to C_ℓ used in Eq. (2.6) for the real and complex cases, respectively.

Lemma 9 (CNOT cost of a multi-controlled complex gRBS gate)**.** *Let* χ_ℓ *be as defined in Lemma 8,* $c_{\mathbf{ctrl}}\,\mathrm{R}^{\mathbf{in}}_{\mathbf{out}}(\theta,\,\phi) \equiv c_{ctrl_1,\ldots,ctrl_\ell}\,\mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}$ *be as in Sec. 2.2, and* $\mu = \ell + m + m'$*. For* $\phi = 0$*, the number of* CNOTs *to compile this* μ*-qubit gate is*

$$2(m + m' - 1) + \min(2\chi_{\mu-2}(\mathrm{R_y}),\, \chi_{\mu-1}(\mathrm{R_y}))\,.$$

If $\phi \neq 0$*, then the cost is*

$$2(m + m' - 1) + \min(4\chi_{\mu-2}(\mathrm{R_y}),\, \chi_{\mu-1}(U))\,.$$

Proof. (i) Case $\phi = 0$: Since the $\mathrm{R_z}(\phi)$ gates are absent, the ℓ controls can act directly on the $\mathrm{R_y}$ gates (see Fig. 2.2). As a result, $c_{ctrl_1,\ldots,ctrl_\ell}\,\mathrm{R}^{in_1,\ldots,in_m}_{out_1,\ldots,out_{m'}}$ is equivalent to $2(m-1) + 2(m'-1) + 2$ CNOTs and two $(\mu-2)$-controlled $\mathrm{R_y}$ gates for the compilation in Fig. 2.1 (*Top*), or $2(m-1) + 2(m'-1) + 2$ CNOTs and a single $(\mu-1)$-controlled $\mathrm{R_y}$ gate for the compilation in Fig. 2.1 (*Bottom*). The claim then follows from compiling the controlled $\mathrm{R_y}$ gates using Lemma 8 and choosing the less costly compilation between the two. *Top* is the best compilation for $\mu \leq 4$, while *Bottom* is the best otherwise.

(ii) Case $\phi \neq 0$: When using the RBS compilation in Fig. 2.1 (*Top*) to build a complex gRBS, it is necessary to perform 2 $(\mu-2)$-controlled $\mathrm{R_y}$ gates as well as 2 $(\mu-2)$-controlled $\mathrm{R_z}$ gates. Since these gates have the same CNOT cost [130], we express the total cost as 4 times the cost of one $(\mu-2)$-controlled $\mathrm{R_y}$ gate. On the other hand, the product $\mathrm{R_z}(\phi)\ \mathrm{R_y}(\theta)$ in the compilation shown in Fig. 2.1 (*Bottom*) corresponds to a single $\mathrm{SU}(2)$ of the form $U = e^{i\lambda W}$, where $\lambda(\theta,\phi) = \arccos(\cos\theta\cos\phi)$ and $W(\theta,\phi) = \frac{1}{\sin\lambda(\theta,\phi)}(-\sin\theta\sin\phi\,X + \sin\theta\cos\phi\,Y + \cos\theta\sin\phi\,Z)$. The proof then proceeds identically to Case (i). *Top* is the best compilation for $\ell = 0$ and $m = m' = 1$, while *Bottom* is the best otherwise. □

Explicit CNOT counts for all the gates used throughout this work are summarized in Table 5.1.

5.1.1.2 CNOT cost of sparse and binary encoders

Using the results of Sec. 5.1.1.1, we numerically investigate the CNOT cost of implementing the sparse and binary encoders from Secs. 2.4 and 2.5, respectively.

