

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

Optimizing and Characterizing Measurement-Only Topological Quantum Computing with Majorana Zero Mode Based Qubits

Permalink

<https://escholarship.org/uc/item/7bj0t5m0>

Author

Tran, Alan

Publication Date

2020

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

**Optimizing and Characterizing Measurement-Only
Topological Quantum Computing with Majorana Zero Mode
Based Qubits**

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Physics

by

Alan D Tran

Committee in charge:

Dr. Parsa Bonderson, Co-Chair
Professor Chetan Nayak, Co-Chair
Professor Andrea Young

September 2020

The Dissertation of Alan D Tran is approved.

Professor Andrea Young

Professor Chetan Nayak, Committee Co-Chair

Dr. Parsa Bonderson, Committee Co-Chair

August 2020

Optimizing and Characterizing Measurement-Only Topological Quantum Computing with
Majorana Zero Mode Based Qubits

Copyright © 2020

by

Alan D Tran

For my brother and sister, my parents, and my grandparents

Acknowledgements

I would like to acknowledge and thank everyone who has made this dissertation and Ph.D. possible.

I am beyond grateful to my advisor, Parsa Bonderson, for all his patience and guidance throughout my time here. He has been a pillar of support and encouragement and I count myself exceedingly fortunate to have him as a mentor.

I would like to thank all the collaborators I have had the pleasure of working with: Alex Bocharov, Bela Bauer, César Galindo, Colleen Delaney, Eric Rowell, Marcus P. da Silva, Meng Cheng, Roger Mong, Steven Flammia, and Zhenghan Wang.

In addition, I want to thank the physics and math community at UCSB who has shaped my experience here. I am lucky to have been able to discuss with and learn from all the students, post-docs, researchers, visitors, and professors at UCSB, KITP, Microsoft, and HRL. Special thanks goes especially to my committee members Chetan Nayak and Andrea Young, and to Sean Fraer who has made every trip to Q all the more enriching.

Finally, I am deeply thankful to my friends and family who have supported me always and brightened each day.

Curriculum Vitæ

Alan D Tran

Education

- 2020 Ph.D. in Physics (Expected), University of California, Santa Barbara.
2017 M.Sc. in Physics, University of California, Santa Barbara.
2014 B.Sc. in Physics, University of California, Los Angeles.

Publications

- A. Tran, B. Bauer, P. Bonderson, S. Flammia, and M. P. da Silva, *Randomized benchmarking of measurement-only Majorana zero mode qubits*, [In preparation]
- A. Tran, A. Bocharov, B. Bauer, and P. Bonderson, *Optimizing Clifford gate generation for measurement-only topological quantum computation with Majorana zero modes*, [SciPost Phys. **8**, 91 (2020)]
- C. Delaney and A. Tran, *A systematic search of knot and link invariants beyond modular data*, [arXiv:1806:02843]
- P. Bonderson, C. Delaney, C. Galindo, E. Rowell, A. Tran, Z. Wang, *On invariants of modular categories beyond modular data*, [Journal of Pure and Applied Algebra **223**(9) (2019)]

Abstract

Optimizing and Characterizing Measurement-Only Topological Quantum Computing with Majorana Zero Mode Based Qubits

by

Alan D Tran

Quantum computing is a revolutionizing technology on many fronts. Topological quantum computing is an especially attractive platform for this due to its innate robustness to local sources of noise and scalability. As the advent of quantum computing technologies nears, it becomes increasingly important to both optimize as much as possible the basic operations and to characterize them. In this thesis we study measurement-only topological quantum computing with Majorana zero mode (MZM) based qubits, particularly on their incarnation as the boundary defects of $1d$ topological superconducting nanowires. The specific qubit device we consider are comprised of collections of such wires connected together on a single island such that 4 or 6 MZMs are hosted (tetron and hexon qubits respectively).

This thesis focuses on optimizing and characterizing the basic operations needed for quantum computing with such MZM qubits, concentrating on hexons which are the smallest MZM qubit allowing measurement-only gates on a single island. We begin by optimizing the Clifford gates for a single hexon qubit and for the controlled-Pauli, W , and SWAP gates on two hexon qubits. A brute-force search is performed with respect to a physically motivated example cost function. Tools and techniques are developed to aid in the compiling and optimization

of measurement sequences. Utilizing the developed tools, we apply it towards the optimization of the stabilizer measurements for the surface code, an especially useful quantum error correcting code for the present MZM setting. Finally, we discuss adapting the technique of randomized benchmarking to the hexon setting, addressing simplifications and extensions that a hexon qubit can offer.

Contents

Curriculum Vitae	vi
Abstract	vii
1 Introduction	1
2 Majorana Zero Modes in nanowires	6
2.1 Toy model	7
2.2 Realizing MZMs	12
2.2.1 Theoretical proposals	13
2.2.2 Experimental progress	15
2.3 MZM nanowires as qubits	16
3 Optimal Clifford gate synthesis for MZM hexon qubits	22
3.1 Introduction	23
3.2 Majorana Hexon Architecture	27
3.2.1 Single-hexon state space and operators	32
3.2.2 Single-qubit gates through measurements	37
3.2.3 Multi-hexon operations	40
3.2.4 Not all measurements are created equal	48
3.2.5 Relabeling Majorana zero modes	52
3.3 Forced-Measurement Methods	55
3.3.1 Forced-measurement protocols for 2-MZM measurements	56
3.3.2 Forced-measurement protocols involving $2N$ -MZM measurements	61
3.3.3 Procrastination methods	63
3.3.4 Adaptive methods	68
3.4 Majorana-Pauli Tracking	69
3.4.1 Proof of Majorana-Pauli tracking	71
3.5 Brute-force Optimization of Measurement-Only Generation of Gates	75
3.5.1 Optimization	75
3.5.1.1 Majorana-Pauli tracking methods	77

3.5.1.2	Forced-measurement methods	79
3.5.2	Gate search	80
3.5.3	Demonstration of Methods	84
3.5.4	Comparative Analysis	87
3.6	Final Remarks	89
4	Optimal stabilizer measurements for measurement-only MZM based surface code	91
4.1	Overview and motivation	92
4.2	Measurement circuit and example compilation	95
4.3	Circuit optimization	102
4.4	Boundary circuits	105
4.5	Interleaving	106
5	Measurement-Only Randomized Benchmarking for MZM Hexon Qubits	113
5.1	Introduction	114
5.2	Review	115
5.2.1	Notation	115
5.2.2	Randomized benchmarking	116
5.3	Randomized benchmarking on a hexon	120
5.4	Simulations	131
5.5	Discussion	134
A	Optimal compilation tools and details	135
A.1	Example of Adaptive Forced-Measurement Protocol	135
A.2	Sequence Morphology	137
A.2.1	Some general formulas	137
A.2.2	Sequence Morphology	140
A.2.2.1	Space-time reflections	141
A.2.2.2	Paulimorphism	143
A.2.2.3	Sequence reduction	147
A.3	Demonstration Details	151
A.3.1	Two-Sided Hexon with Configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$	151
A.3.2	One-Sided Hexon with Configuration $\langle 1, 2, 6, 3, 4, 5 \rangle$	157
A.3.3	One-Sided Hexon with Configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$	164
B	Optimal surface code details	172
B.1	Measurement sequences for an all-hexon surface code	172
B.2	Optimal stabilizer measurement sequence	174
	Bibliography	190

Chapter 1

Introduction

Quantum computing is a paradigm for understanding and manipulating information at a more fundamental, quantum mechanical level [6, 29, 30, 67, 75] and has the potential to be a disruptive, revolutionizing technology. Already, it has birthed new fields of study and connected previously disparate ones. It promises applications in efficient simulation of quantum systems [42], cryptography [82], and general optimization tasks to name a few. These can for example aid in the design of novel drugs [91] or to the engineering of more efficient industrial processes, for instance in nitrogen fixation which accounts for on the order of 2% of world energy consumption [79]. In addition many quantum algorithms have been developed that are lower in complexity than any rival classical algorithm [7, 28, 82]. Efficient prime factorization is one such algorithm and it furthermore breaks many modern encryption schemes. This has necessitated the development of cryptographic channels secure even against quantum computers [8–10]. In addition, such quantum algorithms have also spurred the development

of quantum inspired classical algorithms that perform exponentially better than previous iterations [4, 85]. Notwithstanding, there are hopefully many yet unforeseen possibilities and synergies to be discovered as was the case with the introduction of classical computing. As a result, quantum information and computing has grown significantly in recent years and has attracted interest and investments from almost all sectors, from academic institutions to startups and venture capital firms to established corporations and from governments all around the world.

Quantum information is stored in quantum mechanical systems, the simplest of which are two-level systems known as *qubits* (quantum bits). Qubits are processed via quantum gates which are unitary operations acting on their Hilbert space. Output can be extracted from measuring qubits which give a binary, classical result. A system of N qubits has a 2^N dimensional Hilbert space which requires 2^N complex numbers to represent. Therefore, the resources required to simulate a quantum mechanical system scales polynomially in the number of qubits versus exponentially in the number of bits. It is this key feature which naturally gives quantum computers an edge in simulating quantum systems. The computational power of quantum computers also leads to them being more error prone as well. In contrast to classical bits which are discrete, qubits can continuously interpolate between two basis states. Schematically, a classical bit is pinned to point from the center towards either the north or south pole of a globe while a qubit can point to any position ¹. Small enough perturbations cannot reverse the classical bit's direction but arbitrarily small errors can change the state of a qubit, decohering it

¹A qubit can be mapped to the Bloch sphere [67]: a qubit's state vector can be represented as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \cos(\theta/2)|0\rangle + e^{i\phi}\sin(\theta/2)|1\rangle$ due to the normalization constraint and neglecting an overall phase. θ and ϕ correspond to a point on a the unit 2-sphere.

as errors are accumulated during its lifetime. To combat this, quantum error correcting codes are used [36]. These encode k logical qubits into n physical qubits. The extra redundancy spreads errors and allows for their detection and correction, at the cost of additional physical resources and more difficult to implement logical level gates. Provided the physical error rate is below some threshold, a quantum error correcting code can yield arbitrarily small logical error rates. Thus, all quantum computing platforms need to be scalable enough to achieve a sufficiently low logical error rate within some maximal number of physical qubits (e.g. as set by experimental constraints, monetary cost, etc.).

Topological quantum computing is an especially attractive platform for this. In this scheme, the aim is to form robust, topologically protected qubits and gates via a topological phase of matter [43, 44, 65]. A topological phase can host exotic quasiparticles known as non-Abelian anyons. Quantum information (e.g. a qubit or a qudit) can be stored in the non-degenerate state space of a collection of these non-Abelian anyons. The different states are indistinguishable from one another by any local observables, up to exponentially suppressed parameters. Thus, when the quasiparticles are far apart, the non-local storage of information provides protection from local sources of noise, exponentially suppressed by the distance between. Quantum information can be manipulated through the application of quantum gates. In this setting, gates are performed via the braiding of quasiparticles. Here, the topological protection stems from the fact that the particular path of a braid is unimportant, it is the topological class of the braiding that matters. Although more challenging to engineer, topological qubits are inherently more robust to noise. Despite this topological protection, in the near term at least, quantum error

correcting codes, such as the surface code, must still be employed in order to suppress the overall error rate.

Though there are many proposals for quantum computing including: spin qubits, trapped ion qubits, superconducting qubits, photonic qubits, etc. In this thesis we study topological quantum computing with Majorana zero mode (MZM) based qubits, particularly in their incarnation as the boundary defects of $1d$ topological superconducting nanowires and with measurement-only braiding gates. The specific qubit devices we consider are such nanowires connected onto a single island, hosting 4 or 6 MZMs. These are termed tetrons and hexons, respectively and quantum gates proceed by sequences of parity measurements on different pairs of MZMs. Our focus in this thesis is on optimizing and characterizing the basic operations needed for quantum computing with such MZM qubits, specifically for hexons which are the smallest MZM qubit allowing measurement-only gates on a single island.

In Ch. 2 we give an overview of particular example of topological quantum computing, reviewing a simple toy model hosting MZMs, experimental realizations, and how qubits can be formed and measurement-only gates applied. In Ch. 3 we turn our attention to optimizing the Clifford gates for a single hexon qubit and for the controlled-Pauli, W , and SWAP gates on two hexon qubits. A brute-force search is performed with respect to a physically motivated example cost function. Significant improvements from the naïve method of first finding optimal measurement sequences realizing the basic braids and then composing these are reported. Tools and techniques are developed to aid in the compiling and optimization of measurement sequences—these are especially useful when a brute force search may not be possible or for

greater number of qubits. Using the tools developed, in Ch. 4 we apply it towards the problem of optimizing stabilizer measurements for the surface code, a class of quantum error correcting code especially useful in the present MZM setting. Here the optimization for a full surface code cycle additionally requires that different stabilizer measurements are compatible with one another and as simultaneous as possible. We report such a result for a mixed tetron-hexon layout and provide some useful unoptimized results for a denser all hexon layout. Finally, in Ch. 5 we discuss randomized benchmarking in the hexon setting, addressing simplifications and extensions that a hexon qubit can offer. In particular, we utilize the ability to frame track Clifford gates and perform measurements in any Pauli basis to obviate the final inversion gate of the original randomized benchmarking protocol. Additionally, we can make use of the random measurement outcomes by sampling from the Clifford gateset up to an overall Pauli with the random outcomes then generating the full Clifford group. This reduces the sampling space for n qubits by a factor of 4^n . Finally, since all hexon operations are composed of measurements, we explore the use of randomized benchmarking for deducing the average measurement fidelity.

Chapter 2

Majorana Zero Modes in nanowires

In this chapter we review Majorana zero modes (MZM), specifically in their incarnation as boundary defects in $1d$ nanowires. We begin by introducing a simple $1d$ toy model due to Kitaev [46] that hosts MZMs at its boundary. Though this toy model cannot be directly realized, there are proposals to engineer the key features of this model through combining relatively achievable components [61, 68]. We discuss these proposals and give a brief overview of their experimental progress. Finally, assuming that MZMs are able to be produced, we review topological quantum computing with MZMs.

2.1 Toy model

As a preliminary we lay out some key properties. Formally, Majorana operators are self-adjoint operators

$$\gamma_j = \gamma_j^\dagger \quad (2.1)$$

satisfying the anticommutation relations

$$\{\gamma_i, \gamma_j\} = 2\delta_{ij}. \quad (2.2)$$

In contrast, a fermionic annihilation and creation operators c_j, c_j^\dagger satisfy the anticommutation relations

$$\{c_i, c_j\} = 0 \quad \text{and} \quad \{c_i, c_j^\dagger\} = \delta_{ij}. \quad (2.3)$$

The two are related via

$$c_j = \frac{1}{2}(\gamma_{j,a} - i\gamma_{j,b}), \quad (2.4)$$

$$\gamma_{j,a} = c_j + c_j^\dagger, \quad (2.5)$$

$$\gamma_{j,b} = \frac{c_j - c_j^\dagger}{i} \quad (2.6)$$

from which the number occupation operator follows

$$c_j^\dagger c_j = \frac{1}{2}(1 + i\gamma_{j,a}\gamma_{j,b}) \quad (2.7)$$

with $c_j^\dagger c_j$ taking values 0 or 1 and $i\gamma_{j,a}\gamma_{j,b}$ taking values ± 1 .

A qubit is a 2-state quantum mechanical system. We can, for example, ascribe these two states to the unoccupied $|0\rangle$ and occupied $|1\rangle$ states of a spinless electron. There are two types of error which may affect this qubit, bit flip X errors and phase Z errors:

$$X = |0\rangle\langle 1| + |1\rangle\langle 0| \quad Z = |0\rangle\langle 0| - |1\rangle\langle 1|. \quad (2.8)$$

In a system of N such qubits, the conservation of fermion parity prohibits a single X type error from occurring. Furthermore, a Z type error which corresponds to $1 - 2c_j^\dagger c_j = -i\gamma_{j,a}\gamma_{j,b}$ would be suppressed if the underlying Majorana operators $\gamma_{j,a}$ and $\gamma_{j,b}$ were spatially separated.

Spurred by this, we come to Kitaev's model [46] of a $1d$ spinless p -wave superconductor

$$H = -\mu \sum_{j=1}^L c_j^\dagger c_j - \frac{1}{2} \sum_{j=1}^L \left(t c_j^\dagger c_{j+1} + \Delta c_j c_{j+1} + \text{h.c.} \right). \quad (2.9)$$

It describes a chain of spinless electrons at sites $j = 1 \dots L$ with chemical potential μ , hopping t , and p -wave pairing Δ . This Hamiltonian is mathematically equivalent to the transverse field Ising model via a Jordan-Wigner transformation [45].

Using Eq. 2.4 to rewrite this in terms of Majorana operators we get

$$H = -\frac{\mu}{2} \sum_{j=1}^L (1 + i\gamma_{j,a}\gamma_{j,b}) - \frac{1}{4} \sum_{j=1}^{L-1} [(\Delta + t)i\gamma_{j,b}\gamma_{j+1,a} + (\Delta - t)\gamma_{j,a}i\gamma_{j+1,b}]. \quad (2.10)$$

μ now corresponds to an on-site pairing between Majoranas while $\Delta \pm t$ correspond to inter-site Majorana pairing terms as illustrated in Fig. 2.1.

This Hamiltonian admits two phases, a trivial one and a non-trivial topological one. This can be seen by first imposing periodic boundary conditions on the chain in order to study its bulk properties [3]. Fourier transforming to momentum space, the spectrum of the Hamiltonian is

$$E_{\text{bulk}} = \sqrt{(\mu + t \cos(k))^2 + \Delta^2 \sin^2(k)}. \quad (2.11)$$

It is gapped everywhere except $\mu = \pm t$ giving three regions, $\mu < -t$, $|\mu| < t$, and $\mu > t$. The first and last are related by a particle-hole transformation so there are two distinct phases. A trivial, strong-pairing phase and a topological, weak-pairing phase. The topological character of the latter can be understood via the map $\mathbf{h}(k)$ from the Brillouin zone to the Hamiltonian. At non-gapless points, this map is from the Brillouin zone to the unit sphere, $\mathbf{h}(k) \rightarrow \hat{\mathbf{h}}(k)$. Due to particle-hole symmetry, the map is constrained so that $\hat{\mathbf{h}}(0)$ is either parallel or anti-parallel to $\hat{\mathbf{h}}(\pi)$. Thus as k is varied from 0 to π , there are two options: either we begin at one pole and end at the same pole, or we begin at one pole and end at the opposite pole. These correspond to the trivial and topological phases respectively.

For open boundary conditions this topological character manifests as the existence of zero-energy Majorana end states with a two-fold ground state degeneracy (Fig. 2.1). We elucidate this by studying the system at fine-tuned points deep within each phase. In the trivial phase $\mu \rightarrow \infty$, we see that the Hamiltonian (Eq. 2.10) reduces to

$$H_{\text{trivial}} = -\frac{\mu}{2} \sum_{j=1}^L (1 + i\gamma_{j,a}\gamma_{j,b}) \quad (2.12)$$

which is an atomic insulator with the two Majoranas at each site paired so that $i\gamma_{j,a}\gamma_{j,b} = +1$.

On the other hand, when $\mu = 0$ and $\Delta = t$, the Hamiltonian reduces to

$$H_{\text{topological}} = -\frac{\Delta}{2} \sum_{j=1}^{L-1} i\gamma_{j,b}\gamma_{j+1,a}. \quad (2.13)$$

Note in particular that $\gamma_{1,a}$ and $\gamma_{L,b}$ do not appear and so commute with this Hamiltonian. They are the Majorana zero modes appearing at the ends of the finite chain and can be paired to have either $i\gamma_{1,a}\gamma_{L,b} = \pm 1$ giving rise to the two-fold ground state degeneracy. Away from this fine-tuned point, the two ground states gain an exponential splitting that scales like $e^{-L/\xi}$ with ξ the correlation length. In this setting we can find analogous boundary modes $\gamma_{L,R}$ such that [3,45]

$$[H_{\text{topological}}, \gamma_{L,R}] \sim e^{-L/\xi}. \quad (2.14)$$

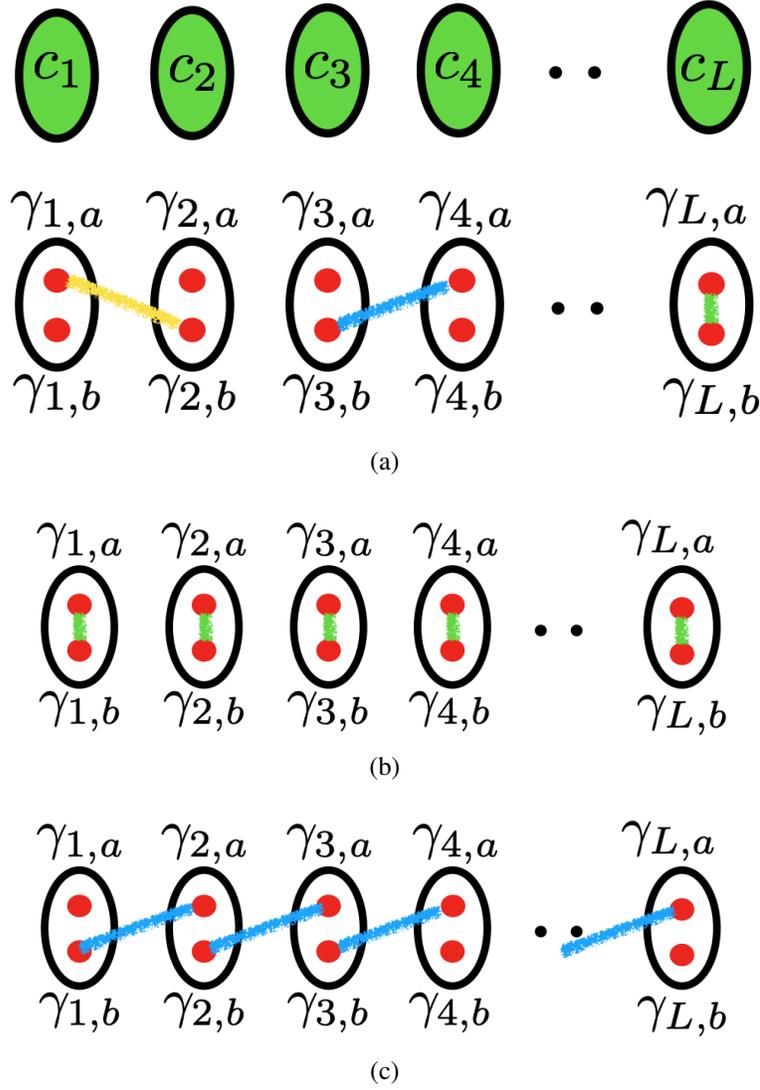


Figure 2.1: Kitaev chain with fermionic annihilation operators on each site (green ovals) labeled c_1 through c_L . These can be rewritten in terms of Majorana operators (red dots). (a) In the Majorana form of the Hamiltonian (Eq. 2.10) the chemical potential μ corresponds to an on-site Majorana pairing term (green) while $\Delta \pm t$ corresponds to different intersite Majorana pairing terms (blue/yellow). (b) The trivial phase corresponding to all Majoranas being paired in an on-site fashion with no ground-state degeneracy. (c) The topological phase where Majoranas are paired between sites, exhibiting a two-found ground state degeneracy due to the unpaired Majorana modes at the ends of the chain.

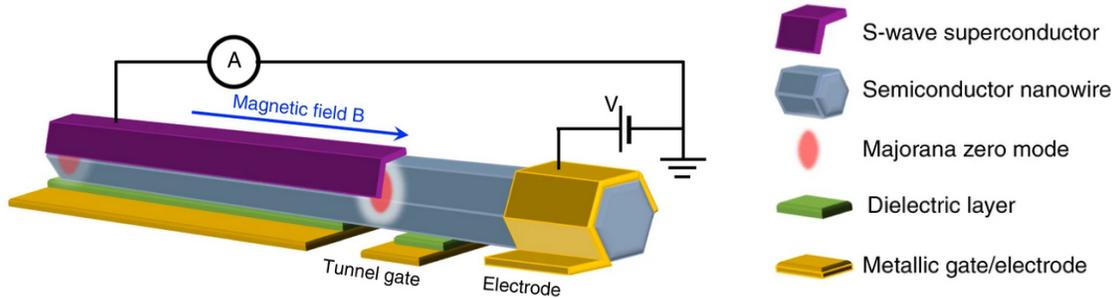


Figure 2.2: Majorana device for tunnel-gate control. Schematic experimental setup of the simplest Majorana nanowire device for tunneling spectroscopy measurement. (Source: Ref. [92]. Licensed under CC BY 4.0).

2.2 Realizing MZMs

The previous section reviewed a simple model showcasing spatially separated MZMs pinned to the ends of a $1d$ chain. However, this toy model requires p -wave pairing and spinless electrons. The first does not occur naturally and the second is rare. Remarkably it was shown that the topological phase of this system could be effectively engineered using semiconducting nanowires with strong spin-orbit coupling, s -wave superconductors, and an external magnetic field [61, 68]. Though there are other variants involving involving semiconducting and superconducting heterostructures, or topological insulators, and in greater dimensions [35, 80], we concentrate on the nanowire proposal here as that platform is the focus of this thesis. In this section we review these proposals and give a brief overview of its experimental progress.

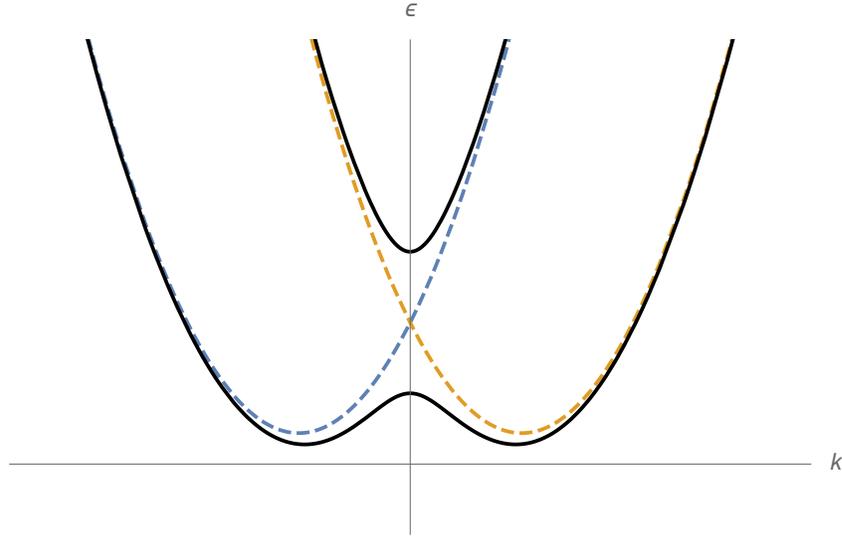


Figure 2.3: Spectrum (Eq. 2.18) for a semiconducting nanowire with spin-orbit coupling under the presence of a magnetic field. When the chemical potential lies within the induced gap, the system appears effectively spinless.

2.2.1 Theoretical proposals

The Hamiltonian for a nanowire with spin-orbit coupling proximitized to an s -wave superconductor is

$$H = H_{\text{wire}} + H_{\Delta} \quad (2.15)$$

$$H_{\text{wire}} = \int dx \psi_{\sigma}^{\dagger}(x) \left(-\frac{\partial_x^2}{2m} - \mu - i\alpha\sigma_y\partial_x + h\sigma_z \right) \psi_{\sigma}(x) \quad (2.16)$$

$$H_{\Delta} = \int dx \Delta(\psi_{\uparrow}\psi_{\downarrow} + \text{h.c.}) \quad (2.17)$$

where $\psi^{\dagger}(x)_{\sigma}$ adds an electron with spin σ to the wire, $\alpha > 0$ is the strength of the spin-orbit coupling, and h is the strength of the magnetic field aligned in the z direction.

In the $\Delta = 0$ case, the spectrum is given by

$$\epsilon_{\pm}(k) = \frac{k^2}{2m} - \mu \pm \sqrt{(\alpha k)^2 + h^2}. \quad (2.18)$$

As illustrated in Fig. 2.3, the spin-orbit coupling produces two bands shifted in momentum with the magnetic field then splitting the crossing at $k = 0$. When the chemical potential lies within this regime, the nanowire appears spinless.

When the nanowire is proximitized to an s -wave superconductor, we can understand the effect by projecting into the lower band which is valid when $\Delta \ll h$. Due to the competition from spin-orbit coupling, the magnetic field is not able to fully polarize the electrons. This allows for the s -wave pairing to still effect them leading to an effective p -wave pairing when the electrons are nearly-olarized. This analysis yields quasiparticle energies

$$\epsilon'_{\pm}(k) = \sqrt{\Delta^2 + \frac{\epsilon_+^2 + \epsilon_-^2}{2} \pm (\epsilon_+ - \epsilon_-) \sqrt{\frac{(h\Delta)^2}{(\alpha k)^2 + h^2} + \mu^2}}. \quad (2.19)$$

At $h = \sqrt{\Delta^2 + \mu^2}$ the gap closes at $\epsilon'_-(0) = 0$. Below this critical value the wire no longer appears spinless and is in a trivial phase. Above this value the wire is in a topological superconducting phase with MZMs localized at the endpoints and is smoothly connected to the phase in the previous section [3].

In summary, the MZM physics of Kitaev's toy model can be engineered with three simple phenomena that are relatively simple to achieve experimentally: a semiconducting nanowire with strong spin-orbit coupling (e.g. InAs, InSb), an s -wave superconductor (e.g. Al), and an

external magnetic field. Variations on this setup include using magnetic impurities on top of an *s*-wave superconductor [69], rotating the magnetic field along the wire [47], or using planar Josephson junctions [71]. Though the ingredients for this recipe can be simple, its gestalt requires surmounting material and engineering challenges.

2.2.2 Experimental progress

Since the original proposals there has been a flurry of experimental effort towards demonstrating the existence of MZMs in synthetic topological superconducting nanowires (see [60] for a recent review). Experiments thus far have concentrated on constructing semiconducting nanowires of InAs or InSb with a thin Al or NbTiN superconducting shell. This is because both have large spin-orbit coupling, Lande *g*-factor, and allow for high quality interfaces with the superconducting shell. Depending on the fabrication technique, the dimensions range from a wire diameter of roughly 80-100nm, lengths up a few micron long, and a shell thickness of roughly 10nm.

Zero conductance bias peaks (ZCBP) [32, 53, 83] are one of the most quoted experimental signatures for MZMs. When the nanowire is coupled to a metallic lead, single electrons can tunnel through a barrier into the wire as a Cooper pair, reflecting a hole back into the lead (Andreev reflection). At zero bias voltage, in the trivial phase, such a process is suppressed and the electron will be reflected back into the lead. In contrast, in the topological phase, at zero bias voltage, the electron can tunnel into the wire via the zero energy MZM at the boundary, leading to resonant Andreev reflection. This results in a ZCBP quantized to $2e^2/h$

for a range of parameters consistent with the nanowire being in the topological phase. Such an experiment was first carried out in [64] and subsequently reproduced by many groups [20, 23, 26, 31]. However, alternate explanations for ZCBPs with non-topological origins were soon proposed. For example, due to trivial Andreev bound states forming at the ends of the nanowire, subgaps density of states, or disorder related effects due to the unclear interface between the semiconductor and superconductor [5, 54, 55, 58, 59, 72, 84]. Much of the recent progress on eliminating these false-positives [19] have been due to improved material fabrication and sample quality.

Though not a “smoking gun”, ZCBPs in conjunction with other experiments [60, 74, 92] can provide strong evidence for MZMs. Further, it is the robustness of the signature across parameters that is the hallmark of a topological phase. Perhaps the most direct demonstration of MZMs would be fusion or braiding experiments that could directly probe their topological properties [1, 87].

2.3 MZM nanowires as qubits

Assuming MZMs can be engineered, we discuss how to perform Majorana based topological quantum computing. In this setting, quantum information is encoded via the fermion parity of some collection of an even number of MZMs. Braiding these MZMs enacts quantum gates. As mentioned previously, MZMs are good candidates for qubits because they enjoy exponentially suppressed error rates that scale as $e^{-L/\xi}$ and $e^{-\Delta/T}$ where L is the distance between the two MZMs, ξ is the correlation length, Δ is the superconducting gap, and T is the temperature.

Since fermion parity is always conserved, a single pair of MZMs will be in a definite state of either

$$|0\rangle_2 = |i\gamma_1\gamma_2 = +1\rangle, \quad (2.20)$$

$$|1\rangle_2 = |i\gamma_1\gamma_2 = -1\rangle. \quad (2.21)$$

Therefore, at least four MZMs are needed for a qubit. In this case, the 2-dimensional state space can be defined to have overall even parity and an ancillary pair, e.g. $i\gamma_1\gamma_2$, having either even or odd parity.

$$|0\rangle_4 = |i^2\gamma_1\gamma_2\gamma_3\gamma_4 = +1, i\gamma_1\gamma_2 = +1\rangle, \quad (2.22)$$

$$|1\rangle_4 = |i^2\gamma_1\gamma_2\gamma_3\gamma_4 = +1, i\gamma_1\gamma_2 = -1\rangle. \quad (2.23)$$

Similarly, the state space for six MZMs can be defined as

$$|0\rangle_6 = |i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6 = +1, i\gamma_3\gamma_4 = +1, i\gamma_1\gamma_2 = +1\rangle, \quad (2.24)$$

$$|1\rangle_6 = |i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6 = +1, i\gamma_3\gamma_4 = +1, i\gamma_1\gamma_2 = -1\rangle. \quad (2.25)$$

We call the system of six MZMs a hexon, and a system of four MZMs a tetron.

The extra MZMs in a hexon allows for braiding via measurements-only. This is the focus of this thesis and we give a high-level overview of it here with more in-depth discussion in the next chapter.

There is a unifying framework for this in the language of Majorana stabilizer codes. An $[[n, k, d]]$ Majorana stabilizer code encode k qubits into n Majoranas with distance d [18, 39]¹. The k qubits sit in a 2^k dimensional subspace within the the full $2^{n/2}$ dimensional space of the n Majoranas. We term this subspace the code space or logical space. It is defined as the simultaneous $+1$ eigenspace of a group of mutually commuting stabilizer operators, \mathcal{S} . The logical operators for the k encoded qubits are defined to be all the operators that commute with \mathcal{S} but are not themselves part of \mathcal{S} , up to equivalence by operators in \mathcal{S} . For example, hexons as described above are $[[6, 1, 2]]$ Majorana stabilizer codes. The stabilizer group is generated by the overall island parity and the parity of an ancillary pair

$$\mathcal{S}_6 = \langle i^3 \gamma_1 \gamma_2 \gamma_3 \gamma_4 \gamma_5 \gamma_6, i \gamma_3 \gamma_4 \rangle \quad (2.26)$$

and the logical operators are

$$\mathcal{L}_6 = \{i \gamma_1 \gamma_2 = i \gamma_5 \gamma_6, i \gamma_1 \gamma_6 = i \gamma_2 \gamma_5\} \quad (2.27)$$

which can be arbitrarily labeled as Pauli $Z_6 = i \gamma_1 \gamma_2$ or $X_6 = i \gamma_1 \gamma_6$. Measurements deform this code space and update the stabilizers and logical operators in a consistent way [37]. A sequence of measurements that begins and ends with the same stabilizer group will implement a map between the initial and final set of logical operators which means a quantum gate has been implemented [12–15, 70, 86].

¹A distance d code can detect fewer than d errors and correct fewer than $d/2$ errors.

More explicitly, a measurement of $i\gamma_a\gamma_b$ on a state ρ gives an outcome $s_{ab,j} = \pm 1$ corresponds to a projector

$$\Pi_{s_{ab}}^{(ab)} = \frac{\mathbb{1} + i\gamma_a\gamma_b}{2} \quad (2.28)$$

with probability

$$\Pr(s_{ab} = s) = \text{Tr} [\Pi_s^{(ab)} \rho \Pi_s^{(ab)}], \quad (2.29)$$

and transforms the state as

$$\rho \rightarrow \frac{\Pi_s^{(ab)} \rho \Pi_s^{(ab)}}{\Pr(s_{ab} = s)}. \quad (2.30)$$

For this thesis we consider the specific qubit devices and architectures proposed in Ref. [41]. These are comprised of several MZM nanowires connected via a trivial superconducting backbone. They can be connected in either a two-sided or one-sided fashion as shown in Fig. 2.4. Furthermore, a scalable architecture of qubits can be arrayed (Fig. 2.5 that allows for all 2-MZM parity measurements on a single island or 4-MZM measurements between pairs on neighboring islands. This enables measurement-only quantum computing on the full array of topological qubits. These parity measurements can be performed by measuring a parity-dependent energy shift as described in Fig. 2.6. This can be detected via energy level spectroscopy, quantum dot charge, or differential capacitance measurements [41].

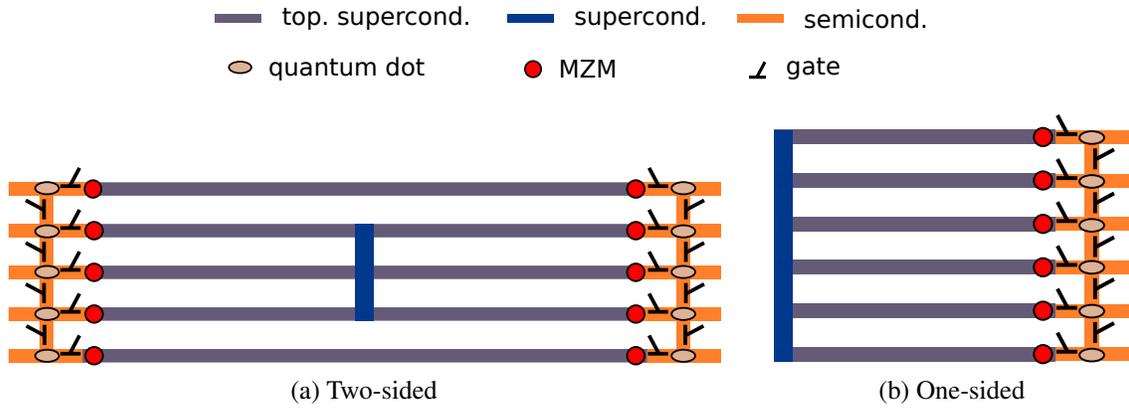


Figure 2.4: We consider two types of Majorana hexon architectures in detail: (a) the two-sided architecture and (b) the one-sided architecture. Shown here is a single qubit of each architecture with the required semiconducting quantum dots, cutter gates, and superconducting coherent links (top and bottom wire in the two-sided hexon) needed to perform all pairwise MZM measurements. The relative lengths of the vertical and horizontal dimensions are not to scale, and likely to be relatively much longer in the horizontal direction.

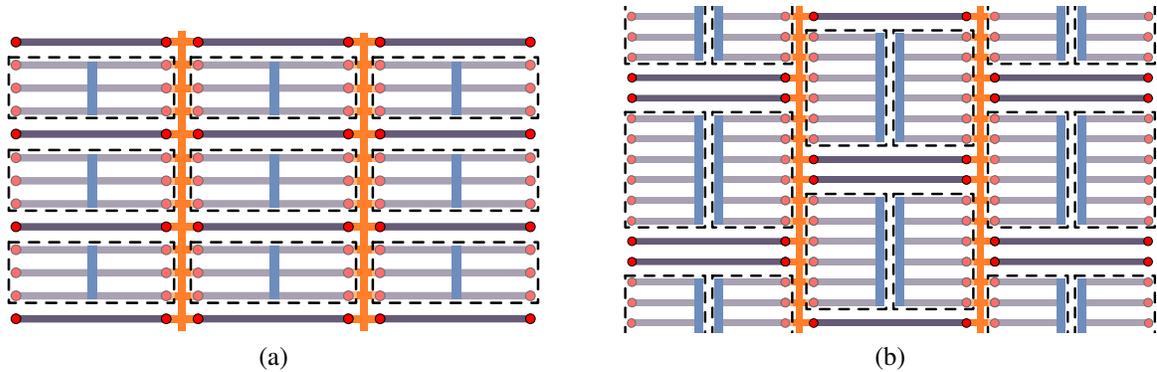


Figure 2.5: Arrays of hexons, where each hexon is shown enclosed in a dashed-line rectangle. (a) Two-sided hexons can be tiled regularly onto a rectangular lattice. (b) One-sided hexons can be tiled onto a squashed rectangular lattice, with left-facing on one sublattice and right-facing hexons on the other. In both architectures, the ability to physically implement two-qubit gates will not be equally difficult in the different directions. For example, utilizing coherent links will generally increase the difficulty.

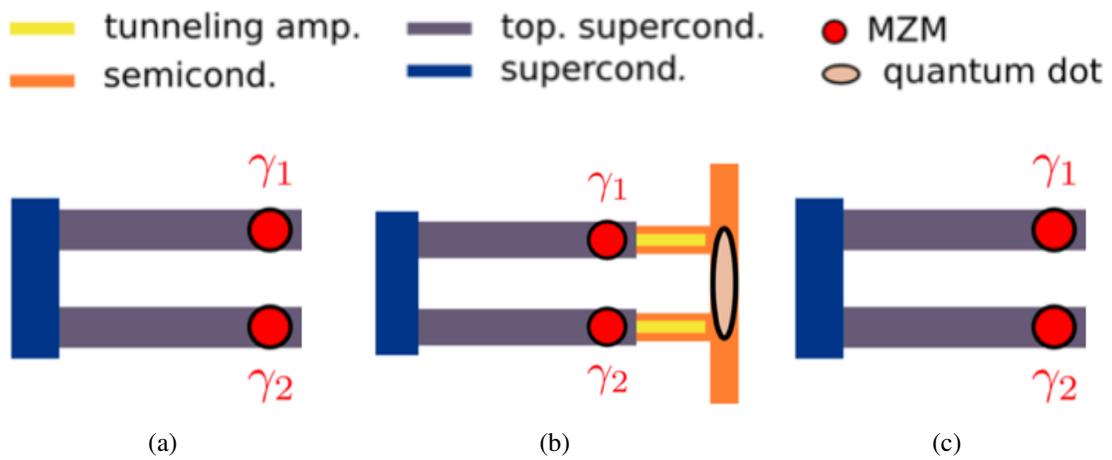


Figure 2.6: A 2-MZM parity measurement proceeds in three steps. (a) Before the measurement all couplings are turned off. (b) During the measurement couplings are turned on along the path between the two MZMs and the quantum dot. (c) After the measurement couplings are turned off again.

Chapter 3

Optimal Clifford gate synthesis for MZM

hexon qubits

Having reviewed topological MZM qubits manifested as collections of topological superconducting nanowires, we now turn our attention to the optimal compilation of measurement-only Clifford gates for the MZM hexon qubit in particular.

One of the main challenges for quantum computation is that while the number of gates required to perform a non-trivial quantum computation may be very large, decoherence and errors in realistic quantum architectures limit the number of physical gate operations that can be performed coherently. Therefore, an optimal mapping of the quantum algorithm into the physically available set of operations is of crucial importance. We examine this problem for a measurement-only topological quantum computer based on Majorana zero modes, where gates are performed through sequences of measurements. Such a scheme has been proposed as a

practical, scalable approach to process quantum information in an array of topological qubits built using Majorana zero modes. Building on previous work that has shown that multi-qubit Clifford gates can be enacted in a topologically protected fashion in such qubit networks, we discuss methods to obtain the optimal measurement sequence for a given Clifford gate under the constraints imposed by the physical architecture, such as layout and the relative difficulty of implementing different types of measurements. Our methods also provide tools for comparative analysis of different architectures and strategies, given experimental characterizations of particular aspects of the systems under consideration.

The results presented in this chapter were previously published in “Optimizing Clifford gate generation for measurement-only topological quantum computation with Majorana zero modes” by Alan Tran, Alex Bocharov, Bela Bauer, and Parsa Bonderson, *SciPostPhys.*8.6.091 (2020) [86] with minor modification. Licensed under CC BY 4.0.

3.1 Introduction

Recent experimental progress has established the existence of Majorana zero modes (MZMs) [46, 63, 78], in particular in their incarnation in hybrid semiconductor-superconductor heterostructures [60], as one of the most promising platforms for realizing topological quantum computation [65]. As the evidence for the successful experimental realization of such topological phases mounts [60], the question arises how to assemble a network of topological superconductors in a way that allows practical quantum information processing on many qubits. While several proposals have been put forward [1, 40, 41, 52], we will focus here on the measurement-

only approach of Ref. [41].

Measurement-only topological quantum computation, first proposed in Refs. [14, 15, 70, 77], appears particularly favorable in the context of MZMs since it avoids having to physically move the MZMs, which are bound to macroscopic defects (such as the ends of wires) and may be difficult to move without strongly disturbing the system. Instead, braiding transformations are effectively generated through a series of (potentially non-local) measurements on sets of MZMs involving the MZMs that encode the computational state that is to be manipulated, and another set of MZMs that serve as ancillary degrees of freedom. In the architectures proposed in Ref. [41], the required measurements are performed by coupling groups of MZMs to quantum dots, thus affecting the energy spectrum of the dot in a way that can be measured using established techniques. Importantly, the encoded quantum state as well as the operations being performed remain topologically protected, i.e. errors due to a large class of experimental imperfections are exponentially suppressed in system size and the topological gap.

The first challenge in compiling a given quantum circuit for a topological quantum computer based on MZMs is that their topologically protected operations are not by themselves computationally universal [65]: they can only produce multi-qubit Clifford gates, a subgroup of the unitary group that is efficiently simulatable on a classical computer [37]. To perform universal quantum computation, they need to be augmented by one additional non-Clifford gate. A typical choice is the so-called T -gate (or $\pi/8$ -phase gate), which can be implemented by preparing and injecting a “magic state,” which in turn can be prepared to high fidelity using distillation protocols [17]. However, this distillation is very resource-intensive and likely to be

the bottleneck of quantum computation using MZMs.¹ It should be noted, however, that surface codes – one of the leading proposals for error correction based on conventional qubits – suffer from the same problem [34]. The set of available computational gates in our envisioned architecture will thus comprise some subset of the topologically-protected Clifford gates (such as all single-qubit operations together with two-qubit operations between all adjacent qubits), augmented by the T -gate, which will not be topologically protected. When compiling a given quantum algorithm from this gate set, the primary challenge is to reduce the number of T -gates. This problem has been the focus of much attention [11, 81].

In this chapter, we focus on a second problem, which is particular to the measurement-only approach: synthesizing the topologically-protected Clifford gates from a sequence of measurements. The previously espoused strategy (see e.g. Ref. [41]) for generating Clifford gates in the measurement-only approach to topological quantum computing with MZMs was to first generate minimal-length measurement sequences for the basic (nearest-neighbor) braiding transformations for each qubit, and a measurement sequence for a two-qubit entangling gate between all pairs of qubits (or at least between all adjacent pairs of qubits), and then use the resulting gate set as the generating gate set used to synthesize any other Clifford gates. From the perspective of the fundamental operators, i.e. measurements, this strategy may be inefficient, as there may exist shorter sequences of measurements that compile to the same gate. We will describe different strategies and protocols for optimizing the generation of computational gates via measurement sequences with the physical measurements themselves as the generating set of operations. We introduce a weighting system for different measurements in a given topolog-

¹For estimates for a relevant problem, see e.g. Ref. [79].

ical quantum computing architecture that provides a more meaningful metric than number of measurements with respect to which optimization is performed. We provide a demonstration of our methods using brute-force search to find optimal measurement sequence realizations for single-qubit Clifford gates and for two-qubit controlled-Pauli gates. Our methods may also be used to provide a comparative analysis of different strategies and architectures that are being considered for implementation.

The outline of the chapter is as follows. In Sec. 3.2, we review the physical architectures for measurement-only topological quantum computing using islands of six MZMs – the so-called “hexon” qubit architectures. We discuss the fact that physically performing different measurements will have different levels of difficulty, and describe a systematic approximation of such. We also discuss the possible advantages of different encodings of the computational and ancillary degrees of freedom in the physical MZMs. In Sec. 3.3, we describe the “forced-measurement” protocol, and several strategies to improve upon it. In Sec. 3.4, we describe the Majorana-Pauli tracking method that allows us to circumvent the use of forced-measurement protocols by tracking the measurement outcomes and their effect on the computation. Tracking methods are a more efficient alternative to forced-measurements, but they may only be employed when the measurement outcomes correspond to Abelian anyons. This is always the case for MZM-based architectures, which is the main focus of this chapter. In Sec. 3.5, we discuss optimization and search strategies for measurement-only gates in the various architectures and methods that can be utilized. We provide a demonstration of our methods utilizing brute-force search to find optimizations of measurement sequences for all one-qubit and a

cillary degrees of freedom that facilitate measurement-based operations. While a qubit can also be formed from four MZMs (referred to as a tetron when on an isolated superconducting island), due to the absence of ancillary MZMs, such a qubit by itself does not permit any topologically-protected unitary gate operations. A hexon, on the other hand, allows for the full set of single-qubit Clifford gates to be implemented with topological protection. Therefore, we focus on the hexon architecture, though many of the techniques we develop here can be adapted to systems of several tetrans, where some tetrans serve as ancillary qubits.

For MZMs that emerge at the ends of nanowires, a hexon is formed by joining several Majorana nanowires via a spine made from trivial (*s*-wave) superconductor, as shown in Fig. 3.1. We consider both a two-sided hexon architecture and a one-sided hexon architecture, as shown in Figs. 3.1(a) and (b), respectively. In the two-sided architecture, three wires are joined by a spine in the middle and MZMs are present at both ends of the wire. In the one-sided architecture, six wires are joined at one of their ends and, thus, MZMs are present only at the other end. An important benefit of these architectures is that a single qubit island is galvanically isolated (except for weak coupling to dots, see below), and thus Coulomb interactions give rise to a finite charging energy E_C for the island. This helps to prevent (extrinsic) quasiparticle poisoning, as the probability for an electron to tunnel onto or off of the island from outside is exponentially suppressed in the ratio of the charging energy E_C to temperature, $\exp(-E_C/k_B T)$. Decoherence of topologically protected states due to thermally excited quasiparticles on the island is suppressed by $\exp(-\Delta/k_B T)$, where Δ is the topological gap. Degeneracy splitting due to virtual tunneling of fermions between MZMs is suppressed by $\exp(-L/\xi)$, where L is

the separation of MZMs and ξ the superconducting coherence length.

Projective measurements of the joint fermionic parity of any two MZMs (2-MZM measurements) can be carried out by enabling weak coherent single-electron tunneling between the MZMs and adjacent quantum dots, forming an interference loop. Projective measurements of the collective fermionic parity of $2N$ -MZMs may be performed similarly, though care must be taken to ensure that the interference loop always involves all $2N$ MZMs, e.g. fermions cannot pass directly between the various quantum dots involved. These couplings give rise to shifts in the energy spectrum and charge occupation of the dot that depend on the fermionic parity of the MZMs. These shifts can, in turn, be measured using established techniques developed for charge and spin qubits, such as charge sensing or quantum capacitance measurements. Importantly, the measurement is topologically protected in the sense that the operator that is being measured is known up to corrections that are exponentially small in the distance separating the MZMs through the superconducting region (nanowire and spine). However, similar to other quantum non-demolition measurements, the measurement fidelity is limited by the achievable signal-to-noise ratio and decoherence of the qubit in other channels. Additionally, the calibration of a signal's correlation to even or odd parity is a choice of convention whose effect on the final outcome of a compilation is a Pauli factor as we elucidate in Sec. 3.3.3.

While both hexon architectures considered here allow, in principle, 2-MZM measurements between any pair of MZMs, it is clear that, depending on the layout, certain measurements will be more difficult to perform than others. For example, in the two-sided architecture, some measurements are between MZMs on the same side (left or right) of the island, while others

are on opposite sides. For measurements involving MZMs that are in close proximity to each other, such as ones that are on the same side of the hexon, one can adjust the electrostatic gates in the semiconducting regions to define a single quantum dot that the MZMs being measured are coupled together. However, when the MZMs are farther separated (e.g. on opposite sides), enabling coherent single-electron tunneling between these MZMs and a common quantum dot is much more challenging, as their distance may exceed the phase coherence length of realistic semiconducting wires. In such cases, a coherent superconducting link can be used to span the distance, but this increases the complexity of the device and the required tuning necessary to perform such a measurement. In Sec. 3.2.4, we will discuss the matter of measurement difficulty in more detail and provide a model for assigning “difficulty” weights to different measurements, which will be incorporated in our gate synthesis optimization strategies.

Multiple hexons can be arranged into an array, and multi-qubit operations are performed by weakly coupling MZMs from different islands to common quantum dots. Since the coupling between MZMs and quantum dots is weak, the charging energy protection against quasi-particle poisoning remains effective during such operations. This restricts the operators that can be measured to ones that commute with the charging energy (or total parity) on each island, which are precisely the measurements involving an even number of Majorana operators on each island. We will focus on 4-MZM measurements, as measurements involving larger numbers of MZMs appear unrealistic to achieve in practice.

In the multi-hexon arrays shown in Fig. 3.2, we see that the most realistic 4-MZM measurements involving pairs of hexons give rise to rectangular lattice connectivity graphs of qubits.

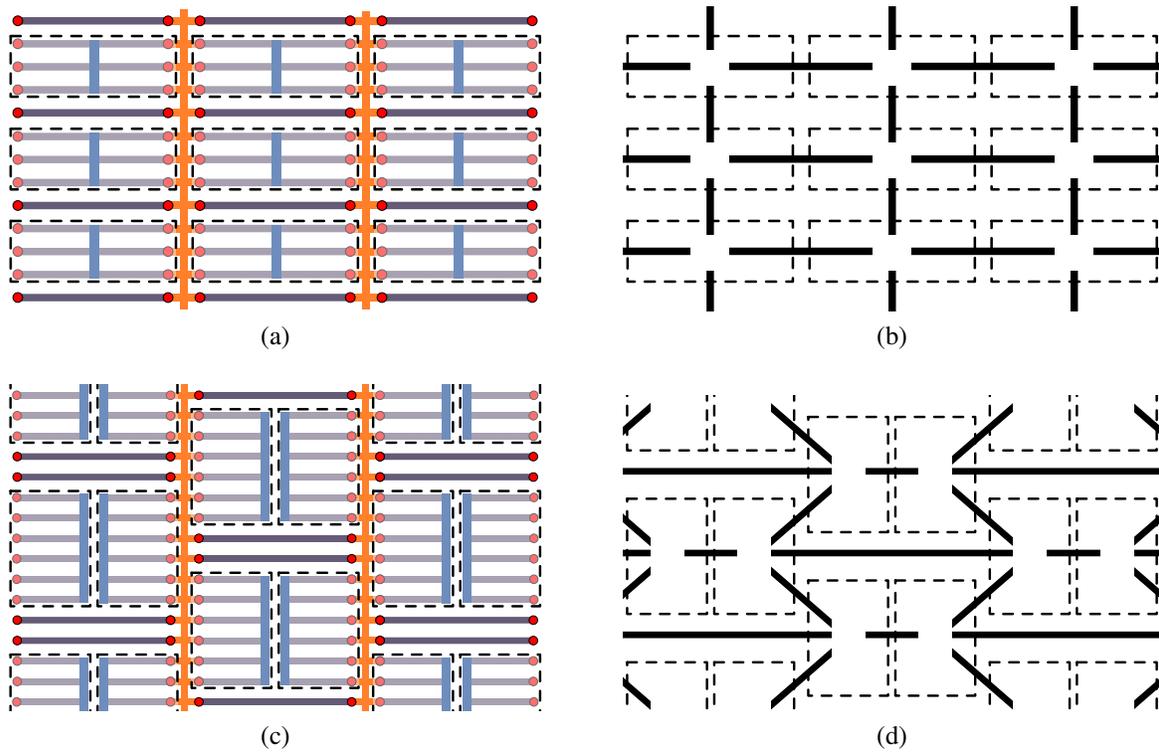


Figure 3.2: Arrays of hexons, where each hexon is shown enclosed in a dashed-line rectangle. (a) Two-sided hexons can be tiled regularly onto a rectangular lattice. (b) The connectivity of this two-sided hexon array, indicating which pairs of hexons can be acted on by joint 4-MZM measurements, is shown by solid black lines connecting the dashed rectangles. (c) One-sided hexons can be tiled onto a squashed rectangular lattice, with left-facing on one sublattice and right-facing hexons on the other. (d) The connectivity of this one-sided hexon array, indicating which pairs of hexons can be acted on by joint 4-MZM measurements, is shown by solid black lines connecting the dashed rectangles. Examples of measurements that yield the shown connectivity can be found in Figs. 3.3 and 3.4. In both architectures, the ability to physically implement two-qubit gates will not be equally difficult in the different directions. For example, utilizing coherent links will generally increase the difficulty.

Even within this rectangular lattice connectivity, certain 4-MZM measurements will be more difficult to perform than others. This can lead to better or worse connectivity between qubits in the four different directions (up, down, left, and right), and may even prevent some 4-MZM measurements from having realistic implementations. In Sec. 3.2.4, we will illustrate some of the measurements that will be utilized (see Figs. 3.3 and 3.4).

3.2.1 Single-hexon state space and operators

We label the positions of the six MZMs in a hexon $1, \dots, 6$, and associate a Majorana fermionic operator γ_j to the MZM at the j th position. These operators obey the usual fermionic anticommutation relations $\{\gamma_j, \gamma_k\} = 2\delta_{jk}$. For any ordered pair of MZMs j and k , their joint fermionic parity operator is given by $i\gamma_j\gamma_k = -i\gamma_k\gamma_j$, which has eigenvalues $p_{jk} = \pm 1$ for even and odd parity, respectively. (The conventions in this thesis will differ slightly from those of Ref. [41].) The corresponding projection operator onto the subspace with parity $s = p_{jk} = \pm 1$ is given by

$$\Pi_s^{(jk)} = \Pi_{-s}^{(kj)} = \frac{1}{2} (\mathbb{1} + s i\gamma_j\gamma_k). \quad (3.1)$$

The operator $i\gamma_j\gamma_k$ can then be expressed as

$$i\gamma_j\gamma_k = \Pi_+^{(jk)} - \Pi_-^{(jk)}, \quad (3.2)$$

where we use the shorthand \pm for ± 1 , the even-parity (vacuum) and odd-parity (fermion) channels, respectively.

In this way, we can write basis states $|p_{12}, p_{34}, p_{56}\rangle$ for a system of six MZMs in terms of the fermionic parities for some choice of how to pair them together. Due to the finite charging energy of the island, the system generically has ground states only in either the even or the odd collective fermion parity sector, which can be tuned using the gate voltage; without loss of generality, we here assume that the system is tuned to have ground states with even collective fermionic parity, i.e. $p_{12}p_{34}p_{56} = +1$, while states with odd collective parity are excited states

associated with quasiparticle poisoning. (The discussion and results for $p_{12}p_{34}p_{56} = -1$ is straightforwardly similar, but we will not focus on it in this chapter.) In this way, the low-energy state space of the hexon is 4-dimensional, with basis states

$$|+, +, +\rangle, \quad (3.3)$$

$$|-, +, -\rangle = i\gamma_2\gamma_5 |+, +, +\rangle, \quad (3.4)$$

$$|+, -, -\rangle = i\gamma_4\gamma_5 |+, +, +\rangle, \quad (3.5)$$

$$|-, -, +\rangle = i\gamma_2\gamma_3 |+, +, +\rangle. \quad (3.6)$$

Viewing this as a two-qubit system with the first qubit encoded in p_{34} and the second qubit encoded in p_{12} , the above basis states are $|0, 0\rangle, |0, 1\rangle, |1, 0\rangle, |1, 1\rangle$, in order. We can then express the MZM parity operators in terms of Pauli operators on these two qubits as

$$\begin{aligned} i\gamma_1\gamma_2 &= \mathbb{1} \otimes Z, & i\gamma_1\gamma_3 &= X \otimes Y, & i\gamma_1\gamma_4 &= -Y \otimes Y, & i\gamma_1\gamma_5 &= Z \otimes Y, \\ i\gamma_1\gamma_6 &= \mathbb{1} \otimes X, & i\gamma_2\gamma_3 &= X \otimes X, & i\gamma_2\gamma_4 &= -Y \otimes X, & i\gamma_2\gamma_5 &= Z \otimes X, \\ i\gamma_2\gamma_6 &= -\mathbb{1} \otimes Y, & i\gamma_3\gamma_4 &= Z \otimes \mathbb{1}, & i\gamma_3\gamma_5 &= Y \otimes \mathbb{1}, & i\gamma_3\gamma_6 &= X \otimes Z \\ i\gamma_4\gamma_5 &= X \otimes \mathbb{1}, & i\gamma_4\gamma_6 &= -Y \otimes Z, & i\gamma_5\gamma_6 &= Z \otimes Z, \end{aligned} \quad (3.7)$$

where the Pauli matrices are

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \quad (3.8)$$

We use the convention in which the MZMs 3 and 4 serve as the ancillary MZMs with definite joint parity, e.g. $p_{34} = +1$, and the computational qubit is encoded in p_{12} . The remaining parity is correlated with the other two as $p_{56} = p_{12}p_{34}$, so when the ancillary pair has $p_{34} = +1$, the computational basis states are

$$|0\rangle = |p_{12} = p_{56} = +\rangle, \quad |1\rangle = |p_{12} = p_{56} = -\rangle, \quad (3.9)$$

and when $p_{34} = -1$, the computational basis states are

$$|0\rangle = |p_{12} = -p_{56} = +\rangle, \quad |1\rangle = |p_{12} = -p_{56} = -\rangle. \quad (3.10)$$

Another way to view this is that a hexon is a Majorana stabilizer code which encodes a single logical qubit in six MZMs [18, 39]. In this language, logical qubits are defined to be in the simultaneous $+1$ eigenspace of a group of operators, called the stabilizer group. The logical gates which act on this space are operators which commute with the stabilizer group but are not themselves stabilizers. For the case of a hexon, the stabilizer group is generated by the total parity of the island $i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6$ and the parity of the ancillary pair $i\gamma_3\gamma_4$. The logical Pauli operators are taken to be $\bar{Z} = i\gamma_1\gamma_2$ and $\bar{X} = i\gamma_1\gamma_6$. We will initially focus on the case where we require $p_{34} = +1$ for the (initial and final) computational basis, but will allow the ancillary qubit to have either parity in Sec. 3.4.

We will often make use of a diagrammatic calculus, which allows us to perform algebra in the topological state space by manipulating diagrams, see e.g. Refs [16, 44, 75]. In this

diagrammatic formalism, isotopy invariance allows us to freely stretch or slide around strands so long as the topology of diagrams remains fixed, i.e. open endpoints of lines are held fixed and trivalent junction do not pass each other when slid along lines. Additional rules for reconnection and braiding of diagrams is incorporated by the so-called F -symbols and R -symbols.

In the diagrammatic formalism, the projectors can be represented as²

$$\Pi_+^{(jk)} = \text{---}, \quad \Pi_-^{(jk)} = \text{---}, \quad \Pi_s^{(jk)} = \text{---} \quad (3.11)$$

where $+$ (vacuum) is diagrammatically represented as no line, $-$ (fermion) is represented by a wavy red line, and an unspecified fusion channel $s = \pm 1$ is represented by a magenta line.

The $p_{34} = +1$ computational qubit basis states of hexons are diagrammatically represented as

$$\left| \frac{1-a}{2} \right\rangle = |a, +, a\rangle = \begin{array}{c} \gamma_1 \ \gamma_2 \ \gamma_3 \ \gamma_4 \ \gamma_5 \ \gamma_6 \\ \text{---} \text{---} \text{---} \\ \text{---} \\ a \end{array} \quad (3.12)$$

where $a = \pm 1$. A general computational qubit state $|\Psi\rangle$ will be denoted as

$$|\Psi\rangle := \begin{array}{c} \gamma_1 \ \gamma_2 \ \gamma_3 \ \gamma_4 \ \gamma_5 \ \gamma_6 \\ \text{---} \text{---} \text{---} \\ \text{---} \\ |\Psi\rangle \end{array} := \sum_{a=+1,-1} \Psi_a \begin{array}{c} \gamma_1 \ \gamma_2 \ \gamma_3 \ \gamma_4 \ \gamma_5 \ \gamma_6 \\ \text{---} \text{---} \text{---} \\ \text{---} \\ a \end{array} . \quad (3.13)$$

Operator multiplication is given by stacking diagrams. The identity operator acting on two

²In this thesis, we use the diagrammatic normalizations such that a closed loop of either fermion line or MZM line evaluates to 1. Consequently, straightening out bends in the MZM lines will yield nontrivial constant factors, but these will always result in overall constants that can be neglected in the context where they occur in this thesis.

MZMs is written as

$$\mathbb{1}_{jk} = \left| \quad \right| = \left(\quad \right) + \left(\quad \right), \quad (3.14)$$

so the lines are just extended when identity is applied.

The fermionic parity operator $i\gamma_j\gamma_k$ is diagrammatically represented as a fermion line connecting strands j and k ; it can also be written as an antisymmetric combination of its two projectors (cf. Eq. (3.2))

$$i\gamma_j\gamma_k = \left| \quad \right| = \left(\quad \right) - \left(\quad \right). \quad (3.15)$$

Since $\gamma_j^2 = \mathbb{1}$, a fermion line connecting a single strand to itself with no additional fermion lines connected in between can be freely removed

$$\left| \quad \right| = \left| \quad \right|. \quad (3.16)$$

From $(i\gamma_j\gamma_k)(i\gamma_k\gamma_l) = -(i\gamma_k\gamma_l)(i\gamma_j\gamma_k) \propto i\gamma_j\gamma_l$ we see that sliding endpoints of fermion lines past one another along a MZM line incurs a minus sign, and also that fermion lines compose

$$\left| \quad \right| \left| \quad \right| = - \left| \quad \right| \left| \quad \right| \propto \left| \quad \right|. \quad (3.17)$$

$(i\gamma_j\gamma_k)^2 = \mathbb{1}$ is expressed diagrammatically as

$$\left| \quad \right| \left| \quad \right| = \left| \quad \right| \left| \quad \right|. \quad (3.18)$$

Lastly, we pick up a phase of i when we flip the side of a MZM line to which a fermion attaches

$$\begin{array}{|c} \text{wavy} \\ \hline \end{array} = i \begin{array}{|c} \text{wavy} \\ \hline \end{array}. \quad (3.19)$$

3.2.2 Single-qubit gates through measurements

Single-qubit Clifford gates can be implemented on the encoded qubit in a topologically protected manner via a “measurement-only” braiding protocol [14]. The braiding transformations are represented in term of Majorana operators as

$$R^{(jk)} = \frac{1}{\sqrt{2}}(\mathbb{1} + \gamma_j \gamma_k) = \begin{array}{cc} j & k \\ \diagdown & \diagup \\ & \\ \diagup & \diagdown \end{array} \quad (3.20)$$

for the counterclockwise exchange of MZMs at positions j and k . Using the measurement-only protocol, the single-qubit braiding gates are realized by sequentially measuring the joint fermionic parity of MZM pairs, subject to the following constraints: (1) the first measurement must involve exactly one MZM from the ancillary pair, (2) subsequent measurements must involve exactly one MZM from the preceding measured pair, and (3) the final measurement must involve the (original) ancillary pair and the measurement outcome must equal the ancillary pair’s initial joint parity, which (for now) is taken to be $p_{34} = +$. As such, sequential measurements will correspond to anticommuting parity operators, i.e. measurements of pairs (jk) and (lm) are allowed to follow one another if and only if $i\gamma_j\gamma_k i\gamma_l\gamma_m = -i\gamma_l\gamma_m i\gamma_j\gamma_k$. These conditions ensure that the measurements do not read-out any information about the state of the encoded computational qubit. Another way of viewing this process is that one is performing a

sequence of anyonic teleportations [14], where, in each step, the encoded qubit state is being re-encoded in a different set of MZMs and the measured pair of MZMs temporarily becomes the ancillary pair. In this view, the sequence of teleportations defines the braiding “path” and enacts the corresponding braiding transformation on the encoded state.

In order to ensure that the final measurement outcome of the ancillary pair is the same as its initial value, and to deterministically control which computational gate is produced by such a process involving measurements, one may use a “forced-measurement” protocol for each measurement step [14]. This is a repeat-until-success procedure involving the ancillary degrees of freedom that allows one to end with a desired measurement outcome. In other words, a forced-measurement of $i\gamma_j\gamma_k$ onto a specific fusion channel s effectively acts on the state space as the projector $\Pi_s^{(jk)}$. In this protocol, if we get an undesired result, we can perform a different measurement that effectively “resets” the state of the system to allow for the target measurement to be performed again with a new probability of obtaining the desired outcome (see Sec. 3.3 for more details). As the measurement outcomes involved in this process should (ideally) have equal probability of both outcomes, the probability of needing more than some number of attempts to succeed is exponentially suppressed in the number of attempts. The average number of attempts needed to achieve the desired outcome is 2. Thus, while probabilistically determined, the number of measurements needed for a given forced measurement can be treated as a constant, on average.

In Sec. 3.3, we will describe the forced-measurement procedure in more detail, as well as a refinement of the strategy of applying forced measurements at each measurement step.

However, since we have this repeat-until-success method of effectively producing a desired measurement outcome (via a sequence of physical measurements) at each step, we will initially discuss the measurement-only gate synthesis in terms of the projectors corresponding to the desired measurement outcomes, rather than the full sequence of physical measurements involved.

A sequence of projectors on a hexon subject to the above constraints generates a single-qubit Clifford gate acting on the encoded computational qubit. For example,

$$\Pi_+^{(34)}\Pi_+^{(13)}\Pi_+^{(23)}\Pi_+^{(34)} \propto \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} = \Pi_+^{(34)} \otimes S, \quad (3.21)$$

where S is the $\pi/4$ -phase gate. (Here, we have included an initial $\Pi_+^{(34)}$, which is redundant when assuming the ancillary MZMs are properly initialized, but which is convenient for evaluating the operator the sequence will effect.) This relation can be checked algebraically in terms of the Majorana operators by expanding each projector. Similarly, the gate $B = S^\dagger H S^\dagger$ (where H is the Hadamard gate) acting on the qubit can be produced from the projector sequence

$$\Pi_+^{(34)}\Pi_-^{(35)}\Pi_-^{(23)}\Pi_+^{(34)} \propto \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \otimes \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -i \\ -i & 1 \end{bmatrix} = \Pi_+^{(34)} \otimes B. \quad (3.22)$$

We note that the gate set $\{S, B\}$ generates all single-qubit Clifford gates C_1 .

The Clifford gates can be directly related to the braiding transformations, for example $S = R^{(12)}$ and $B = R^{(25)}$. These relations can be made more visually transparent by viewing

the projector sequences applied to the hexon in the diagrammatic representation

$$\Pi_+^{(34)} \Pi_+^{(13)} \Pi_+^{(23)} |\Psi\rangle = \text{Diagram} \quad (3.23)$$

and

$$\Pi_+^{(34)} \Pi_+^{(35)} \Pi_+^{(23)} |\Psi\rangle = \text{Diagram} \quad (3.24)$$

Here, the ancillary MZM pair is explicitly initialized to $p_{34} = +$, so the redundant initial projector $\Pi_+^{(34)}$ is not included.

3.2.3 Multi-hexon operations

The Hilbert space of two hexon units is the tensor product of that of the two individual hexons. We label the first hexon's MZMs $1, \dots, 6$ and the second hexon's MZMs by $1', \dots, 6'$, and ascribe Majorana operators to them, accordingly. In this way, we have the two-hexon qubit basis states

$$|a, b\rangle = |a\rangle \otimes |b\rangle = |p_{12} = a, p_{34} = +, p_{56} = a\rangle \otimes |p_{1'2'} = b, p_{3'4'} = +, p_{5'6'} = b\rangle. \quad (3.25)$$

In order to generate entangling two-qubit gates, we need to include measurements of the collective fermionic parity of four MZMs, two (labeled j and k) from the first hexon and two (labeled l' and m') from the second hexon. We write the 4-MZM joint parity projector as

$$\Pi_s^{(jk;l'm')} = \frac{1}{2} (\mathbb{1} - s \gamma_j \gamma_k \gamma_{l'} \gamma_{m'}) = \Pi_+^{(jk)} \Pi_s^{(l'm')} + \Pi_-^{(jk)} \Pi_{-s}^{(l'm')}, \quad (3.26)$$

where we use semicolons to separate labels corresponding to different hexons. We re-emphasize that the order of MZM labels matters, since the Majorana operators anti-commute. We also emphasize that these projectors will not change the total fermionic parity of either hexon island. Diagrammatically, these projectors can be represented as

$$\Pi_+^{(jk;l'm')} = \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad (3.27a)$$

$$\Pi_-^{(jk;l'm')} = \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} + \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \end{array} \quad (3.27b)$$

where the first projector in each term acts on the MZMs at positions j and k of the first hexon, while the second projector acts on MZMs l' and m' of the second hexon.

Two-qubit gates can similarly be generated from sequences of 2-MZM and 4-MZM projection operators. Since the particle number on each island should be preserved, all measurement operators need to involve an even number of Majorana operators on each island. In the case of two-qubit operations, the measurement sequences must also begin and end with both ancillary pairs in their initialized state. In other words, the sequences of projectors begin and end with

$\Pi_+^{(\text{anc})} = \Pi_+^{(34)} \Pi_+^{(3'4')}$. However, if either $\Pi_+^{(34)}$ or $\Pi_+^{(3'4')}$ commutes with every term in the measurement sequence, then the final measurement of the ancillary pairs does not need to involve the corresponding pair of MZMs, since they will already be in the desired final ancillary state.

More generally, a system of N hexons encodes N computational qubits and N ancillary qubits. When we specify an ordered set \mathcal{M} of $2r$ MZMs, we define the corresponding fermionic parity and projection operator to be

$$\Gamma_{\mathcal{M}} = i^r \prod_{a \in \mathcal{M}} \gamma_a, \quad (3.28)$$

$$\Pi_s^{(\mathcal{M})} = \frac{1}{2} (\mathbb{1} + s \Gamma_{\mathcal{M}}), \quad (3.29)$$

where the order of Majorana operators in the product respects the order of the set. Multi-hexon measurements only ever need to involve two MZMs from each hexon involved, since the overall fermionic parity of each hexon island is fixed, giving the relation $i^3 \gamma_1 \gamma_2 \gamma_3 \gamma_4 \gamma_5 \gamma_6 = +1$ on the ground state space. This allows the product of four of the MZMs from a hexon to be replaced by the product of the other two (with appropriate phase factors).

The general condition for a sequence of fermionic parity measurements involving N hexons to compile to a unitary gate acting on the computational qubits is that the measurements (which range from 2-MZM to $2N$ -MZM measurements) should not read information out of the computational state, i.e. the corresponding projectors should not reduce the rank of any encoded computational state. Any subsequence of the projector sequence must therefore *not* multiply out to an operator of rank less than 2^N . Additionally, the final measurement in a

sequence must project the ancillary MZMs into the initialized state.

In order to translate this general condition into more explicit constraints on the allowed measurements, it is helpful for the case of MZMs to utilize the stabilizer formalism, as may be adapted from Ref. [37]. (This, of course, also works for the single-qubit measurement-only gates, but is overkill for that case.) In this picture, we view the system of N hexons as a Majorana stabilizer code that encodes N logical qubits in $6N$ MZMs. Each hexon island has a fixed total parity throughout the measurement-only sequence, which translates into the fixed stabilizer $i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6$. Each hexon island initially has an additional ancillary qubit stabilizer corresponding to the parity operator $i\gamma_3\gamma_4$. Thus, these are the generators of the initial stabilizer group of a hexon $S_0 = \langle i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6, i\gamma_3\gamma_4 \rangle$, which is isomorphic to $\mathbb{Z}_2 \times \mathbb{Z}_2$. The corresponding logical Pauli operators (acting on the logical qubit) for a hexon island are $\bar{X} = [i\gamma_1\gamma_6]$, $\bar{Y} = [-i\gamma_2\gamma_6]$, and $\bar{Z} = [i\gamma_1\gamma_2]$, where the equivalence classes contain all parity operators related by multiplication by a stabilizer, that is $[\Gamma_{\mathcal{M}}] = \{\Gamma_{\mathcal{N}}, \mid \exists Q \in S : \Gamma_{\mathcal{N}} = Q\Gamma_{\mathcal{M}}\}$. The initial stabilizer group and operators for the N hexon system is obtained by taking products of each hexon's stabilizer group and operators, i.e. $S = \prod_{\alpha=1}^N S_0^{(\alpha)} \cong \mathbb{Z}_2^{2N}$, so there are 4^N stabilizers.

In order for a measurement to neither act trivially on nor read information out of the encoded logical state, the operator being measured must not commute with all of the stabilizers. Since the measured parity operator $\Gamma_{\mathcal{M}}$ and the stabilizers are all products of Majorana operators, this means $\Gamma_{\mathcal{M}}$ must commute with exactly half of the stabilizers and anticommute with the other half. After performing such a measurement, the stabilizer group and logical operators

must be updated. The updated stabilizer group is obtained by removing all of the stabilizers that anticommute with $\Gamma_{\mathcal{M}}$, and then adding $\Gamma_{\mathcal{M}}$ as a new stabilizer and using them to generate the new stabilizer group. We can write this in terms of the following steps [37]:

1. Write $S = S_C \cup S_A$, where S_C is the subgroup of stabilizers that commute with $\Gamma_{\mathcal{M}}$ and S_A is the set of stabilizers that anticommute with $\Gamma_{\mathcal{M}}$.
2. Update the stabilizer group to: $S' = S_C \times \langle \Gamma_{\mathcal{M}} \rangle$.
3. Write each logical Pauli operator \bar{P} as $\bar{P} = \bar{P}_C \cup \bar{P}_A$, where \bar{P}_C is the subset of parity operators in the equivalence class that commute with $\Gamma_{\mathcal{M}}$ and \bar{P}_A is the subset of parity operators in the equivalence class that anticommute with $\Gamma_{\mathcal{M}}$.
4. Update each logical Pauli operator to: $\bar{P}' = \bar{P}_C \cup \bar{P}_C \Gamma_{\mathcal{M}} = [P_C]'$, for any $P_C \in \bar{P}_C$, where $[\cdot]'$ is the equivalence class under multiplication by the updated stabilizer S' .

In this way, each step in the measurement-only sequence may be viewed as a deformation of the Majorana stabilizer code (updating the stabilizer group and logical operators) of the N hexon system [12].

For computational purposes, it is typically more convenient to work with a minimal set of generators of the stabilizer group and a single representative of the logical operators. Let J be a minimal set of generators of the stabilizer, i.e. $\langle J \rangle = S$ and $|J| = 2N$. Let $P \in \bar{P}$ be a representative element of the logical Pauli operator. These objects are updated after measuring $\Gamma_{\mathcal{M}}$ according to the following steps:

1. Identify all elements $A_1, \dots, A_n \in J$ that anticommute with $\Gamma_{\mathcal{M}}$.

2. Update the generating set the stabilizer group to: $J' = J \cup \{\Gamma_{\mathcal{M}}, A_1 A_2, \dots, A_1 A_n\} \setminus \{A_1, \dots, A_n\}$.
3. Update the representative element of each logical Pauli operator to: $P' = P$ if P commutes with $\Gamma_{\mathcal{M}}$, or to $P' = A_1 P$ if P anticommutes with $\Gamma_{\mathcal{M}}$.

It should be clear that $S' = \langle J' \rangle$, $|J'| = |J|$, and $P' \in \bar{P}' = [P']'$. We emphasize that the labeling order of the elements A_1, \dots, A_n is arbitrary and the choice of A_1 is not special.

For a measurement-only sequence applied to a single hexon, each measurement step may select from 8 possible pairs of MZMs to measure. For two hexons, there are 16 possible 2-MZM measurements and 176 4-MZM measurements that are allowed to select from at each step. If a sequence of measurements ends with the final stabilizer group equal to the initial stabilizer group, then the sequence yields a logical gate acting on the original logical state space, which is determined by the transformation of the logical Pauli operators.

We will use \mathcal{G} to denote a specific sequence of projectors, corresponding to a specific sequence of measurements and outcomes (or forced measurements), used to generate a gate with a measurement-only protocol, as

$$\mathcal{G} = \Pi_+^{(\text{anc})} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_+^{(\text{anc})}, \quad (3.30)$$

where the labels \mathcal{M}_μ are used to denote an allowed ordered set of (an even number of) MZMs whose joint fermionic parity is being projected onto corresponding parity s_μ at the μ th projector in the sequence. The ancillary projector gives the projection of all involved hexons' ancillary

pair of MZMs into the $+$ state, that is

$$\mathbf{\Pi}_+^{(\text{anc})} = \mathbf{\Pi}_+^{(34)} \otimes \dots \otimes \mathbf{\Pi}_+^{(3'\dots'4'\dots')}. \quad (3.31)$$

The resulting unitary gate acting on the encoded computational state space will be written as G , where

$$\mathcal{G} \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes G. \quad (3.32)$$

We emphasize that the relation between projection operator sequences and computational gates is many-to-one.