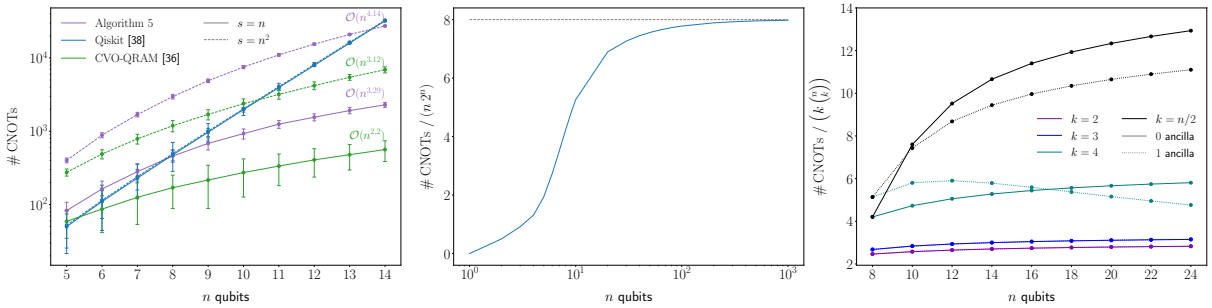

Figure 5.1: (*Left*) **Comparison of the average** CNOT **cost of Alg. 5 with Refs. [64, 62].** CNOT-gate count (in log scale) as a function of the number of qubits n. We compare the costs of Alg. 5 (purple lines) with the algorithms from Refs. [62] (green lines) and [64] (blue lines). We present results averaged over 100 Haar-random states $\in \mathbb{C}^s$, each embedded onto a data vector $\mathbf{x} \in \mathbb{C}^{2^n}$. Sparsity levels $s \in \{n,\, n^2\}$ are represented by solid and dashed lines, respectively. Error bars indicate one standard deviation. (*Middle*) CNOT**-gate count of the binary encoder.** Count based on Eq. (2.15) in Lemma 7 as a function of number of qubits n (in log scale). We used the compilation of controlled $\mathrm{R_y}$ rotations from Tab. 5.1. We observe an asymptotic behavior towards $8\,n\,2^n$ for large n. (*Right*) CNOT **cost of Alg. 3 with the addition of one ancilla.** We compare the CNOT cost of Alg. 3 (solid lines) with the CNOT cost of the same algorithm with the additional step of including a clean ancillary qubit and using it to heuristically remove redundant controlled operations (dotted lines). The comparison is performed for Hamming weights $k \in \{2,\, 3,\, 4,\, n/2\}$ For $k \in \{2,\, 3\}$ (purple and blue lines, respectively), the ancilla does not help decrease the CNOT significantly, if at all, independently of n. For $k = 4$ (green lines), the addition of an ancillary qubit increases the number of CNOTs in the circuits for $n \leq 18$. This is due to the cost of extra controlled Toffoli gates required. For $n > 18$, the heuristic algorithm starts removing CNOTs of the original circuit, and the advantage with respect to the ancilla-free circuit increases with n. For $k = n/2$ (black lines), we can see that the contribution of the one clean ancilla appears at $n > 12$, and increases with n.

Fig. 5.1(a) compares, as a function of the number of qubits n, the average CNOT cost of the sparse encoder presented in Alg. 5 with the QRAM-like sparse encoder from Ref. [62] as well as the ancilla-free state preparation from Ref. [64]. While the former is implemented in the `qclib` package [63], the latter is currently implemented in the `Qiskit` package [65]. We used the `Qibo` package [75] to implement Alg. 5. The CNOT cost is averaged over the encoding of 100 random sparse vectors $\mathbf{y}$ of dimension 2^n and sparsity $s \in \{n,\, n^2\}$. Each $\mathbf{y}$ was obtained by sampling a Haar-random vector $\mathbf{x} \in \mathbb{C}^s$ and embedding it into a 2^n-dimensional sparse data vector by associating each entry x_j with a random computational basis state $|b_j\rangle$

sampled from the uniform distribution. Ref. [64] displayed a scaling of $\mathcal{O}(2^n)$ in CNOT-gate count independent of the sparsity s (solid and dashed blue lines). Even though Ref. [64] provides an ancilla-free method for state preparation, which is shown not to be the most suitable for the preparation of sparse states. The sparse encoder from Ref. [62] presented a scaling of $\sim \mathcal{O}\left(n^{2.2}\right)$ for $s = n$ (solid greed line), and as $\sim \mathcal{O}\left(n^{3.12}\right)$ for $s = n^2$ (dashed green line). This is a significant improvement when compared with Ref. [64]. However, the algorithm from Ref. [62] requires $\mathcal{O}(n)$ ancillary qubits to harness that improvement. Meanwhile, the sparse encoder in Alg. 5 scales as $\sim \mathcal{O}\left(n^{3.29}\right)$ for $s = n$ (solid purple line), and as $\sim \mathcal{O}\left(n^{4.14}\right)$ for $s = n^2$ (dashed purple line). These scalings are $\sim \mathcal{O}(n)$ inferior to the ones from Ref. [62] for both sparsity levels tested. We highlight, however, that Alg. 5 is ancilla-free. This points to a trade-off between ancillary qubits and CNOT-gate cost. Algorithm 5 allows one to have a sparse amplitude encoder that is both $\mathrm{poly}(n)$ in CNOT cost and ancilla-free, at the price of having to implement linearly deeper circuits.

Fig. 5.1(b) shows the asymptotic behavior $\sim 8\,n\,2^n$ of the total gate count for the binary encoder (see Lemma 7 in the main text).

5.1.2 Explicit examples

Here, we present three tables illustrating with explicit examples the execution of the classical algorithms used to generate the quantum circuits for our HW encoder algorithms. Each table corresponds to one of the figures present in the main text, namely: Table 5.2 illustrates the real dense encoder (Alg. 3) and corresponds to Fig. 2.3; Table 5.3 illustrates the real sparse encoder (Alg. 5) and corresponds to Fig. 2.4; finally, Table 5.4 illustrates the binary encoder that combines the dense HW-k for all possible $k \in [0, n]$ and corresponds to Fig. 2.5.

5.1.3 Using one ancilla to reduce CNOT costs

Here, we present a heuristic method that uses one ancilla to remove controlled operations in the later stages of the circuits generated by Algs. 3 and 4 as mentioned in Sec. 2.3. The

results are shown in Fig. 5.1(c). For a given k, the maximum number of indices in the **ctrl** set is $k-1$. Thus, consecutive controlled RBS gates that share the same $k-1$ indices in their respective **ctrl** sets can be implemented together. The *"ancilla trick"* goes as follows. First, we add one ancillary qubit to the system. Then, a controlled Toffoli gate is added before the target consecutive controlled RBSs. This Toffoli gate acts on the ancilla and is controlled by the same qubits that are in the aforementioned **ctrl** set. The next step is the replacement of the set of controls **ctrl** of the target RBS gates by $ctrl = \{n+1\}$, where $n+1$ is the index of the newly added ancillary qubit. Finally, the same Toffoli gate is added to the circuit after the stacked RBS gates, undoing the previous Toffoli operation. The same "trick" can be applied to stacked RBS gates that have the same $k-2$ indices in $ctrl$, and so on. The process ends when adding the two controlled Toffoli gates is more costly than implementing the original gates.