An example of a two-qubit entangling gate generated from 2-MZM and 4-MZM projectors is

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.33)$$

which can be obtained from the sequence of projectors, as in Ref. [41]:

$$\mathbf{\Pi}_+^{(34)} \mathbf{\Pi}_{s_3}^{(35)} \mathbf{\Pi}_{s_2}^{(56;1'2')} \mathbf{\Pi}_{s_1}^{(45)} \mathbf{\Pi}_+^{(\text{anc})} \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes W^{-s_1 s_2 s_3}, \quad (3.34)$$

where either W ($-s_1 s_2 s_3 = +1$) or its inverse ($-s_1 s_2 s_3 = -1$) is obtained, depending on the measurement outcomes. The first term in the tensor product acts on the ancillary qubits and the second acts on the computational qubits. Note that $\mathbf{\Pi}_+^{(3'4')}$ commutes with every term above, so

the final projector only needs to act on MZMs 3 and 4. For example, diagrammatically,

$$\begin{aligned}
 W |\Psi_1, \Psi_2\rangle &\propto \Pi_+^{(34)} \Pi_-^{(35)} \Pi_+^{(56;1'2')} \Pi_+^{(45)} |\Psi_1, \Psi_2\rangle \\
 &= \sum_{s_2} \left(\text{Diagram 1} \right) \left(\text{Diagram 2} \right). \tag{3.35}
 \end{aligned}$$

To see this relation, we first use the fact that $\Pi_+^{(3'4')}$ commutes with every other projector in the sequence, and hence can be factored out and ignored for this calculation. We expand

$$\begin{aligned}
 \Pi_+^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56;1'2')} \Pi_{s_1}^{(45)} \Pi_+^{(34)} &= \Pi_+^{(34)} \frac{\mathbb{1} + i s_3 \gamma_3 \gamma_5}{2} \frac{\mathbb{1} - s_2 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'}}{2} \frac{\mathbb{1} + i s_1 \gamma_4 \gamma_5}{2} \Pi_+^{(34)} \\
 &= 2^{-3} \Pi_+^{(34)} (\mathbb{1} + i s_1 \gamma_4 \gamma_5 + i s_3 \gamma_3 \gamma_5 - s_2 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'} + s_1 s_3 \gamma_3 \gamma_4 \\
 &\quad + i s_1 s_2 \gamma_4 \gamma_6 \gamma_{1'} \gamma_{2'} - i s_2 s_3 \gamma_3 \gamma_6 \gamma_{1'} \gamma_{2'} + s_1 s_2 s_3 \gamma_3 \gamma_4 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'}) \Pi_+^{(34)} \\
 &= 2^{-3} \Pi_+^{(34)} (\mathbb{1} - i s_1 s_3 \mathbb{1} - s_2 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'} - i s_1 s_2 s_3 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'}) \Pi_+^{(34)} \\
 &= 2^{-5/2} \Pi_+^{(34)} e^{-i \frac{\pi}{4} s_1 s_3} (\mathbb{1} - i s_1 s_2 s_3 \gamma_5 \gamma_6 \gamma_{1'} \gamma_{2'}) \Pi_+^{(34)} \\
 &= 2^{-2} e^{i \frac{\pi}{4} s_1 s_3 (s_2 - 1)} \Pi_+^{(34)} \otimes W^{-s_1 s_2 s_3}. \tag{3.36}
 \end{aligned}$$

Here, we used the facts that $\Pi_+^{(34)} \gamma_3 \gamma_j = \gamma_3 \gamma_j \Pi_-^{(34)}$ and $\Pi_+^{(34)} \gamma_4 \gamma_j = \gamma_4 \gamma_j \Pi_-^{(34)}$ for $j \neq 3, 4$, that $\Pi_+^{(34)}$ projects onto the subspace with $\gamma_3 \gamma_4 = -i$, and $(i \gamma_5 \gamma_6)(i \gamma_{1'} \gamma_{2'}) = Z \otimes Z \otimes \mathbb{1} \otimes Z$.

We note that, using the methods of this thesis, we can find more efficient projector sequences

for this gate, such as

$$\Pi_+^{(34)} \Pi_{s_2}^{(35)} \Pi_{s_1}^{(36;1'2')} \Pi_+^{(\text{anc})} \propto \Pi_+^{(\text{anc})} \otimes W^{s_1 s_2}. \quad (3.37)$$

The gate set $\{S, B, W\}$, where the single-qubit gates can act on any qubit and the two-qubit gates can act on any (nearest-neighbor) pair of qubits, generates all N -qubit Clifford gates C_N . For instance, the controlled- Z gate can be obtained as $C(Z) = (S^\dagger \otimes S^\dagger)W$, and $C(X)$ can be obtained from $C(Z)$ by conjugating the target qubit by $H = SBS$. It is well-known that $\{S, H, C(Z)\}$ generates the entire set of N -qubit Clifford gates for any N , so $\{S, B, W\}$ does as well.

3.2.4 Not all measurements are created equal

Experimentally, certain measurements will be more difficult to perform than others. For example, measurements on nearby MZMs can be expected to be less faulty and require less resources than measurements involving distant MZMs. We can account for this by using a cost function that assigns “difficulty” weights to the specific measurement operations that are utilized throughout a computation. In this way, a sequence of measurements, used e.g. to generate computational gates, will have a corresponding difficulty weight.

We use the ambiguous term “difficulty” primarily as a stand-in for error-rate, but also to encapsulate resource requirements and other complexities, until a more accurate picture of these matters is obtained through physical experiments. We will provide extremely rough, but systematic and physically motivated estimates of the difficulty weights for the measurements,

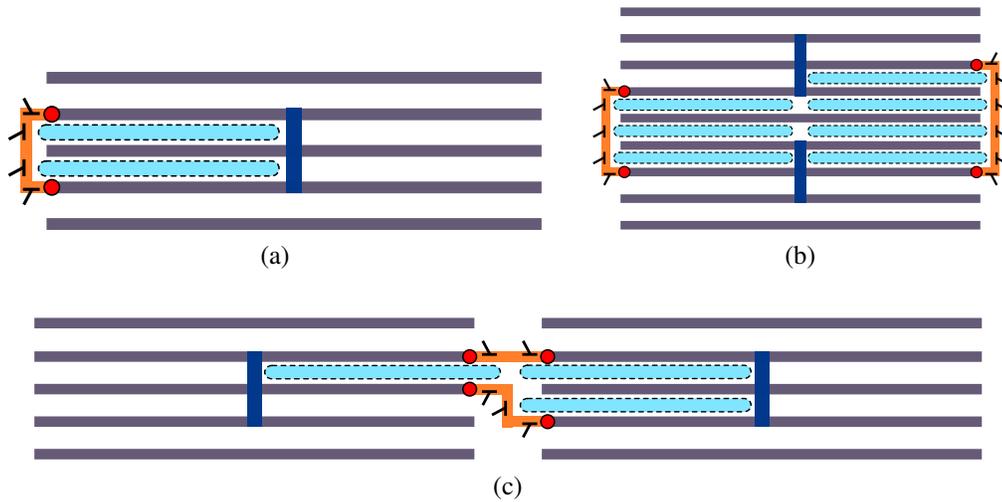


Figure 3.3: Various fermionic parity measurement configurations for the two-sided hexon architecture. (a) A 2-MZM measurement with $n_c = 2$ vertical cutter gates opened, $n_a = 2$ units of area enclosed by the interference loop, and $n_t = 2$ tunneling junctions to MZMs. (b) A 4-MZM measurement on vertically displaced hexons, with $n_c = 7$, $n_a = 7$, and $n_t = 4$. (c) A 4-MZM measurement on horizontally displaced hexons, with $n_c = 1$, $n_a = 3$, and $n_t = 4$.

to provide quantitative demonstrations of our methodology.

Cutter gates — In the hexon architecture, measurements are performed by coupling different MZMs to quantum dots, which effectively form interference loops delineated by the paths connecting the MZMs through the hexon and the paths connecting MZMs through the dots, as shown in Figs. 3.3 and 3.4. To select the interference paths, electrostatic depletion gates are tuned which effectively connect or disconnect different parts of the semiconductor, and define quantum dots in it. We will refer to these gates as *cutter gates*. These cutter gates affect the measurement difficulty in two ways: (i) It appears likely that disorder in the region where the cutters are deposited will locally decrease the phase coherence of the semiconductor, and thus reduce the visibility of the measurement. (ii) The overall length of the semiconducting path will affect phase coherence, and its volume affects properties of the dot such as its charging

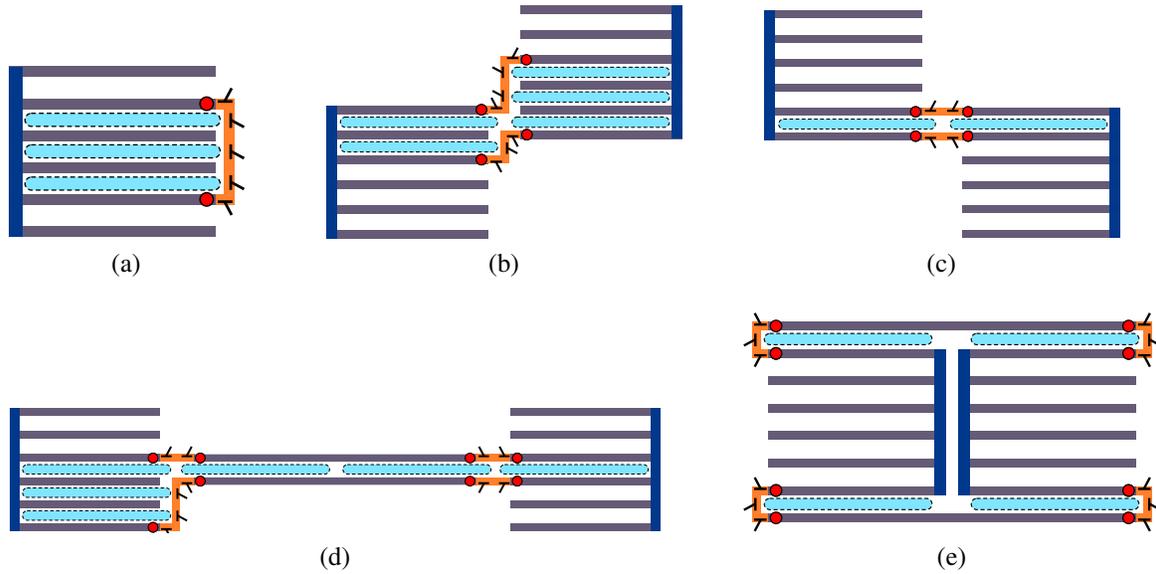


Figure 3.4: Various fermionic parity measurement configurations for the one-sided hexon architecture. (a) A 2-MZM measurement with $n_c = 3$ vertical cutter gates opened, $n_a = 3$ units of area enclosed by the interference loop, and $n_t = 2$ tunneling junctions to MZMs. (b) A 4-MZM measurement in the upward direction, with $n_c = 3$, $n_a = 5$, and $n_t = 4$. (c) A 4-MZM measurement in the downward direction, with $n_c = 0$, $n_a = 2$, and $n_t = 4$. (d) A 4-MZM measurement in the rightward direction, with $n_c = 2$, $n_a = 6$, and $n_t = 8$. (e) A 4-MZM measurement in the leftward direction, with $n_c = 4$, $n_a = 4$, and $n_t = 8$.

energy and level spacing. In general, the measurement will be easier for smaller dots. We use the number of *vertical* cutter gates involved in a measurement as simple placeholder for the length of the semiconducting region.

Tunnel junctions — Wherever a MZM couples to the semiconductor, the coupling must be carefully tuned by a depletion gate forming a tunnel junction. In contrast to cutter gates between semiconducting regions, which will generally be either fully opened or closed, it is important to tune the coupling to MZMs carefully such that its ratio with the charging energy E_C is in a favorable regime where the effect on the quantum dot is quickly and reliably measurable, while not suppressing the charging energy of the dot and increasing the probability

of quasiparticle poisoning. Realistically, the visibility of the signal will be reduced with each tunnel junction, and noise in the tunnel gate can affect the measurement signal. Furthermore, as part of the measurement protocol, this coupling must be tuned from 0 to its target value on a time-scale that is at the same time fast compared to the measurement time and slow enough to avoid inducing diabatic corrections; this must be achieved even in the presence of non-monotonic pinch-off curves due to bound states near the gate. Finally, note that MZMs that are far away from each other can be connected using superconducting coherent links, themselves made from topological superconductors and requiring additional tunnel junctions. The number of tunneling junctions is equal to the number of MZMs involved in a measurement, which may be larger than the number of MZMs being measured if using superconducting coherent links. It is also equal to the number of horizontal cutter gates.

Flux noise — The energy shift of the quantum dot depends on the magnetic flux enclosed in the loop. Noise in the enclosed flux, either from noise in the background field or any flux lines used to tune local fields, will make the measurement more challenging. As the flux noise will depend on the enclosed area, we account for this area, assuming that the geometries are such that the relevant areas for such errors are approximately partitioned into integer multiples of some unit area.

Number of islands — The difficulty of a measurement will also depend on the number N of hexons involved. This is because the measurement visibility will be significantly affected by how well the system can be tuned to the resonant tunneling point, and also because the operations utilized in a measurement can cause errors that transfer fermions between the different

hexons.

Given the factors described above, we define the difficulty weight of a fermionic parity measurement of $2N$ -MZMs $\mathcal{M} = (jk; l'm'; \dots)$ involving N hexons to be

$$w(\mathcal{M}) = w_c^{n_c(\mathcal{M})} w_t^{n_t(\mathcal{M})} w_a^{n_a(\mathcal{M})} f(N), \quad (3.38)$$

where n_c is the number of vertical cutter gates that are opened for the measurement, n_t is the number of tunneling junctions involved in the measurement, which is equal to the number of MZMs involved in the measurement (including those of coherent links), and n_a is the (integer) amount of unit area enclosed by the interferometry loop delineated by the measurement. The quantities w_c , w_t , and w_a are the difficulty weights associated with the corresponding factors described above. (The weights w_t associated with the tunneling junctions will also include the contribution from horizontal cutter gates, since these are used to control the tunneling in a manner somewhat different from the way the vertical cutter gates are used to define the quantum dots.) The difficulty associated with the number of hexons involved in the measurement is likely a more complicated (though quickly growing) function of N that we denote as $f(N)$. All of these quantities must be determined by the experimental setup being utilized.

3.2.5 Relabeling Majorana zero modes

In our discussion thus far, we have labeled the MZMs in a hexon $1, \dots, 6$ and assigned them particular roles according to these labels. For example, in the computational basis, the MZMs labeled as 3 and 4 serve as the ancillary pair, while MZMs 1, 2, 5, and 6 collectively

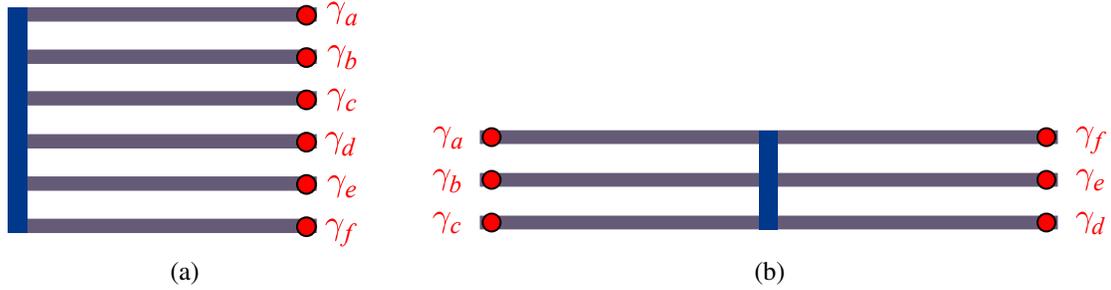


Figure 3.5: A labeling configuration $\langle a, b, c, d, e, f \rangle$ of MZMs shown for (a) one-sided hexons, which follow the labeling order from top to bottom, and (b) two-sided hexons, which follow the labeling order counterclockwise from top-left to top-right.

encode the computational qubit. However, we are free to choose how the six labels are assigned to the physical MZMs of a hexon. We will now discuss briefly how this choice can affect the difficulty of different measurements, and hence measurement-only gate synthesis.

Let $\langle a, b, c, d, e, f \rangle$ denote the configuration of MZMs within a hexon, where (i) for one-sided hexons, the labeling goes from top to bottom, and (ii) for two-sided hexons, the labeling goes counterclockwise from the top-left to the top-right. A possible configuration for either hexon architecture, which was used in Ref. [41], is $\langle 1, 2, 3, 4, 5, 6 \rangle$. Here, MZMs 1 and 6 are on opposite ends of the hexon. On the other hand, in the configuration $\langle 1, 6, 2, 3, 4, 5 \rangle$, these two MZMs are adjacent. In this way, different configurations of MZMs will result in different assignments of difficulty weights to a measurement. For example, a measurement of MZMs (16) will have weights $w(16)_{\langle 1,6,2,3,4,5 \rangle} < w(16)_{\langle 1,2,3,4,5,6 \rangle}$ for these two configurations. Thus, if this measurement occurs very frequently in a computation, the configuration $\langle 1, 6, 2, 3, 4, 5 \rangle$ may be advantageous. We will take this into account when we numerically optimize measurement sequences in Sec. 3.5 and discuss examples of optimal configurations of the labels under certain assumptions about the weights.

Note that there are certain symmetry relations for each architecture, which reduce the number of inequivalent configurations that must be considered. A two-sided hexon has horizontal and vertical reflection symmetry, reducing the number of inequivalent configurations from $6! = 720$ to 180. One-sided hexons have horizontal reflection symmetry, so the number of configurations that we consider is reduced from 720 to 360.

In order for the gate generation methods to be scalable, the full array of hexons in the system should utilize labeling configurations that are periodic in the array. In this chapter, we consider the simplest case, where each hexon in the array uses the same labeling configuration. However, one could imagine finding benefits from assigning different configurations to different hexons, e.g. one configuration for all right-facing one-sided hexons and a different configuration for all left-facing one-sided hexons.

Depending on the architecture and labeling configuration used, the different 4-MZM measurements can have significantly different difficulty weights. Moreover, the measurements involving hexon pairs that are neighbors in different directions may have different difficulty levels. For example, in the case of one-side hexon arrays, the measurements connecting vertical neighbors shown in Fig. 3.4(b) and (c) will generally be less difficult than those connecting horizontal neighbors shown in Fig. 3.4(d) and (e). However, the geometry can make certain 4-MZM measurements essentially impossible (or prohibitively difficult). For example, the measurements involving vertical neighbors must always involve the top-most MZM of one hexon and the bottom-most MZM of the other hexon.

3.3 Forced-Measurement Methods

In the measurement-only approach to topological quantum computation, the desired sequences of projection operators that yield computational gates are physically generated by performing measurements on the system. When the joint fermionic parity operator $\Gamma_{\mathcal{M}}$ of an ordered set \mathcal{M} of MZMs is measured in a system in a pure state $|\Psi\rangle$, the measurement outcome $s = \pm$ will be obtained with probability $p_s = \langle \Psi | \Pi_s^{(\mathcal{M})} | \Psi \rangle$, and one obtains the corresponding post-measurement state

$$|\Psi\rangle \mapsto \frac{1}{\sqrt{p_s}} \Pi_s^{(\mathcal{M})} |\Psi\rangle. \quad (3.39)$$

For general states described by a density matrix ρ , the measurement outcome s is obtained with probability $p_s = \text{Tr} [\Pi_s^{(\mathcal{M})} \rho]$, and the post-measurement state is

$$\rho \mapsto \frac{1}{p_s} \Pi_s^{(\mathcal{M})} \rho \Pi_s^{(\mathcal{M})}. \quad (3.40)$$

The probabilistic nature of measurements can be dealt with in the measurement-only approach (where ancillary degrees of freedom are being utilized) by using forced-measurement protocols. When the outcome of a measurement in a measurement-only sequence is a non-Abelian anyon, the use of a forced-measurement protocol is necessary. On the other hand, when the measurement outcomes is an Abelian anyon (different from the “desired” measurement outcome at a given step), then one can use tracking methods as a more efficient alternative, as will be described in Sec. 3.4. In the case of MZMs, one can always use tracking methods instead of forced-measurements. However, we will nonetheless use the example of MZMs to

discuss forced-measurement methods in this section, since the basic ideas carry over to more general non-Abelian anyons, with straightforward modifications.

3.3.1 Forced-measurement protocols for 2-MZM measurements

In order to get a desired projector $\Pi_s^{(jk)}$ in the measurement-only scheme, we can utilize a repeat-until-success “forced-measurement” procedure. When the measurement of $i\gamma_j\gamma_k$ is performed, the probability of obtaining the desired outcome is $1/2$ (except for the initial projector on the ancillary MZMs, which should have deterministic outcome). If an undesired measurement outcome is obtained, we can essentially undo this measurement by performing a parity measurement on the pair of MZMs measured in the previous step, and then perform the measurement of $i\gamma_j\gamma_k$ again. Each such measurement of $i\gamma_j\gamma_k$ yields a new probability of $1/2$ of obtaining the desired measurement outcome. This repeated attempt and reset process does not collapse the encoded computational state, because we are utilizing ancillary MZMs and measurements in a manner similar to the quantum state teleportation protocol. In other words, the measurements simply alter which subset of the system encodes the computational state. On average, the number of attempts needed (including the first one) to obtain the desired measurement outcome in this way is 2. The likelihood of not succeeding to obtain the desired outcome within n attempts is 2^{-n} , so failure is exponentially suppressed.

For example, suppose we wish to implement the S gate via the sequence of projection

operators $\Pi_+^{(34)}\Pi_+^{(23)}\Pi_+^{(13)}$:

$$(3.41)$$

Imagine that we perform the first step's measurement of $i\gamma_1\gamma_3$ with the desired outcome $s_1 = +$, but for the second step's measurement of $i\gamma_2\gamma_3$, we obtain the undesired outcome $s_2 = -$. At this point, we can repeat the measurement of $i\gamma_1\gamma_3$ (the outcome of which is irrelevant) and then repeat the measurement of $i\gamma_2\gamma_3$, with another 1/2 probability of obtaining the desired outcome $s_2 = +$. If the undesired measurement outcome is obtained again, we repeat this process until the desired outcome is obtained. This example is depicted in the following (recall that the purple line indicates unspecified measurement outcomes, here the outcome does not affect the result):

$$(3.42)$$

Notice that the measurements corresponding to $\Pi_-^{(23)}$ and $\Pi_{s_3}^{(13)}$ are rendered inconsequential by the forcing procedure. Diagrammatically, this can be verified by applying isotopy invariance (bending and straightening lines) on the the diagrams and using the fact that fermions (red

wavy lines) whose ends both connect to the same MZM line give rise to an overall phase at most, and can thus be removed without changing the state. Algebraically, this can be verified by checking that

$$\Pi_s^{(jk)} \Pi_p^{(kl)} \Pi_q^{(jk)} \Pi_r^{(kl)} \propto \Pi_s^{(jk)} \Pi_r^{(kl)}, \quad (3.43)$$

through a straightforward manipulation of Majorana operators.

In order to distinguish the application of a forced-measurement operation from projectors associated with a physical measurement, we denote the application of this forced-measurement protocol applied to the MZM pair (jk) in a sequence following a measurement of (kl) as $\overset{\leftarrow}{\Pi}_s^{(jk)}$. In terms of the sequence of projectors with the desired measurement outcome s obtained at the n th attempt, we have

$$\overset{\leftarrow}{\Pi}_s^{(jk)} \Pi_r^{(kl)} = \Pi_s^{(jk)} \Pi_{r_{n-1}}^{(kl)} \Pi_{s_{n-1}}^{(jk)} \dots \Pi_{r_3}^{(kl)} \Pi_{s_2}^{(jk)} \Pi_{r_2}^{(kl)} \Pi_{s_1}^{(jk)} \Pi_r^{(kl)}, \quad (3.44)$$

where $s_a \neq s$ for $a = 1, \dots, n-1$, and the measurement outcomes r_a are irrelevant.

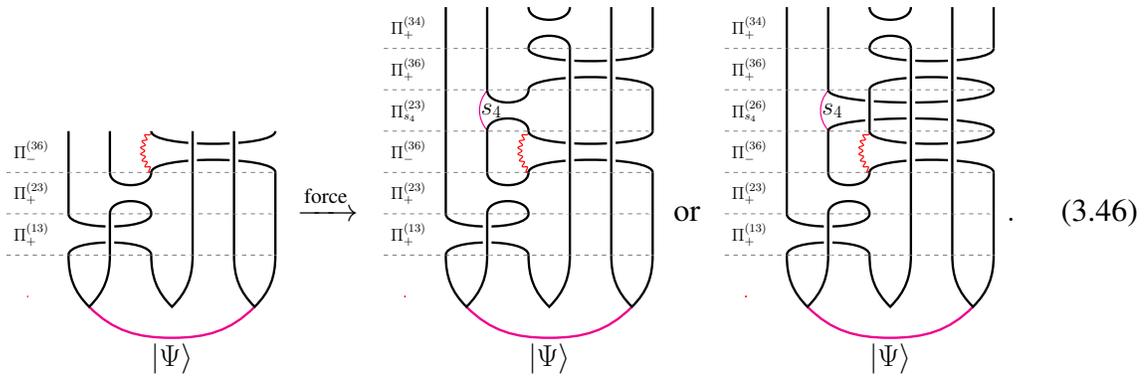
The difficulty weight of a sequence of measurements is simply the product of difficulty weights of each measurement in the sequence. Since a forced measurement involves a probabilistically determined number of measurements, we define the difficulty weight associated with an application of a forced measurement to be the geometric mean (over the distribution for n) of the difficulty weight of the sequence. In other words, the difficulty weight of this

forced measurement is taken to be

$$\overleftarrow{w}(jk) = \exp \left(\sum_{n=1}^{\infty} 2^{-n} \ln [w(jk)^n w(kl)^{n-1}] \right) = w(jk)^2 w(kl). \quad (3.45)$$

This is equal to the difficulty weight of the average case sequence, i.e. $\langle n \rangle = 2$ attempts.³

There is an alternative to this forced-measurement protocol that similarly achieves the desired measurement outcome within a measurement-only sequence. When the measurement of the MZM pair (jk) immediately following a measurement of the MZM pair (kl) yields an undesired outcome, instead of resetting by repeating the previous measurement of (kl) , we can instead reset by measuring the MZM pair (jl) . This is shown in the following diagrammatic representation for the desired projector sequence $\Pi_+^{(34)} \Pi_+^{(36)} \Pi_+^{(23)} \Pi_+^{(13)}$, when an undesired measurement outcome occurs for the measurement of MZMs (36) :



That this procedure works as claimed can be verified diagrammatically by applying isotopy invariance and removing fermion lines, as allowed. Algebraically, this can be verified by check-

³For more general non-Abelian anyons, the probability factors 2^{-n} and corresponding average number of attempts $\langle n \rangle = 2$ need to be replaced with the outcome probabilities particular to the type of anyons involved.

ing that

$$\Pi_s^{(jk)} \Pi_p^{(jl)} \Pi_q^{(jk)} \Pi_r^{(kl)} \propto \Pi_s^{(jk)} \Pi_r^{(kl)}, \quad (3.47)$$

through a straightforward manipulation of Majorana operators.

In order to differentiate the application of this alternative forced-measurement protocol from the previous one (and from an ordinary projector), we denote the application of this forced-measurement protocol applied to the MZM pair (jk) in a sequence following a measurement of (kl) as $\widehat{\Pi}_s^{(jk)}$. In terms of the sequence of projectors with the desired measurement outcome s obtained at the n th attempt, we have

$$\widehat{\Pi}_s^{(jk)} \Pi_r^{(kl)} = \Pi_s^{(jk)} \Pi_{p_{n-1}}^{(jl)} \Pi_{s_{n-1}}^{(jk)} \cdots \Pi_{p_3}^{(jl)} \Pi_{s_2}^{(jk)} \Pi_{p_2}^{(jl)} \Pi_{s_1}^{(jk)} \Pi_r^{(kl)}, \quad (3.48)$$

where $s_a \neq s$ for $a = 1, \dots, n-1$, and the measurement outcomes p_a are irrelevant. Similar to the case of the previous forced-measurement protocol, the difficulty weight associated with an application of this alternative forced measurement is defined to be the geometric mean of the difficulty weight of the sequence, and is equal to the difficulty weight of the average case sequence, i.e. $\langle n \rangle = 2$ attempts. This is given by

$$\widehat{w}(jk) = w(jk)^2 w(jl). \quad (3.49)$$

This alternative forcing protocol would be preferable to the previous one in situations where parity measurements of MZMs (jl) are physically less difficult to perform than those of MZMs (kl) , i.e. when $w(jl) < w(kl)$.

3.3.2 Forced-measurement protocols involving $2N$ -MZM measurements

We now discuss similar forced-measurement strategies for $2N$ -MZM measurements, in particular 4-MZM measurements, as well as 2-MZM measurements that follow a 4-MZM measurement.

In general, the required condition for a forced measurement on \mathcal{M}_2 following a measurement of \mathcal{M}_1 to be possible is the following:

$$\Pi_{s_4}^{(\mathcal{M}_2)} \Pi_{s_3}^{(\mathcal{M}_3)} \Pi_{s_2}^{(\mathcal{M}_2)} \Pi_{s_1}^{(\mathcal{M}_1)} \propto \Pi_{s_4}^{(\mathcal{M}_2)} \Pi_{s_1}^{(\mathcal{M}_1)}, \quad (3.50)$$

for some choice of \mathcal{M}_3 . This, of course, assumes the subsequent projectors in this sequence do not commute, so $\Gamma_{\mathcal{M}_1} \Gamma_{\mathcal{M}_2} = -\Gamma_{\mathcal{M}_2} \Gamma_{\mathcal{M}_1}$ and $\Gamma_{\mathcal{M}_2} \Gamma_{\mathcal{M}_3} = -\Gamma_{\mathcal{M}_3} \Gamma_{\mathcal{M}_2}$, as otherwise they would not interact in a way for which a forcing protocol can be actualized. As such, we see that

$$\begin{aligned} \Pi_{s_4}^{(\mathcal{M}_2)} \Pi_{s_3}^{(\mathcal{M}_3)} \Pi_{s_2}^{(\mathcal{M}_2)} \Pi_{s_1}^{(\mathcal{M}_1)} &= \Pi_{s_4}^{(\mathcal{M}_2)} \frac{\mathbb{1} + s_3 \Gamma_{\mathcal{M}_3}}{2} \Pi_{s_2}^{(\mathcal{M}_2)} \Pi_{s_1}^{(\mathcal{M}_1)} \\ &= \frac{1}{2} \Pi_{s_4}^{(\mathcal{M}_2)} \left(\Pi_{s_2}^{(\mathcal{M}_2)} + \Pi_{-s_2}^{(\mathcal{M}_2)} s_3 \Gamma_{\mathcal{M}_3} \right) \Pi_{s_1}^{(\mathcal{M}_1)} \\ &= \frac{1}{2} \Pi_{s_4}^{(\mathcal{M}_2)} (s_3 \Gamma_{\mathcal{M}_3})^{\frac{1-s_2 s_3}{2}} \Pi_{s_1}^{(\mathcal{M}_1)}. \end{aligned} \quad (3.51)$$

It is clear that Eq. (3.50) will hold if either

$$\mathcal{M}_3 = \mathcal{M}_1 \quad \text{or} \quad \mathcal{M}_3 = (\mathcal{M}_1 \cup \mathcal{M}_2) \setminus (\mathcal{M}_1 \cap \mathcal{M}_2), \quad (3.52)$$

i.e. if $\Gamma_{\mathcal{M}_3} = \Gamma_{\mathcal{M}_1}$ or $\Gamma_{\mathcal{M}_3} \propto \Gamma_{\mathcal{M}_2} \Gamma_{\mathcal{M}_1}$, since the projectors will then allow $\Gamma_{\mathcal{M}_3}$ to be replaced by a constant.

This provides a generalization of the two different forcing protocols described in the previous subsection. Note that the latter condition can lead to invalid measurement sequences that would collapse the qubit states or to measurements of greater than $2N$ -MZMs if \mathcal{M}_1 and \mathcal{M}_2 contain more than two elements each. This is a case we want to avoid, as the cost of doing multi-MZM measurements grows dramatically in the number of MZMs. On the other hand, the case of $\mathcal{M}_3 = \mathcal{M}_1$ is always permissible and so forced measurements are always possible when needed.

More explicitly, for measurement sequences involving 4-MZM measurements, some of the possible forced-measurement protocols include

$$\widehat{\prod}_s^{(ac)} \prod_r^{(ab; x' y')} = \prod_s^{(ac)} \prod_{p_{n-1}}^{(bc; x' y')} \prod_{s_{n-1}}^{(ac)} \dots \prod_{p_2}^{(bc; x' y')} \prod_{s_1}^{(ac)} \prod_r^{(ab; x' y')}, \quad (3.53a)$$

$$\overset{+}{\prod}_s^{(ac)} \prod_r^{(ab; x' y')} = \prod_s^{(ac)} \prod_{r_{n-1}}^{(ab; x' y')} \prod_{s_{n-1}}^{(ac)} \dots \prod_{r_2}^{(ab; x' y')} \prod_{s_1}^{(ac)} \prod_r^{(ab; x' y')}, \quad (3.53b)$$

$$\overset{+}{\prod}_s^{(ab; x' y')} \prod_r^{(ac)} = \prod_s^{(ab; x' y')} \prod_{r_{n-1}}^{(ac)} \prod_{s_{n-1}}^{(ab; x' y')} \dots \prod_{r_2}^{(ac)} \prod_{s_1}^{(ab; x' y')} \prod_r^{(ac)}, \quad (3.53c)$$

$$\overset{+}{\prod}_s^{(ac; w' z')} \prod_r^{(ab; x' y')} = \prod_s^{(ac; w' z')} \prod_{r_{n-1}}^{(ab; x' y')} \prod_{s_{n-1}}^{(ac; w' z')} \dots \prod_{r_2}^{(ab; x' y')} \prod_{s_1}^{(ac; w' z')} \prod_r^{(ab; x' y')}, \quad (3.53d)$$

which have the corresponding difficulty weights

$$\widehat{w}(ac) = w(ac)^2 w(bc; x'y'), \quad (3.54a)$$

$$\overset{\mp}{w}(ac) = w(ac)^2 w(ab; x'y'), \quad (3.54b)$$

$$\overset{\mp}{w}(ab; x'y') = w(ab; x'y')^2 w(ac), \quad (3.54c)$$

$$\overset{\mp}{w}(ac; w'z') = w(ac; w'z')^2 w(ab; x'y'). \quad (3.54d)$$

3.3.3 Procrastination methods

The forced-measurement protocols of the previous subsections provides control over which fermionic parities are projected upon at each step, which allows us to effectively implement a projector sequence that generates a specified target computational gate. In principle, one can apply a forced-measurement protocol for every projector in a given projector sequence. In practice, this turns out to be an inefficient strategy, since the different projectors in the sequence may have a correlated effect on the resulting gate. This subsection outlines theoretical tools for determining which projectors in a sequence have a correlated effect and, therefore, which specific measurements can tolerate any outcome and which are required to be forced in order to obtain the intended computational gate. We will show that the measurement outcomes can only change the final gate by an overall Pauli operator for the case of MZMs. By the anti-commutation properties of Pauli operators, we need only apply a forced-measurement protocol for at most 3 of the projectors for each hexon in a measurement-only projector sequence in order to realize a specified target gate.

Diagrammatically, this can be understood by recalling that a measurement with outcome s , corresponding to the projector $\Pi_s^{(jk)}$, is represented by a cap and cup in the MZM lines corresponding to γ_j and γ_k , with a fermion line connecting the cap and cup for outcomes $s = -$, as shown in Eq. (3.11). For every $s = -$ projector in a measurement-only sequence of projectors, we can slide the corresponding fermion line (that terminates on two MZM lines) up to the top of the diagram using the diagrammatic rules. Each such fermion line that has been slid to the top of the diagram simply connects two MZM lines a and b , i.e. it results in a parity operator $i\gamma_a\gamma_b$. If every measurement sequence starts and ends with a forced $\Pi_+^{(\text{anc})}$, the fermion lines will not connect to the ancillary MZMs' lines when pushed to the top of the diagram, i.e. a and b do not correspond to ancillary MZMs. Thus, the fermion lines slid to the top of the diagram correspond to the following Pauli operators (cf. Eq. (3.7)):

$i\gamma_a\gamma_b$	Pauli	
$i\gamma_1\gamma_2$	$\mathbb{1} \otimes Z$	
$i\gamma_1\gamma_5$	$Z \otimes Y$	
$i\gamma_1\gamma_6$	$\mathbb{1} \otimes X$	(3.55)
$i\gamma_2\gamma_5$	$Z \otimes X$	
$i\gamma_2\gamma_6$	$-\mathbb{1} \otimes Y$	
$i\gamma_5\gamma_6$	$Z \otimes Z$	

In other words, the complete operation effected on the computational subspace by a measurement-only sequence will be a braiding transformation (hence a Clifford gate) determined by which

MZMs were measured in the sequence, followed by a Pauli gate determined by the measurement outcomes.

Thus, we see that a single hexon projector sequence

$$\mathcal{G} = \Pi_+^{(34)} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_+^{(34)}, \quad (3.56)$$

(with projection channels s_μ that need not all be $+$) compiling to gate G can be rewritten as

$$\mathcal{G} = (i\gamma_{j_q} \gamma_{k_q} \dots i\gamma_{j_1} \gamma_{k_1}) \mathcal{G}_+ \quad (3.57)$$

$$\propto (Z^p \otimes P) \left(\Pi_+^{(\text{anc})} \otimes G_+ \right) = \Pi_+^{(\text{anc})} \otimes P G_+ \quad (3.58)$$

where

$$\mathcal{G}_+ = \Pi_+^{(34)} \Pi_+^{(\mathcal{M}_{n-1})} \dots \Pi_+^{(\mathcal{M}_1)} \Pi_+^{(34)}, \quad (3.59)$$

is the projector sequence obtained from \mathcal{G} by switching all its projectors to have $s_\mu = +$, and q is the number of $s_\mu = -$ projectors in the sequence \mathcal{G} . Furthermore, the product of fermionic parity operators corresponding to the fermion lines after sliding them to the top of the diagram is equal to $i\gamma_{j_q} \gamma_{k_q} \dots i\gamma_{j_1} \gamma_{k_1} = Z^p \otimes P$, where p is an integer and $P \in \{\mathbb{1}, X, Y, Z\}$ is a Pauli gate. Thus, the effect of the measurement outcomes s_μ in a single hexon projector sequence is to change the resulting compiled gate by at most a Pauli gate.

A useful example to consider is the following projector sequence, which can realize any of

the Pauli gates, depending on the measurement outcomes:

$$\mathcal{P} = \Pi_+^{(34)} \Pi_{s_5}^{(23)} \Pi_{s_4}^{(13)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(34)} \Pi_{s_1}^{(35)} \propto \Pi_+^{(\text{anc})} \otimes P, \quad (3.60)$$

$$P = Z^{\frac{1-s_5}{2}} Z^{\frac{1-s_3}{2}} X^{\frac{1-s_2}{2}}. \quad (3.61)$$

Notice that the resulting gate P is independent of s_1 and s_4 . Diagrammatically, this result is easily obtained, as follows.

$$\mathcal{P} |\Psi\rangle = \begin{array}{c} \Pi_+^{(34)} \\ \Pi_{s_5}^{(23)} \\ \Pi_{s_4}^{(13)} \\ \Pi_{s_3}^{(23)} \\ \Pi_{s_2}^{(34)} \\ \Pi_{s_1}^{(35)} \end{array} \begin{array}{c} \text{Diagram 1: Braided MZM lines with measurement outcomes } s_1, s_2, s_3, s_4, s_5 \text{ indicated by pink circles.} \\ \text{Diagram 2: Straightened MZM lines with measurement outcomes } s_1, s_2, s_3, s_4, s_5 \text{ indicated by pink bars.} \\ \text{Diagram 3: Further simplified straightened MZM lines with measurement outcomes } s_1, s_2, s_3, s_4, s_5 \text{ indicated by pink bars.} \end{array} \propto \begin{array}{c} \Pi_+^{(34)} \\ \Pi_+^{(23)} \\ \Pi_+^{(13)} \\ \Pi_+^{(23)} \\ \Pi_+^{(34)} \\ \Pi_+^{(35)} \end{array} \begin{array}{c} \text{Diagram 4: Final simplified braided MZM lines.} \\ \text{Diagram 5: Final simplified straightened MZM lines.} \end{array}, \quad (3.62)$$

where the equality is up to overall phases. Notice that isotopy of the MZM lines allows them to be straightened out, leaving no nontrivial braiding, and hence $P_+ = \mathbb{1}$, where \mathcal{P}_+ is the sequence \mathcal{P} with all measurement outcomes $s_\mu = +$ and P_+ is the gate \mathcal{P}_+ compiles to. Also notice that both ends of the s_1 line connect to the $j = 5$ MZM line when straightened, and both ends of the s_4 line connect to the $j = 1$ MZM line when straighten, so the s_1 and s_4 lines can be removed without affecting the resulting computational gate, i.e. $\gamma_1 \gamma_1 = \gamma_5 \gamma_5 = \mathbb{1}$. Finally,

after straightening out the MZM lines and sliding the s_μ lines to the top of the diagram, we see that $s_2 = -$ would contribute the operator $i\gamma_2\gamma_5 = Z \otimes X$, $s_3 = -$ would contribute $i\gamma_1\gamma_2 = \mathbb{1} \otimes Z$, and $s_5 = -$ would contribute $i\gamma_1\gamma_2 = \mathbb{1} \otimes Z$. Thus, the compiled gate is $P = Z^{\frac{1-s_5}{2}} Z^{\frac{1-s_3}{2}} X^{\frac{1-s_2}{2}}$, as claimed.

Some specific realizations include

$$\mathbb{1} \quad X \quad Y \quad Z \quad (3.63)$$

The figure shows four diagrams, labeled $\mathbb{1}$, X , Y , and Z , representing the realization of Pauli gates. Each diagram consists of a vertical stack of projection channels, indicated by horizontal dashed lines and labeled with $\Pi_+^{(ij)}$ for various pairs (ij) . The channels are connected by MZM lines, which are represented by vertical lines with loops. The initial state is $|\Psi\rangle$, shown at the bottom of each diagram. The X gate is realized by a red dashed line connecting the second and fifth channels. The Y gate is realized by red dashed lines connecting the second and fifth channels, and the first and fourth channels. The Z gate is realized by a red dashed line connecting the first and fourth channels.

Similar arguments apply for the case of multi-hexon projector sequences, which demonstrate that the different choices of projection channels s_μ change the compiled gate by at most a multi-qubit Pauli gate. A more general argument that verifies this is given in Sec. 3.4.

Finally, by tracking the effects of the projection channels s_μ on the resulting compiled gate in this manner, we can see which measurements in the sequence need to be forced in order to obtain the desired gate. In particular, for a single hexon, when we slide all the fermion lines in a projector sequence to the top of the diagram, each line can either be removed or end up in one of the six configurations connecting MZM lines represented by the fermion parity operators $i\gamma_j\gamma_k$ listed in Eq. (3.55). In turn, this determines which Pauli operator a given measurement

outcome contributes to P in the decomposition $G = PG_+$. In this way, it is clear that a measurement sequence generating a Clifford gate requires *at most* three of its measurements in each hexon to be forced – one of which is needed to end with the proper final state $+$ of the ancillary MZMs (via the projector $\Pi_+^{(\text{anc})}$) and at most two of which are needed to ensure that the desired P is obtained in the sequence. For instance, in the example above, we see that sequence \mathcal{P} can generate a particular desired Pauli gate for any values of s_1 , s_3 , and s_4 , by choosing s_2 and s_5 appropriately, i.e. by applying forced measurements for the corresponding steps.

3.3.4 Adaptive methods

While forced measurements and procrastination are, strictly speaking, adaptive protocols, it is worth considering adaptive methods that change the sequence of projectors/forced measurements in a more complex manner. This could potentially find utility when the projector sequence requires a measurement that is particularly difficult, but which we wish to avoid including in forcing protocols, as doing so would increase the number of times this costly measurement will need to be performed, on average. However, this strategy generally increases the total number of measurements needed, so the most likely instances that could benefit from its use would involve multi-hexon measurements whose difficulty outweighs that of several single-hexon measurements. For an example of such an adaptive approach proving beneficial, see Appendix A.1.

3.4 Majorana-Pauli Tracking

When using MZMs for measurement-only topological quantum computing, it is possible to forego the use of forced measurements by instead tracking the measurement outcomes, the different possibilities of which only change the resulting transformation by Pauli gates. [50] More generally, a similar tracking strategy can be employed when the measurement outcomes are always guaranteed to be Abelian anyons, e.g. when using Parafermions (parafermion zero modes), as was applied for measurement-only braiding transformations in Ref. [93]. The tracking methods allow for the use of fewer physical measurement operations and makes the sequence of measurement operations used for topological gate operations completely deterministic. The cost of using such methods is the need to classically track the measurement outcomes and utilize adaptive methods when non-Clifford gates are introduced.

For a system of N hexons, we now write a sequence of projection operators that compiles to a gate $G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}$ acting on the computational state space as

$$\mathcal{G}_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)} = \Pi_{\mathbf{s}_n}^{(\text{anc})} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_{\mathbf{s}_0}^{(\text{anc})} \propto \Upsilon_{\mathbf{s}_n \mathbf{s}_0} \Pi_{\mathbf{s}_0}^{(\text{anc})} \otimes G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}, \quad (3.64)$$

where \mathcal{M}_μ are the ordered sets of (up to $2N$) MZMs whose collective fermionic parity is being projected onto $s_\mu = \pm$ (collectively denoted as \vec{s} , where \mathbf{s}_0 and \mathbf{s}_n are themselves vectors), and the ancillary projectors take the form

$$\Pi_{\mathbf{s}_n}^{(\text{anc})} = \Pi_{s_{n,1}}^{(34)} \otimes \dots \otimes \Pi_{s_{n,N}}^{(3' \dots 4' \dots')}. \quad (3.65)$$

Since we are allowing for the initial and final ancillary projectors to be inequivalent, we introduced the operator

$$\Upsilon_{\mathbf{s}_n \mathbf{s}_0} = \bigotimes_{j=1}^N (i\gamma_{4,j}\gamma_{5,j})^{\frac{1-s_{n,j}s_{0,j}}{2}} = \bigotimes_{j=1}^N (X_j \otimes \mathbb{1}_j)^{\frac{1-s_{n,j}s_{0,j}}{2}}, \quad (3.66)$$

where $\gamma_{a,j}$ is the a th MZM of the j th hexon. This operator flips the state of each ancillary qubit whose initial and final projections differ. In other words, $\Upsilon_{\mathbf{s}_n \mathbf{s}_0} \Pi_{\mathbf{s}_0}^{(\text{anc})} \Upsilon_{\mathbf{s}_n \mathbf{s}_0} = \Pi_{\mathbf{s}_n}^{(\text{anc})}$.

It is straightforward to show using the diagrammatic formalism that the sequence of projectors in Eq. (3.64) must reduce to an operator with the form of the right hand side of that expression. The only task is to determine the operator $G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}$. By definition, we only consider a projector sequence to be a valid measurement-only sequence if $G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}$ is unitary, i.e. does not reduce the rank of the computational subspace. This translates to the requirement that the measurement to be performed anticommutes with at least one of the stabilizer generators as elucidated in 3.2.3.

In the following, we show that different projection channels $(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)$ for a fixed sequence of MZM sets \mathcal{M}_μ will, at most, change the compiled gate $G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}$ by a multi-qubit Pauli gate, assuming it does not reduce the rank.⁴ These Pauli gate differences are determined by the corresponding sequences of projections. In other words, for the same \mathcal{M}_μ with another

⁴It is possible that changing the projection channels will yield a sequence that projects to zero. This merely indicates that such a sequence of projection channels cannot occur as a result of measurements, i.e. it would have probability zero. A trivial example of this would be if we let $\mathcal{M}_\mu = \mathcal{M}_{\mu+1}$ and $s_\mu = -s_{\mu+1}$, but it is possible for more subtle cancellations to occur in a measurement-only sequence if care is not taken to use only valid measurements.

sequence of projection channels $(\mathbf{r}_n, \vec{r}, \mathbf{r}_0)$ that does not project to zero, we have

$$G_{(\mathbf{r}_n, \vec{r}, \mathbf{r}_0)} = P_{(\mathbf{r}_n, \vec{r}, \mathbf{r}_0; \mathbf{s}_n, \vec{s}, \mathbf{s}_0)} G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}, \quad (3.67)$$

where $P_{(\mathbf{r}_n, \vec{r}, \mathbf{r}_0; \mathbf{s}_n, \vec{s}, \mathbf{s}_0)}$ is an N -qubit Pauli gate.

Thus, if we perform a measurement-only sequence of measurements for a desired gate and track the measurement outcomes, rather than using forced measurements, we will have a known Pauli gate correction. If the non-Clifford gates that we utilize in a quantum computation are single qubit phase gates (in any of the Pauli bases), we can also push the Pauli gate correction through the phase gates with at most a Clifford gate correction that can be dealt with by updating the subsequent Clifford gate in the computation to absorb the Clifford correction. When non-Clifford phase gates are implemented by injecting states, such a Clifford correction will be necessary anyway, so this would not be a significantly greater burden.

3.4.1 Proof of Majorana-Pauli tracking

We now prove Eq. (3.67). In the stabilizer picture, for a given sequence of measurements, different measurement outcomes can only affect the updates of logical operators and stabilizers up to a sign at each step. It therefore follows that for a given measurement sequence, the final stabilizer and logical operators will be the same up to signs for all measurement outcomes. Since the action of Pauli operators is to change the sign of Pauli operators, this completes the proof.

A more explicit proof follows by taking the product

$$\begin{aligned} \mathcal{G}_{(\mathbf{r}_n, \vec{r}, \mathbf{r}_0)} \mathcal{G}_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}^\dagger &= \prod_{\mathbf{r}_n}^{(\text{anc})} \prod_{r_{n-1}}^{(\mathcal{M}_{n-1})} \dots \prod_{r_1}^{(\mathcal{M}_1)} \prod_{\mathbf{r}_0}^{(\text{anc})} \prod_{\mathbf{s}_0}^{(\text{anc})} \prod_{s_1}^{(\mathcal{M}_1)} \dots \prod_{s_{n-1}}^{(\mathcal{M}_{n-1})} \prod_{\mathbf{s}_n}^{(\text{anc})} \\ &\propto \delta_{\mathbf{r}_0, \mathbf{s}_0} \Upsilon_{\mathbf{r}_n \mathbf{s}_n} \prod_{\mathbf{s}_n}^{(\text{anc})} \otimes G_{(\mathbf{r}_n, \vec{r}, \mathbf{r}_0)} G_{(\mathbf{s}_n, \vec{s}, \mathbf{s}_0)}^\dagger, \end{aligned} \quad (3.68)$$

and recursively using relations that will reduce the product of projectors.

For this, we will utilize the relation

$$\prod_{q_1}^{(\mathcal{B}_1)} \dots \prod_{q_k}^{(\mathcal{B}_k)} + z \prod_{-q_1}^{(\mathcal{B}_1)} \dots \prod_{-q_k}^{(\mathcal{B}_k)} = (q_1 \Gamma_{\mathcal{B}_1})^{\frac{1-z}{2}} \prod_{q_{1:2}}^{(\mathcal{B}_{1:2})} \dots \prod_{q_{k-1:k}}^{(\mathcal{B}_{k-1:k})}, \quad (3.69)$$

that holds for ordered sets \mathcal{B}_μ of even numbers of MZMs such that the $\prod_{q_\mu}^{(\mathcal{B}_\mu)}$ all commute with each other, i.e. $|\mathcal{B}_\mu \cap \mathcal{B}_\nu|$ is even for all μ and ν , where $z, q_\alpha = \pm 1$. In this expression, we define $q_{\mu:\nu} = \pm 1$ and the ordered sets $\mathcal{B}_{\mu:\nu}$ obtained by taking the symmetric difference $(\mathcal{B}_\mu \cup \mathcal{B}_\nu) \setminus (\mathcal{B}_\mu \cap \mathcal{B}_\nu)$, and ordering its elements such that

$$\prod_{q_{\mu:\nu}}^{(\mathcal{B}_{\mu:\nu})} = \frac{\mathbb{1} + q_{\mu:\nu} \Gamma_{\mathcal{B}_{\mu:\nu}}}{2} = \frac{\mathbb{1} + q_\mu q_\nu \Gamma_{\mathcal{B}_\mu} \Gamma_{\mathcal{B}_\nu}}{2}. \quad (3.70)$$

We notice that, since $|\mathcal{B}_\mu \cap \mathcal{B}_\nu|$ is even, the operator $\frac{1}{2}(\mathbb{1} + q_\mu q_\nu \Gamma_{\mathcal{B}_\mu} \Gamma_{\mathcal{B}_\nu})$ will always be a projector for a joint fermionic parity operator $\Gamma_{\mathcal{B}_{\mu:\nu}}$.

We can establish Eq. (3.69) inductively, starting by noticing that

$$\prod_q^{(\mathcal{B})} + z \prod_{-q}^{(\mathcal{B})} = \frac{\mathbb{1} + q \Gamma_{\mathcal{B}}}{2} + z \frac{\mathbb{1} - q \Gamma_{\mathcal{B}}}{2} = (q \Gamma_{\mathcal{B}})^{\frac{1-z}{2}}. \quad (3.71)$$

For $k = 2$, we see that

$$\begin{aligned}
\Pi_{q_1}^{(\mathcal{B}_1)} \Pi_{q_2}^{(\mathcal{B}_2)} + z \Pi_{-q_1}^{(\mathcal{B}_1)} \Pi_{-q_2}^{(\mathcal{B}_2)} &= \frac{\mathbb{1} + q_1 \Gamma_{\mathcal{B}_1}}{2} \frac{\mathbb{1} + q_2 \Gamma_{\mathcal{B}_2}}{2} + z \frac{\mathbb{1} - q_1 \Gamma_{\mathcal{B}_1}}{2} \frac{\mathbb{1} - q_2 \Gamma_{\mathcal{B}_2}}{2} \\
&= \frac{1 + z}{2} \frac{\mathbb{1} + q_1 q_2 \Gamma_{\mathcal{B}_1} \Gamma_{\mathcal{B}_2}}{2} + \frac{1 - z}{2} \frac{q_1 \Gamma_{\mathcal{B}_1} + q_2 \Gamma_{\mathcal{B}_2}}{2} \\
&= (q_1 \Gamma_{\mathcal{B}_1})^{\frac{1-z}{2}} \Pi_{q_{1:2}}^{(\mathcal{B}_{1:2})}. \tag{3.72}
\end{aligned}$$

If Eq. (3.69) holds for $k \geq 2$, then

$$\begin{aligned}
(q_1 \Gamma_{\mathcal{B}_1})^{\frac{1-z}{2}} \Pi_{q_{1:2}}^{(\mathcal{B}_{1:2})} \dots \Pi_{q_{k:k+1}}^{(\mathcal{B}_{k:k+1})} &= \left(\Pi_{q_1}^{(\mathcal{B}_1)} \dots \Pi_{q_k}^{(\mathcal{B}_k)} + z \Pi_{-q_1}^{(\mathcal{B}_1)} \dots \Pi_{-q_k}^{(\mathcal{B}_k)} \right) \Pi_{q_{k:k+1}}^{(\mathcal{B}_{k:k+1})} \\
&= \left(\Pi_{q_1}^{(\mathcal{B}_1)} \dots \Pi_{q_k}^{(\mathcal{B}_k)} + z \Pi_{-q_1}^{(\mathcal{B}_1)} \dots \Pi_{-q_k}^{(\mathcal{B}_k)} \right) \left(\Pi_{q_k}^{(\mathcal{B}_k)} \Pi_{q_{k+1}}^{(\mathcal{B}_{k+1})} + \Pi_{-q_k}^{(\mathcal{B}_k)} \Pi_{-q_{k+1}}^{(\mathcal{B}_{k+1})} \right) \\
&= \Pi_{q_1}^{(\mathcal{B}_1)} \dots \Pi_{q_{k+1}}^{(\mathcal{B}_{k+1})} + z \Pi_{-q_1}^{(\mathcal{B}_1)} \dots \Pi_{-q_{k+1}}^{(\mathcal{B}_{k+1})} \tag{3.73}
\end{aligned}$$

shows that it holds for $k + 1$, and this completes the induction argument.

Returning to the product of projectors in Eq. (3.68), each step of the recursion involves a product in the middle of the string of projectors that takes the form

$$\Pi_r^{(\mathcal{M})} \Pi_{q_1}^{(\mathcal{A}_1)} \dots \Pi_{q_k}^{(\mathcal{A}_k)} \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_l}^{(\mathcal{C}_l)} \Pi_s^{(\mathcal{M})}, \tag{3.74}$$

where $|\mathcal{A}_\alpha \cap \mathcal{A}_{\alpha'}|$, $|\mathcal{A}_\alpha \cap \mathcal{C}_\beta|$, $|\mathcal{C}_\beta \cap \mathcal{C}_{\beta'}|$, and $|\mathcal{M} \cap \mathcal{C}_\beta|$ are all even, while $|\mathcal{M} \cap \mathcal{A}_\alpha|$ are all odd. In other words, the projectors $\Pi_{q_\alpha}^{(\mathcal{A}_\alpha)}$ and $\Pi_{p_\beta}^{(\mathcal{C}_\beta)}$ all commute with each other, $\Pi_s^{(\mathcal{M})}$ com-

mutates with $\Pi_{p_\nu}^{(\mathcal{C}_\beta)}$, and $\Gamma_{\mathcal{M}}$ anticommutes with all $\Gamma_{\mathcal{A}_\alpha}$. From this, we find

$$\begin{aligned}
\Pi_r^{(\mathcal{M})} \Pi_{q_1}^{(\mathcal{A}_1)} \dots \Pi_{q_k}^{(\mathcal{A}_k)} \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_l}^{(\mathcal{C}_l)} \Pi_s^{(\mathcal{M})} &= \Pi_r^{(\mathcal{M})} \Pi_{q_1}^{(\mathcal{A}_1)} \dots \Pi_{q_k}^{(\mathcal{A}_k)} \Pi_s^{(\mathcal{M})} \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_l}^{(\mathcal{C}_l)} \\
&= \frac{1}{2} \left(\Pi_{q_1}^{(\mathcal{A}_1)} \dots \Pi_{q_k}^{(\mathcal{A}_k)} + r s \Pi_{-q_1}^{(\mathcal{A}_1)} \dots \Pi_{-q_k}^{(\mathcal{A}_k)} \right) \Pi_s^{(\mathcal{M})} \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_l}^{(\mathcal{C}_l)} \\
&= \frac{1}{2} (q_1 \Gamma_{\mathcal{A}_1})^{\frac{1-rs}{2}} \Pi_{q_{1:2}}^{(\mathcal{A}_{1:2})} \Pi_{q_{2:3}}^{(\mathcal{A}_{2:3})} \dots \Pi_{q_{k-1:k}}^{(\mathcal{A}_{k-1:k})} \Pi_s^{(\mathcal{M})} \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_l}^{(\mathcal{C}_l)}, \tag{3.75}
\end{aligned}$$

where we expanded $\Pi_r^{(\mathcal{M})} = \frac{\mathbb{1} + r \Gamma_{\mathcal{M}}}{2}$ and anticommuted $\Gamma_{\mathcal{M}}$ through the $\Gamma_{\mathcal{A}_\alpha}$ to obtain the second line, and then used Eq. (3.69). We notice that all the projectors in the last line of Eq. (3.75) commute with each other, since $|\mathcal{A}_{\alpha:\alpha+1} \cap \mathcal{M}|$ is even.

Recursively applying Eq. (3.75) to Eq. (3.68) and moving extra fermionic parity operators (e.g. $\Gamma_{\mathcal{A}_1}$) through the remaining projectors to the left (which will flip the projection channels when $|\mathcal{A}_1 \cap \mathcal{M}_\mu|$ is odd), we find a final result of the form

$$\mathcal{G}_{(r_n, \vec{r}, r_0)} \mathcal{G}_{(s_n, \vec{s}, s_0)}^\dagger \propto \delta_{r_0, s_0} \Gamma_B \Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_m}^{(\mathcal{C}_m)}, \tag{3.76}$$

where the projectors all commute with each other and Γ_B is the fermionic parity operator corresponding to some ordered set of MZM labels \mathcal{B} that is determined by the projector sequences.

When this does not project to zero, it must be proportional to $\Upsilon_{r_n s_n} \mathbf{\Pi}_{s_n}^{(\text{anc})} \otimes G_{(r_n, \vec{r}, s_0)} G_{(s_n, \vec{s}, s_0)}^\dagger$, which implies that $\Pi_{p_1}^{(\mathcal{C}_1)} \dots \Pi_{p_m}^{(\mathcal{C}_m)} = \mathbf{\Pi}_{s_n}^{(\text{anc})}$ and that $G_{(r_n, \vec{r}, s_0)} G_{(s_n, \vec{s}, s_0)}^\dagger$ is a multi-qubit Pauli operator.

Applying the same argument to $\mathcal{G}_{(r_n, \vec{r}, r_0)}^\dagger \mathcal{G}_{(s_n, \vec{s}, s_0)}$ shows that, when this sequence does not project to zero, $G_{(s_n, \vec{s}, s_0)}^\dagger G_{(r_n, \vec{r}, r_0)}$ is a multi-qubit Pauli operator. Combining the results for

these two cases establishes that when $G_{(r_n, \vec{r}, r_0)}$ and $G_{(s_n, \vec{s}, s_0)}$ are nonzero, they are related by a multi-qubit Pauli gate. This proves Eq. (3.67).

3.5 Brute-force Optimization of Measurement-Only Generation of Gates

In this section, we discuss optimization strategies for measurement-generated gates and then carry out numerical searches for the optimal measurement-only realizations of gates. We exhaustively search all valid projector sequences, i.e. those that do not collapse the encoded computational state, up to some pre-determined length for single-qubit and two-qubit gates. This is used to determine the optimal measurement sequences for all single-qubit gates. For two-qubit gates, the search space is much larger and we limit our focus on optimization of the controlled-Pauli, W , and SWAP gates.

In Appendix A.2, we will discuss techniques whose computational costs scale better than brute-force search, but which are not guaranteed to find (globally) optimal measurement sequences.

3.5.1 Optimization

There are many possible strategies and layers of optimization that may be employed in an effort to optimize the implementation of computational gates.

The crucial first step is deciding on the metric with respect to which optimization is per-

formed. A simple choice would be the length of measurement sequences, which would provide useful results if all measurements are approximately equally difficult to implement. The difficulty weights introduced in Sec. 3.2.4 provide a more physically realistic cost function for optimization. The difficulty weight $w(\mathcal{M})$ in Eq. (3.38) provides a systematic estimation of the error and resource costs of a joint parity measurement of MZMs \mathcal{M} . For each measurement-only gate implemented by a sequence of physical measurements corresponding to $\mathcal{M}_1, \dots, \mathcal{M}_n$, we assign the sequence a difficulty weight defined as the product of its component measurements' weights:

$$w(\{\mathcal{M}_1, \dots, \mathcal{M}_n\}) = \prod_{\mu=1}^n w(\mathcal{M}_\mu). \quad (3.77)$$

Here, \mathcal{M}_n constitutes the ancillary MZMs that actually need to be measured at the final step, i.e. the ones whose projectors do not commute with the rest of the projector sequence, and actually may represent multiple measurements, since each hexon's ancillary pair are projected/measured separately. (We do not include a contribution for the initial ancillary measurement at step $\mu = 0$, since that is provided by previous operations.) The individual weight factors in Eq. (3.38) will need to be determined through experimental characterization of the physical systems.

Another key aspect of optimization is deciding which set of computational gates to optimize, as all gates cannot be simultaneously optimized. This choice should take into consideration how the quantum computing system is primarily going to be used. For example, if it is implementing certain algorithms or error-correction protocols that call certain gates with high frequency, then it would be natural to optimize the implementation for that set of gates. Some

typical choices include the controlled-Pauli gates, the Hadamard gate, and/or all single-qubit Clifford gates. When averaging the sequence weights over the target set, we use the geometric mean due to the multiplicative nature of the weights.

In determining how to appropriately search for optimal measurement sequences, one needs to decide whether one is utilizing Majorana-Pauli tracking methods or forced-measurement methods, as the optimization goals and relation between projector sequences and measurement sequences differ between these two cases, which we detail in the following. In most quantum computing contexts, it will be preferable to utilize the tracking methods, as they generally provide significantly better efficiency than forced-measurement methods. We will demonstrate our optimization methods for both approaches.

As seen in Sec. 3.2.5, the difficulty weights for different measurements will depend on the MZM labeling configuration used for a given hexon architecture. Thus, the labeling configurations represents another set of parameters over which one can optimize. We carry out the gate optimization analysis within a fixed labeling configuration, and then do so for each possible labeling configuration, up to symmetry. In this way, we can compare and determine which configuration(s) provide the optimal implementation of the relevant gate set.

3.5.1.1 Majorana-Pauli tracking methods

In the case where we use Majorana-Pauli tracking, when we write the measurement-only compilation of a gate G in terms of a projector sequence

$$\mathcal{G}_{(s_n, \vec{s}, s_0)} = \Pi_{s_n}^{(\text{anc})} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \cdots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_{s_0}^{(\text{anc})}, \quad (3.78)$$

the sequence of physical measurements to be performed is exactly the sequence $\mathcal{M}_1, \dots, \mathcal{M}_n$ specified in the projector sequence. When the physical measurement outcomes do not match the specified projector channels s_μ , the resulting gate will differ from G by at most a Pauli gate, which we track and compensate for at a later time in a more efficient manner. As such, this measurement-only realization of G is assigned the difficulty weight

$$w(\mathcal{G}) = \prod_{\mu=1}^n w(\mathcal{M}_\mu), \quad (3.79)$$

where \mathcal{M}_n corresponds to the measurements of ancillary MZMs.

When Majorana-Pauli tracking is being utilized, it is useful to group together Clifford gates into their Pauli cosets, given by the collections of Clifford gates that are equivalent up to multiplication by an overall (multi-qubit) Pauli gate, i.e. the Pauli coset of a N -qubit Clifford gate G is defined to be

$$[G] = \{G' \in \mathcal{C}_N \mid \exists P \in \mathcal{P}_N : G' = PG\}. \quad (3.80)$$

When using tracking, we do not need to be able to generate every Clifford gate; we only need one gate from each Pauli coset, as all differences by Pauli gates are dealt with by the tracking methods. Thus, we can use the most easily realized gate in a given Pauli coset to implement the entire class of gates. In this way, optimization of $[G]$ is carried out by optimizing all of its elements and selecting the one with lowest difficulty weight to use when any of the gates in $[G]$

is called in a computation. Thus, we define

$$w([G]) = \min_{\mathcal{G} \in [G]} w(\mathcal{G}). \quad (3.81)$$

3.5.1.2 Forced-measurement methods

If, for some reason, one wanted to implement the braiding Clifford gates exactly instead of up to a Pauli correction, forced-measurement protocols would be utilized to ensure the desired gate. (More generally, forced-measurement protocols are actually necessary when the fusion channels include non-Abelian anyons, which is not the case for MZMs.) In this case, one must decide which of the forced-measurement methods to utilize. In our demonstrations, we utilize both forced-measurement protocols and procrastination, but not more complicated adaptive methods.

Using these forced-measurement methods, when we write the measurement-only compilation of a gate G in terms of a projector sequence

$$\mathcal{G}_{(+,\vec{s},+)} = \Pi_+^{(\text{anc})} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_+^{(\text{anc})}, \quad (3.82)$$

we must determine the minimal set of projectors in the sequence that must be forced in order to generate the desired gate (there are at most three per hexon involved in the gate). We then follow the procrastination method outlined in Sec. 3.3.3 and convert the projector sequence into a measurement sequence by utilizing forced-measurement protocols only for the generation of these projectors that must be forced, and standard measurements for the rest. For each of the

projectors that must be forced, we assess which of the two forced-measurement protocols has the smaller difficulty weight, as given by Eqs. (3.45), (3.49), and (3.54), and we use the lesser weight protocol to implement that forced projector.

The corresponding physical measurement sequence obtained from the projector sequence will be probabilistically determined. As such, we consider the geometric average of the difficulty weight of the physical measurement sequence. This is obtained by starting with the expression for the difficulty weight of the projector sequence and replacing the weights of the projectors that must be forced with the average difficulty weight corresponding to the forced-measurement protocol used. This gives the average difficulty weight of this forced-measurement implementation of G :

$$w(\mathcal{G}_{(+,\vec{s},+)}) = \prod_{\mu=1}^n w(\mathcal{M}_\mu) \prod_{\mu \in F_1} \frac{\overleftarrow{w}(\mathcal{M}_\mu)}{w(\mathcal{M}_\mu)} \prod_{\mu \in F_2} \frac{\widehat{w}(\mathcal{M}_\mu)}{w(\mathcal{M}_\mu)}, \quad (3.83)$$

where F_1 is the set of projectors in the sequence to be implemented by forced measurements of the first type and F_2 is the set of projectors in the sequence to be implemented by forced measurements of the second type.

In this way, the optimization analysis when using forced-measurement methods is still processed via projector sequences.

3.5.2 Gate search

Single-qubit gates For single-qubit gates, we first determine which sequences of measurements/projectors are valid, i.e. which sequences of \mathcal{M}_μ do not collapse the computational

state. (For single-qubit gates, this is irrespective of the corresponding projection channels s_μ at each step.) As discussed in Sec. 3.2.2, valid single-qubit measurement sequences have the constraint that consecutive 2-MZM measurements must have exactly one MZM in common, so each measurement step involves choosing one MZM from the previous measurement pair and one from the four remaining MZMs, leading to 8 possible measurements to choose from. The n th measurement in the sequence is fully constrained, as it must be of the ancillary pair of MZMs (3,4). The penultimate measurement is also constrained, as it must involve one MZM from the antepenultimate measurement pair and one from the ancillary pair (3,4), leading to 4 possible choices for the penultimate measurement. Thus, the size of the search space for single-hexon measurement sequences of length n is 2^{3n-4} . Even though this scaling is exponential in n , we are able to consider sufficiently long sequences for all single-qubit gates in order to determine their optimal measurement-only sequences.

Once we have determined which measurement/projector sequences are valid, we evaluate the resulting computational gates $G_{(s_n, \vec{s}, s_0)}$ for all possible measurement outcomes/projection channels s_μ .

For the single-qubit gates, when forced-measurement methods are being utilized, we can determine the minimal set of projectors that need to be forced using the tools developed in Sec. 3.3.3. Moreover, when Majorana-Pauli tracking methods are being utilized, the same methods allow us to determine the overall Pauli gate correction.

In the following, we searched up to $n = 9$ and found that the lowest weight sequences occur at $n \leq 5$. For the estimated weight factors used in this thesis, this constitutes an exhaustive

search for the minimal weight single-qubit gates, because longer sequences for the same gates are guaranteed to have larger difficulty weight values. In other words, we have found the globally optimal measurement-only implementations of the single-qubit gates.

Two-qubit gates The set of two-qubit Clifford gates has 11,520 elements or 720 Pauli cosets, making it impractical to report optimal sequences for each element. For the purpose of this thesis, we focus on controlled-Pauli gates $\{C(X), C(Y), C(Z)\}$ and these will be the only two-qubit gates with respect to which we optimize the labeling configurations. We then also report results for the W and SWAP gates within these labeling configurations.

As discussed in Sec. 3.2.3, valid measurements on two-hexons must be of operators that anticommute with at least one stabilizer of the code. There are thus 16 valid 2-MZM measurements and 176 valid 4-MZM measurements at every step. As before, the final set of stabilizers must match the initial one. This condition fixes the final measurement and also partially fixes the penultimate measurement in a sequence. Thus, the search space is roughly $176^k 16^{n-k-1}$ depending on the number k of 4-MZM measurements involved in a sequence and their placements. In addition to helping reduce the size of the search space, limiting the number of 4-MZM projectors can be physically motivated by the assumption that they would typically be significantly more costly than 2-MZM measurements.

We have carried out such a search up to length $n = 5$ and included at most $k = 3$ 4-MZM measurements. For each measurement sequence that compiles to $C(X), C(Y), C(Z)$ up to a Pauli operator, we calculate its difficulty weight and compare it with other sequences that realize these gates up to Pauli cosets, recording the minimal weight sequence found for each

labeling configuration.

Within each $C(P)$ optimized labeling configuration we also search for minimal weight sequences compiling to W and SWAP. Note that certain 4-MZM measurements are not possible in the one-sided geometries. In these cases, gates such as SWAP require longer measurement sequences though, at the same time, the restriction allows us to search to a greater length $n = 6$.

Both minimal weight forced-measurement and minimal weight tracked-measurement sequences for $C(P)$ gates were found at $n = 4$ involving only a single 4-MZM measurement. No $C(P)$ gates were found for sequences of length $n < 4$ though W gates occur at $n = 3, k = 1$. A naïve compilation for SWAP is $\text{SWAP} = C(X)_{12}C(X)_{21}C(X)_{12}$ which in the present compilation would require three 4-MZM measurements. However, our search reveals a more direct compilation for SWAP requiring only two 4-MZM measurements.

In the case of two-qubit gates, we do not have a simple way of determining the Pauli corrections or which projectors need to be forced, so we use a brute-force method. In particular, for a given measurement sequence that can realize a desired gate, we evaluate the sequence using all possible measurement/projector channels s_μ . This immediately gives the Pauli correction gate for Majorana-Pauli tracking, and can be used to determine which projectors need to be forced when using forced-measurement methods. This is done by first grouping together projector sequences that yield the same gate. For each such set of projector sequences, we first check which projector channels s_μ are the same across all elements of the set; these projectors must be forced. We then look for correlations between the remaining measurement outcomes, which may require further forced measurements. We start from the first projector that does not

have fixed projection channel, which we denote as s_ν , and consider separately the subsets of projector sequences where this outcome is $s_\nu = +1$ or -1 . Within each subset, we check if any subsequent measurement has fixed outcome; if so, it must be forced onto a channel that is correlated with s_ν , and if not we recursively apply the procedure to this measurement. For example, we find that the measurement sequence $\Pi_+^{(34)}\Pi_{s_3}^{(35)}\Pi_{s_2}^{(56)}\Pi_{s_1}^{(35;1'6')}$ compiles to $C(X)$ exactly when $s_2 = +$ and $s_3 = s_1$, i.e. the following two projector sequences yield the same gate:

$$\Pi_+^{(34)}\Pi_+^{(35)}\Pi_+^{(56)}\Pi_+^{(35;1'6')}, \quad (3.84a)$$

$$\Pi_+^{(34)}\Pi_-^{(35)}\Pi_+^{(56)}\Pi_-^{(35;1'6')}, \quad (3.84b)$$

which indicates that the $\mu = 2, 3, 4$ projectors need to be forced.

3.5.3 Demonstration of Methods

We now demonstrate the use of our methods for the various cases of interest. For the purposes of producing a quantitative demonstration, we will very roughly estimate the difficulty weight factors to be: $w_c = 1.25$, $w_t = 1.65$, $w_a = 1.01$, and $f(N) = (\prod_{n=1}^N n!)^{(N-1)!}$. The results obtained for these weight factor values should not be misconstrued as being universal. For practical applications, the analysis will need to be performed again using weight factors that are more accurately estimated from experiments on the physical system being utilized.

We note that multiple measurement-only sequences may yield the same computational gate

with the same difficulty weight. When this is the case for minimal weight sequences, we only present one representative of the set of minimal weight sequences for a gate or Pauli class. Similarly, multiple MZM labeling configurations may yield equally optimal minimal difficulty weights for the relevant gates, and we will only present one of the optimal configurations.

For two-sided hexon architectures, in both the case of using forced-measurement methods and the case of using Majorana-Pauli tracking methods, we find that the MZM labeling configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$ yields the optimal results within our search for each of the following gates or gate sets, independently: the single-qubit Hadamard gate, the geometric average of all single-qubit Clifford gates, the geometric average of $C(X)$ acting in all four directions, and in the geometric average over all $C(P)$ gates in all four directions.

For one-sided hexon architectures, in the case of using forced-measurement methods, we find that: (a) the MZM labeling configuration $\langle 1, 2, 6, 3, 4, 5 \rangle$ yields the optimal results for the Hadamard gate and the geometric average of all single-qubit Clifford gates; (b) the MZM labeling configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$ yields the optimal results within our search for the geometric average over $C(X)$ acting in all four directions and the geometric average over all $C(P)$ gates in all four directions.

For one-sided hexon architectures, in the case of using Majorana-Pauli tracking methods, we find that the MZM labeling configuration $\langle 1, 2, 6, 3, 4, 5 \rangle$ yields the optimal results within our search for each of the following gates or gate sets, independently: the single-qubit Hadamard gate, the geometric average of all single-qubit Clifford gates, the geometric average of $C(X)$ acting in all four directions, and in the geometric average over all $C(P)$ gates in all four direc-

Configuration	H	C_1	$C(X)$	$C(P)$	W	SWAP
$\langle 3, 4, 1, 2, 6, 5 \rangle_2$	1.39×10^8	7.72×10^6	8.10×10^8	7.78×10^8	3.81×10^6	1.50×10^{12}
$\langle 1, 2, 6, 3, 4, 5 \rangle_1$	9.99×10^5	1.45×10^5	2.85×10^8	3.02×10^8	3.92×10^6	2.95×10^{14}
$\langle 3, 4, 1, 2, 6, 5 \rangle_1$	9.99×10^5	1.89×10^5	2.69×10^8	2.39×10^8	6.25×10^6	5.93×10^{14}

Table 3.1: Optimal MZM labeling configurations for two-sided $\langle 3, 4, 1, 2, 6, 5 \rangle_2$ and one-sided hexon architectures $\langle 1, 2, 6, 3, 4, 5 \rangle_1$, $\langle 3, 4, 1, 2, 6, 5 \rangle_1$ when using forced-measurement methods. The difficulty weights or geometric average of weights are reported for the gates: the Hadamard gate H , the set of single-qubit Clifford gates C_1 , the controlled-not gate $C(X)$, the set of controlled-Pauli gates $C(P)$, the W gate, and the SWAP gate. The weights of the two-qubit gates are averaged over the four connectivity directions.

Configuration	$[H]$	$[C_1]$	$[C(X)]$	$[C(P)]$	$[W]$	$[SWAP]$
$\langle 3, 4, 1, 2, 6, 5 \rangle_2$	1.76×10^2	5.44×10^2	4.20×10^3	4.20×10^3	1.05×10^3	7.42×10^4
$\langle 1, 2, 6, 3, 4, 5 \rangle_1$	5.13×10^1	9.66×10^1	4.25×10^3	4.42×10^3	1.10×10^3	1.18×10^6
$\langle 3, 4, 1, 2, 6, 5 \rangle_1$	8.17×10^1	1.16×10^2	4.78×10^3	4.59×10^3	1.39×10^3	1.48×10^6

Table 3.2: Optimal MZM labeling configurations for two-sided $\langle 3, 4, 1, 2, 6, 5 \rangle_2$ and one-sided hexon architectures $\langle 1, 2, 6, 3, 4, 5 \rangle_1$, $\langle 3, 4, 1, 2, 6, 5 \rangle_1$ when using Majorana-Pauli tracking methods. The difficulty weights or geometric average of weights are reported for the Pauli cosets of gates: the Hadamard gate H , the set of single-qubit Clifford gates C_1 , the controlled-not gate $C(X)$, the set of controlled-Pauli gates $C(P)$, the W gate, and the SWAP gate. The weights of the two-qubit gates are averaged over the four connectivity directions.

tions.

In Table 3.1, we present a summary of the minimal difficulty weights of gates for the case when forced-measurement methods (including procrastination) are being utilized for the mentioned configurations. In Table 3.2, we present a summary of the minimal difficulty weights of Pauli cosets of gates for the case when Majorana-Pauli tracking methods are being utilized for the mentioned configurations. Details of the measurement-only sequences and corresponding difficulty weights for the specific gates or Pauli cosets of gates can be found in Appendix A.3. We also provide the detailed Pauli gate corrections that arise for the presented

optimal measurement-only gate sequences when using Majorana-Pauli tracking methods.

3.5.4 Comparative Analysis

The methods in this thesis can be used to compare different approaches and architectures to determine preferences between them. Here, we discuss some of the comparative analyses that can be made.

Measurements: forced vs. tracked It is clear without a detailed analysis that utilizing the Majorana-Pauli tracking methods will be more efficient than utilizing forced-measurement methods. Our optimization analysis serves to more precisely quantify the difference, when such a comparison is desired. This can be done for our demonstration by comparing the results in Tables 3.1 and 3.2, which exhibit substantial benefit for using tracking methods.

Scalable architectures: one-sided hexons vs. two-sided hexons It will be important to eventually determine which scalable architectures are preferable. Our methods can help this assessment, once sufficient experimental data is collected for all architectures under consideration to provide an accurate comparison between the different options. (We emphasize that the difficulty weight factors w_c , w_t , w_a , and $f(N)$ might even differ between different architectures.) An important aspect of this comparison is also knowing how the quantum computing device will be utilized, i.e. which gates are relevant to the optimization problem. This can already be observed in the results of our demonstration (with the caution that the speculative weight factors were assumed to be identical for one-sided and two-sided hexon architectures).

For example, in the case where tracking methods are utilized, Table 3.2 shows that the one-sided hexon architecture has a notable advantage for single-qubit gates, but that the two-sided hexon architecture has a slight advantage for controlled-Pauli gates and a major advantage for SWAP gates.

Measurement-only gate synthesis: measurements vs. gates/braids The primary premise of this thesis is that, for measurement-only topological quantum computing, there will be a significant benefit by optimizing gate synthesis with respect to the physical measurements, rather than optimizing with respect to a generating set of gates or braiding operators, each of which is implemented through a measurement-only sequence. In order to make this benefit quantitative, we perform a similar analysis using the “natural” generating set of Clifford gates $\langle S, H, C(Z) \rangle$ or braiding gates $\langle S, B, W \rangle$, where the difficulty weights of these generators are determined by their optimal measurement-only sequence realizations. The detailed comparison is presented in the tables in Appendix A.3. Here, we summarize the comparison for the case where Majorana-Pauli tracking methods are used for two-sided hexon architectures in Table 3.3 and for one-sided hexon architectures in Table 3.4. The benefit is even more dramatic when forced-measurement methods are utilized.