5.1.4 Clifford Data Regression

Clifford Data Regression (CDR) is a data-driven protocol to mitigate errors on expectation values estimated on NISQ devices [58, 59]. The protocol to implement CDR takes as inputs a quantum circuit $\mathcal{C}$ and an expectation value $\mu^{(0)}$ and repeats the following primitive: *(i)* construct a near-Clifford circuit by randomly replacing most, if not all, non-Clifford gates in the original circuit with Clifford gates; *(ii)* run both classical simulation and noisy implementation of the near-Clifford circuit, obtaining a set $\mathcal{S}$ containing new expectation values $\mu_{\text{noiseless}}$ and μ_{noisy} coming from the noiseless and noisy circuit, respectively,

$$\mathcal{S} := \left\{ \left(\mu^{(j)}_{\text{noisy}}, \mu^{(j)}_{\text{noiseless}} \right) \right\}_{j \in [\,|\mathcal{S}|\,]} , \tag{5.1}$$

where $|\mathcal{S}|$ is the cardinality of the set $\mathcal{S}$. The next step is to fit a regression model f on $\mathcal{S}$ such that

$$\mu^{(j)}_{\text{noiseless}} \approx f_{\mathcal{S}} \left(\mu^{(j)}_{\text{noisy}} \right) , \forall\, j \in [\,|\mathcal{S}|\,] \, . \tag{5.2}$$

With that at hand, after executing the original circuit on the noisy hardware and obtaining the noisy expectation value $\mu^{(0)}_{noisy}$, the last step is to use the fitted model to get an error-mitigated

version of $\mu^{(0)}_{\text{noisy}}$, *i.e.*

$$\mu^{(0)}_{\text{mitigated}} \approx f_{\mathcal{S}}\left(\mu^{(0)}_{\text{noisy}}\right) . \tag{5.3}$$

The error-mitigated expectation value $\mu^{(0)}_{\text{mitigated}}$ is then the final empirical estimator of $\mu^{(0)}$.

In our proof-of-principle demonstration described in Sec. 2.6.1, we implemented CDR following the protocol detailed above, training a linear model for each expectation value separately. In doing so, we first compiled the original circuit in a way that the only non-Clifford gates present were $\mathrm{R_y}$ rotations with at most one control qubit. We call the replacement rate, $r \in (0, 1)$, the number of $\mathrm{R_y}$ gates replaced in the original circuit to generate each near-Clifford circuit j. We created our set of near-Clifford circuits using replacement rates $r \in \{0.79,\, 0.82,\, 0.89,\, 0.93,\, 1.00\}$ and sampling 300 circuits per replacement rate, totaling $|\mathcal{S}| = 1500$ data points. The random Cliffords to be inserted were sampled uniformly from the set of all possible single- and two-qubit Clifford gates. We also uniformly sampled the positions in the circuit of the $\mathrm{R_y}$ gates that were replaced. The set (5.1) was generated by classically simulating noiseless circuits locally while simulating the same circuits using the IonQ 's proprietary noise models on IonQ 's Quantum Cloud [76].3 Every probability estimated from the hardware experiment was treated as an expectation value over a rank-1 projector in the computational basis and the mitigation procedure followed as described above by fitting one regression model per probability.

q_6	q_5	q_4	q_3	q_2	q_1	Gate
$\overline{1}$	$\overline{1}$	0	0	0	0	$\cancel{c_6}\,\mathrm{R}_1^5(\theta_1)$
$\overline{1}$	0	$\overline{0}$	$\overline{0}$	$\overline{0}$	1	$\cancel{c_6}\,\mathrm{R}_2^1(\theta_2)$
$\overline{1}$	0	$\overline{0}$	$\overline{0}$	1	0	$\cancel{c_6}\,\mathrm{R}_3^2(\theta_3)$
$\overline{1}$	0	$\overline{0}$	1	0	0	$\cancel{c_6}\,\mathrm{R}_4^3(\theta_4)$
$\overline{1}$	0	1	0	0	0	$c_4\,\mathrm{R}_5^6(\theta_5)$
0	$\overline{1}$	$\overline{1}$	0	0	0	$c_5\,\mathrm{R}_1^4(\theta_6)$
0	$\overline{1}$	0	$\overline{0}$	$\overline{0}$	1	$c_5\,\mathrm{R}_2^1(\theta_7)$
0	$\overline{1}$	0	$\overline{0}$	1	0	$c_5\,\mathrm{R}_3^2(\theta_8)$
0	$\overline{1}$	0	1	0	0	$c_3\,\mathrm{R}_4^5(\theta_9)$
0	0	$\overline{1}$	$\overline{1}$	0	0	$c_4\,\mathrm{R}_1^3(\theta_{10})$
0	0	$\overline{1}$	0	$\overline{0}$	1	$c_4\,\mathrm{R}_2^1(\theta_{11})$
0	0	$\overline{1}$	0	1	0	$c_2\,\mathrm{R}_3^4(\theta_{12})$
0	0	0	$\overline{1}$	$\overline{1}$	0	$c_3\,\mathrm{R}_1^2(\theta_{13})$
0	0	0	$\overline{1}$	0	1	$c_1\,\mathrm{R}_2^3(\theta_{14})$
0	0	0	0	1	1	