Generating set	$[H]$	$[C_1]$	$[C(X)]$	$[C(P)]$	$[SWAP]$	$[W]$
$\langle \Pi_s^{(jk)}, \Pi_s^{(jk;l'm')} \rangle$	1.76×10^2	5.44×10^2	4.20×10^3	4.20×10^3	7.42×10^4	1.05×10^3
$\langle S, H, C(Z) \rangle$	1.76×10^2	1.10×10^4	1.30×10^8	1.30×10^8	2.21×10^{24}	1.30×10^8
$\langle S, B, W \rangle$	6.88×10^6	1.33×10^4	4.98×10^{16}	1.38×10^{12}	1.23×10^{50}	1.05×10^3

Table 3.3: Difficulty weights of Pauli cosets of gates for the case where Majorana-Pauli tracking methods are utilized for two-sided hexon architectures with the $\langle 3, 4, 1, 2, 6, 5 \rangle$ labeling configuration. We compare the weights for gates synthesized from the generating sets of operations given by MZM measurements $\langle \Pi_s^{(jk)}, \Pi_s^{(jk;l'm')} \rangle$, Clifford gates $\langle S, H, C(Z) \rangle$, or braiding operations $\langle S, B, W \rangle$, respectively.

Generating set	$[H]$	$[C_1]$	$[C(X)]$	$[C(P)]$	$[SWAP]$	$[W]$
$\langle \Pi_s^{(jk)}, \Pi_s^{(jk;l'm')} \rangle$	5.13×10^1	9.66×10^1	4.25×10^3	4.42×10^3	1.18×10^6	1.10×10^3
$\langle S, H, C(Z) \rangle$	5.13×10^1	1.20×10^3	1.12×10^7	1.12×10^7	1.40×10^{21}	1.12×10^7
$\langle S, B, W \rangle$	1.70×10^5	1.44×10^3	3.19×10^{13}	1.04×10^{10}	3.25×10^{40}	1.10×10^3

Table 3.4: Difficulty weights of Pauli cosets of gates for the case where Majorana-Pauli tracking methods are utilized for one-sided hexon architectures with the $\langle 1, 2, 6, 3, 4, 5 \rangle$ labeling configuration. We compare the weights for gates synthesized from the generating sets of operations given by MZM measurements $\langle \Pi_s^{(jk)}, \Pi_s^{(jk;l'm')} \rangle$, Clifford gates $\langle S, H, C(Z) \rangle$, or braiding operations $\langle S, B, W \rangle$, respectively.

3.6 Final Remarks

The methods introduced in this chapter can be applied more generally to topological quantum computation with other non-Abelian anyons or defects. For example, the difficulty analysis and optimization can be applied for different and mixed architectures, such as tetron, octons, etc., systems with different topological orders, and to other measurement-based operations, such as the injection of non-Clifford gates.

The procrastination and tracking methods can only be applied when the measurement outcomes correspond to fusion channels that are Abelian [13, 93], e.g. for Ising anyons, MZMs, and Parafermions (parafermionic zero modes). When fusion channels may be non-Abelian,

leaving the corresponding projectors in a measurement-only sequence of measurements will eventually lead to measurements extracting information regarding the computational state, (at least partially) collapsing it. Thus, when the measurement have non-Abelian fusion channels, one must use forced-measurement protocols to ensure all the projection channels are Abelian.

Acknowledgements

We thank M. Beverland, N. Delfosse, J. Haah, T. Karzig, C. Knapp, D. Pikulin, and M. P. da Silva for enlightening discussions.

Chapter 4

Optimal stabilizer measurements for measurement-only MZM based surface code

Despite the topological protection, a scalable quantum computer built from topological qubits will still require error correction to achieve the desired logical error rates for nontrivial quantum computation. The surface code [27], which is a topological quantum error correction code in the broader class of stabilizer codes, represents one of the most promising proposals for scalable quantum error correction. Generally, stabilizer codes map very favorably onto Majorana-based quantum computers [52, 73, 88, 89]; however, the surface code requires measuring products of four Pauli operators, which can be challenging. In this chapter we focus on optimizing the stabilizer measurements needed to implement the surface code. Starting from

standard techniques to do these measurements using ancillary qubits, we use the methods developed in Ch. 3 to propose an optimal measurement sequence that implements the desired operations in an array of Majorana-based qubits.

Part of the results presented in this chapter were previously published in “Optimizing Clifford gate generation for measurement-only topological quantum computation with Majorana zero modes” by Alan Tran, Alex Bocharov, Bela Bauer, and Parsa Bonderson, SciPost-Phys.8.6.091 (2020) [86] with minor modification. Licensed under CC BY 4.0.

4.1 Overview and motivation

An important class of error correcting codes are stabilizer codes. In error correcting codes, the *logical* qubit state is encoded into a carefully chosen subspace of the Hilbert space of many *physical* qubits. The extra redundancy lends to the error correction properties. In the case of stabilizer codes, this subspace is defined as the simultaneous $+1$ eigenspace of some number of commuting multi-qubit Pauli operators, referred to as the stabilizers. Errors are detected by repeatedly measuring the stabilizers; deviations from the expected outcome of $+1$ indicate errors. The collection of errors can be classified into different syndromes of the code which can be decoded and corrected [24, 25].

Within the class of stabilizer codes, the surface code [27] is one of the most promising proposals for large-scale error correction. The simplest realization is defined on a rectangular lattice of qubits, whose plaquettes are divided into two sublattices in a checkerboard pattern. There is one stabilizer for each plaquette: for one sublattice, it is given by the product of the

four Pauli X operators of the data qubits around a plaquette; for the other sublattice, it is given by the corresponding product of four Pauli Z operators.

Since the measurement of Pauli operators translates into topologically protected parity measurements in the MZM-based architectures discussed in the previous chapter (Ch. 3), Pauli stabilizer codes map ideally onto such architectures. Some of the ideas for using these architectures build upon those of Ref. [73], which suggested an implementation of a particular stabilizer code using MZMs, but relies on a physical 8-MZM measurement involving four neighboring topological islands. This approach was generalized in Ref. [57], which however still relies on higher-weight measurements for the implementation of stabilizers. Since such measurements are likely to be prohibitively difficult to implement, it is worth seeking a MZM-based surface code implementation that utilizes physical measurements involving at most 4 MZMs (two topological islands) at a time.

In most practical proposals for implementing the surface code, the measurement of the product of four Pauli operators is achieved by adding an additional ancillary qubit to each plaquette, entangling it in a particular way with its adjacent data qubits, and finally performing a single-qubit measurement on the ancillary qubit. In this section, we propose a specific MZM-based architecture layout, sketched in Fig. 4.1, that can be used to implement precisely such a scheme in an efficient and topologically protected fashion. The required measurements are all 2-MZM or 4-MZM measurements on single and nearest-neighbor islands, respectively, and are natural to carry out in the architecture. We will use the techniques for optimizing compilations introduced in Ch. 3.5 as well as App. A.2 to obtain an optimized measurement sequence

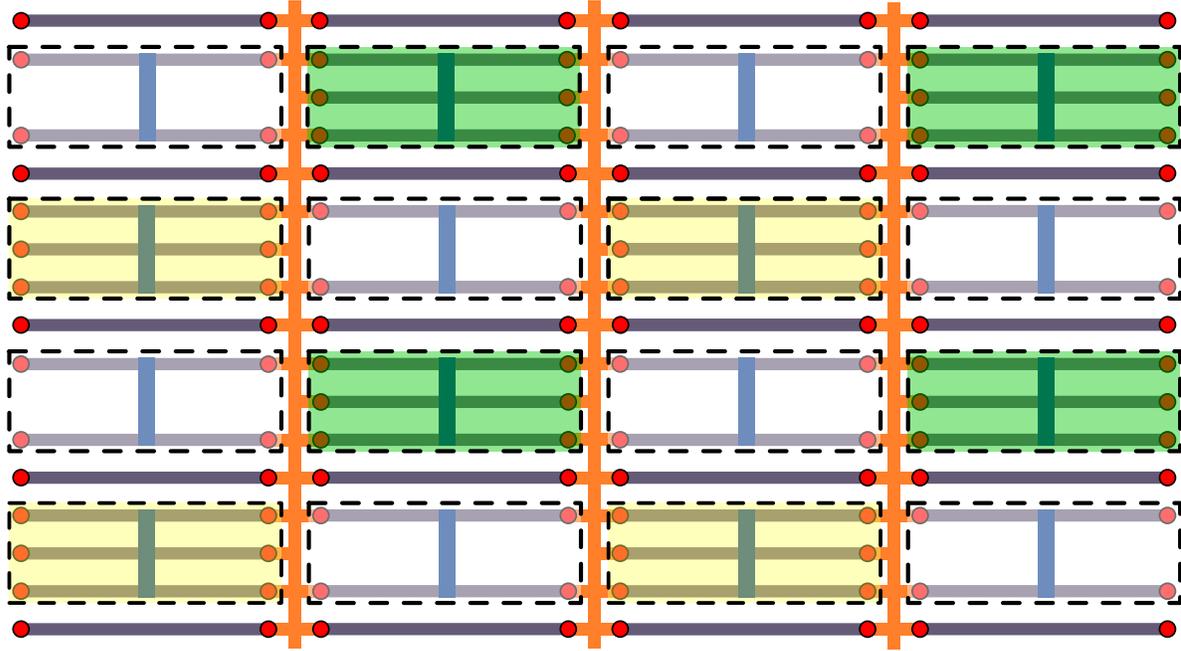


Figure 4.1: A proposed architecture layout for implementing a surface code. The tetrons (shown enclosed in unshaded dashed rectangles) play the role of data qubits in the surface code, while hexons (shown enclosed in shaded dashed rectangles) play the role of ancillary qubits used to facilitate the stabilizer measurement. The yellow and green shading of rectangles correspond to the M_X and M_Z hexons, respectively, which facilitate measuring the $X^{\otimes 4}$ and $Z^{\otimes 4}$ stabilizers on their nearest-neighbor data qubits. Coherent links (shown between vertical neighboring islands) are a necessary aid to enable the full set of Pauli measurements.

implementing the stabilizer measurements.

The proposed architecture makes use of an additional MZM-based qubit design referred to as a tetron. As opposed to a hexon, which has 6 MZMs on a single island, a tetron has 4 MZMs on a single island. Therefore, its state space in a fixed total parity sector of the island is two-dimensional instead of the four-dimensional state space of a hexon. As such, it cannot be used to perform Clifford operations on its own. However, pairs of tetrons together can be used for such an end, as discussed in Ref. [41]. We will see that for the purpose of implementing a surface code, a mixed architecture of tetron and hexon islands is sufficient

and has certain advantages. In our proposal, tetrons will play the role of data qubits in the surface code, while the hexons are used as ancillary qubits that facilitate unitary operations and the implementation of the $X^{\otimes 4}$ and $Z^{\otimes 4}$ stabilizer measurements. In order to avoid confusion between the term “ancillary qubit” used in reference to the second qubit encoded within a hexon and in reference to the qubits used to facilitate stabilizer measurements in the surface code, we will refer to the ancillary qubits of the surface code explicitly as ancillary hexons (or, when generalizing to tetrons or hexons, as ancillary islands). We refer to the ancillary hexons that facilitate measurements of the $X^{\otimes 4}$ stabilizers as “ M_X -hexons” and the ones that facilitate measurements of the $Z^{\otimes 4}$ stabilizers as “ M_Z -hexons.”

It is worth pointing out that the tetrons can trivially be replaced by hexons in the proposed architecture; one can simply ignore the extra degrees of freedom, or even utilize them for a physically denser surface code. Depending on how logical gate operations are performed, it may be favorable to utilize hexons for the data qubits. For example, if transversal gates are used, the ability of hexons to perform single-qubit Clifford gates using only 2-MZM measurements may be useful. We provide example measurement-sequences compiling to a few useful gates for this in App. B.1.

4.2 Measurement circuit and example compilation

The two surface code stabilizers are measured as follows.

For the $X^{\otimes 4}$ stabilizers, the protocol is:

1. Initialize a M_X -hexon qubit into the $|X = +1\rangle$ state, with the hexon’s ancillary qubit in

an arbitrary, but definite state (i.e. into a $|i\gamma_1\gamma_6 = +1, i\gamma_3\gamma_4 = p_{34}\rangle$ state).

2. Apply the sequence of CNOTs: $C(X)_{(h_x, t_4)}C(X)_{(h_x, t_3)}C(X)_{(h_x, t_2)}C(X)_{(h_x, t_1)}$, controlled on the M_X -hexon (labeled h_x) and targeting the four nearest-neighboring tetrons (labeled t_j).
3. Measure the M_X -hexon qubit in the X -basis (i.e. measure $i\gamma_1\gamma_6$).

The effect of this sequence of steps is a measurement of $X^{\otimes 4}$ of the four data tetrons. The outcome of the final measurement (in step 3), is the outcome of this stabilizer measurement.

For the $Z^{\otimes 4}$ stabilizers, the protocol is:

1. Initialize a M_X -hexon qubit into the $|0\rangle$ ($Z = +1$) state, with the hexon's ancillary qubit in an arbitrary, but definite state (i.e. into a $|i\gamma_1\gamma_2 = +1, i\gamma_3\gamma_4 = p_{34}\rangle$ state).
2. Apply the sequence of CNOTs: $C(X)_{(t_4, h_z)}C(X)_{(t_3, h_z)}C(X)_{(t_2, h_z)}C(X)_{(t_1, h_z)}$, controlled on the four nearest-neighboring tetrons (labeled t_j) and targeting the M_Z -hexon (labeled h_z).
3. Measure the M_Z -hexon in the Z -basis (i.e. measure $i\gamma_1\gamma_2$).

The effect of this sequence of steps is a measurement of $Z^{\otimes 4}$ of the four data tetrons. The outcome of the final measurement (in step 3), is the outcome of this stabilizer measurement.

In order to accomplish this as efficiently as possible, we search for optimized compilations of these circuits. Since steps 1 and 3 are simply measurements (two needed for step 1 and one for step 3), they leave no room for optimizing. Thus, we need only focus on step 2, and

search for optimal measurement sequences realizing the two sequences of CNOT gates, which we denote as

$$L_X = C(X)_{(h_x, t_4)} C(X)_{(h_x, t_3)} C(X)_{(h_x, t_2)} C(X)_{(h_x, t_1)} \quad (4.1)$$

$$L_Z = C(X)_{(t_4, h_z)} C(X)_{(t_3, h_z)} C(X)_{(t_2, h_z)} C(X)_{(t_1, h_z)}. \quad (4.2)$$

The search space for a system of one hexon and four tetrons is quite large for a brute-force search. Another way to proceed is by first finding measurement sequences compiling the individual CNOT gates $C(X)_{(h,t)}$ and $C(X)_{(t,h)}$, build from these to construct a full measurement-only circuit for L_X and L_Z , and finally attempt to reduce the length of the sequence with the methods of Appendix A.2.

In this case, it is helpful to find measurement sequence compilations by identifying the stabilizers and logical operators in a system comprising a hexon and tetron, and updating them appropriately as a sequence of measurements is performed (as described in Ch. 3.2.3). If the set of stabilizers at the end of a sequence of measurements is the same as the initial set of stabilizers, the sequence will yield a logical gate that is determined by the transformation of the logical Pauli operators¹. A given measurement sequence will compile to the target gate $C(X)_{(a,b)}$ if the logical Pauli operators transform the same way as they do under conjugation

¹When a measurement sequence is between different stabilizers, the result is an operation which transforms the initial stabilizer group to the final one and then applies a logical Clifford. The logical Clifford will depend on how the logical Pauli operators are defined within both the initial and final code spaces.

by $C(X)_{(a,b)}$, that is

$$\begin{array}{ccc}
 X_a I_b & & X_a X_b \\
 Z_a I_b & \xrightarrow{C(X)_{(a,b)}} & Z_a I_b \\
 I_a X_b & & I_a X_b \\
 I_a Z_b & & Z_a Z_b
 \end{array} . \tag{4.3}$$

Recall from Ch. 3.2.3 that a hexon encodes one logical qubit in six MZMs and is stabilized by the total parity of the island $i^3 \gamma_1 \gamma_2 \gamma_3 \gamma_4 \gamma_5 \gamma_6 = +1$ and restricted to a further ancillary parity sector, which we choose to initialize as $i \gamma_3 \gamma_4 = p_{34} = \pm 1$. The set of generators for the initial hexon stabilizer group is therefore $S_{\text{hex}} = \langle i^3 \gamma_1 \gamma_2 \gamma_3 \gamma_4 \gamma_5 \gamma_6, i \gamma_3 \gamma_4 \rangle$. The corresponding logical Pauli operators (acting on the logical qubit) for a hexon island are $\bar{X}_{\text{hex}} = [i \gamma_1 \gamma_6]$, $\bar{Y}_{\text{hex}} = [-i \gamma_2 \gamma_6]$, and $\bar{Z}_{\text{hex}} = [i \gamma_1 \gamma_2]$, where the equivalence classes contain all parity operators related by multiplication by a stabilizer. The 2-MZM parity operators for hexons can be mapped back to Pauli operators via Eq. (3.7).

Similarly, a tetron encodes one logical qubit in four MZMs and is stabilized by the total parity of the island $i^2 \gamma_1 \gamma_2 \gamma_3 \gamma_4$. The stabilizer group is therefore $S_{\text{tet}} = \langle i^2 \gamma_1 \gamma_2 \gamma_3 \gamma_4 \rangle$. The corresponding logical Pauli operators are $\bar{X}_{\text{tet}} = [i \gamma_1 \gamma_4]$, $\bar{Y}_{\text{tet}} = [-i \gamma_2 \gamma_4]$, and $\bar{Z}_{\text{tet}} = [i \gamma_1 \gamma_2]$.

The 2-MZM pairty operators for tetrons can be mapped back to Pauli operators via

$$\begin{aligned}
 i\gamma_1\gamma_2 &= Z, & i\gamma_1\gamma_3 &= Y, & i\gamma_1\gamma_4 &= X, \\
 i\gamma_2\gamma_3 &= X, & i\gamma_2\gamma_4 &= -Y, \\
 i\gamma_3\gamma_4 &= Z.
 \end{aligned} \tag{4.4}$$

When a measurement of the operator Γ_M is performed, the stabilizers and logical operators are updated according to the rules in Ch. 3.2.3. In discussing stabilizers for the purposes of gate synthesis, we can assume the total parity of each island is always fixed (this is only violated by quasiparticle poisoning errors [48, 49, 88, 89] that flip the parity of an island, which we neglect), so the stabilizers corresponding to total island parity ($i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6 = +1$ for hexons and $i^2\gamma_1\gamma_2\gamma_3\gamma_4 = +1$ for tetrons) will be left implicit.

An example of a measurement sequence realizing $C(X)_{(h,t)}$ is the following: We use the

Step	Measurement of	Stabilizer	$\bar{X}_{\text{hex}}\bar{I}_{\text{tet}}$	$\bar{Z}_{\text{hex}}\bar{I}_{\text{tet}}$	$\bar{I}_{\text{hex}}\bar{X}_{\text{tet}}$	$\bar{I}_{\text{hex}}\bar{Z}_{\text{tet}}$
0	—	34 $\circ\circ$	16 $\circ\circ$	12 $\circ\circ$	$\circ\circ$ 14	$\circ\circ$ 12
1	46 14	46 14	25 $\circ\circ$	12 $\circ\circ$	$\circ\circ$ 14	34 12
2	56 $\circ\circ$	56 $\circ\circ$	13 14	12 $\circ\circ$	$\circ\circ$ 14	34 12
3	46 $\circ\circ$	46 $\circ\circ$	13 14	12 $\circ\circ$	$\circ\circ$ 14	12 12
4	34 $\circ\circ$	34 $\circ\circ$	25 14	12 $\circ\circ$	$\circ\circ$ 14	12 12
			$\bar{X}_{\text{hex}}\bar{X}_{\text{tet}}$	$\bar{Z}_{\text{hex}}\bar{I}_{\text{tet}}$	$\bar{I}_{\text{hex}}\bar{X}_{\text{tet}}$	$\bar{Z}_{\text{hex}}\bar{Z}_{\text{tet}}$

shorthand $ab|cd$ to mean $(i\gamma_a\gamma_b)_{\text{hex}} \otimes (i\gamma_c\gamma_d)_{\text{tet}}$ and $\circ\circ$ to mean that the corresponding hexon or tetron is not involved. As mentioned, the overall island parity stabilizers are left implicit, since they are assumed to be fixed throughout the process. Furthermore, we do not explicitly account for signs in the stabilizers or logical operators. For example, $(i\gamma_1\gamma_2)(i\gamma_1\gamma_3) = -i\gamma_2\gamma_3$, but would be recorded as 23. The effect of these signs is to alter the compiled gate by an overall

Pauli operator, which can be determined by Pauli tracking, as discussed in Ch. 3.4.

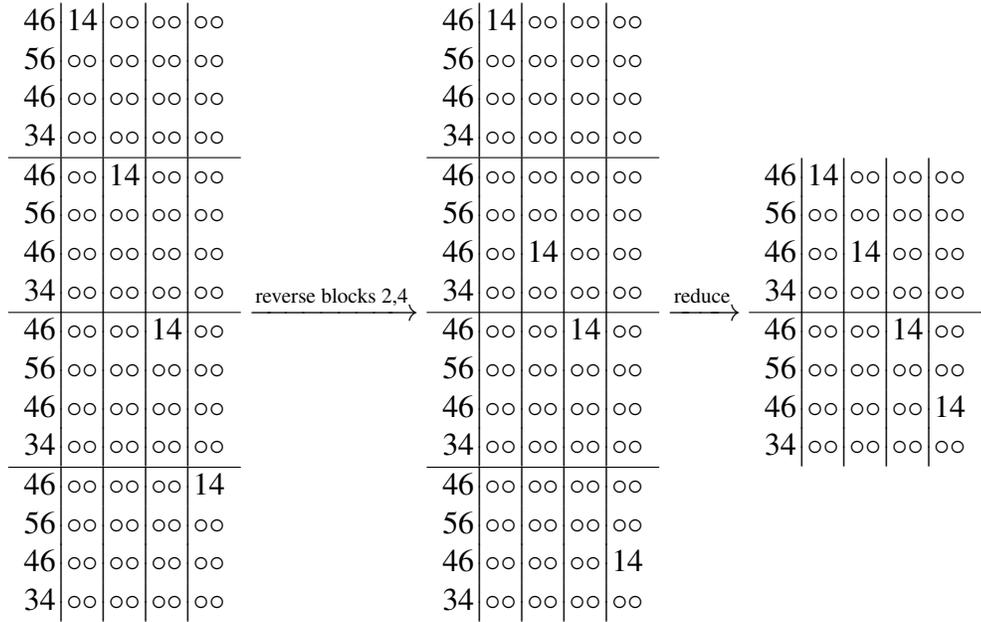
We see that the effect of this measurement sequence is to apply a $C(X)_{(h,t)}$ gate controlled on the hexon and targeting a tetron, up to a Pauli operator. We can build up the full L_X circuit by concatenating variations of this circuit for each of the four tetrons. Then we can improve the efficiency by using the sequence manipulation and reduction tools developed in Appendix A.2. The same can be done for $C(X)_{(t,h)}$ gates and L_Z circuits.

More specifically, we know that reversing a measurement sequence yields the inverse of the compiled gate. Since $C(X)^\dagger = C(X)$, we can freely reverse the corresponding measurement sequence (we assume an initialization of $i\gamma_3\gamma_4$, so all sequences implicitly start with a $i\gamma_3\gamma_4$ stabilizer that we leave implicit from now on) Immediate repetitions of the same mea-

$$\begin{array}{ccc} 46 \left| 14 & & 46 \left| \circ\circ \\ 56 \left| \circ\circ & \xleftrightarrow{\text{reverse}} & 56 \left| \circ\circ \\ 46 \left| \circ\circ & & 46 \left| 14 \\ 34 \left| \circ\circ & & 34 \left| \circ\circ \end{array} \cdot$$

surement can be reduced, since $\Pi_r^{(M)}\Pi_s^{(M)} = \delta_{r,s}\Pi_s^{(M)}$. Furthermore, triplets of measurements of M_1 , M_2 , and then M_1 , where $\{\Gamma_{M_1}, \Gamma_{M_2}\} = 0$ can be reduced, since $\Pi_r^{(M_1)}\Pi_s^{(M_2)}\Pi_t^{(M_1)} \propto (\delta_{r,t} + s\Gamma_{M_2}\delta_{-r,t})\Pi_t^{(M_1)}$ for such measurements.

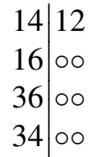
A full L_X circuit can then be compiled and reduced in the following way: Here, the first column corresponds to the hexon and the next four columns correspond to each of the neighboring tetrons. This reduces the naïve length 16 measurement sequence to a length 8 measurement sequence, where each tetron is involved in only a single 4-MZM measurement. We conjecture that this is the minimum number of measurements required to implement L_X . (It is clearly the



minimum number of 4-MZM measurements required.)

The same steps can be applied to construct an optimized implementation of the L_Z circuit.

The starting point is a single $C(X)_{(t,h)}$, which can be implemented by



Following the same steps as for the L_X circuit, i.e. appropriately combining four $C(X)_{(t,h)}$ gates and reducing them yields the following implementation of L_Z :

14	12	oo	oo	oo
16	oo	oo	oo	oo
14	oo	12	oo	oo
34	oo	oo	oo	oo
14	oo	oo	12	oo
16	oo	oo	oo	oo
14	oo	oo	oo	12
34	oo	oo	oo	oo

This also reduces the naïve length 16 measurement sequence to a length 8 sequence, where each tetron is involved in only one 4-MZM measurement.

4.3 Circuit optimization

We can apply cost functions, such as the difficulty weight assignment scheme of Ch. 3.2.4, to find optimized encodings of hexons, tetrons and optimized L_X and L_Z circuit compilations, similar to the optimizations performed in the previous chapter.

For the sequence optimization, we recognize that the L_X and L_Z circuits naturally divide into two segments, each of which involves two applications of $C(X)$ that can be manipulated as a pair and reduced. With this in mind, we first search for all length-4 measurement sequences that alternate between 4-MZM measurements and 2-MZM measurements (each 4-MZM measurement is pairing the hexon with a tetron in a different direction on the lattice, either upwards, rightwards, leftwards, or downwards) and which compile to $C(X)_{(h,t_j)}C(X)_{(h,t_k)}$ and $C(X)_{(t_j,h)}C(X)_{(t_k,h)}$, up to overall Pauli factors. There are 8 possible MZM pairs that can be chosen for the hexon for each measurement step along with $\binom{4}{2} = 6$ MZM pairs for the selected tetron. The search space for a 4-MZM, 2-MZM, 4-MZM, 2-MZM measurement se-

quence with the constraint that the final 2-MZM measurement is on $i\gamma_3\gamma_4$ of the hexon is therefore over $(8 \times 6) \times (7 \times 24 + 1 \times 48) = 10,368$ measurement-only sequences. For each pair of directions, j and k , we find 64 sequences for $C(X)_{(h,t)}$, and similarly for $C(X)_{(t,h)}$. We then combine these to form measurement-only compilations of L_X and L_Z . This produces a list of all L_X and L_Z circuits obtained through optimized compilations of $C(X)_{(h,t_j)}C(X)_{(h,t_k)}$ and $C(X)_{(t_j,h)}C(X)_{(t_k,h)}$. A search over all length-8 measurement sequences that alternate between 4-MZM and 2-MZM measurements has yet to be carried out; the search space in this case has is over $(48 \times 8)^2 \times 48 \times 9 \times 24 \times 1 = 1,528,823,808$ measurement-only sequences.

As in the case of hexons (see Ch. 3.2.5), the MZMs of tetrons may also be relabeled, reflecting a different encoding choice. We use the analogous notation of $\langle a, b, c, d \rangle$ to denote the labeling configuration of MZMs within a two-sided tetron where the labeling goes counterclockwise from the top-left to the top-right. The next step in the optimization is the following: for each tetron labeling configuration, search over all hexon labeling configurations and record the lowest weight L_X sequence and the hexon configuration for which it is realized, and likewise for L_Z . For each tetron labeling configuration, this gives a M_X -hexon labeling configuration, L_X measurement sequence, and corresponding difficulty weight, as well as a M_Z -hexon labeling configuration, L_Z measurement sequence and weight. Defining the tetron labeling configuration weight to be the geometric average of its L_X and L_Z weights, we can pick out the best configuration.

Doing this for the same choice of weights as in the previous section, we find eight tetron labeling configurations that have difficulty weight 1.67×10^{10} . This is clearly an improvement

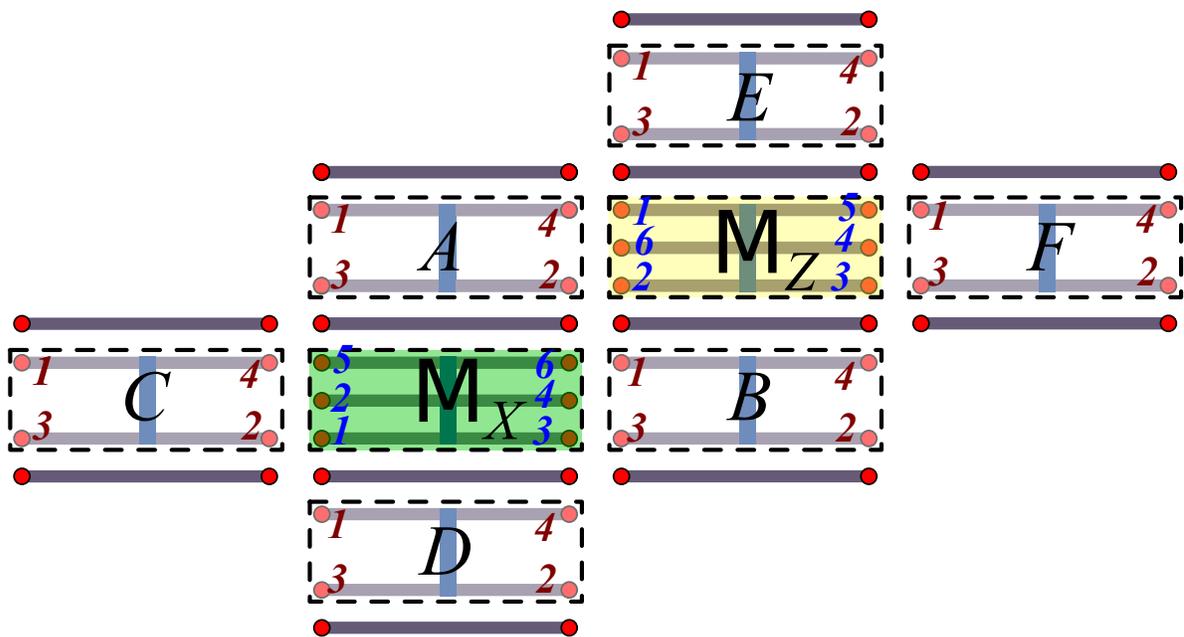


Figure 4.2: An example of an optimized labeling configuration for the proposed architecture. The tetrons (labeled A, B, C, D, E, F) all use the $\langle 1, 3, 2, 4 \rangle$ configuration, the M_X hexon (green) uses the $\langle 5, 2, 1, 3, 4, 6 \rangle$ configuration, and the M_Z hexon (yellow) uses the $\langle 1, 6, 2, 3, 4, 5 \rangle$ configuration.

over the naïve concatenation of four $C(X)$ measurement sequences, which has a total difficulty on weight on the order of 10^{14} (see Table 3.2). An example of an optimized labeling configuration is shown in Fig. 4.2, where the tetrons using the $\langle 1, 3, 2, 4 \rangle$ configuration, the M_X hexons using the $\langle 5, 2, 1, 3, 4, 6 \rangle$ configuration, and the M_Z hexons using the $\langle 1, 6, 2, 3, 4, 5 \rangle$ configuration. The associated optimal measurement sequences are where the first column is

$$L_X = \begin{array}{c|c|c|c|c|c|c|c} M_X & M_Z & A & B & C & D & E & F \\ \hline 24 & \text{oo} & 23 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ 12 & \text{oo} \\ 13 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & 14 & \text{oo} & \text{oo} \\ 34 & \text{oo} \\ 13 & \text{oo} & \text{oo} & 23 & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ 12 & \text{oo} \\ 13 & \text{oo} & \text{oo} & \text{oo} & 23 & \text{oo} & \text{oo} & \text{oo} \\ 34 & \text{oo} \end{array}, \quad L_Z = \begin{array}{c|c|c|c|c|c|c|c} M_X & M_Z & A & B & C & D & E & F \\ \hline \text{oo} & 13 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} & 34 \\ \text{oo} & 16 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ \text{oo} & 13 & 34 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ \text{oo} & 34 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ \text{oo} & 14 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & 12 & \text{oo} \\ \text{oo} & 16 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ \text{oo} & 36 & \text{oo} & 12 & \text{oo} & \text{oo} & \text{oo} & \text{oo} \\ \text{oo} & 34 & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} & \text{oo} \end{array}$$

the M_X hexon, the second column is the M_Z hexon, and the remaining columns correspond to tetrons A, B, C, D, E, F as shown, for example, in Fig. 4.2.

4.4 Boundary circuits

The stabilizer measurements at the boundaries of a surface code will involve fewer data qubits than in the bulk, so we consider these for completeness. For the case of stabilizer measurements involving two data qubits, the tetrons may be measured directly, or via the sequence reduced $C(X)$ circuits studied above, or through the use of GHZ states as described in [57]. For the case of stabilizer measurements involving three data qubits, a direct measurement is hypothesized to be significantly difficult due to the number of islands involved. Further, the se-

quence reduction techniques utilized in the previous section are not applicable for this case. A brute-force search may, however, be performed for length six measurement sequences that alternate between 4-MZM and 2-MZM measurements. An example of a measurement sequence for applying three $C(X)_{(h,t)}$ operations (i.e. a three data qubit version of L_X) found in this way is:

$$\begin{array}{c|c|c|c}
 13 & 23 & \circ\circ & \circ\circ \\
 12 & \circ\circ & \circ\circ & \circ\circ \\
 15 & \circ\circ & 23 & \circ\circ \\
 56 & \circ\circ & \circ\circ & \circ\circ \\
 45 & \circ\circ & \circ\circ & 23 \\
 34 & \circ\circ & \circ\circ & \circ\circ
 \end{array}$$

4.5 Interleaving

To further optimize a stabilizer measurement cycle on the surface code, we can attempt to interleave the measurement sequences so that the X and Z stabilizers are measured at the same time thus reducing the total number of time steps. For measurement sequences

$$\begin{array}{c}
 a_1 \qquad b_1 \qquad c_1 \\
 a_2 \qquad b_2 \qquad c_2 \\
 L_a = \cdot, \quad L_b = \cdot, \quad L_c = \cdot, \dots \\
 \cdot \qquad \cdot \qquad \cdot \\
 \cdot \qquad \cdot \qquad \cdot
 \end{array}$$

where a_j is the j 'th row or measurement prescription in L_a , define their ordered-interleaving as

$$I(\mathbb{L}_a, \mathbb{L}_b, \mathbb{L}_c, \dots) = \begin{array}{c|c|c} a_1 & b_1 & c_1 \\ a_2 & b_2 & \dots \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array}$$

When

$$I(\mathbb{L}_a, \mathbb{L}_b, \mathbb{L}_c, \dots) = I(\sigma(\mathbb{L}_a, \mathbb{L}_b, \mathbb{L}_c, \dots)) \tag{4.5}$$

for all permutations of its arguments σ , then we can write the interleaving more compactly as and perform all measurements in a row simultaneously. In this case we call $\mathcal{I}(\mathbb{L}_a, \mathbb{L}_b, \mathbb{L}_c, \dots)$ a

$$\mathcal{I}(\mathbb{L}_a, \mathbb{L}_b, \mathbb{L}_c, \dots) = \begin{array}{c|c|c} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ \cdot & \cdot & \cdot \dots \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array}$$

valid interleaving.

A necessary further check is to ensure that the simultaneous measurements in a row do not address the same hexons or tetrons . If so, the involved measurements would not be able to be performed in parallel, and an additional time step would be required. We call such events “collisions”. Towards this end, we define two variants of a k -shifted-ordered-interleaving between two sequences

$$I_{k,1}(\mathbb{L}_a, \mathbb{L}_b) = \begin{array}{c} a_1 \\ \cdot \\ \cdot \\ a_k \\ a_{k+1} \\ a_{k+2} \\ \cdot \\ \cdot \\ \cdot \end{array} \left| \begin{array}{c} b_1 \\ \cdot \end{array} \right. , \quad I_{k,2}(\mathbb{L}_a, \mathbb{L}_b) = \begin{array}{c} a_1 \\ \cdot \\ \cdot \\ a_k \\ a_{k+1} \\ a_{k+2} \\ \cdot \\ \cdot \\ \cdot \end{array} \left| \begin{array}{c} b_1 \\ \cdot \end{array} \right. .$$

When $I_{k,1}(\mathbb{L}_a, \mathbb{L}_b) = I_{k,2}(\mathbb{L}_a, \mathbb{L}_b)$, we call this $I_k(\mathbb{L}_a, \mathbb{L}_b)$ and when $I_k(\mathbb{L}_a, \mathbb{L}_b) = I_k(\mathbb{L}_b, \mathbb{L}_a)$, we call this $\mathcal{I}_k(\mathbb{L}_a, \mathbb{L}_b)$ and deem it a valid k -shifted interleaving.

To check whether the full stabilizer measurement cycle is interleavable, it is sufficient to check it for a small patch of the bulk containing 4 M_X hexons, 4 M_Z hexons, and their 16 neighboring tetrons as in Fig. 4.3. Here we further require that

- 1) $\mathbb{L}_{X_{abcd}} = \mathcal{I}(\mathbb{L}_{X_a}, \mathbb{L}_{X_b}, \mathbb{L}_{X_c}, \mathbb{L}_{X_d})$ and $\mathbb{L}_{Z_{abcd}} = \mathcal{I}(\mathbb{L}_{Z_a}, \mathbb{L}_{Z_b}, \mathbb{L}_{Z_c}, \mathbb{L}_{Z_d})$ are valid interleavings and equivalent to doing the stabilizers sequentially, $\mathbb{L}_{X_{abcd}} = \mathbb{L}_{X_a} \mathbb{L}_{X_b} \mathbb{L}_{X_c} \mathbb{L}_{X_d}$, $\mathbb{L}_{Z_{abcd}} = \mathbb{L}_{Z_a} \mathbb{L}_{Z_b} \mathbb{L}_{Z_c} \mathbb{L}_{Z_d}$.
- 2) $\mathcal{I}(\mathbb{L}_{X_{abcd}}, \mathbb{L}_{Z_{abcd}})$ is a valid interleaving and $\mathcal{I}(\mathbb{L}_{X_{abcd}}, \mathbb{L}_{Z_{abcd}}) = \mathbb{L}_{X_{abcd}} \mathbb{L}_{Z_{abcd}}$.

Performing this check for the optimized sequences given in Eqn. 4.5, we see that the interleaving is valid but not collisionless. There is a collision on tetron 4 between \mathbb{L}_{X_a} and \mathbb{L}_{Z_a} (recall that the 4-MZM measurement direction orders are up-down-right-left and right-left-up-down for M_X and M_Z hexons respectively). However, one can further check that the shifted interleaving $\mathcal{I}_k(\mathbb{L}_{X_{abcd}}, \mathbb{L}_{Z_{abcd}})$ is valid and collisionless for $k = 1, 3, 4$. At a shift of $k = 4$, the

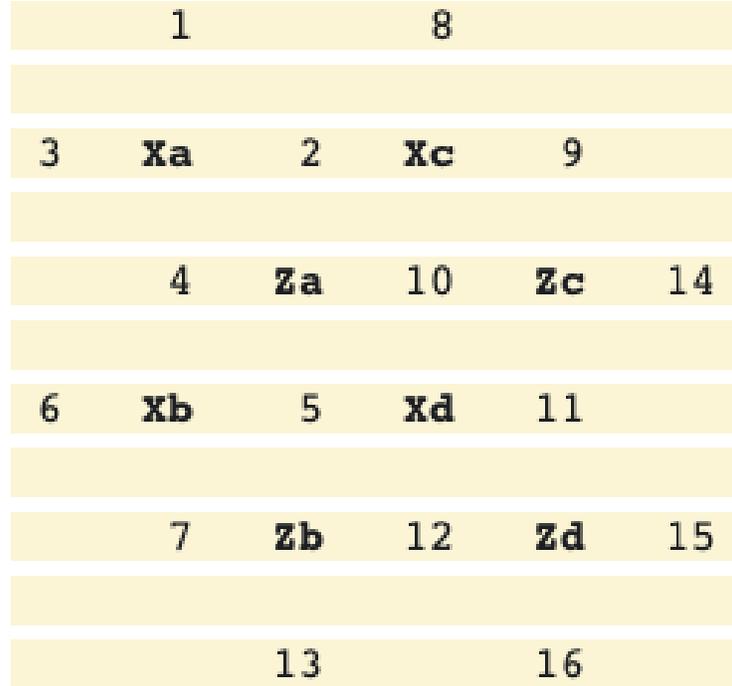


Figure 4.3: A small patch of the square lattice for checking interleaving. Hexons are $X_a, X_b, X_c, X_d, Z_a, Z_b, Z_c, Z_d$ and tetrons are labeled from 1-16.