Table 5.2: **Illustration of the dense HW-k encoder (Alg. 3) for $n = 6$ and $k = 2$.** Data vector $\mathbf{x} \in \mathbb{R}^d$, with $d = \binom{6}{2} = 15$. The order of the bitstrings is given by Alg. 1. Marked bits are represented with an overline. Qubit indices are enumerated from right to left, $c_{ctrl}\,\mathrm{R}_{out}^{in}$ denotes an RBS on input qubit in, output qubit out controlled by qubit(s) $ctrl$, and θ_j given by Eq. (2.7). Unnecessary $ctrl$ qubits removed by Alg. 2 are represented by $\cancel{c_{ctrl}}$. A gate in a given row represents what needs to be added to the circuit to add to the superposition the computational basis state represented by the bitstring in the next row.

5.2 Appendices of Chapter 3

5.2.1 Details on tensor network description

It is well known that each layer of CNOT gates in the hardware-efficient architecture can be written as an MPO with bond dimension $\chi_C = 2$. To see this, we decompose the gate as

$$\begin{aligned}\mathrm{CNOT} &= \sum_{j\in\{0,1\}} |j\rangle\langle j| \otimes X^j \\ &= \left(\sum_{j\in\{0,1\}} |j\rangle\langle j| \otimes \langle j| \otimes \mathbb{1}\right)\left(\mathbb{1} \otimes \sum_{l\in\{0,1\}} |l\rangle \otimes X^l\right) \\ &= (L \otimes \mathbb{1})(\mathbb{1} \otimes R)\,,\end{aligned} \tag{5.4}$$

where X denotes the Pauli-X gate, and the last line defines the operators L and R. Now we define the MPO core $C_{j,l} := (\langle j| \otimes \mathbb{1})R\,L(\mathbb{1} \otimes |l\rangle) \in \mathbb{C}^{2\times 2}$. The unitary operator L_{CNOT} of

q_6	q_5	q_4	q_3	q_2	q_1	Gate
0	0	0	1	1	1	$c_{1,2}\,\mathrm{R}_4^3(\theta_1)$
0	0	1	0	1	1	$c_{2,4}\,\mathrm{R}_3(\theta_2)^1$
0	0	1	1	1	0	$c_2\,\mathrm{R}_{1,5}^{3,4}(\theta_3)$
0	1	0	0	1	1	$c_{2,5}\,\mathrm{R}_4^1(\theta_4)$
0	1	1	0	1	0	$\mathrm{R}_{1,3,6}^{2,4,5}(\theta_5)$
1	0	0	1	0	1	$c_6\,\mathrm{R}_{2,4,5}^{1,3}(\theta_6)$
1	1	1	0	1	0	

Table 5.3: **Illustration of the sparse encoder (Alg. 5) for $n = 6$ and $s = 7$.** Data vector $\mathbf{y} \in \mathbb{R}^{\mathbf{d}}$, with $d = 2^6$ and $s = 7$ non-zero entries. The addresses b_j are sorted in ascending order of HW (a necessary condition). As a byproduct, this sorting also minimizes the CNOT cost of Alg. 2. The angles θ_j are given by Eq.(2.7). A gate in a given row represents what needs to be added to the circuit to add to the superposition the computational basis state represented by the bitstring in the next row. The first 6 elements have Hamming weight $k = 3$, leading to the initial controlled gRBS gates being HW-preserving. Meanwhile, the last one has $k = 4$ and requires a gRBS with $m \neq m'$ to increase the HW by 1.

a 1D layer of CNOT gates on n qubits with periodic boundary conditions and even n can be written as

$$\sum_{\substack{j_1,j_2,\ldots \\ l_1,l_2,\ldots}} \mathrm{Tr}\left(C_{j_1,l_1} C_{l_2,j_2}, C_{j_3,l_4} \cdots C_{j_{n-1},l_n}\right) |l_1, l_2, \ldots\rangle\langle l_1, l_2, \ldots| \,. \tag{5.5}$$

By construction, L_{CNOT} is an MPO of bond dimension 2. As a consequence $\phi(z, g) = (\phi_k(z, g))_{k\in[n]}$, illustrated in Fig. 5.2 is a TT of bond dimension at most $\chi_C^{2D} = 4^D$.

The operator $\tilde{S}$ can also be written as a shallow circuit of operators acting on quantum channels. Assuming local invariance, also $f = (f_k)_{k\in[n]} \in \mathbb{R}^n$ has a corresponding circuit representation. The rank of the resulting TT description of f is increased by every layer of entangling gates and non-local noise. For fairly general noise affecting the shallow circuits, f is of low TT rank. To this end, it is instructive to discuss a scenario with strictly local and mostly gate-dependent noise. Let us assume that $\Omega_j^{(l)}(U)$ is the CPT map describing the noisy implementation of the gate $U \in \mathrm{U}(2)$ when applied on qubit l in the circuit layer j. This noise model is general enough to also capture local noise and errors affecting the read-out and entangling gates. These noise effects can be included in the respective neighboring layer of single-qubit rotations. The local channel twirl with a group G including noise is formally described by the

q_6	q_5	q_4	q_3	q_2	q_1	Operator	k
0	0	0	0	0	0	$\mathrm{R_y}^6(\theta_1)$	0
$\overline{1}$	0	0	0	0	0	Load_{B_1}	1
⋮	⋮	⋮	⋮	⋮	⋮	$\{\theta_2, ..., \theta_6\}$	1
0	1	0	0	0	0	$c_5\,\mathrm{R_y}^6(\theta_7)$	1
$\overline{1}$	$\overline{1}$	0	0	0	0	Load_{B_2}	2
⋮	⋮	⋮	⋮	⋮	⋮	$\{\theta_8, ..., \theta_{21}\}$	2
0	0	0	0	1	1	$c_{1,2}\,\mathrm{R_y}^3(\theta_{22})$	2
$\overline{0}$	$\overline{0}$	$\overline{0}$	1	1	1	Load_{B_3}	3
⋮	⋮	⋮	⋮	⋮	⋮	$\{\theta_{23}, ..., \theta_{41}\}$	3
1	1	1	0	0	0	$c_{4,5,6}\,\mathrm{R_y}^3(\theta_{42})$	3
$\overline{1}$	$\overline{1}$	$\overline{1}$	$\overline{1}$	0	0	Load_{B_4}	4
⋮	⋮	⋮	⋮	⋮	⋮	$\{\theta_{43}, ..., \theta_{56}\}$	4
0	0	1	1	1	1	$c_{1,2,3,4}\,\mathrm{R_y}^5(\theta_{57})$	4
$\overline{0}$	1	1	1	1	1	Load_{B_5}	5
⋮	⋮	⋮	⋮	⋮	⋮	$\{\theta_{58}, ..., \theta_{62}\}$	5
1	1	1	1	1	0	$c_{2,3,4,5,6}\,\mathrm{R_y}^1(\theta_{63})$	5
1	1	1	1	1	1		6

Table 5.4: **Illustration of the binary encoder for $n = 6$ qubits.** A gate (or collection of gates) in a given row represents what needs to be added to the circuit to add to the superposition the computational basis state represented by the bitstring in the next row. The initial bitstring has $k = 0$, and (controlled) $\mathrm{R_y}$ gates are used to increase the HW by 1. The first bitstring with HW-$(k+1)$ has a one-bit difference from the final bitstring of the Ehrlich algorithm with HW-k. Operators Load_{B_k} fill up the subspace of HW-k, and the algorithm stops after reaching $k = n$. Angles θ_j are given by Eq.(2.7), θ_j are used in $\mathrm{R_y}$ gates and θ_j are used in Load_{B_k}. Total number of parameters is $2^n - 1$.

non-commutative Fourier transform of the map $\Omega_j^{(l)}$ evaluated at the representation of unitary channels $\omega(U) = U \otimes \bar{U}$, *i.e.*

$$\hat{\Omega}_j^{(l)}[\omega] = \int_G \bar{U} \otimes U \otimes \Omega_j^{(l)}(U) \mathrm{d}\mu(U)\,. \tag{5.6}$$

We will here simply use $\hat{\Omega}_j^{(l)}[\omega]$ as a convenient symbol for the "noisy channel twirl". Note that $\hat{\Omega}_j^{(l)}[\omega]$ is a linear operator on superoperators, a "super-duper-operator" if one insists. We can also introduce a corresponding operator for the entangling layer that conjugates a quantum channel with unitary channels of a layer of CNOT gates. The resulting MPO, $\mathcal{C} = \bar{L}_{\mathrm{CNOT}} \otimes$