4-MZM measurement for both M_X and M_Z hexons are towards the same direction at each time step hence no collisions occur. For $k = 1, 3$, the 4-MZM measurements for M_X occur when a 2-MZM measurement occurs for M_Z and vice versa. Taking into account the preparation and measurement part of the circuits, then one cycle of stabilizer measurements would take $11 + k$ timesteps and D cycles would take $11D + k$ time steps.

However, there are physical interferences to consider as well. Namely, whether quantum dots or semiconducting junctions would overlap when physically performing measurements. If there are such obstacles then simultaneous measurements would not be possible and additional time steps would be required. Indeed, there are physical overlaps for our current example. One

way to remedy this with only a modest increase in the difficulty weight.

First note that:

- 1) All 2-MZM measurements are on the same side of a hexon. This means that no coherent links are required. Likewise, the prepare X and prepare Z measurements can be chosen to be on the side as well. For example, the M_X hexon in the configuration $\langle 5, 2, 1, 3, 4, 6 \rangle$ has the measurement of $i\gamma_3\gamma_4$ on the same side and although $i\gamma_1\gamma_6$ spans the left and right, $i\gamma_2\gamma_5$ can be used instead and is on the left. So, a prepare X sequence can be measure $i\gamma_3\gamma_4$ then measure $i\gamma_2\gamma_5$; similarly for preparing Z .
- 2) An odd shift delays the 4-MZM measurements so that they occur when a 2-MZM measurement on the other type of hexon occurs.
- 3) In the tetron picture $\bar{X}_{\text{tet}} = i\gamma_1\gamma_2 = i\gamma_3\gamma_4$ and $\bar{Z}_{\text{tet}} = i\gamma_1\gamma_4 = i\gamma_2\gamma_3$ so 4-MZM measurements involving these pairs can be swapped for a more convenient choice. E.g. measuring $12_{\text{hex}}|12_{\text{tet}}$ is the same as $12_{\text{hex}}|34_{\text{tet}}$.
- 4) We can horizontally or vertically flip a hexon labeling configuration at the cost of modest additional difficulty weight.

With this in mind, we adapt our L_X and L_Z sequences and the labeling configuration as follows:

- 1) Every other column of M_Z hexons is flipped horizontally and every even column of M_X hexons is flipped vertically (Fig. 4.4).

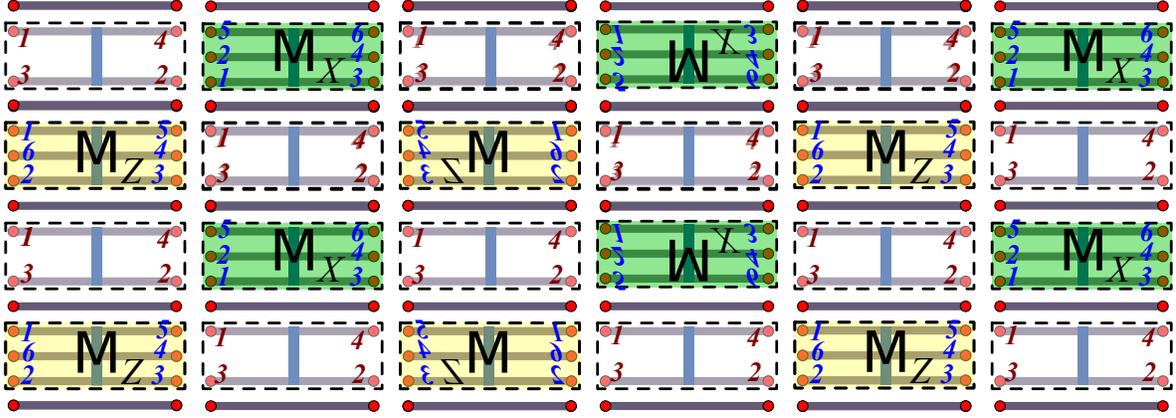


Figure 4.4: Every other column of M_Z hexons is flipped vertically and every even column of M_X hexons is flipped horizontally

- 2) Use a shift of $k = 3$, leading with the L_Z circuit. Also, for the L_X circuit, reverse the order of the up-down part and for the L_Z circuit, also reverse the order of the up-down part but keep it as is for the column where the M_Z hexon is flipped. The overall sequence including preparation and measurement is:

Step	M_X	M'_X	M_Z	M'_Z
1)	13 23 _l	13 14 _l	34 ○○	34 ○○
2)	34 ○○	34 ○○	12 ○○	12 ○○
3)	25 ○○	25 ○○	13 34 _r	13 12 _r
4)	34 ○○	34 ○○	16 ○○	16 ○○
5)	25 ○○	25 ○○	13 34 _l	13 12 _l
6)	13 14 _d	13 14 _d	34 ○○	34 ○○
7)	12 ○○	12 ○○	36 34 _d	14 12 _u
8)	24 23 _u	24 23 _u	16 ○○	16 ○○
9)	34 ○○	34 ○○	14 34 _u	36 12 _d
10)	13 23 _r	13 14 _r	34 ○○	34 ○○
11)	12 ○○	12 ○○	12 ○○	12 ○○
12)	13 23 _l	13 14 _l	34 ○○	34 ○○
13)	34 ○○	34 ○○	12 ○○	12 ○○
14)	25 ○○	25 ○○	13 34 _r	13 12 _r

where M'_X and M'_Z are the vertically and horizontally flipped M_X and M_Z hexons respec-

tively, the notation $ab|cd_x$ means measure jointly $i\gamma_a\gamma_b$ on the hexon and $i\gamma_c\gamma_d$ on the tetron in the x direction from the hexon (either upward, downward, rightward, or leftward), and the grayed out entries correspond to continuations of the measurement cycle measurements (see App. B.2). Further, extra difficulty (relative to the optimized measurement of a single stabilizer, Eq. 4.5) is picked up from flipping the labeling configurations. Namely, for M'_X on step 6 there is an additional factor of $1.25^4 \cdot 1.01^4 \sim 2.5$. Finally, the shift of 3 gives $11D + 3$ time steps for D stabilizer measurement cycles.

Acknowledgements

We thank M. Beverland, N. Delfosse, J. Haah, T. Karzig, C. Knapp, D. Pikulin, and M. P. da Silva for enlightening discussions.

Chapter 5

Measurement-Only Randomized

Benchmarking for MZM Hexon Qubits

As quantum computing ripens towards fruition, the accurate characterization of the underlying qubits and of the operations on those qubits grows more and more crucial. A full characterization of a quantum process is possible through quantum state tomography. However, quantum state tomography is resource intensive with the number of experimental configurations needed growing exponentially in the system size. An alternative is randomized benchmarking (RB) which extracts the average fidelity of a quantum gate set in an efficient manner.

A promising qubit platform makes use of Majorana zero modes (MZM) 1. MZMs carry non-Abelian topological defects which offers topological protection of the qubit itself and topologically protected gate operations via the braiding of these MZMs. Recent experiments in semiconductor-superconductor heterostructures have shown signatures of such MZMs [60].

We establish randomized benchmarking for measurement-only Majorana zero mode qubits composed of six Majorana zero-modes. Such MZM hexon qubits have logical gates compiled from a sequence of non-commuting measurements of pairs of MZMs. The ability to measure arbitrary pairs of MZMs allows us to simplify the RB procedure. The ability to compile logical Cliffords out of arbitrary length measurement sequences allows us to extract the average contribution of a measurement to the average fidelity of a compiled Clifford gate.

This chapter is based on work in progress with Bela Bauer, Parsa Bonderson, Steve Flammia, and Marcus P. Da Silva, content reproduced with permission of the authors.

5.1 Introduction

In this chapter, we adapt randomized benchmarking to the case of MZM hexon qubits which utilize measurement-only braiding to perform Clifford gates. The structure of the paper is as follows. First, we introduce notation and review the standard randomized benchmarking protocol (Sec. 5.2). In Ch. 5.3 we study RB on hexons and discuss simplifications and extensions for this platform. In particular, we utilize frame-tracking and the ability to measure in any Pauli basis. Further, since gates are implemented via measurements, we propose a direct, measurement-only randomized benchmarking (DMRB) protocol to explore whether the average measurement fidelity can be extracted. We perform simulations in Ch. 5.4. Finally we conclude in Ch. 5.5.

5.2 Review

5.2.1 Notation

First we lay out some basic notation. Operators are denoted with Roman fonts and the quantum channel \mathcal{U} corresponding to a unitary operator U is $\mathcal{U}(\rho) = U\rho U^\dagger$ and denoted with calligraphic font. Noisy implementations of an ideal channel are denoted with over-tildes, $\tilde{\mathcal{U}}$. We will also use the Pauli transfer matrix representation of quantum channels. Let d be the dimension of the Hilbert space of the density matrix (e.g. for a qubit $d = 2$). Let $\{e_1, \dots, e_{d^2}\}$ be the canonical orthonormal basis of \mathbb{C}^{d^2} and $\{B_1, \dots, B_{d^2}\}$ a trace-orthomoral basis (that is $\text{Tr}(B_i^\dagger B_j) = \delta_{ij}$) of $\mathbb{C}^{d \times d}$ which we can take to be the normalized Pauli operators $B_i = \frac{1}{\sqrt{d}} P_i$ and the identity $B_1 = \mathbb{1}_d / \sqrt{d}$. Now we define the linear map $|\cdot\rangle\rangle : \mathbb{C}^{d \times d} \rightarrow \mathbb{C}^{d^2}$ to be

$$|\rho\rangle\rangle = \sum_j \text{Tr}(B_j^\dagger \rho) e_j \quad (5.1)$$

and the adjoint map by $\langle\langle \rho | = |\rho\rangle\rangle^\dagger$ so that $\langle\langle A | B \rangle\rangle = \text{Tr}(A^\dagger B)$. A channel \mathcal{C} takes ρ to $\mathcal{C}(\rho)$ and has a matrix representation

$$\mathcal{C} = \sum_j |\mathcal{C}(B_j)\rangle\rangle \langle\langle B_j|. \quad (5.2)$$

The following is a basic correspondence:

$$\begin{aligned} \rho &\rightarrow |\rho\rangle\rangle \\ \mathcal{C}(\rho) = C\rho C^\dagger &\rightarrow \mathcal{C}|\rho\rangle\rangle = |\mathcal{C}(\rho)\rangle\rangle \end{aligned} \tag{5.3}$$

Composition of channels is simply matrix multiplication, $\mathcal{C}_2 \circ \mathcal{C}_1(\rho) = \mathcal{C}_2\mathcal{C}_1|\rho\rangle\rangle$. `endalign`

5.2.2 Randomized benchmarking

The randomized benchmarking protocol is

1. Choose a positive integer m .
2. Choose uniformly at random a sequence of m gates $(G_1, \dots, G_m) \in G^m$. Set $G_{m+1} = (G_m \dots G_1)^\dagger$. The gate-set G must form a unitary 2-design, a common choice in practice is the uniform distribution over the set of Clifford gates [22].
3. Prepare an initial state $\rho = |\psi\rangle\langle\psi|$, apply the sequence of gates, and measure the observable $E = |\psi\rangle\langle\psi|$.
4. Repeat steps 2-3 to obtain an estimate of the average survival probability for the sequence length m , \hat{q}_m .
5. Repeat steps 1–4 for various m and fit to the model

$$\bar{q}(m) = Ap^m + B \quad (5.4)$$

where p is related to the average fidelity of the gate-set and A and B are SPAM-dependent constants.

Randomized benchmarking [62] seeks to estimate the average fidelity between ideal operations $\mathcal{G}_i \in \mathcal{G}$ from a unitary 2-design \mathcal{G} and their noisy, physical realizations $\tilde{\mathcal{G}}_i = \Lambda_i \mathcal{G}_i$. This is done by subjecting an initially prepared state $\rho = |\psi\rangle\langle\psi|$ to a random sequence of gates $\tilde{\mathcal{G}}_{m:1}(\rho) = \tilde{\mathcal{G}}_{m+1} \circ \tilde{\mathcal{G}}_m \circ \dots \circ \tilde{\mathcal{G}}_1(\rho)$ such that the sequence ideally compiles to the identity, $G_{m+1}^\dagger = G_m \dots G_1$. A measurement $E = |\psi\rangle\langle\psi|$ is then performed and whether or not the initial state was left invariant by the sequence is recorded. Repeating this yields an estimated

survival probability p_m for this sequence length. This data (m, p_m) is then fit to the model $Ap^m + B$ where the decay rate p is related to the average gate fidelity of the gate set G via $p = \frac{d\bar{F}(\Lambda)-1}{d-1}$ with A and B nuisance parameters related to the state-preparation and measurement (SPAM).

Randomized benchmarking relies on two key properties. First, that the gate-set G to be tested forms a unitary 2-design. This means twirling a channel Λ over G is equivalent to twirling over the full unitary group $U(d)$

$$\frac{1}{|G|} \sum_{j=1}^{|G|} G_j \Lambda(G_j^\dagger \rho G_j) G_j^\dagger = \int_{U(d)} dU U \Lambda(U^\dagger \rho U) U, \quad (5.5)$$

in the Pauli transfer matrix representation

$$\frac{1}{|G|} \sum_{j=1}^{|G|} G \Lambda G^\dagger = \int_{U(d)} dU U \Lambda U^\dagger. \quad (5.6)$$

Second, twirling a channel $\Lambda(\cdot)$ over $U(d)$ produces the unique depolarizing channel $\mathcal{D}\{\Lambda\}(\cdot)$

$$\mathcal{D}\{\Lambda\}(\rho) = p\rho + (1-p)\mathbb{1}/d \quad (5.7)$$

with the same average fidelity as Λ [66] giving

$$\bar{F}(\Lambda) = \bar{F}(\mathcal{D}\{\Lambda\}) = p + \frac{1-p}{d}, \quad (5.8)$$

where the average fidelity is defined as

$$\bar{F}(\mathcal{E}) = \int d\psi \langle \psi | \mathcal{E}(\psi) | \psi \rangle. \quad (5.9)$$

The average error rate or infidelity is

$$r = 1 - \bar{F}(\Lambda) = \frac{(d-1)(1-p)}{d}. \quad (5.10)$$

We will now prove the randomized benchmarking decay model for the case where the noise $\Lambda = \Lambda_j$ is gate independent. For sequence length m the the expression for the average survival probability in the Pauli transfer matrix representation is

$$\begin{aligned} \bar{q}(m) &= \mathbb{E}_{G_1, \dots, G_m} \langle \langle \tilde{E} | \tilde{\mathcal{G}}_{m+1} \dots \tilde{\mathcal{G}}_1 | \tilde{\rho} \rangle \rangle \\ &= \mathbb{E}_{G_1, \dots, G_m} \langle \langle \tilde{E} | \Lambda \mathcal{G}_{m+1} \Lambda \mathcal{G}_m \dots \Lambda \mathcal{G}_1 | \tilde{\rho} \rangle \rangle \\ &= \mathbb{E}_{G_1, \dots, G_m} \langle \langle \tilde{E} | \Lambda \mathcal{G}_1^\dagger \dots \mathcal{G}_m^\dagger \Lambda \mathcal{G}_m \dots \Lambda \mathcal{G}_1 | \tilde{\rho} \rangle \rangle \\ &= \mathbb{E}_{G_1, \dots, G_{m-1}} \langle \langle \tilde{E} | \Lambda \mathcal{G}_1^\dagger \dots [\mathbb{E}_{G_m} \mathcal{G}_m^\dagger \Lambda \mathcal{G}_m] \dots \Lambda \mathcal{G}_1 | \tilde{\rho} \rangle \rangle \\ &= \mathbb{E}_{G_1, \dots, G_{m-2}} \langle \langle \tilde{E} | \Lambda \mathcal{D}\{\Lambda\} \mathcal{G}_1^\dagger \dots [\mathbb{E}_{G_{m-1}} \mathcal{G}_{m-1}^\dagger \Lambda \mathcal{G}_{m-1}] \dots \Lambda \mathcal{G}_1 | \tilde{\rho} \rangle \rangle \\ &= \langle \langle \tilde{E} | \Lambda \mathcal{D}\{\Lambda\}^m | \tilde{\rho} \rangle \rangle \\ &= p^m \langle \langle \tilde{E} | \tilde{\rho} \rangle \rangle + \frac{(1-p^m)}{d} \langle \langle \tilde{E} | \mathbb{1} \rangle \rangle \\ &= Ap^m + B \end{aligned} \quad (5.11)$$

where we used the fact that the depolarizing channel commutes with all unitary channels to

sequentially perform the twirl from the middle out [2] and absorbed $\langle\langle \tilde{E} | \Lambda$ back into $\langle\langle \tilde{E} |$.

5.3 Randomized benchmarking on a hexon

As described in the previous chapters, we can form Clifford gates acting on the logical space of a MZM hexon qubit as a sequence of compatible parity measurement. Therefore, we can straightforwardly apply the usual Clifford randomized benchmarking and its variant protocols to the case of measurement-only MZM hexon qubits. In this section we will discuss simplifications and extensions for randomized benchmarking in the hexon setting.

Since measurements can be performed in any Pauli basis, there is no need for a final inversion gate as in the original protocol. Instead, we can track through the action of Clifford gates and measure in the appropriate final basis. For example, if the RB experiment is initialized with a $|+Z\rangle$ state and the random sequence of Cliffords compiles to an overall Hadamard gate, then we can do a final measurement in the $|+X\rangle$ basis instead of performing H^\dagger and measuring in $|+Z\rangle$ again. This modifies the nuisance parameters but gives the same exponential decay form, $A\rho^m + B$, as one can check by following the derivation of Eq. 5.11 with $\langle\langle \tilde{E} | \rightarrow \langle\langle \tilde{E}_{\text{tracked}} | = \langle\langle \rho | \mathcal{G}_1^\dagger \dots \mathcal{G}_m^\dagger \Lambda_{\text{SPAM}}^\dagger$.

Another simplification comes from the fact that different measurement outcomes in a measurement sequence changes the compiled gate only up to an overall Pauli. Instead of sampling from the full Clifford gateset, we can instead sample from the Clifford gates up to Paulis, with the random measurement outcomes then generating the full Clifford gateset [2]. This reduces the sampling space for n qubits by 4^n .

A further feature is that we can leverage the fact that all gates are performed via measurements to extract from the RB infidelity a parameter for the effective measurement infidelity. Challenges include the fact that the set of measurement operators do not themselves form a 2-design, so much of the established machinery does not apply. Additionally, the compatibility requirement between measurements induces a restriction on the distribution of measurements which can be drawn at any given time step which can potentially correlate errors. Furthermore, the hexon encodes a logical qubit in six MZMs so leakage can occur. Though the measurements are not amenable to randomized benchmarking on an individual level, we can group them together into blocks of length- k measurement sequences and study their effect, similar to just performing Clifford randomized benchmarking. By varying k we can then extract some notion of a per-measurement error parameter. To wit, we provide a direct, measurement-only randomized benchmarking (DMRB) protocol [76]:

1. Choose a positive integer m .
2. Prepare a random initial hexon state ρ by selecting uniformly at random two commuting pairs of MZMs to measure (the overall island parity is assumed to be fixed at all times). E.g. $\rho = \Pi_+^{(12)} \Pi_+^{(34)} \Pi_+^{(56)}$ or $\rho = \Pi_+^{(16)} \Pi_+^{(23)} \Pi_+^{(45)}$. Variant note: a specific initial state e.g. $\rho = \Pi_+^{(34)} \Pi_+^{(12)} \Pi_+^{(56)}$ can be chosen instead.
3. Choose uniformly at random a sequence of m compatible measurements. Variant note: instead of choosing a random measurement sequence of length- m , a length- m measurement sequence can be drawn according to some distribution $\Omega(m)$ instead [76].
4. Apply the sequence of measurements and, via tracking, measure the two stabilizers of the final state, recording whether they match (success) or do not (failure).
5. Repeat steps 2-4 to obtain an estimate of the average survival probability for the sequence length m , \hat{q}_m .
6. Repeat steps 1–5 for various m and fit to the model

$$\bar{q}(m) = Ap^m + \frac{1}{2} \quad (5.12)$$

where p is related to the fidelity of measurements and A is a SPAM-dependent nuisance parameter.

To see this, we first show that for any $\delta > 0$ there exists an N such that after applying N random, compatible measurements to an initial stabilizer state, the final state will have equal probability of being in any other stabilizer state. The stabilizer state can be represented as an unordered pair of MZM operators $\{i\gamma_a\gamma_b, i\gamma_c\gamma_d\}$ or as an ordered pair of ancillary encoding with a logical Pauli operator: $(a, l) = (i\gamma_a\gamma_b, i\gamma_c\gamma_d)$. For example, in the previous chapters we used $i\gamma_3\gamma_4$ as the ancillary encoding and $i\gamma_1\gamma_2 = i\gamma_5\gamma_6, i\gamma_1\gamma_6 = i\gamma_2\gamma_5$ as the logical operators. Though arbitrary, we set $i\gamma_1\gamma_2 = \bar{Z}_{34}$ and $i\gamma_1\gamma_6 = \bar{X}_{34}$ and through this choice of representation, we fixed the action of different measurement sequences to logical Clifford operators. Thus, using the measurement update rules of Sec. 3.2.3, we can construct a 180×180 logical Pauli transition matrix (180 comes from the 15×2 different ancillary encoding frames and the choice of 6 logical operators within each of those)

$$M_{(a,l),(a',l')} = \Pr\left((a, l) \middle| (a', l')\right) \quad (5.13)$$

where the probability $\Pr\left((a, l) \middle| (a', l')\right)$ depends on how one chooses the distribution of compatible measurements (Step 3). For example, in the case where all compatible measurements are allowed and equally likely $\Pr\left((a, l) \middle| (a', l')\right) = 1/16$ when there is a measurement which can take (a', l') to (a, l) since there are 8 measurements compatible with (a', l') and 2 possible outcomes, ± 1 . On the other hand, the compatible measurements could be restricted so that $i\gamma_5\gamma_6$ is not allowed, for example.

We will work with the case where

$$M_{(a,l),(a',l')} = \begin{cases} \frac{1}{16} & \text{if there is a measurement that can take } (a, l) \text{ to } (a', l') \\ 0 & \text{if there is no measurement that can take } (a, l) \text{ to } (a', l') \end{cases} \quad (5.14)$$

for the remainder of this paper. By construction, this matrix is symmetric and doubly stochastic which by the Perron-Frobenius theorem then guarantees that the uniform vector is a non-degenerate eigenvector attaining the largest eigenvalue of 1; the second largest eigenvalue is $1/2$. This matrix tells us how logical Paulis in different encodings change probabilistically as measurements are applied. For large N , $M^N \approx \mathbb{1}/180$ with decay rate 2^{-N} . This means that an initial logical Pauli operator in a particular ancillary encoding will have nearly equal probability of becoming any logical Pauli in any ancillary encoding after sufficiently many measurements. The distribution of measurement sequences of a certain length implements a Pauli mixing which uniformly permutes an input logical Pauli operator in some encoding to any logical Pauli in any encoding. This implies that, to a good approximation, a uniformly random Clifford operation has been applied along with a random encoding frame change [21, 38, 90]. More specifically, given an input canonical unit basis vector $e_{(a,l)}$ for any $\delta > 0$ there exists an $N > 0$ such that

$$\left\| M^N e_{(a,l)} - \frac{\mathbb{1}}{180} \right\|_p < \delta. \quad (5.15)$$

where any norm will do but we use the 1-norm.

As a check, we can restrict to the case where the initial ancillary encoding is $\Pi_+^{(34)}$ and the final can be $\Pi_{\pm}^{(34)}$ and (a) randomly sample from the set of all compatible measurement sequences of length n with the restriction that the n 'th measurement is on $i\gamma_3\gamma_4$ to compare against (b) the bound given in Eq. 5.15. Prepare an initial state $\rho = \Pi_+^{(34)}\Pi_+^{(12)}\Pi_+^{(56)}$ and define the probability distribution of final states from (a) to be $f_n^{(a)}$ so that

$$\delta_n^{(a)} = \left\| f_n^{(a)} - \frac{\mathbb{1}}{12} \right\|_1 \quad (5.16)$$

$$\delta_n^{(b)} = \left\| \text{Tr}_{(a,l) \neq (i\gamma_3\gamma_4, l')} M_{\text{final34}} M_{\text{penult34}} M^{n-2} e_{(i\gamma_3\gamma_4, i\gamma_1\gamma_2)} - \frac{\mathbb{1}}{12} \right\|_1 \quad (5.17)$$

where M_{final34} , M_{penult34} correspond to modified M such that the final measurement is fixed to be $i\gamma_3\gamma_4$ and the penultimate measurement is compatible with that, and we trace out outcomes that are not in the $i\gamma_3\gamma_4$ encoding frame. We find (for 10^6 samples)

n	$\delta^{(a)}$	$\delta^{(b)}$
4	0.4995	0.5000
5	0.1215	0.1211
6	0.1253	0.1245
7	0.0300	0.0306

(5.18)

which is rather close. This shows that M can accurately model the tracking of a measurement sequence.

A projector sequence, as mentioned in previous chapters, acts on a state ρ in an initial

encoding of $\Pi_{s_i}^{(i)}$ in two parts: a part that transforms the initial encoding to the final encoding, $T_{f,i} : \Pi_{s_i}^{(i)} \rightarrow \Pi_{s_f}^{(f)}$, and a part that projects into and acts as a logical Clifford operation $\bar{C}_{[f]}$ in this encoding space

$$\Pi_{s_f}^{(f)} \dots \Pi_{s_i}^{(i)} = \bar{C}_{[f]} T_{f,i}. \quad (5.19)$$

The specific Clifford that is compiled depends on the choice of Pauli frames in both the initial and final encodings. Since we are concerned with attaining a uniform distribution over the set of Clifford gates for the purpose of twirling over a 2-design, the particular representations will not be important.

We can now argue for the decay Eq. 5.12 given above. First, we choose a precision δ and use Eq. 5.15 to find a minimal N . Then, for fixed sequence length $m \geq N$, we select k such that $N \leq k \leq m$ and k divides m , $nk = m$. We group the sequence into subsequences of length k which we partition into sectors of fixed initial and final encodings $\{(f_1, i_1), \dots, (f_n, i_n)\}$. The j 'th length- k measurement sequence therefore produces an operation

$$\Pi_{s_{jk}}^{(jk)} \dots \Pi_{s_{(j-1)k+1}}^{((j-1)k+1)} = \bar{C}_{[jk]} \mathcal{T}_{jk,(j-1)k} \quad (5.20)$$

that changes the encoding from the $(j-1)k$ 'th projector to the the jk 'th projector and applies a logical Clifford in the $\Pi_{s_{jk}}^{(jk)}$ encoding; note that we are working in the Pauli transfer matrix representation and have left the $\Pi_{s_{(j-1)k}}^{((j-1)k)}$ projector of the previous subsequence implicit. And for notational simplicity we denote the encoding at the j 'th projector $\Pi_{s_j}^{(j)}$ by its position in

the sequence, j . We will label the sector of encodings by $\vec{T} = (T_{nk,(n-1)k}, \dots, T_{k,0})$, note $\prod_{j=1}^n \vec{T}_j = T_{m,0}$ where the 0'th encoding is the initial one, e.g. $\Pi_+^{(34)}$ in previous chapters.

The non-ideal form of Eq. 5.20 is

$$\tilde{\Pi}_{s_{jk}}^{(jk)} \dots \tilde{\Pi}_{s_{(j-1)k+1}}^{((j-1)k+1)} = \Lambda \Pi_{s_{jk}}^{(jk)} \dots \Lambda \Pi_{s_{(j-1)k+1}}^{((j-1)k+1)} \quad (5.21)$$

$$\approx \Lambda^{(k)} \bar{\mathcal{C}}_{[jk]} \mathcal{T}_{jk,(j-1)k} \quad (5.22)$$

where we have assumed that every measurement has a noise channel Λ immediately proceeding it and that a length- k sequence has an effective noise channel $\Lambda^{(k)}$ depending only on the length of the sequence.

The expression for the average survival probability at length m for an initial state ρ in a sector of fixed \vec{T} is the expectation sequences within those sectors.

$$\bar{q}_k(m, \rho, \vec{T}) = \mathbb{E}_{\text{seq} \in \vec{T}} \langle \langle \tilde{\rho}_{\text{tracked}} | \tilde{\Pi}_{s_m}^{(m)} \dots \tilde{\Pi}_{s_1}^{(1)} | \tilde{\rho} \rangle \rangle \quad (5.23)$$

$$= \mathbb{E}_{\text{seq} \in \vec{T}} \langle \langle \tilde{\rho}_{\text{tracked}} | \left(\tilde{\Pi}_{s_{nk}}^{(nk)} \dots \tilde{\Pi}_{s_{(n-1)k+1}}^{((n-1)k+1)} \right) \dots \left(\tilde{\Pi}_{s_k}^{(k)} \dots \tilde{\Pi}_{s_1}^{(1)} \right) | \tilde{\rho} \rangle \rangle \quad (5.24)$$

$$= \mathbb{E}_{\text{seq} \in \vec{T}} \langle \langle \tilde{\rho}_{\text{tracked}} | \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[nk]} \mathcal{T}_{nk,(n-1)k} \right) \dots \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[k]} \mathcal{T}_{k,0} \right) | \tilde{\rho} \rangle \rangle \quad (5.25)$$

unpacking $\langle\langle \tilde{\rho}_{\text{tracked}} | = \langle\langle \rho | \mathcal{T}_{0,k} \bar{\mathcal{C}}_{[k]}^\dagger \cdots \mathcal{T}_{(n-1)k, nk} \bar{\mathcal{C}}_{[nk]}^\dagger \Lambda_{\text{SPAM}}^\dagger$ and using Eq. 5.15 to convert $\mathbb{E}_{\text{seq} \in \bar{T}}$ to $\mathbb{E}_{\bar{\mathcal{C}}_{[k]}, \dots, \bar{\mathcal{C}}_{[nk]}}$

$$\begin{aligned}
&= \mathbb{E}_{\bar{\mathcal{C}}_{[k]}, \dots, \bar{\mathcal{C}}_{[(n-1)k]}} \langle\langle \rho | \left(\mathcal{T}_{0,k} \bar{\mathcal{C}}_{[k]}^\dagger \cdots \mathcal{T}_{(n-2), (n-1)k} \bar{\mathcal{C}}_{[(n-1)k]}^\dagger \right) \\
&\quad \left(\mathcal{T}_{(n-1)k, nk} \mathbb{E}_{\mathcal{C}_{[nk]}^-} \bar{\mathcal{C}}_{[nk]}^\dagger \Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)} \bar{\mathcal{C}}_{[nk]} \mathcal{T}_{nk, (n-1)k} \right) \\
&\quad \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[(n-1)k]} \mathcal{T}_{(n-1)k, (n-2)k} \right) \cdots \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[k]} \mathcal{T}_{k,0} \right) | \tilde{\rho} \rangle\rangle \quad (5.26)
\end{aligned}$$

recall that for $k > N$, we are within δ of a uniform distribution of the Clifford gate-set and therefore within δ of a 2-design, so now we twirl the middle line to get $\bar{\mathcal{D}}_{[nk]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \}$ which after conjugating by $\mathcal{T}_{(n-1)k, nk}$ becomes $\bar{\mathcal{D}}_{[(n-1)k]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \}$ which is a logical depolarizing channel acting in the $\Pi_{s_{(n-1)k}}^{((n-1)k)}$ encoding and which corresponds to the noise $\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)}$ projected into the $\Pi_{s_{nk}}^{(nk)}$ encoding, $(\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]}$,

$$\begin{aligned}
&= \mathbb{E}_{\bar{\mathcal{C}}_{[k]}, \dots, \bar{\mathcal{C}}_{[(n-1)k]}} \langle\langle \rho | \left(\mathcal{T}_{0,k} \bar{\mathcal{C}}_{[k]}^\dagger \cdots \mathcal{T}_{(n-2), (n-1)k} \bar{\mathcal{C}}_{[(n-1)k]}^\dagger \right) \\
&\quad \left(\bar{\mathcal{D}}_{[(n-1)k]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \} \right) \\
&\quad \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[(n-1)k]} \mathcal{T}_{(n-1)k, (n-2)k} \right) \cdots \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[k]} \mathcal{T}_{k,0} \right) | \tilde{\rho} \rangle\rangle \quad (5.27)
\end{aligned}$$

since the depolarizing channel commutes and identity on a subspace x can be written as $I_{[x]} =$

$T_{x,y}T_{y,x}$ for any y

$$\begin{aligned}
&= \mathbb{E}_{\bar{\mathcal{C}}_{[k]}, \dots, \bar{\mathcal{C}}_{[(n-2)k]}} \langle\langle \rho | \left(\mathcal{T}_{0,k} \bar{\mathcal{C}}_{[k]}^\dagger \dots \mathcal{T}_{(n-3),(n-2)k} \bar{\mathcal{C}}_{[(n-2)k]}^\dagger \right) \\
&\quad \left(\mathcal{T}_{(n-2)k,(n-1)k} \bar{\mathcal{D}}_{[(n-1)k]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \} \mathcal{T}_{(n-1)k,(n-2)k} \right) \\
&\quad \left(\mathcal{T}_{(n-2)k,(n-1)k} \mathbb{E}_{\mathcal{C}_{[(n-1)k]}} \bar{\mathcal{C}}_{[(n-1)k]}^\dagger \Lambda^{(k)} \bar{\mathcal{C}}_{[(n-1)k]} \mathcal{T}_{(n-1)k,(n-2)k} \right) \\
&\quad \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[(n-2)k]} \mathcal{T}_{(n-2)k,(n-3)k} \right) \dots \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[k]} \mathcal{T}_{k,0} \right) | \tilde{\rho} \rangle\rangle \quad (5.28)
\end{aligned}$$

$$\begin{aligned}
&= \mathbb{E}_{\bar{\mathcal{C}}_{[k]}, \dots, \bar{\mathcal{C}}_{[(n-2)k]}} \langle\langle \rho | \left(\mathcal{T}_{0,k} \bar{\mathcal{C}}_{[k]}^\dagger \dots \mathcal{T}_{(n-3),(n-2)k} \bar{\mathcal{C}}_{[(n-2)k]}^\dagger \right) \\
&\quad \left(\bar{\mathcal{D}}_{[(n-2)k]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \} \bar{\mathcal{D}}_{[(n-2)k]} \{ (\Lambda^{(k)})_{[(n-1)k]} \} \right) \\
&\quad \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[(n-2)k]} \mathcal{T}_{(n-2)k,(n-3)k} \right) \dots \left(\Lambda^{(k)} \bar{\mathcal{C}}_{[k]} \mathcal{T}_{k,0} \right) | \tilde{\rho} \rangle\rangle \quad (5.29)
\end{aligned}$$

iterating this we get

$$= \langle\langle \rho | \bar{\mathcal{D}}_{[0]} \{ (\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]} \} \bar{\mathcal{D}}_{[0]} \{ (\Lambda^{(k)})_{[(n-1)k]} \} \dots \bar{\mathcal{D}}_{[0]} \{ (\Lambda^{(k)})_{[k]} \} | \tilde{\rho} \rangle\rangle \quad (5.30)$$

now, we make the further assumption that $\bar{\mathcal{D}}_{[0]}\{(\Lambda^{(k)})_{[jk]}\} = \bar{\mathcal{D}}_{[0],k}$ for all $[jk]$ and thus we can write $\bar{\mathcal{D}}_{[0]}\{(\Lambda_{\text{SPAM}}^\dagger \Lambda^{(k)})_{[nk]}\} = \bar{\mathcal{D}}_{[0],\text{SPAM}} \bar{\mathcal{D}}_{[0],k}$ for all $[nk]$

$$= \langle\langle \rho | \bar{\mathcal{D}}_{[0],\text{SPAM}} \bar{\mathcal{D}}_{[0],k}^n | \tilde{\rho} \rangle\rangle \quad (5.31)$$

$$= p' p_k^n \langle\langle \rho | \tilde{\rho} \rangle\rangle + \frac{1 - p' p_k^n}{2} \langle\langle \rho | \Pi_{s_0}^{(0)} \rangle\rangle \quad (5.32)$$

$$= \left(\langle\langle \rho | \tilde{\rho} \rangle\rangle - \frac{1}{2} \right) p' p_k^n + \frac{1}{2} \quad (5.33)$$

$$= A' p_k^n + \frac{1}{2} \quad (5.34)$$

where p', p_k are the depolarizing strengths corresponding to $\bar{\mathcal{D}}_{[0],\text{SPAM}}, \bar{\mathcal{D}}_{[0],k}^n$ and we used $\langle\langle \rho | \Pi_{s_0}^{(0)} \rangle\rangle =$

1. Averaging over frame changes \vec{T} and initial states ρ gives

$$\bar{q}_k(m) = \mathbb{E}_{\rho, \vec{T}} \bar{q}_k(m, \rho, \vec{T}) = A p_k^{m/k} + \frac{1}{2} \quad (5.35)$$

which corresponds to Steps 1- 5 of the DMRB protocol. The above holds within some function $g(\delta)$ of δ for any $k > N$ via Eq. 5.15 with better precision as $N \rightarrow \infty$. For a fixed sequence, k is a mathematical construct that parametrizes how we chose to form subsequences and thus has no physical effect. The decay is therefore independent of k up to $g(\delta)$ and how the sequence is partitioned into subsequences,

$$\bar{q}(m) = \bar{q}_k(m). \quad (5.36)$$

The general solution to $\bar{q}_{k_1}(m) = \bar{q}_{k_2}(m)$ in the limit of arbitrarily many $k_i \geq N$ and dividing

m is $p_k = p^k$ for some p . Thus,

$$\bar{q}(m) = Ap^m + B. \quad (5.37)$$

This is an exponential decay in the number of measurements, we therefore interpret the corresponding RB number $r = \frac{1-p}{2}$ as the average contribution to the average gate error rate of a single measurement.

5.4 Simulations

We carry out simulations for the DMRB protocol described in the previous section. We select a minimal measurement sequence length of $m = 16$. There are two types of errors we consider

1. **QP** A quasiparticle hopping event where after a measurement, depending on whether the measurement was a two-sided or one-sided two-hexon measurement (assuming a two-sided hexon qubit), there is probability ϵ_{qp1} or ϵ_{qp2} for an electron to hop onto the island at MZM j and exit at a different MZM k . This results in random two-qubit Pauli $i\gamma_j\gamma_k$ being applied with probability ϵ_{qp1} or ϵ_{qp2} . There are 15 measurements in total 6 of which are one-sided and 9 of which are two-sided in the two-sided hexon architecture. For a random measurement sequence then, the effective error rate for quasiparticle hopping is

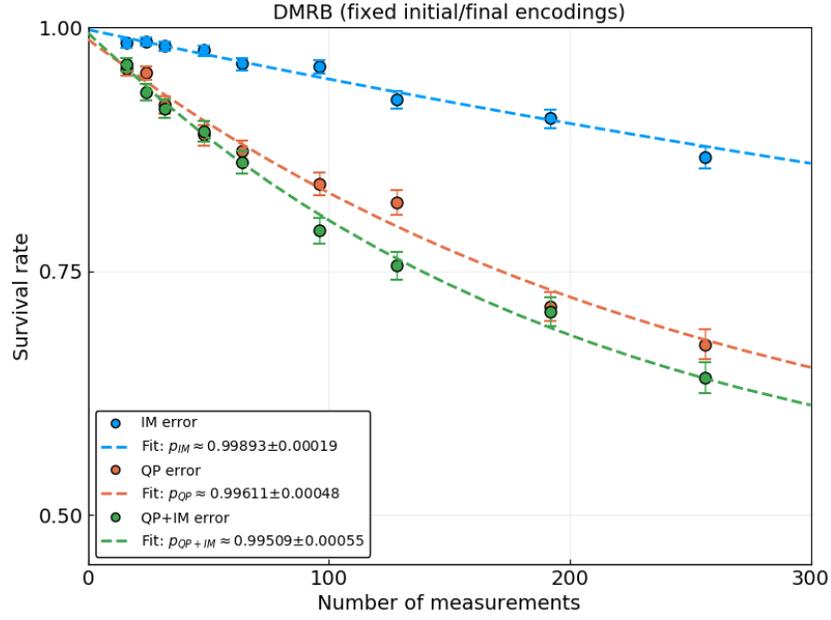
$$\epsilon_{qp} = \frac{6}{15}\epsilon_{qp1} + \frac{9}{15}\epsilon_{qp2} \quad (5.38)$$

2. **IM** Imperfect measurements where the measurement no longer perfectly projects onto $i\gamma_j\gamma_k$ but instead projects into either the positive or negative eigenspace of the operator

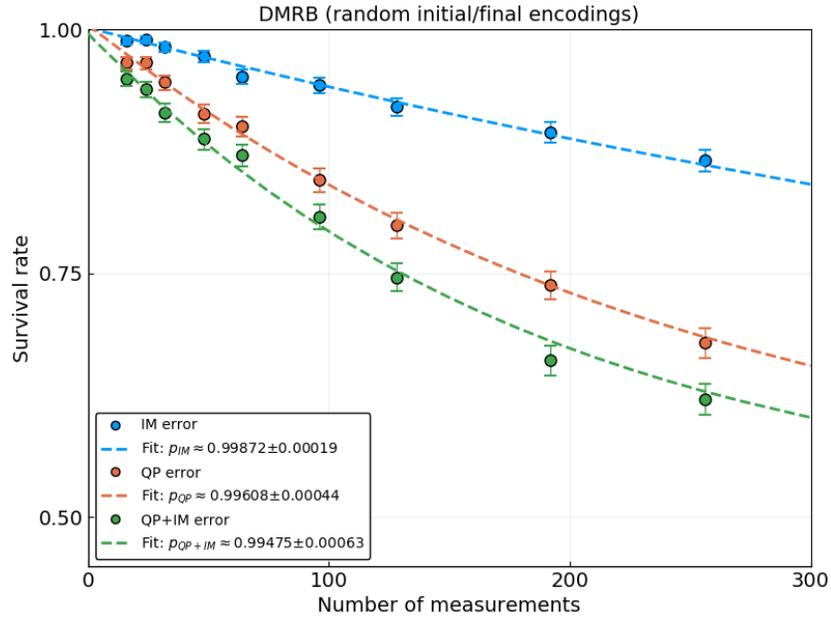
$$i\gamma_j\gamma_k \rightarrow i\gamma_j\gamma_k + \sum_{j' \neq j,k} \exp(-w_{j'k}/W) i\gamma_{j'}\gamma_k + \sum_{k' \neq j,k} \exp(-w_{jk'}/W) i\gamma_j\gamma_{k'} \quad (5.39)$$

where $w_{jk} > 0$ is the difficulty of measuring $i\gamma_j\gamma_k$ and W parametrizes its strength.

We are interested in the cases where the initial state is $\rho = \Pi_+^{(34)}\Pi_+^{(12)}\Pi_+^{(56)}$ and the final measurement is $i\gamma_3\gamma_4$ and where the initial state is random and the final measurement is random. We carry out this simulation for $\epsilon_{qp1} = 0.002, \epsilon_{qp2} = 0.005$ giving $\epsilon_{qp} = 0.0038$ and w_{jk} as given in Ch. 3.2.4 with $W = 1$. As shown in the plots below we get close $(0.9961 \pm 0.0005, 0.9961 \pm 0.0004)$ to the expected survival probability $1 - 0.0038$. and moreover the two error types multiply together when combined, $p_{QP+IM} \approx p_{QP}p_{IM}$.



(a)



(b)

Figure 5.1: Direct, measurement-only randomized benchmarking simulations according to the protocol 5.12. 1000 trials are taken at each sequence length m . The data (m, \hat{q}) is fit to $\ln(\hat{q} - 0.5) = \ln(A') + m \ln(p)$. Error bars are one sigma confidence intervals. QP corresponds to quasiparticle hopping error, IM corresponds to the imperfect measurement error, and QP+IM corresponds to both. (a) Initial state of $\rho = \Pi_+^{(34)} \Pi_+^{(12)} \Pi_+^{(56)}$ and a final measurement on $i\gamma_3\gamma_4$. (b) Random initial state and random final measurement.

5.5 Discussion

We have shown that randomized benchmarking can be performed on a measurement-only MZM hexon qubit. Moreover it offers two advantages in that measurements can be taken of any Pauli basis since any pair of MZMs can be measured. This allows for Pauli tracking which removes the need to apply a final inversion gate as compared to the original RB protocol. Second, sufficiently long measurement sequences are shown to compile nearly uniformly at random to a logical Clifford gate. Putting these together we propose a direct measurement-only randomized benchmarking (DMRB) protocol that has a simple form exponentially decaying in the number of measurements. The extracted DMRB parameter can be interpreted as the average contribution of the set of measurements to the average error rate of randomly compiled Clifford gates. Simulations for simple error models (quasiparticle hopping and imperfect measurements) show this exponential decay. Further work, however, is needed however to rigorously tie the DMRB decay to the average fidelity of a single measurement operation. Moreover, devising experiments that cross-check the validity of this interpretation or bound the average measurement fidelity would be useful.

Appendix A

Optimal compilation tools and details

A.1 Example of Adaptive Forced-Measurement Protocol

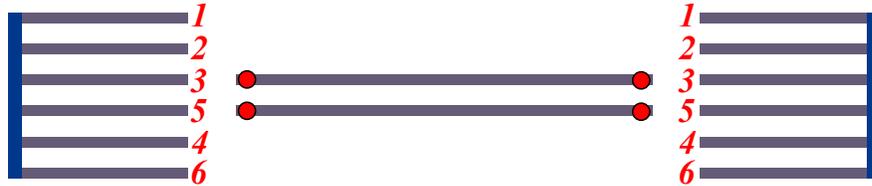


Figure A.1: Two one-sided hexons in the MZM labeling configuration $\langle 1, 2, 3, 5, 4, 6 \rangle$.

For an example of the adaptive method described in Sec. 3.3.4, consider the following measurement-only sequence for compiling $C(Z)$ between horizontal neighboring one-sided hexons for the MZM labeling configuration $\langle 1, 2, 3, 5, 4, 6 \rangle$, as shown in Fig. A.1:

$$\prod_{+}^{\dagger p(34)} \prod_{+}^{\dagger p(3'4')} \hat{\prod}_{-s_2 s_1}^{\dagger p(46)} \prod_{+}^{\dagger p(56)} \prod_{s_2}^{\dagger p(4'6')} \prod_{s_1}^{\dagger p(35;3'5')} \quad (\text{A.1})$$

This sequence has difficulty weight 1.44×10^{11} . Notice that in the forced measurement $\prod_{+}^{\dagger p(56)}$,

the 4-MZM measurement of $(35; 3'5')$ must be repeated when the undesired outcome of the (56)-measurement is obtained.

On the other hand, searching further out in the number of measurements, the measurement-only sequence

$$\prod_{+}^{\dagger p(34)} \prod_{+}^{\dagger p(3'4')} \prod_{s_3 s_2}^{\dagger p(4'6')} \prod_{s_5}^{\dagger p(4'5')} \widehat{\prod}_{-s_3 s_2 s_1}^{(46)} \prod_{s_3}^{(56)} \prod_{s_2}^{(4'6')} \prod_{s_1}^{(35; 3'5')} \quad (\text{A.2})$$

also compiles to $C(Z)$ and has difficulty weight 4.05×10^{10} . Notice that the forcing procedures in this alternative sequence no longer involve the 4-MZM measurement of $(35; 3'5')$.

Simply using this alternative sequence is already an improvement over the first, but it is possible to achieve better results by combining the two in a more complex way. Notice that the first three measurements in these two sequences are identical if $s_3 = +$. This suggests a more optimal protocol for synthesizing $C(Z)$ is: (1) Perform the first three measurements giving the projector sequence $\prod_{s_3}^{(56)} \prod_{s_2}^{(4'6')} \prod_{s_1}^{(35; 3'5')}$. (2a) If $s_3 = +$, finish the measurement sequence as in Eq.(A.1). (2b) If $s_3 = -$, finish the measurement sequence as in Eq.(A.2). This protocol gives a geometric average weight of 3.43×10^9 for the $C(Z)$ gate.

Sequence	Weight
$\prod_{+}^{\dagger p(34)} \prod_{+}^{\dagger p(3'4')} \widehat{\prod}_{-s_2 s_1}^{(46)} \prod_{+}^{(56)} \prod_{s_2}^{(4'6')} \prod_{s_1}^{(35; 3'5')}$	2.90×10^8
$\prod_{+}^{\dagger p(34)} \prod_{+}^{\dagger p(3'4')} \prod_{-s_2}^{\dagger p(4'6')} \prod_{s_5}^{\dagger p(4'5')} \widehat{\prod}_{s_2 s_1}^{(46)} \prod_{-}^{(56)} \prod_{s_2}^{(4'6')} \prod_{s_1}^{(35; 3'5')}$	4.05×10^{10}
Average case	3.43×10^9

(A.3)

In general, such adaptive protocols will be relevant whenever a projector sequence has a forced measurement that involves a particularly high weight measurement (for example, a 4-MZM measurement that uses coherent links) and extending the sequence removes having to repeat that costly measurement.

A.2 Sequence Morphology

In this section, we develop some tools to aid in the optimization of measurement-only gate compilation. We provide a method for generating alternate projector sequences for a specified gate from a given projector sequence and a method for generating projector sequences for all gates in the same conjugacy class as the specified gate. We also provide a protocol for reducing the lengths of projector sequences obtained through gate synthesis.

A.2.1 Some general formulas

Recall that 2-MZM projectors are defined as $\Pi_s^{(jk)} = \frac{\mathbb{1} + si\gamma_j\gamma_k}{2}$ and obey the usual properties of complete orthogonal projectors

$$\Pi_s^{(jk)}\Pi_t^{(jk)} = \delta_{s,t}\Pi_s^{(jk)} \quad (\text{A.4a})$$

$$\Pi_+^{(jk)} + \Pi_-^{(jk)} = \mathbb{1}. \quad (\text{A.4b})$$

From the definition, it also follows that

$$\Pi_+^{(jk)} - \Pi_-^{(jk)} = i\gamma_j\gamma_k, \quad (\text{A.5})$$

$$(i\gamma_j\gamma_k)\Pi_s^{(jk)} = \Pi_s^{(jk)}(i\gamma_j\gamma_k) = s\Pi_s^{(jk)}. \quad (\text{A.6})$$

If $i\gamma_{a_1}\gamma_{a_2}$ and $i\gamma_{b_1}\gamma_{b_2}$ anti-commute with each other, we have the relations

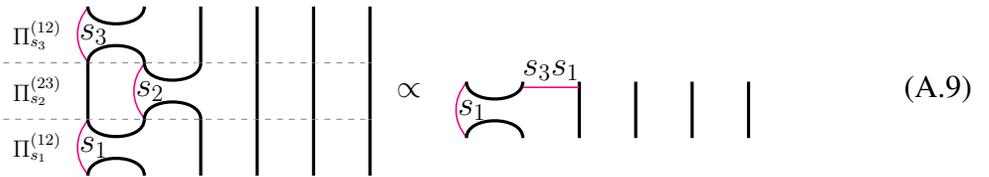
$$\Pi_{s_2}^{(a_1a_2)}\Pi_{s_1}^{(b_1b_2)} = \Pi_{-s_1}^{(b_1b_2)}\Pi_{s_2}^{(a_1a_2)} + \frac{s_1}{2}i\gamma_{b_1}\gamma_{b_2} \quad (\text{A.7a})$$

$$= \Pi_{s_1}^{(b_1b_2)}\Pi_{-s_2}^{(a_1a_2)} + \frac{s_2}{2}i\gamma_{a_1}\gamma_{a_2}. \quad (\text{A.7b})$$

It follows that we can reduce the triplet of projections

$$\Pi_{s_3}^{(a_1a_2)}\Pi_{s_2}^{(b_1b_2)}\Pi_{s_1}^{(a_1a_2)} = \begin{cases} \frac{1}{2}\Pi_{s_1}^{(a_1a_2)} & \text{if } s_3 = s_1 \\ \frac{s_2}{2}(i\gamma_{b_1}\gamma_{b_2})\Pi_{s_1}^{(a_1a_2)} & \text{if } s_3 = -s_1 \end{cases}, \quad (\text{A.8})$$

when $i\gamma_{a_1}\gamma_{a_2}$ and $i\gamma_{b_1}\gamma_{b_2}$ anti-commute. Diagrammatically, this gives identities of the form



$$\Pi_{s_3}^{(12)} \Pi_{s_2}^{(23)} \Pi_{s_1}^{(12)} \propto \Pi_{s_1}^{(12)} \Pi_{s_3 s_1} \quad (\text{A.9})$$

where the magenta line labeled by s_μ between a cup and a cap indicates a projector with unspecified projection channel $s_\mu = \pm 1$, and a magenta line labeled by s_μ connecting MZM

lines j and k corresponds to the operator $(i\gamma_j\gamma_k)^{\frac{1-s_\mu}{2}}$.

In general, for multi-hexon MZM measurements $\Pi_s^{(\mathcal{M})}$

$$\begin{aligned} \Pi_s^{(\mathcal{M}_1)} \Pi_p^{(\mathcal{M}_2)} \Pi_r^{(\mathcal{M}_1)} &= \Pi_s^{(\mathcal{M}_1)} \frac{\mathbb{1} + p \Gamma_{\mathcal{M}_2}}{2} \Pi_r^{(\mathcal{M}_1)} \\ &= \begin{cases} \frac{1}{2} \Pi_r^{(\mathcal{M}_1)} & \text{if } s = r \\ \frac{p}{2} \Gamma_{\mathcal{M}_2} \Pi_r^{(\mathcal{M}_1)} & \text{if } s = -r \end{cases} \end{aligned} \quad (\text{A.10})$$

whenever $\Gamma_{\mathcal{M}_1}$ anticommutes with $\Gamma_{\mathcal{M}_2}$, On the other hand, when $\Gamma_{\mathcal{M}_1}$ commutes with $\Gamma_{\mathcal{M}_2}$

$$\Pi_s^{(\mathcal{M}_1)} \Pi_p^{(\mathcal{M}_2)} \Pi_r^{(\mathcal{M}_1)} = \delta_{s,r} \Pi_s^{(\mathcal{M}_1)} \Pi_p^{(\mathcal{M}_2)} = \delta_{s,r} \Pi_p^{(\mathcal{M}_2)} \Pi_r^{(\mathcal{M}_1)}. \quad (\text{A.11})$$

Additionally, we have the following identities (for a, b, c all distinct)

$$\Pi_{s_3}^{(bc)} \Pi_{s_2}^{(ac)} \Pi_{s_1}^{(ab)} = \frac{1 + i s_1 s_2 s_3}{2} \Pi_{s_3}^{(bc)} \Pi_{s_1}^{(ab)}, \quad (\text{A.12a})$$

$$\Pi_{s_4}^{(ab)} \Pi_{s_3}^{(bc)} \Pi_{s_2}^{(ac)} \Pi_{s_1}^{(ab)} = \begin{cases} \frac{1 + i s_1 s_2 s_3}{4} \Pi_{s_1}^{(ab)} & \text{if } s_4 = s_1 \\ s_3 (i \gamma_b \gamma_c) \frac{1 + i s_1 s_2 s_3}{4} \Pi_{s_1}^{(ab)} & \text{if } s_4 = -s_1 \end{cases}. \quad (\text{A.12b})$$

The first follows from $i \gamma_a \gamma_c = i (i \gamma_b \gamma_c) (i \gamma_a \gamma_b)$ and the second follows from the first and

Eq. (A.8). Diagrammatically, these relations take the form

Diagrammatic equation (A.13a) showing a sequence of projection operators $\Pi_{s_1}^{(12)}$, $\Pi_{s_2}^{(13)}$, and $\Pi_{s_3}^{(23)}$ on the left, and their corresponding braiding relations with vertical lines on the right, separated by a proportionality symbol \propto . The right side shows a sequence of projection operators $\Pi_{s_1}^{(12)}$, $\Pi_{s_3}^{(23)}$, and a vertical line.

Diagrammatic equation (A.13b) showing a sequence of projection operators $\Pi_{s_1}^{(12)}$, $\Pi_{s_2}^{(13)}$, $\Pi_{s_3}^{(23)}$, and $\Pi_{s_4}^{(12)}$ on the left, and their corresponding braiding relations with vertical lines on the right, separated by a proportionality symbol \propto . The right side shows a sequence of projection operators $\Pi_{s_1}^{(12)}$, $\Pi_{s_4}^{(12)}$, and a vertical line.

A.2.2 Sequence Morphology

Given a projection operator sequence \mathcal{G} compiling to G , it is useful to develop tools for constructing alternate sequences compiling to G . This is because measuring certain MZMs may be easier than measuring others. Thus, we wish to come up with as many ways of obtaining a gate G as possible so that we can then pick out the one that is easiest to implement. Furthermore, the ability to compile some gate H when given a compilation for a different gate G is beneficial for expanding our range of operations. This subsection details methodologies for both of these tasks.

For a system of N hexons, we write a sequence of projection operators that compiles to the gate G acting on the computational state space as

$$\mathcal{G} = \Pi_+^{(\text{anc})} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \Pi_{s_1}^{(\mathcal{M}_1)} \Pi_+^{(\text{anc})} \propto \Pi_+^{(\text{anc})} \otimes G, \quad (\text{A.14})$$

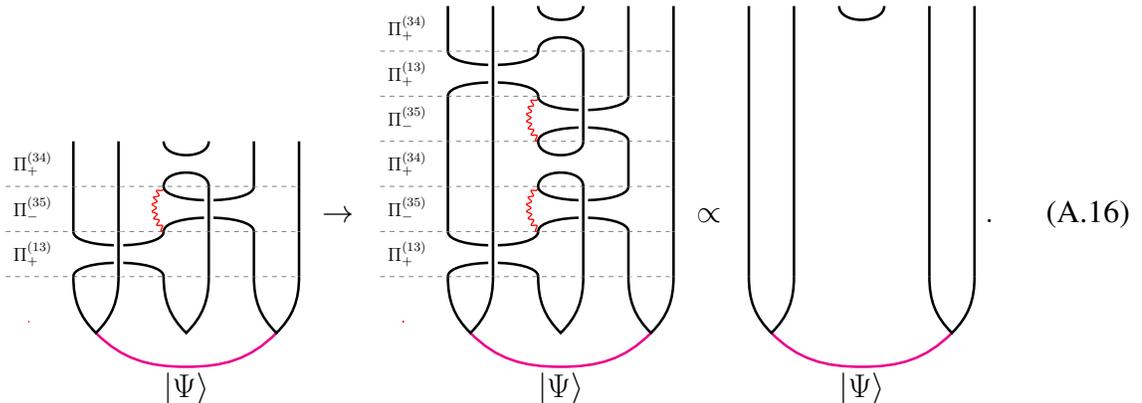
where \mathcal{M}_μ are the sets of (up to $2N$) MZMs whose collective fermionic parity is being projected onto $s_\mu = \pm$. The first term in the tensor product acts on the ancillary qubits and the second term acts on the computational qubits. Given a projector sequence \mathcal{G} that compiles to the target gate G , one can easily construct sequences that compile to the complex conjugate G^* , the inverse gate $G^{-1} = G^\dagger$, the transposed gate $G^T = G^{*\dagger}$, and nontrivial alternate sequences for G .

A.2.2.1 Space-time reflections

By reversing a projector sequence, one generates a compilation for the inverse gate $G^{-1} = G^\dagger$, since the projectors are Hermitian, that is

$$\mathcal{G}^{\text{rev}} = \Pi_+^{(\text{anc})} \Pi_{s_1}^{(\mathcal{M}_1)} \dots \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})} \Pi_+^{(\text{anc})} \propto \Pi_+^{(\text{anc})} \otimes G^\dagger. \tag{A.15}$$

Diagrammatically, this can be seen by first applying \mathcal{G} then \mathcal{G}^\dagger and noting that the stacked diagram can be straightened out and fermion lines canceled, yielding the identity operator; for example



The complex conjugated gate can be constructed by complex conjugating each term in the projector sequence, as follows

$$\mathcal{G}^* = \Pi_+^{(\text{anc})^*} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})^*} \dots \Pi_{s_1}^{(\mathcal{M}_1)^*} \Pi_+^{(\text{anc})^*} \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes G^*. \quad (\text{A.17})$$

In the choice of basis that we are using, complex conjugating the fermionic parity projectors has the effect of potentially changing which parity is being projected onto. In particular, we find that $\Pi_s^{(\mathcal{M})^*} = \Pi_{\pm s}^{(\mathcal{M})}$ when $\Gamma_{\mathcal{M}}^* = \pm \Gamma_{\mathcal{M}}$. In other words, when $\Gamma_{\mathcal{M}}$ is written as a tensor product of Pauli matrices, $\Pi_s^{(\mathcal{M})^*} = \Pi_s^{(\mathcal{M})}$ when the tensor product involves an even number of Y matrices, and $\Pi_s^{(\mathcal{M})^*} = \Pi_{-s}^{(\mathcal{M})}$ when the tensor product involves an odd number of Y matrices. It is straightforward to check that, in our choice of basis, the later occurs for a single hexon projector $\Pi_s^{(jk)}$ whenever $|j - k|$ is even. We emphasize that the action of complex conjugation is basis dependent.

A different nontrivial way to arrive at the complex conjugated gate is by creating the mirror image of the braid sequence. For example,

$$\begin{array}{ccc} \begin{array}{c} \Pi_+^{(34)} \\ \Pi_-^{(35)} \\ \Pi_+^{(13)} \end{array} & \xleftrightarrow{\text{mirror}} & \begin{array}{c} \Pi_+^{(34)} \\ \Pi_-^{(24)} \\ \Pi_+^{(46)} \end{array} = \begin{array}{c} \Pi_+^{(34)} \\ \Pi_-^{(24)} \\ \Pi_+^{(46)} \end{array} \\ |\Psi\rangle & & |\Psi\rangle \end{array} \quad (\text{A.18})$$

On the level of projector sequences, this is implemented for a single hexon via the “mirroring” operation $\Pi_s^{(jk)^M} = \Pi_s^{((7-k)(7-j))}$ applied to each projector in the sequence, where the MZMs

are numbered $1, \dots, 6$ from left to right. We note that the ancillary qubit's projector is invariant under mirroring, i.e. $\Pi_+^{(34)^M} = \Pi_+^{(34)}$. Thus, we have

$$\mathcal{G}^M = \Pi_+^{(\text{anc})^M} \Pi_{s_{n-1}}^{(\mathcal{M}_{n-1})^M} \dots \Pi_{s_1}^{(\mathcal{M}_1)^M} \Pi_+^{(\text{anc})^M} \propto \Pi_+^{(\text{anc})} \otimes G^*. \quad (\text{A.19})$$

Generalizing to the N hexon case, the mirroring operator can be applied to each hexon independently to generate 2^N (potentially) different projector sequences.

Combining the two ways of complex conjugating, we can construct an alternate compilation for a gate G , given \mathcal{G} . Specifically, we can first mirror the sequence to get a compilation \mathcal{G}^M for G^* and then we can complex conjugate each projector to get a projector sequence \mathcal{G}^{M*} . The result compiles to the gate $G^{**} = G$. We dub this operation “mirror-conjugating” a sequence.

For example, both of the following projector sequences will compile to the same gate ZH

$$ZH |\Psi\rangle \propto \begin{array}{c} \Pi_+^{(34)} \\ \Pi_+^{(35)} \\ \Pi_+^{(13)} \\ \hline \end{array} \begin{array}{c} \text{Diagram 1} \\ \hline \end{array} = \begin{array}{c} \Pi_+^{(34)} \\ \Pi_+^{(24)} \\ \Pi_-^{(46)} \\ \hline \end{array} \begin{array}{c} \text{Diagram 2} \\ \hline \end{array}. \quad (\text{A.20})$$

A.2.2.2 Paulimorphism

It follows from the definition of the Clifford group that, given a sequence \mathcal{G} compiling to gate G , one can construct alternate compilations of G , as well as compilations for any other gate $G' \in \text{Conj}_{\text{CN}}(G)$ in the same conjugacy class as G . We define the (projective) stabilizer

of a gate $G \in \mathbf{C}_N$ to be

$$\text{Stab}_{\mathbf{C}_N}(G) = \{A \in \mathbf{C}_N \mid AGA^{-1} = e^{i\phi}G\} \quad (\text{A.21})$$

and the (projective) conjugacy class of $G \in \mathbf{C}_N$ to be

$$\text{Conj}_{\mathbf{C}_N}(G) = \{G' \in \mathbf{C}_N \mid \exists K \in \mathbf{C}_N \text{ s.t. } G' = e^{i\phi}K GK^{-1}\}, \quad (\text{A.22})$$

where the equivalences are up to arbitrary overall phases $e^{i\phi}$ (since we are considering gates, not group elements).

We define the *sequence stabilizer* and *sequence conjugacy class* for a MZM projector sequence \mathcal{G} , acting on the $2N$ qubits corresponding to N hexons, that compiles to the gate G as

$$\text{Stab}(\mathcal{G}) = \left\{ \mathcal{A} \in \mathbf{C}_{2N} \mid \mathcal{A}\mathcal{G}\mathcal{A}^{-1} \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes G \right\} \quad (\text{A.23})$$

$$\text{Conj}(\mathcal{G}) = \left\{ \mathcal{G}' \mid \begin{array}{l} \exists \mathcal{K} \in \mathbf{C}_{2N} \text{ s.t. } \mathcal{G}' = e^{i\phi}\mathcal{K}\mathcal{G}\mathcal{K}^{-1}, \\ \mathcal{G}' \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes G', G' \in \text{Conj}_{\mathbf{C}_N}(G) \end{array} \right\}. \quad (\text{A.24})$$

We note that conjugation by Clifford gates maps fermionic parity projectors to fermionic parity projectors, though possibly changes the number and location of MZMs involved in the projection operator. This follows from the observation that conjugation by Clifford gates maps multi-qubit Pauli operators to multi-qubit Pauli operators, up to possible signs, together with

the bijection between multi-MZM parity operators and the multi-qubit Pauli operators established by Eq. (3.7). Thus, conjugating \mathcal{G} by $\mathcal{A} \in \text{Stab}(\mathcal{G})$ yields the (potentially different) projector sequence

$$\tilde{\mathcal{G}} = \mathcal{A} \mathcal{G} \mathcal{A}^\dagger = \tilde{\Pi}_+^{(\text{anc})} \tilde{\Pi}_{s_{n-1}}^{(\mathcal{M}_{n-1})} \dots \tilde{\Pi}_{s_1}^{(\mathcal{M}_1)} \tilde{\Pi}_+^{(\text{anc})} \propto \Pi_+^{(\text{anc})} \otimes G, \quad (\text{A.25})$$

where

$$\tilde{\Pi}_{s_\mu}^{(\mathcal{M}_\mu)} = \mathcal{A} \Pi_{s_\mu}^{(\mathcal{M}_\mu)} \mathcal{A}^\dagger = \Pi_{s_\mu}^{(\tilde{\mathcal{M}}_\mu)}, \quad (\text{A.26})$$

is the fermionic parity projection operator, corresponding to a new set of MZMs $\tilde{\mathcal{M}}_\mu$, for which the number, order, and locations of MZMs may be different than those of the original set \mathcal{M}_μ . This is generally determined from the transformation of the fermionic parity operators

$$\Gamma_{\tilde{\mathcal{M}}} = \mathcal{A} \Gamma_{\mathcal{M}} \mathcal{A}^\dagger. \quad (\text{A.27})$$

For example, for a single hexon ($N = 1$), the pairwise projectors become

$$\tilde{\Pi}_s^{(jk)} = \mathcal{A} \Pi_s^{(jk)} \mathcal{A}^\dagger = \frac{1}{2} (\mathbb{1} + s \mathcal{A} (i \gamma_j \gamma_k) \mathcal{A}^\dagger) = \Pi_s^{(\tilde{jk})}, \quad (\text{A.28})$$

where (\tilde{jk}) is a potentially different pair of MZM labels than (jk) , and we allow the order of labels in (\tilde{jk}) to be used to absorb changes in the sign of s , i.e. $\Pi_{-s}^{(ab)} = \Pi_s^{(ba)}$. We note that $\mathcal{A} \in \text{Stab}(\mathcal{G})$ has the property that $\mathcal{A} \left(\Pi_+^{(\text{anc})} \otimes \mathbb{1}_{2N} \right) \mathcal{A}^\dagger = \Pi_+^{(\text{anc})} \otimes \mathbb{1}_{2N}$ and the property that $\left(\Pi_+^{(\text{anc})} \otimes \mathbb{1}_{2N} \right) \mathcal{A} \left(\Pi_+^{(\text{anc})} \otimes \mathbb{1}_{2N} \right) = \Pi_+^{(\text{anc})} \otimes A$ for some $A \in \text{Stab}_{\text{C}_N}(G)$.

Conjugating \mathcal{G} and \mathcal{G}^{M^*} by the elements of its stabilizer $\text{Stab}(\mathcal{G})$ yields up to $2|\text{Stab}(\mathcal{G})|$ possible compilations for a target gate G with the same sequence length as \mathcal{G} . In practice, this generates fewer than $2|\text{Stab}(\mathcal{G})|$ distinct projector sequences, because an entire sequence is often invariant with respect to some subgroup of the stabilizer group.

Likewise, given a compilation for a gate G , we also are able to generate compilations for every other element of its conjugacy class $H \in \text{Conj}_{\mathbb{C}_N}(G)$. This is because, by definition, there exists an $X \in \mathbb{C}_n$ such that $H = e^{i\phi} X G X^\dagger$.

For example, for a single hexon, the sequence $\mathcal{S} = \Pi_+^{(34)} \Pi_+^{(23)} \Pi_+^{(13)} \Pi_+^{(34)}$, which compiles to the phase gate S , yields a total of 16 distinct compilations by starting with either \mathcal{S} or \mathcal{S}^{M^*} and conjugating by elements of $\text{Stab}(\mathcal{S})$.

It may be useful to impose a locality constraint that restricts which elements of $\text{Stab}(\mathcal{G})$ and $\text{Conj}(\mathcal{G})$ we utilize, in order to prevent the physical measurements from increasing in complexity, i.e. so that the resulting measurements do not involve a larger number of MZMs nor additional hexons. This can be accomplished by restricting \mathcal{A} and \mathcal{K} in these definitions to the subset of Clifford gates generated by $\{S_{a_j}, S_{q_j}, H_{q_j}, C(Z)_{a_j q_j}\}$, where a_j and q_j labels the j th hexon's ancillary and computational qubits, respectively. If we denote the subset of Clifford gates generated by these gate as the *hexon-local* Clifford gates $\widehat{\mathbb{C}}_{2N} \subset \mathbb{C}_{2N}$, then we define the *hexon-local sequence stabilizer* to be

$$\widehat{\text{Stab}}(\mathcal{G}) = \left\{ \mathcal{A} \in \widehat{\mathbb{C}}_{2N} \mid \mathcal{A} \mathcal{G} \mathcal{A}^{-1} \propto \Pi_+^{(\text{anc})} \otimes G \right\}, \quad (\text{A.29})$$

and the *hexon-local sequence conjugacy class* to be

$$\widehat{\text{Conj}}(\mathcal{G}) = \left\{ \mathcal{G}' \left| \begin{array}{l} \exists \mathcal{K} \in \widehat{\mathbb{C}}_{2N} \text{ s.t. } \mathcal{G}' = e^{i\phi} \mathcal{K} \mathcal{G} \mathcal{K}^{-1}, \\ \mathcal{G}' \propto \mathbf{\Pi}_+^{(\text{anc})} \otimes \mathcal{G}', \mathcal{G}' \in \text{Conj}_{\mathbb{C}_N}(G) \end{array} \right. \right\}. \quad (\text{A.30})$$

Conjugating a fermionic parity operator $\Gamma_{\mathcal{M}}$ by an element $\mathcal{A} \in \widehat{\mathbb{C}}_{2N}$ of the hexon-local Clifford gates yields a fermionic parity operator $\Gamma_{\widetilde{\mathcal{M}}}$ that involves the same number of MZMs from each hexon, though possibly with different locations within each hexon. Hence, the corresponding projectors and measurements for these operators involve the same number of MZMs from each hexon. In other words, the locality with respect to hexons of the corresponding measurement is preserved.

A.2.2.3 Sequence reduction

The goal of this subsection is to find an efficient compilation for a target gate G that can be generated from a generating gate set $\{G_1, \dots, G_N\}$, for which we have the corresponding compilations $\{\mathcal{G}_1, \dots, \mathcal{G}_N\}$ with respective sequence lengths $\{L_1, L_2, \dots, L_N\}$. (We do not count the initial ancillary projector $\mathbf{\Pi}_+^{(\text{anc})}$ in the sequence lengths L_j , since each prior step ends with such a projector.) Our strategy will be to start from a projector sequence obtained by naively taking the product of generating gates' projector sequences and then iteratively reducing the combined sequence length via the reduction formulas outlined in Sec. A.2.1.

The protocol for doing this is as follows:

- 1) For each expression $G = G_{j_m} \dots G_{j_1}$ of the target gate in terms of the generating gates,

concatenate the corresponding projector sequences to obtain a projector sequence

$$\mathcal{G} = \mathcal{G}_{j_m} \dots \mathcal{G}_{j_1} \quad (\text{A.31})$$

that compiles to G . The resulting projector sequence has length $L = \sum_{q=1}^m L_{j_q}$.

- 2) Find all alternate compilations for each generator's projector sequence \mathcal{G}_{j_q} , using the methods of Sec. A.2.2. We denote the distinct projector sequences for generator G_{j_q} as $\mathcal{G}_{j_q}^{(\alpha_q)}$ for $\alpha_q = 1, \dots, K_{j_q}$, where K_{j_q} is the number of distinct sequences. Construct all possible projector sequences (up to scalar factors) by independently replacing each \mathcal{G}_{j_q} in the sequence with the alternates $\mathcal{G}_{j_q}^{(\alpha_q)}$ to get

$$\mathcal{G}^{(\vec{\alpha})} = \mathcal{G}_{j_m}^{(\alpha_m)} \dots \mathcal{G}_{j_1}^{(\alpha_1)}, \quad (\text{A.32})$$

where $\vec{\alpha}$ is used to label the different compilations. In this way, we have produced a (naïve) total of $\prod_{q=1}^m K_{j_q}$ possible compilations for G .

- 3) For each $\mathcal{G}^{(\vec{\alpha})}$, search for and apply all possible reductions of each sequence $\mathcal{G}^{(\vec{\alpha})}$ via the reduction formulas introduced in Sec. A.2.1. Repeat until no further reduction is possible. Each reduction will lower the length of the overall sequence by 1-2 projectors. We denote the fully reduced projector sequence obtained from a projector sequence \mathcal{G} as $\check{\mathcal{G}}$.

If each generator's projector sequence is already fully reduced, i.e. $\mathcal{G}_{j_q}^{(\alpha_q)} = \check{\mathcal{G}}_{j_q}^{(\alpha_q)}$, the

remaining reductions will be found at the locations in the projector sequence where the generator subsequences are concatenated (at least at the initial reduction iterations).

Gates compiled from a large number of generator gates can obtain a significant reduction in the length of their projector sequence using this procedure. Note that single-qubit gates acting on different qubits only benefit from the reduction procedure applied individually within qubits, but it is possible to obtain collective reductions for combinations of single-qubit and two-qubit gates.

As an example, let us apply the reduction procedure to the compilation $\mathcal{G} \propto \Pi_+^{(\text{anc})} \otimes C(Z)$, using the gate compilation $C(Z) = S_2 S_1 W$ and the generating gate projector sequences from Ref. [41]:

$$\mathcal{S}_1 = \Pi_+^{(34)} \Pi_+^{(13)} \Pi_+^{(23)}, \quad (\text{A.33})$$

$$\mathcal{S}_2 = \Pi_+^{(3'4')} \Pi_+^{(1'3')} \Pi_+^{(2'3')}, \quad (\text{A.34})$$

$$\mathcal{W} = \Pi_+^{(34)} \Pi_+^{(35)} \Pi_+^{(56;1'2')} \Pi_+^{(45)}. \quad (\text{A.35})$$

$$(\text{A.36})$$

The naïve compilation $\mathcal{G} = S_2 S_1 W$ has length 10. Applying the reduction procedure, we

obtain a compilation of length 8, as follows:

$$\begin{aligned}
\mathcal{G} &= \left[\Pi_+^{(3'4')} \Pi_+^{(1'3')} \Pi_+^{(2'3')} \right] \left[\Pi_+^{(34)} \Pi_+^{(13)} \Pi_+^{(23)} \right] \left[\Pi_+^{(34)} \Pi_+^{(35)} \Pi_+^{(56;1'2')} \Pi_+^{(45)} \right] \\
&\rightarrow \left[\Pi_+^{(3'4')} \Pi_+^{(1'3')} \Pi_+^{(2'3')} \right] \left[\Pi_+^{(34)} \Pi_-^{(36)} \Pi_+^{(35)} \right] \left[\Pi_+^{(34)} \Pi_+^{(35)} \Pi_+^{(56;1'2')} \Pi_+^{(45)} \right] \\
&\rightarrow \Pi_+^{(3'4')} \Pi_+^{(1'3')} \Pi_+^{(2'3')} \Pi_+^{(34)} \Pi_-^{(36)} \Pi_+^{(35)} \Pi_+^{(56;1'2')} \Pi_+^{(45)}. \tag{A.37}
\end{aligned}$$

In the first step, we replaced \mathcal{S}_1 with the alternate compilation $\tilde{\mathcal{S}}_1 = \Pi_+^{(34)} \Pi_-^{(36)} \Pi_+^{(35)}$. In the second step, we applied the reduction formula $\Pi_+^{(35)} \Pi_+^{(34)} \Pi_+^{(35)} \rightarrow \Pi_+^{(35)}$. For this example, a more thorough search would have yielded a better result, as one can find more efficient compilations of both W and $C(Z)$, with lengths 3 and 4, respectively, given by

$$W = \Pi_+^{(34)} \Pi_+^{(35)} \Pi_+^{(36;1'2')} \tag{A.38}$$

$$C(Z) = \Pi_+^{(34)} \Pi_+^{(46)} \Pi_+^{(56)} \Pi_+^{(46;1'2')} \tag{A.39}$$

However, while brute-force search may be employed in this example, it is not practical to do so in general, as the space of projector sequences grows exponentially in the sequence length. Furthermore, one can in principle consider other cost functions to optimize against, for example taking into account that some projectors can be applied simultaneously and that some measurements may be more difficult than others. We will discuss these points in more detail in Sec. 3.2.4.

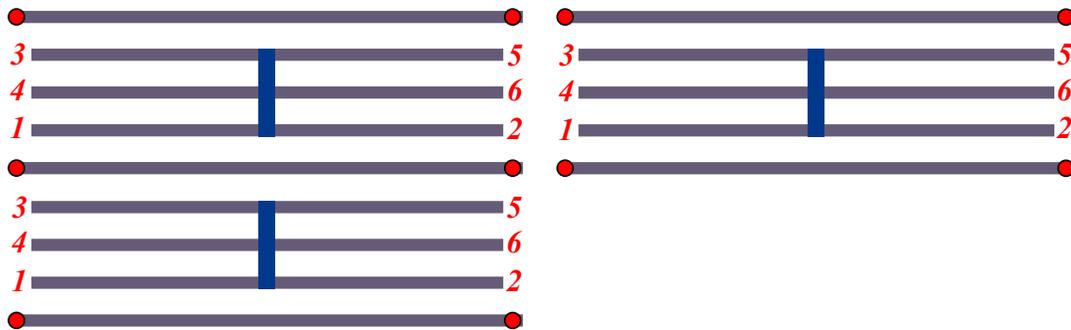
A.3 Demonstration Details

In this appendix, we present the details of the demonstration of our methods outlined in Sec. 3.5.3 for the very roughly estimated weight factor values: $w_c = 1.25$, $w_t = 1.65$, $w_a = 1.01$, and $f(N) = (\prod_{n=1}^N n!)^{(N-1)!}$.

The two options for forced-measurement operations are described in Sec. 3.3.