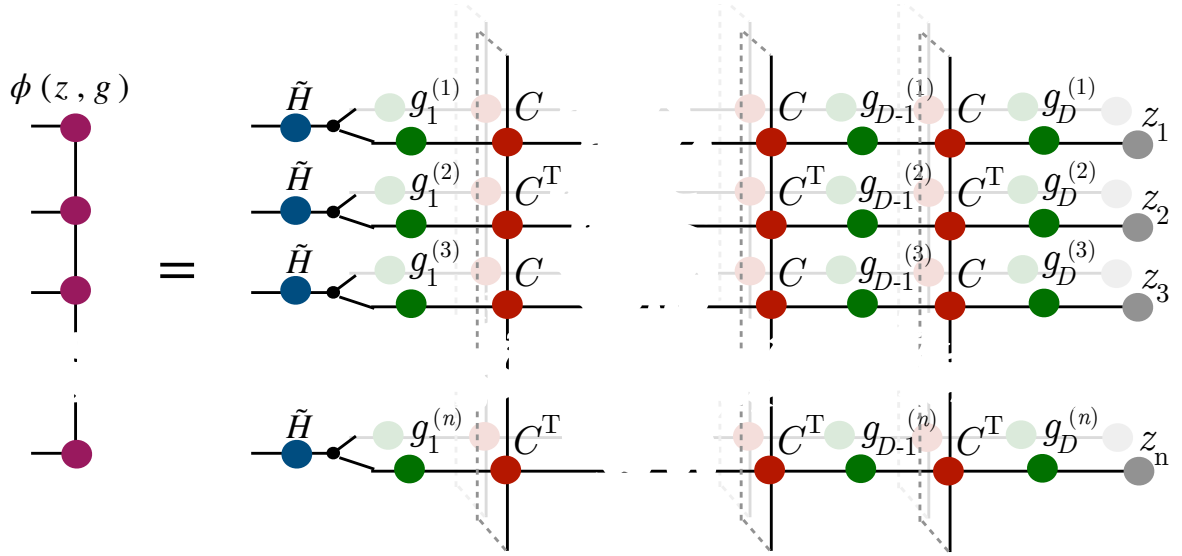

Figure 5.2: **Tensor network of** $\phi(z, g)$. The network in the background is the conjugate of the one in the foreground, determined by the measurement outcome z and the applied random single qubit rotations g. The small black dot denotes the copy-tensor, here projecting on the diagonal elements of a density matrix. The matrix $\tilde{H}$ denotes the Hadamard-Walsh transform. Here it is mapping the diagonal elements of the density matrix to the expectation values of multi-qubit Z observables. The entangling layer with MPO core C and C^T has bond dimension 2. As a consequence, the TT rank of $\phi(z, g)$ is 4^D.

$L_{\mathrm{CNOT}} \otimes L_{\mathrm{CNOT}} \otimes \bar{L}_{\mathrm{CNOT}}$, has bond dimension 16. The read-out is locally described by $\mathcal{B}_2 = \sum_{z\in[2]} |\Pi_z)(\Pi_z|$, the local dephasing channel in computational basis. Finally, to ensure the local invariance of $\tilde{S}$, we assume gate-independent left noise $\Omega_j^{(1)}(U) = \Lambda_j \omega(U)$ for the first layer acting on the state under consideration, with Λ_j being a quantum channel.

Figure 5.3 depicts the resulting expression for f. Without any further assumption, we conclude that f has a TT rank of at most 16^D. The rank is even smaller if all noise channels are Pauli noise. Then, we can fully restrict the circuit in Fig. 5.3 as acting only on the subspace of Pauli noise channels, *i.e.* diagonal channels in Pauli operator basis. To see this, note that the local dephasing channel is a Pauli noise channel. Under the assumption of Pauli noise, the noisy channel twirl maps Pauli noise channels to Pauli noise channels. Being a Clifford unitary, the entangling layer acts as a two-local permutation on the diagonal of the Pauli-noise channel. The representation of $\mathcal{C}$ restricted to Pauli-noise channels has (MPO) bond dimension 4. Hence, f is described by a TT of rank at most $\chi = 4^D$.

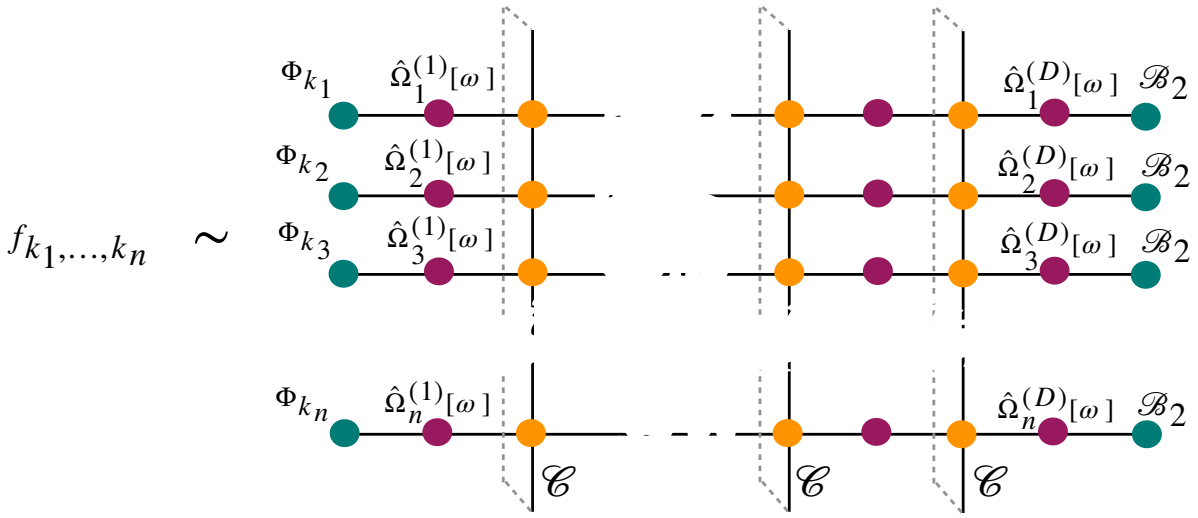

Figure 5.3: **Tensor network of f for strictly local noise, mostly gate-dependent noise.** The entangling layer acting here as a "super-duper operator" on quantum channels has an MPO bond-dimension 16, and restricted to Pauli noise channels, it has an MPO bond dimension 4. Correspondingly, the TT rank of f is 16^D for local noise and 4^D for local Pauli noise.

The argument for one-local noise can be generalized to quasi-local noise by introducing, *e.g.* , MPO descriptions of the implementation map modeling cross-talk effects, or modeling a noisy entangling layer $\mathcal{C}$ with a potentially larger bond dimension. In general, the TT rank of f will be increased due to correlated noise effects. If these correlations are sufficiently short-ranged, f still has an effective low-rank description. We expect that this will often be the case on actual quantum hardware.

5.2.2 Incoherent noise model

We model the incoherent noise as an unitary channel $\Lambda := \exp(i\,\gamma\,H_{\mathrm{g}}/8)$, where H_{g} is a random Hamiltonian sampled from the *Gaussian Unitary Ensemble* (GUE) [131, 101], γ is the so-called *strength parameter*, and the constant 8 is a normalization factor (see Sect. IV-B of Ref. [101]). By sampling $H \sim$ GUE, we guarantee that, on average, that is no preferential direction in which the noise acts. Even though $\gamma \in [0,\, 8]$, we focus on $\gamma < 2$ based on reported error rates for the CNOT gate across multiple commercial quantum computing platforms.

5.2.3 Worst-case bias measure

Consider the case where the half-sided noisy frame operator is still diagonal, *i.e.* $\tilde{S} = \sum_a \tilde{S}_a |w_a)(w_a|$ with w_a, $a \in \mathbb{F}_2^{2n}$, Frobenius-normalized Pauli operators such that $\|w_a\|_F = 1$. We define the bias of standard shadow estimation of an observable from noisy shadow measurements in state ϱ as

$$\mathrm{bias}(O,\, \varrho) \coloneqq |(O \,|\, \varrho) - (O|S^{-1}\tilde{S}|\varrho)| \,. \tag{5.7}$$

The norm difference $\|S^{-1}\tilde{S} - I\|_\infty$ has the interpretation of the worst-case bias for estimating a single Pauli observable $O_a = \sqrt{d}\, w_a$. This can be directly seen from the following expression

$$\max_{\varrho, a \in \mathbb{F}_2^{2n}} \mathrm{bias}(O_a, \varrho) = \sqrt{d} \max_{a,\varrho} |(w_a|I - S^{-1}\tilde{S}|w_a)(w_a \,|\, \varrho)| \leq \|I - S^{-1}\tilde{S}\|_\infty \,, \tag{5.8}$$

where we used the diagonality of $\tilde{S}$ in Pauli basis and that $|(w_a \,|\, \varrho)| \leq \|w_a\|_\infty \|\varrho\|_1 \leq 1/\sqrt{d}$ for all a, ϱ, and $\|\cdot\|_1$ denoting the trace-norm (nuclear / Schatten-1 norm). Most importantly, this bound is saturated for ϱ in an eigenstate of w_a, the observable suffering the maximal bias.

5.2.4 Worst-case bias per Pauli support

In Figure 5.4 we show the same worst-case bias of standard non-mitigated shadow estimation for Pauli observables, under the same noise conditions. This is presented in shaded lines. However, this time we also present the worst-case bias for fixed Pauli supports $\mathrm{PS} = 1, 2$. We observe that the Pauli worst-case biases for these Pauli supports seem to hit a plateau for both noise levels. However, for $r \sim 10^{-2}$, even though stable, the biases are still sizable.

5.2.5 Overlap estimation based on Huggins *et al.* (2022)

The optimization protocol for quantum chemistry problems implemented by Ref. [35] involves estimating the wavefunction overlaps $\langle\phi|\Psi_T\rangle$. This is done by preparing $|\Psi_T\rangle$ in a quantum processor, constructing its classical shadow representation, and evaluating the overlap with multiple $|\phi\rangle$ via post-processing. The state $|\Psi_T\rangle$ is called a *trial function*, and $|\phi\rangle$ is named

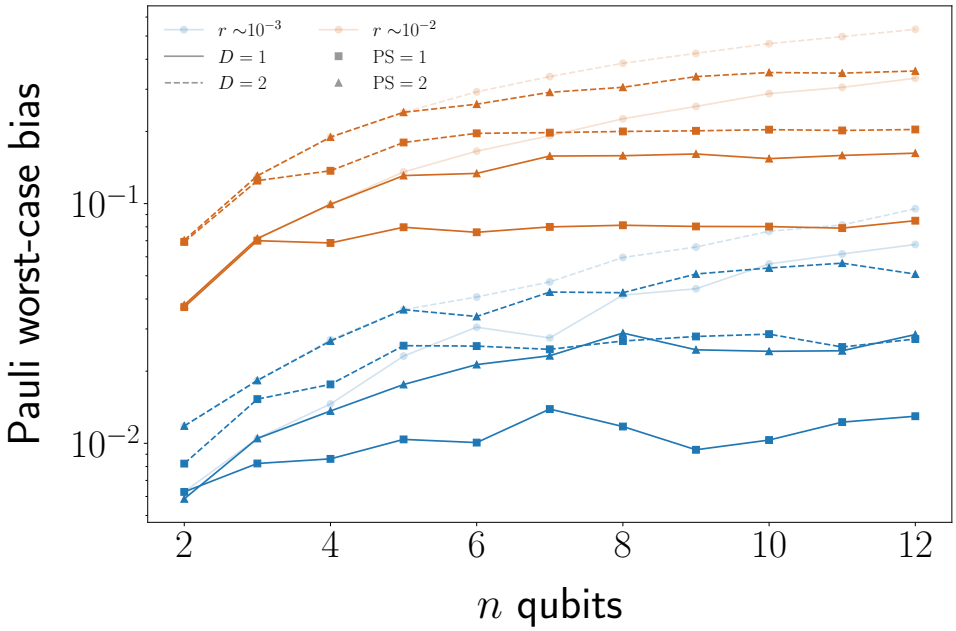

Figure 5.4: **Worst-case bias from Fig. 3.2 for different Pauli supports.**

a *walker function*. We refer to Ref. [35] for more details about $|\Psi_T\rangle$ and $|\phi\rangle$ in the quantum chemistry context. For our intents and purposes, we set a phase $\theta = \pi/\sqrt{2}$, and

$$|\Psi_T\rangle = \cos^2(\theta)\,|11001100\rangle + \frac{1}{2}\sin(2\theta)\,(|11000011\rangle + |00111100\rangle) + \sin^2(\theta)\,|00110011\rangle\ . \tag{5.9}$$

Moreover, we define

$$|\phi_1\rangle = \frac{1}{2}\,(|11001100\rangle + |11000011\rangle + |00111100\rangle + |00110011\rangle)\ . \tag{5.10}$$

and

$$|\phi_2\rangle = \frac{1}{\sqrt{2}}\,|11001100\rangle + \frac{1}{\sqrt{6}}\,(|11000011\rangle + |00111100\rangle + |00110011\rangle)\ . \tag{5.11}$$

We simulate the perfect preparation of $|\Psi_T\rangle$ and its reconstruction via classical shadows with noisy measurements. The noise model chosen is based on the one from the Sycamore processor [5, 6], since Ref. [35] implements this protocol in that quantum hardware.

5.2.6 Data histograms for Fig. 3.5 (a)

As explained in the main text, given the sequence $\mathcal{S}$ of snapshots obtained using the quantum hardware, the k-th eigenvalue of the noisy frame operator $\tilde{S}$ is estimated by calculating

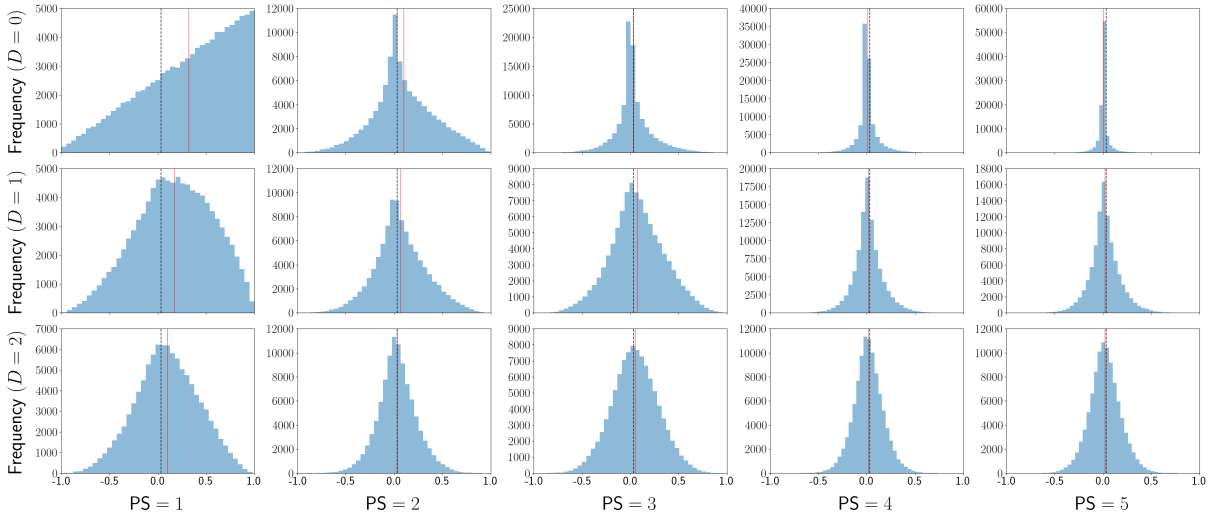

Figure 5.5: **Histograms of experimental realization on IBM Quantum .** Distribution of samples used to estimate eigenvalues of the noisy frame $\mathcal{S}$ for different Pauli supports $1 \leq \mathrm{PS} \leq 5$, over $|\mathcal{S}| = 10^5$ samples from the `ibm_nairobi` quantum processor, for $n = 5$ and $D \in \{0, 1, 2\}$. The average over the 10^5 samples gives our estimates of the eigenvalues of $\tilde{S}$ shown in Fig. 3.5 (a). We indicate the corresponding eigenvalue by dashed red lines, while the dashed black lines indicate the $1/(d+1)$ level.

$\mathbb{E}_{(z,g)\in\mathcal{S}}\ \phi_k(z,g), \forall\, k$, with ϕ_k being defined in Eq. (3.5).

In Figure 5.5, we present the distributions of the data used to estimate the eigenvalues of the noisy frame operator obtained on the `ibm_nairobi` quantum processor using the IBM Quantum platform. As in Fig. 3.5 (a), we show one eigenvalue per Pauli support PS and per depth $D \in \{0, 1, 2\}$. Moreover, we use a sequence of size $|\mathcal{S}| = 10^5$.

We observe that multiple eigenvalues significantly deviate from the value for global rotations, still for $D = 2$. Furthermore, the bias significantly differs for different eigenvalues. Thus, mitigating the bias by rescaling the estimator with a single "global" scalar, or equivalently, only relying on ratio estimators, does not remove the bias.