A.3.1 Two-Sided Hexon with Configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$

The MZM labeling configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$ is optimal for the two-sided hexon architecture, when using either the forced-measurement methods or the Majorana-Pauli tracking methods, for the gates or Pauli cosets of gates: the Hadamard gate H , the geometric average of single-qubit Clifford gates C_1 , the controlled- X gate $C(X)$, and the geometric average of the controlled-Pauli gates $C(P)$. This configuration within an array looks like:



Gate	Forced Measurement Sequence	Weight	$\langle S, H \rangle$	Weight	$\langle S, B \rangle$	Weight
X	$\prod_{+}^{(34)} \prod_{s_3}^{(36)} \prod_{-}^{(34)} \prod_{s_1}^{(14)}$	3.67×10^5	$HSSH$	1.38×10^{26}	BB	1.82×10^{10}
Y	$\prod_{+}^{(34)} \prod_{s_3}^{(35)} \prod_{-}^{(34)} \prod_{s_1}^{(14)}$	2.30×10^5	$HSSHSS$	9.85×10^{35}	$BBSS$	1.30×10^{20}
Z	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{s_2}^{(12)} \prod_{s_1}^{(14)}$	2.30×10^5	SS	7.14×10^9	SS	7.14×10^9
S	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(24)}$	8.45×10^4	S	8.45×10^4	S	8.45×10^4
XS	$\prod_{+}^{(34)} \prod_{s_4}^{(35)} \prod_{-}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(24)}$	1.39×10^8	$HSSHHS$	1.17×10^{31}	BBS	1.54×10^{15}
YS	$\prod_{+}^{(34)} \prod_{s_4}^{(35)} \prod_{-}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(24)}$	1.39×10^8	$HSSHHS^\dagger$	1.17×10^{31}	SBB	1.54×10^{15}
ZS	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{s_1}^{(24)}$	8.45×10^4	S^\dagger	8.45×10^4	S^\dagger	8.45×10^4
H	$\prod_{+}^{(34)} \prod_{-s_1}^{(35)} \prod_{s_2}^{(56)} \prod_{s_2}^{(25)} \prod_{s_1}^{(35)}$	1.39×10^8	H	1.39×10^8	SBS	9.64×10^{14}
XH	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{-s_1}^{(35)}$	1.07×10^5	HSS	9.92×10^{17}	$S^\dagger B^\dagger S$	9.64×10^{14}
YH	$\prod_{+}^{(34)} \prod_{-s_1}^{(35)} \prod_{-s_2}^{(56)} \prod_{s_2}^{(25)} \prod_{s_1}^{(35)}$	1.39×10^8	$SSHSS$	7.09×10^{27}	$SB^\dagger S$	9.64×10^{14}
ZH	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(35)}$	1.07×10^5	SSH	9.92×10^{17}	$S^\dagger BS$	9.64×10^{14}
SH	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{-s_1}^{(24)} \prod_{s_1}^{(46)}$	8.16×10^7	SH	1.17×10^{13}	$B^\dagger S^\dagger$	1.14×10^{10}
XSH	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{s_1}^{(24)} \prod_{s_1}^{(46)}$	8.16×10^7	$S^\dagger HSS$	8.39×10^{22}	BS^\dagger	1.14×10^{10}
YSH	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(24)} \prod_{s_1}^{(46)}$	8.16×10^7	$SHSS$	8.39×10^{22}	$B^\dagger S$	1.14×10^{10}
ZSH	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{-s_1}^{(24)} \prod_{s_1}^{(46)}$	8.16×10^7	$S^\dagger H$	1.17×10^{13}	BS	1.14×10^{10}
HS	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{-s_1}^{(35)} \prod_{s_1}^{(36)}$	6.46×10^7	HS	1.17×10^{13}	$S^\dagger B^\dagger$	1.14×10^{10}
XHS	$\prod_{+}^{(34)} \prod_{-s_1}^{(13)} \prod_{s_1}^{(35)} \prod_{s_1}^{(36)}$	6.46×10^7	HS^\dagger	1.17×10^{13}	SB	1.14×10^{10}
YHS	$\prod_{+}^{(34)} \prod_{-s_1}^{(13)} \prod_{-s_1}^{(35)} \prod_{s_1}^{(36)}$	6.46×10^7	$SSHHS^\dagger$	8.39×10^{22}	$S^\dagger B$	1.14×10^{10}
ZHS	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(35)} \prod_{s_1}^{(36)}$	6.46×10^7	$SSHHS$	8.39×10^{22}	SB^\dagger	1.14×10^{10}
SHS	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(46)}$	1.35×10^5	SHS	9.92×10^{17}	B^\dagger	1.35×10^5
$XSHS$	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{s_1}^{(46)}$	1.35×10^5	$S^\dagger HS^\dagger$	9.92×10^{17}	B	1.35×10^5
$YSHS$	$\prod_{+}^{(34)} \prod_{-s_2 s_1}^{(14)} \prod_{-s_2}^{(12)} \prod_{s_2}^{(26)} \prod_{s_1}^{(23)}$	1.76×10^8	SHS^\dagger	9.92×10^{17}	$B^\dagger SS$	9.64×10^{14}
$ZSHS$	$\prod_{+}^{(34)} \prod_{s_2 s_1}^{(14)} \prod_{s_2}^{(12)} \prod_{s_2}^{(26)} \prod_{s_1}^{(23)}$	1.76×10^8	$S^\dagger HS$	9.92×10^{17}	BSS	9.64×10^{14}
Average		7.72×10^6		6.04×10^{18}		2.25×10^{11}

Table A.1: Minimal difficulty weight measurement sequences for each single-qubit Clifford gate when using forced-measurement methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H \rangle$ Decomp.	Weight	$\langle S, B \rangle$ Decomp.	Weight
$[S]$	$\prod_{s_3}^{(34)} \prod_{s_2}^{(24)} \prod_{s_1}^{(14)}$	1.76×10^2	S	1.76×10^2	S	1.76×10^2
$[H]$	$\prod_{s_3}^{(34)} \prod_{s_2}^{(35)} \prod_{s_1}^{(13)}$	1.76×10^2	H	1.76×10^2	SBS	6.88×10^6
$[SH]$	$\prod_{s_4}^{(34)} \prod_{s_3}^{(36)} \prod_{s_2}^{(35)} \prod_{s_1}^{(13)}$	2.63×10^3	SH	3.10×10^4	BS	3.91×10^4
$[HS]$	$\prod_{s_4}^{(34)} \prod_{s_3}^{(36)} \prod_{s_2}^{(13)} \prod_{s_1}^{(35)}$	2.63×10^3	HS	3.10×10^4	SB	3.91×10^4
$[SHS]$	$\prod_{s_4}^{(34)} \prod_{s_3}^{(46)} \prod_{s_2}^{(14)}$	2.22×10^2	SHS	5.45×10^6	B	2.22×10^2
Average		5.44×10^2		1.10×10^4		1.33×10^4

Table A.2: Minimal difficulty weight measurement sequences for each Pauli coset of single-qubit Clifford gates when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Gate	Tracked Measurement Sequence
$Y \frac{1-s_3 s_0}{2} Z \frac{1-s_2 s_1}{2} S$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(24)} \Pi_{s_0}^{(34)}$
$Y \frac{1-s_3 s_0}{2} Y \frac{1-s_2 s_1 s_0}{2} ZH$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(13)} \Pi_{s_1}^{(35)} \Pi_{s_0}^{(34)}$
$Z \frac{1-s_3 s_0}{2} Z \frac{1-s_3 s_2 s_0}{2} X \frac{1-s_2 s_1 s_0}{2} XSH$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(36)} \Pi_{s_2}^{(35)} \Pi_{s_1}^{(13)} \Pi_{s_0}^{(34)}$
$Y \frac{1-s_3 s_0}{2} Y \frac{1-s_3 s_2 s_0}{2} X \frac{1-s_2 s_1 s_0}{2} ZHS$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(13)} \Pi_{s_2}^{(35)} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(34)}$
$Y \frac{1-s_3 s_0}{2} X \frac{1-s_2 s_1}{2} SHS$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(46)} \Pi_{s_0}^{(34)}$

Table A.3: Pauli gate corrections tracked for the corresponding single-qubit gates implemented using Majorana-Pauli tracking methods. The implicit action on the ancillary qubit is $X \frac{1-s_n s_0}{2} \Pi_{s_0}^{(34)}$ for sequences of length n .

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
$C(X)$ (u)	$\prod_{+}^{(34)} \prod_{+}^{(35)} \prod_{+}^{(56)} \prod_{+}^{(35;1'6')}$	1.79×10^9	${}^H C(Z)$	2.16×10^{25}	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	2.56×10^{36}
(d)	$\prod_{+}^{(3'4')} \prod_{+}^{(3'5')} \prod_{+}^{(2'5')} \prod_{+}^{(12;3'5')}$	2.26×10^9		2.16×10^{25}		2.56×10^{36}
(r)	$\prod_{+}^{(3'4')} \prod_{-s}^{(1'4')} \prod_{+}^{(56;3'4')} \prod_{+}^{(3'6')}$	3.69×10^8		8.89×10^{24}		4.88×10^{36}
(l)	$\prod_{+}^{(34)} \prod_{-s}^{(14)} \prod_{+}^{(12)} \prod_{+}^{(14;2'5')}$	2.88×10^8		8.89×10^{24}		4.88×10^{36}
(a)	$\prod_{+}^{(34)} \prod_{-s}^{(14)} \prod_{+}^{(12)} \prod_{+}^{(14;2'5')}$	8.10×10^8		1.39×10^{25}		3.54×10^{36}
$C(Y)$ (u)	$\prod_{+}^{(34)} \prod_{+}^{(35)} \prod_{+}^{(56)} \prod_{+}^{(35;1'5')}$	2.85×10^9	${}^{SH} C(Z)$	1.55×10^{35}	$B_2^\dagger W^\dagger S_2 B_2 S_1$	3.59×10^{26}
(d)	$\prod_{+}^{(3'4')} \prod_{+}^{(1'4')} \prod_{+}^{(1'5')} \prod_{+}^{(12;3'5')}$	3.60×10^9		1.55×10^{35}		3.59×10^{26}
(r)	$\prod_{+}^{(3'4')} \prod_{+}^{(1'4')} \prod_{+}^{(56;3'4')} \prod_{+}^{(3'5')}$	2.32×10^8		6.34×10^{34}		6.83×10^{26}
(l)	$\prod_{+}^{(34)} \prod_{-s}^{(14)} \prod_{+}^{(12)} \prod_{+}^{(14;2'6')}$	1.81×10^8		6.34×10^{34}		6.83×10^{26}
(a)	$\prod_{+}^{(34)} \prod_{-s}^{(14)} \prod_{+}^{(12)} \prod_{+}^{(14;2'6')}$	8.10×10^8		9.91×10^{34}		4.95×10^{26}
$C(Z)$ (u)	$\prod_{+}^{(34)} \prod_{+}^{(35)} \prod_{+}^{(56)} \prod_{+}^{(35;1'2')}$	1.12×10^9	$C(Z)$	1.12×10^9	$S_1 S_2 W^\dagger$	1.97×10^{16}
(d)	$\prod_{+}^{(3'4')} \prod_{+}^{(3'5')} \prod_{+}^{(5'6')} \prod_{+}^{(12;3'5')}$	1.12×10^9		1.12×10^9		1.97×10^{16}
(r)	$\prod_{+}^{(3'4')} \prod_{+}^{(1'3')} \prod_{+}^{(1'2')} \prod_{+}^{(56;1'3')}$	4.60×10^8		4.60×10^8		3.75×10^{16}
(l)	$\prod_{+}^{(34)} \prod_{+}^{(13)} \prod_{+}^{(12)} \prod_{+}^{(13;5'6')}$	4.60×10^8		4.60×10^8		3.75×10^{16}
(a)	$\prod_{+}^{(34)} \prod_{+}^{(13)} \prod_{+}^{(12)} \prod_{+}^{(13;5'6')}$	7.18×10^8		7.18×10^8		2.72×10^{16}
Average (u)		1.79×10^9		1.55×10^{23}		2.63×10^{26}
(d)		2.09×10^9		1.55×10^{23}		2.63×10^{26}
(r)		3.40×10^8		6.38×10^{22}		5.00×10^{26}
(l)		2.88×10^8		6.38×10^{22}		5.00×10^{26}
(a)		7.78×10^8		9.96×10^{22}		3.62×10^{26}

Table A.4: Minimal difficulty weight measurement sequences for controlled-Pauli two-qubit gates when using forced-measurement methods. The labels (u), (d), (r), and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r), and (l). The average difficulty weight of the four directions is labeled by (a). For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight			
[C(X)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'6')}$	${}^H C(Z)$	1.63×10^8	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	4.23×10^{16}			
	(d)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3'}^{(3'5')} \Pi_{s_2'}^{(2'5')} \Pi_{s_1'}^{(12;3'5')}$					6.64×10^3	1.63×10^8	4.23×10^{16}
	(r)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(3'6')}$					2.66×10^3	1.04×10^8	5.86×10^{16}
	(l)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;2'5')}$					2.66×10^3	1.04×10^8	5.86×10^{16}
	(a)	4.20×10^3					1.30×10^8	4.98×10^{16}	
[C(Y)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'5')}$	${}^{SH} C(Z)$	5.04×10^{12}	$B_2^\dagger W^\dagger S_2 B_2 S_1$	1.37×10^{12}			
	(d)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3'}^{(3'5')} \Pi_{s_2'}^{(1,2;1'5')} \Pi_{s_1'}^{(1'4')}$					8.38×10^3	5.04×10^{12}	1.37×10^{12}
	(r)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(3'5')}$					2.11×10^3	3.23×10^{12}	1.89×10^{12}
	(l)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;2'6')}$					2.11×10^3	3.23×10^{12}	1.89×10^{12}
	(a)	4.20×10^3					4.04×10^{12}	1.61×10^{12}	
[C(Z)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'2')}$	$C(Z)$	5.26×10^3	$S_1 S_2 W^\dagger$	2.77×10^7			
	(d)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3'}^{(3'5')} \Pi_{s_2'}^{(5'6')} \Pi_{s_1'}^{(12;3'5')}$					5.26×10^3	5.26×10^3	2.77×10^7
	(r)	$\Pi_{s_4'}^{(3'4')} \Pi_{s_3}^{(2'3')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(1'4')}$					3.36×10^3	3.36×10^3	3.84×10^7
	(r)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(34;5'6')} \Pi_{s_1}^{(14)}$					3.36×10^3	3.36×10^3	3.84×10^7
	(a)	4.20×10^3					4.20×10^3	3.26×10^7	
Average	(u)	6.64×10^3		1.63×10^8		1.17×10^{12}			
	(d)	6.64×10^3		1.63×10^8		1.17×10^{12}			
	(r)	2.66×10^3		1.04×10^8		1.62×10^{12}			
	(l)	2.66×10^3		1.04×10^8		1.62×10^{12}			
	(a)	4.20×10^3		1.30×10^8		1.38×10^{12}			

Table A.5: Minimal difficulty weight measurement sequences for Pauli cosets of controlled–Pauli two-qubit gates when using Majorana–Pauli tracking methods. The labels (u) , (d) , (r) , and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r) , and (l) . The average difficulty weight of the four directions is labeled by (a) . For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Gate	Tracked Measurement Sequence
$\left(\begin{array}{l} Z^{\frac{1-s_3 s_1}{2}} \otimes X^{\frac{1-s_2 s_0}{2}} \\ Z^{\frac{1-s_2 s'_0}{2}} \otimes X^{\frac{1-s_3 s_1}{2}} \end{array} \right) C(X)_u$ $\left(\begin{array}{l} Z^{\frac{1-s_2 s'_0}{2}} \otimes X^{\frac{1-s_3 s_1}{2}} \\ Z^{\frac{1+s_3 s_1 s'_0}{2}} \otimes Y^{\frac{1-s'_4 s'_0}{2}} X^{\frac{1-s_2 s_0 s'_0}{2}} \end{array} \right) C(X)_d$ $\left(\begin{array}{l} Z^{\frac{1+s_3 s_1 s'_0}{2}} \otimes Y^{\frac{1-s'_4 s'_0}{2}} X^{\frac{1-s_2 s_0 s'_0}{2}} \\ Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes X^{\frac{1-s_2}{2}} \end{array} \right) C(X)_r$ $\left(\begin{array}{l} Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes X^{\frac{1-s_2}{2}} \\ Y^{\frac{1-s_3 s_1 s'_0}{2}} \otimes Y^{\frac{1-s_2 s_0}{2}} \end{array} \right) C(X)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'6')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'5')} \Pi_{s_2}^{(2'5')} \Pi_{s_1}^{(12;3'5')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(3'6')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;2'5')} \Pi_{s_0}^{(\text{anc})}$
$\left(\begin{array}{l} Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes Y^{\frac{1-s_2 s_0}{2}} \\ Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1+s_2 s'_0}{2}} \end{array} \right) C(Y)_u$ $\left(\begin{array}{l} Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1+s_2 s'_0}{2}} \\ Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1-s'_4 s_2 s_0}{2}} \end{array} \right) C(Y)_d$ $\left(\begin{array}{l} Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1-s'_4 s_2 s_0}{2}} \\ Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1-s_2}{2}} \end{array} \right) C(Y)_r$ $\left(\begin{array}{l} Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1-s_2}{2}} \\ Y^{\frac{1-s_3 s_1 s'_0}{2}} \otimes Y^{\frac{1-s_2 s_0}{2}} \end{array} \right) C(Y)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'5')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'5')} \Pi_{s_2}^{(1,2;1'5')} \Pi_{s_1}^{(1'4')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(3'5')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;2'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(\begin{array}{l} Z^{\frac{1-s_3 s_1}{2}} \otimes Z^{\frac{1-s_2 s_0}{2}} \\ Z^{\frac{1-s_2 s'_0}{2}} \otimes Z^{\frac{1-s_3 s_1}{2}} \end{array} \right) C(Z)_u$ $\left(\begin{array}{l} Z^{\frac{1-s_2 s'_0}{2}} \otimes Z^{\frac{1-s_3 s_1}{2}} \\ Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes X^{\frac{1-s'_4 s'_0}{2}} Z^{\frac{1-s_2 s_0 s'_0}{2}} \end{array} \right) C(Z)_d$ $\left(\begin{array}{l} Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes X^{\frac{1-s'_4 s'_0}{2}} Z^{\frac{1-s_2 s_0 s'_0}{2}} \\ X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_2 s_0 s'_0}{2}} \otimes Z^{\frac{1-s_3 s_1 s_0}{2}} \end{array} \right) C(Z)_r$ $\left(\begin{array}{l} X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_2 s_0 s'_0}{2}} \otimes Z^{\frac{1-s_3 s_1 s_0}{2}} \\ X^{\frac{1-s_3 s_1 s'_0}{2}} \otimes Z^{\frac{1-s_2 s_0}{2}} \end{array} \right) C(Z)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(56)} \Pi_{s_1}^{(35;1'2')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'5')} \Pi_{s_2}^{(5'6')} \Pi_{s_1}^{(12;3'5')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(2'3')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(1'4')} \Pi_{s_0}^{(\text{anc})}$ $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(34;5'6')} \Pi_{s_1}^{(14)} \Pi_{s_0}^{(\text{anc})}$

Table A.6: Pauli gate corrections tracked for the corresponding controlled-Pauli gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X^{\frac{1-s_n s_0}{2}} \Pi_{s_0}^{(34)} \otimes X^{\frac{1-s'_n s'_0}{2}} \Pi_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
SWAP (u)	$\prod_{+}^{(34)} \prod_{-s_1}^{(14)} \prod_{+}^{(16;1'6')} \prod_{s_1}^{(35;1'5')}$	1.20×10^{12}	$C(X)^3$	1.01×10^{76}	$C(X)^3$	1.68×10^{109}
(d)	$\prod_{+}^{(3'4')} \prod_{-s_1}^{(1'4')} \prod_{+}^{(16;1'6')} \prod_{s_1}^{(15;3'5')}$	1.20×10^{12}		1.01×10^{76}		1.68×10^{109}
(r)	$\prod_{+}^{(3'4')} \prod_{s_1}^{(1'4')} \prod_{+}^{(26;1'5')} \prod_{s_1}^{(56;2'3')}$	1.87×10^{12}		7.03×10^{74}		1.16×10^{110}
(l)	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{+}^{(15;2'6')} \prod_{s_1}^{(23;5'6')}$	1.87×10^{12}		7.03×10^{74}		1.16×10^{110}
(a)	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{+}^{(15;2'6')} \prod_{s_1}^{(23;5'6')}$	1.50×10^{12}		2.66×10^{75}		4.42×10^{109}
W (u)	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(23;1'2')}$	2.76×10^6	$S_1 S_2 C(Z)$	8.00×10^{18}	W	2.76×10^6
(d)	$\prod_{+}^{(3'4')} \prod_{s_1}^{(1'3')} \prod_{s_1}^{(12;2'3')}$	2.76×10^6		8.00×10^{18}		2.76×10^6
(r)	$\prod_{+}^{(3'4')} \prod_{s_1}^{(1'3')} \prod_{s_1}^{(56;2'3')}$	5.25×10^6		3.28×10^{18}		5.25×10^6
(l)	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(23;5'6')}$	5.25×10^6		3.28×10^{18}		5.25×10^6
(a)	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(23;5'6')}$	3.81×10^6		5.13×10^{18}		3.81×10^6
Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
[SWAP] (u)	$\prod_{s_4}^{(34)} \prod_{s_3}^{(14)} \prod_{s_2}^{(15;1'5')} \prod_{s_1}^{(23;1'2')}$	6.79×10^4	$C(X)^3$	4.33×10^{24}	$C(X)^3$	7.57×10^{49}
(d)	$\prod_{s_4}^{(3'4')} \prod_{s_3}^{(1'4')} \prod_{s_2}^{(15;1'5')} \prod_{s_1}^{(12;2'3')}$	6.79×10^4		4.33×10^{24}		7.57×10^{49}
(r)	$\prod_{s_4}^{(3'4')} \prod_{s_3}^{(1'4')} \prod_{s_2}^{(26;1'5')} \prod_{s_1}^{(56;2'3')}$	8.11×10^4		1.12×10^{24}		2.01×10^{50}
(l)	$\prod_{s_4}^{(34)} \prod_{s_3}^{(14)} \prod_{s_2}^{(15;2'6')} \prod_{s_1}^{(23;5'6')}$	8.11×10^4		1.12×10^{24}		2.01×10^{50}
(a)	$\prod_{s_4}^{(34)} \prod_{s_3}^{(14)} \prod_{s_2}^{(15;2'6')} \prod_{s_1}^{(23;5'6')}$	7.42×10^4		2.21×10^{24}		1.23×10^{50}
[W] (u)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(13)} \prod_{s_1}^{(23;1'2')}$	8.95×10^2	$S_1 S_2 C(Z)$	1.63×10^8	W	8.95×10^2
(d)	$\prod_{s_3}^{(3'4')} \prod_{s_2}^{(1'3')} \prod_{s_1}^{(12;2'3')}$	8.95×10^2		1.63×10^8		8.95×10^2
(r)	$\prod_{s_3}^{(3'4')} \prod_{s_2}^{(2'3')} \prod_{s_1}^{(56;1'3')}$	1.24×10^3		1.04×10^8		1.24×10^3
(l)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(23)} \prod_{s_1}^{(13;5'6')}$	1.24×10^3		1.04×10^8		1.24×10^3
(a)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(23)} \prod_{s_1}^{(13;5'6')}$	1.05×10^3		1.30×10^8		1.05×10^3

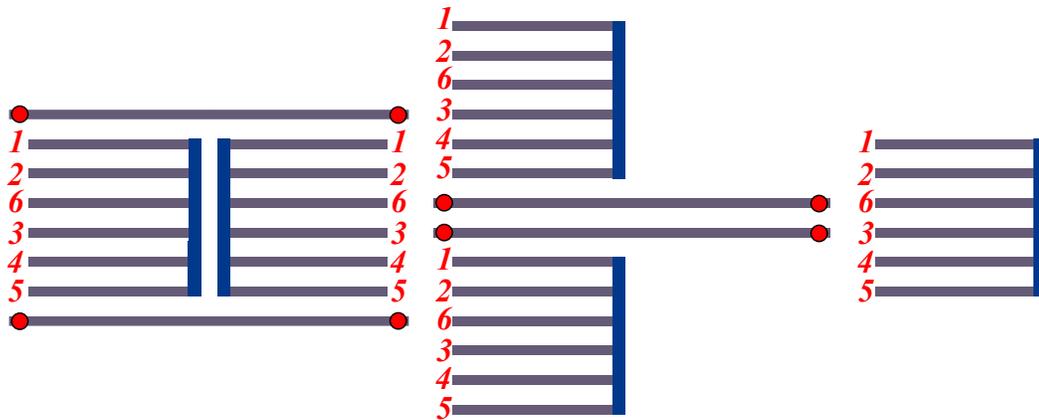
Table A.7: Minimal difficulty weight measurement sequences for the two-qubit SWAP and W gates when using forced-measurement methods and the Pauli cosets of SWAP and W when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using $\text{SWAP} = C(X)_{12}C(X)_{21}C(X)_{12}$ with $C(X)_{12}$ as given in Tables A.4 and A.5.

Gate	Tracked Measurement Sequence
$(Y \otimes \mathbb{1}) \frac{1-s_4 s_0}{2} (Z \otimes Z) \frac{1-s_2 s_0 s'_0}{2} (Y \otimes Y) \frac{1-s_3 s_1 s_0}{2}$ SWAP $_u$	$\prod_{s_4}^{(34)} \prod_{s_3}^{(14)} \prod_{s_2}^{(15;1'5')} \prod_{s_1}^{(23;1'2')} \prod_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Y) \frac{1-s'_4 s'_0}{2} (Z \otimes Z) \frac{1-s_2 s_0 s'_0}{2} (Y \otimes Y) \frac{1-s_3 s_1 s'_0}{2}$ SWAP $_d$	$\prod_{s'_4}^{(3'4')} \prod_{s_3}^{(1'4')} \prod_{s_2}^{(15;1'5')} \prod_{s_1}^{(12;2'3')} \prod_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Y) \frac{1-s'_4 s'_0}{2} (Z \otimes Z) \frac{1+s_2 s'_0}{2} (Y \otimes Y) \frac{1-s_3 s_1 s_0 s'_0}{2}$ SWAP $_r$	$\prod_{s'_4}^{(3'4')} \prod_{s_3}^{(1'4')} \prod_{s_2}^{(26;1'5')} \prod_{s_1}^{(56;2'3')} \prod_{s_0}^{(\text{anc})}$
$(Y \otimes \mathbb{1}) \frac{1-s_4 s_0}{2} (Z \otimes Z) \frac{1+s_2 s_0}{2} (Y \otimes Y) \frac{1-s_3 s_1 s_0 s'_0}{2}$ SWAP $_l$	$\prod_{s_4}^{(34)} \prod_{s_3}^{(14)} \prod_{s_2}^{(15;2'6')} \prod_{s_1}^{(23;5'6')} \prod_{s_0}^{(\text{anc})}$
$(Y \otimes \mathbb{1}) \frac{1-s_3 s_0}{2} (Z \otimes Z) \frac{1-s_2 s_1}{2}$ W_u	$\prod_{s_3}^{(34)} \prod_{s_2}^{(13)} \prod_{s_1}^{(23;1'2')} \prod_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Y) \frac{1-s'_3 s'_0}{2} (Z \otimes Z) \frac{1-s_2 s_1}{2}$ W_d	$\prod_{s'_3}^{(3'4')} \prod_{s_2}^{(1'3')} \prod_{s_1}^{(12;2'3')} \prod_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes X) \frac{1-s'_3 s'_0}{2} (Z \otimes Z) \frac{1+s_2 s_1 s_0}{2}$ W_r	$\prod_{s'_3}^{(3'4')} \prod_{s_2}^{(2'3')} \prod_{s_1}^{(56;1'3')} \prod_{s_0}^{(\text{anc})}$
$(X \otimes \mathbb{1}) \frac{1-s_3 s_0}{2} (Z \otimes Z) \frac{1+s_2 s_1 s'_0}{2}$ W_l	$\prod_{s_3}^{(34)} \prod_{s_2}^{(23)} \prod_{s_1}^{(13;5'6')} \prod_{s_0}^{(\text{anc})}$

Table A.8: Pauli gate corrections tracked for the corresponding SWAP gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X \frac{1-s_n s_0}{2} \prod_{s_0}^{(34)} \otimes X \frac{1-s'_n s'_0}{2} \prod_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

A.3.2 One-Sided Hexon with Configuration $\langle 1, 2, 6, 3, 4, 5 \rangle$

The MZM labeling configuration $\langle 1, 2, 6, 3, 4, 5 \rangle$ is optimal for the one-sided hexon architecture, when using the forced-measurement methods for the gates: the Hadamard gate H , the geometric average of single-qubit Clifford gates C_1 ; or when using the Majorana-Pauli tracking methods, for the Pauli cosets of gates: the Hadamard gate H , the geometric average of single-qubit Clifford gates C_1 , the controlled- X gate $C(X)$, and the geometric average of the controlled-Pauli gates $C(P)$. This configuration within an array looks like:



Gate	Forced Measurement Sequence	Weight	$\langle S, H \rangle$	Weight	$\langle S, B \rangle$	Weight
X	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(45)} \widehat{\Pi}_{s_1}^{(34)} \widehat{\Pi}_{s_1}^{(23)}$	3.10×10^4	$HSSH$	8.14×10^{19}	BB	1.30×10^8
Y	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_2}^{(26)} \widehat{\Pi}_{s_1}^{(36)}$	1.95×10^4	$HSSHSS$	6.64×10^{27}	$BBSS$	1.06×10^6
Z	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(36)} \widehat{\Pi}_{-s_1}^{(34)} \widehat{\Pi}_{s_1}^{(45)}$	1.95×10^4	SS	8.15×10^7	SS	8.15×10^7
S	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(46)} \widehat{\Pi}_{s_1}^{(45)}$	9.03×10^3	S	9.03×10^3	S	9.03×10^3
XS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_2}^{(26)} \widehat{\Pi}_{s_1}^{(16)} \widehat{\Pi}_{s_1}^{(36)}$	9.99×10^5	$HSSHHS$	7.35×10^{23}	BBS	1.17×10^{12}
YS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{-s_2}^{(26)} \widehat{\Pi}_{s_2}^{(16)} \widehat{\Pi}_{s_1}^{(36)}$	9.99×10^5	$SHSSH$	7.35×10^{23}	SBB	1.17×10^{12}
ZS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(46)} \widehat{\Pi}_{s_1}^{(45)}$	9.03×10^3	S^\dagger	9.03×10^3	S^\dagger	9.03×10^3
H	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(45)} \widehat{\Pi}_{-s_1}^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	9.99×10^5	H	9.99×10^5	SBS	9.30×10^{11}
XH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	7.16×10^3	HSS	8.15×10^{13}	$S^\dagger B^\dagger S$	9.30×10^{11}
YH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(45)} \widehat{\Pi}_{-s_1}^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	9.99×10^5	$SSHSS$	6.64×10^{21}	$SB^\dagger S$	9.30×10^{11}
ZH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_2}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	7.16×10^3	SSH	8.15×10^{13}	$S^\dagger BS$	9.30×10^{11}
SH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(35)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	4.63×10^5	SH	9.02×10^9	$B^\dagger S^\dagger$	1.03×10^8
XSH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(35)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	4.63×10^5	$S^\dagger HSS$	7.36×10^{17}	BS^\dagger	1.03×10^8
YSH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(35)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	4.63×10^5	$SHSS$	7.36×10^{17}	$B^\dagger S$	1.03×10^8
ZSH	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(35)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	4.63×10^5	$S^\dagger H$	9.02×10^9	BS	1.03×10^8
HS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)} \widehat{\Pi}_{s_1}^{(13)}$	5.85×10^5	HS	9.02×10^9	$S^\dagger B^\dagger$	1.03×10^8
XHS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{-s_1}^{(23)} \widehat{\Pi}_{s_1}^{(13)}$	5.85×10^5	HS^\dagger	9.02×10^9	SB	1.03×10^8
YHS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(23)} \widehat{\Pi}_{s_1}^{(13)}$	5.85×10^5	$SSH S^\dagger$	7.36×10^{17}	$S^\dagger B$	1.03×10^8
ZHS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{-s_1}^{(23)} \widehat{\Pi}_{s_1}^{(13)}$	5.85×10^5	$SSH S$	7.36×10^{17}	SB^\dagger	1.03×10^8
SHS	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_1}^{(13)}$	1.14×10^4	SHS	8.15×10^{13}	B^\dagger	1.14×10^4
$XSHS$	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{s_1}^{(13)}$	1.14×10^4	$S^\dagger HS^\dagger$	8.15×10^{13}	B	1.14×10^4
$YSHS$	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{s_1}^{(36)} \widehat{\Pi}_{-s_2}^{(26)} \widehat{\Pi}_{s_2}^{(12)} \widehat{\Pi}_{s_1}^{(13)}$	1.26×10^6	SHS^\dagger	8.15×10^{13}	$B^\dagger SS$	1.14×10^4
$ZSHS$	$\widehat{\Pi}_+^{(34)} \widehat{\Pi}_{-s_1}^{(36)} \widehat{\Pi}_{s_2}^{(26)} \widehat{\Pi}_{s_2}^{(12)} \widehat{\Pi}_{s_1}^{(13)}$	1.26×10^6	$S^\dagger HS$	8.15×10^{13}	BSS	1.14×10^4
Average		1.45×10^5		3.28×10^{14}		1.12×10^9

Table A.9: Minimal difficulty weight measurement sequences for each single-qubit Clifford gate when using forced-measurement methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Pauli Class	Unforced Measurement Sequence	Weight	$\langle S, H \rangle$ Decomp.	Weight	$\langle S, B \rangle$ Decomp.	Weight
$[S]$	$\widehat{\Pi}_{s_3}^{(34)} \widehat{\Pi}_{s_2}^{(36)} \widehat{\Pi}_{s_1}^{(35)}$	5.13×10^1	S	5.13×10^1	S	5.13×10^1
$[H]$	$\widehat{\Pi}_{s_3}^{(34)} \widehat{\Pi}_{s_2}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	5.13×10^1	H	5.13×10^1	SBS	1.70×10^5
$[SH]$	$\widehat{\Pi}_{s_4}^{(34)} \widehat{\Pi}_{s_3}^{(35)} \widehat{\Pi}_{s_2}^{(36)} \widehat{\Pi}_{s_1}^{(23)}$	2.22×10^2	SH	2.63×10^3	BS	3.32×10^3
$[HS]$	$\widehat{\Pi}_{s_4}^{(34)} \widehat{\Pi}_{s_3}^{(36)} \widehat{\Pi}_{s_2}^{(35)} \widehat{\Pi}_{s_1}^{(23)}$	2.22×10^2	HS	2.63×10^3	SB	3.32×10^3
$[SHS]$	$\widehat{\Pi}_{s_4}^{(34)} \widehat{\Pi}_{s_3}^{(35)} \widehat{\Pi}_{s_2}^{(23)}$	6.47×10^1	SHS	1.35×10^5	B	6.47×10^1
Average		9.66×10^1		1.20×10^3		1.44×10^3

Table A.10: Minimal difficulty weight measurement sequences for each Pauli coset of single-qubit Clifford gates when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Gate	Tracked Measurement Sequence
$Z \frac{1-s_3 s_0}{2} Z \frac{1+s_2 s_1 s_0}{2} S$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(36)} \Pi_{s_1}^{(35)} \Pi_{s_0}^{(34)}$
$Z \frac{1-s_3 s_0}{2} Y \frac{1-s_2 s_1}{2} ZH$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(36)} \Pi_{s_1}^{(23)} \Pi_{s_0}^{(34)}$
$Z \frac{1-s_3 s_2 s_0}{2} X \frac{1-s_2 s_1}{2} ZSH$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(35)} \Pi_{s_2}^{(36)} \Pi_{s_1}^{(23)} \Pi_{s_0}^{(34)}$
$Z \frac{1-s_4 s_0}{2} Z \frac{1-s_3 s_2 s_0}{2} Y \frac{1-s_2 s_1 s_0}{2} YHS$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(36)} \Pi_{s_2}^{(35)} \Pi_{s_1}^{(23)} \Pi_{s_0}^{(34)}$
$X \frac{1+s_2 s_1 s_0}{2} SHS$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(35)} \Pi_{s_1}^{(23)} \Pi_{s_0}^{(34)}$

Table A.11: Pauli gate corrections tracked for the corresponding single-qubit gates implemented using Majorana-Pauli tracking methods. The implicit action on the ancillary qubit is $X \frac{1-s_n s_0}{2} \Pi_{s_0}^{(34)}$ for sequences of length n .

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
$C(X)(u)$	$\overset{+}{\Pi}_+(3'4') \overset{-}{\Pi}_-(4'5') \overset{+}{\Pi}_+(2'5') \overset{-}{\Pi}_-(12;4'5')$	2.88×10^7	${}^H C(Z)$	1.43×10^{19}	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	1.51×10^{29}
(d)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(45) \overset{+}{\Pi}_+(56) \overset{-}{\Pi}_-(45;1'6')$	2.28×10^7		1.43×10^{19}		1.51×10^{29}
(r)	$\overset{+}{\Pi}_+(3'4') \overset{-}{\Pi}_-(3'6') \overset{+}{\Pi}_+(1'6') \overset{-}{\Pi}_-(56;3'6')$	1.30×10^9		2.08×10^{21}		2.19×10^{31}
(l)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(23) \overset{+}{\Pi}_+(12) \overset{-}{\Pi}_-(14;2'5')$	7.76×10^9		1.24×10^{22}		2.64×10^{32}
(a)		2.85×10^8		2.69×10^{20}		3.39×10^{30}
$C(Y)(u)$	$\overset{+}{\Pi}_+(3'4') \overset{-}{\Pi}_-(4'5') \overset{+}{\Pi}_+(1'5') \overset{-}{\Pi}_-(12;4'5')$	5.76×10^7	${}^{SH} C(Z)$	1.16×10^{27}	$B_2^\dagger W^\dagger S_2 B_2 S_1$	1.85×10^{21}
(d)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(45) \overset{+}{\Pi}_+(56) \overset{-}{\Pi}_-(45;1'5')$	9.22×10^7		1.16×10^{27}		1.85×10^{21}
(r)	$\overset{+}{\Pi}_+(3'4') \overset{-}{\Pi}_-(3'6') \overset{+}{\Pi}_+(2'6') \overset{-}{\Pi}_-(56;3'6')$	6.47×10^8		1.69×10^{29}		2.69×10^{23}
(l)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(23) \overset{+}{\Pi}_+(12) \overset{-}{\Pi}_-(14;1'5')$	4.87×10^9		1.01×10^{30}		3.23×10^{24}
(a)		3.60×10^8		4.16×10^{22}		4.15×10^{22}
$C(Z)(u)$	$\overset{+}{\Pi}_+(3'4') \overset{-}{\Pi}_-(4'5') \overset{+}{\Pi}_+(5'6') \overset{-}{\Pi}_-(12;4'5')$	1.43×10^7	$C(Z)$	1.43×10^7	$S_1 S_2 W^\dagger$	1.43×10^{13}
(d)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(45) \overset{+}{\Pi}_+(56) \overset{-}{\Pi}_-(45;1'2')$	1.43×10^7		1.43×10^7		1.43×10^{13}
(r)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(23) \overset{+}{\Pi}_+(12) \overset{-}{\Pi}_-(23;5'6')$	2.08×10^9		2.08×10^9		2.07×10^{15}
(l)	$\overset{+}{\Pi}_+(34) \overset{-}{\Pi}_-(23) \overset{+}{\Pi}_+(12) \overset{-}{\Pi}_-(14;5'6')$	1.24×10^{10}		1.24×10^{10}		2.49×10^{16}
(a)		2.69×10^8		2.69×10^8		3.20×10^{14}
Average(u)		2.87×10^7		6.19×10^{17}		1.59×10^{21}
(d)		3.11×10^7		6.19×10^{17}		1.59×10^{21}
(r)		1.20×10^9		9.01×10^{19}		2.30×10^{23}
(l)		7.77×10^9		5.38×10^{20}		2.77×10^{24}
(a)		3.02×10^8		1.17×10^{19}		3.56×10^{22}

Table A.12: Minimal difficulty weight measurement sequences for controlled-Pauli two-qubit gates when using forced-measurement methods. The labels (u) , (d) , (r) , and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r) , and (l) . The average difficulty weight of the four directions is labeled by (a) . For comparison, we also present the corresponding realization of the gates formed from the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
[C(X)]	(u) $\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(2'5')} \Pi_{s_1}^{(12;4'5')}$	1.24×10^3	${}^H C(Z)$	2.58×10^6	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	6.55×10^{12}
	(d) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'6')} \Pi_{s_1}^{(36)}$	1.24×10^3		2.58×10^6		6.55×10^{12}
	(r) $\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(3'6')} \Pi_{s_2}^{(1'6')} \Pi_{s_1}^{(56;3'6')}$	9.34×10^3		3.11×10^7		7.89×10^{13}
	(l) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;2'5')} \Pi_{s_1}^{(36)}$	2.28×10^4		7.58×10^7		3.07×10^{14}
	(a) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)}$	4.25×10^3		1.12×10^7		3.19×10^{13}
[C(Y)]	(u) $\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(1'5')} \Pi_{s_1}^{(12;4'5')}$	1.56×10^3	${}^{SH} C(Z)$	6.78×10^9	$B_2^\dagger W^\dagger S_2 B_2 S_1$	2.49×10^9
	(d) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)}$	2.49×10^3		6.78×10^9		2.49×10^9
	(r) $\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(3'6')} \Pi_{s_2}^{(2'6')} \Pi_{s_1}^{(56;3'6')}$	7.40×10^3		8.17×10^{10}		3.00×10^{10}
	(l) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)}$	1.81×10^4		1.99×10^{11}		1.17×10^{11}
	(a) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)}$	4.77×10^3		2.94×10^{10}		1.21×10^{10}
[C(Z)]	(u) $\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(12;5'6')} \Pi_{s_1}^{(3'6')}$	9.79×10^2	C(Z)	9.79×10^2	$S_1 S_2 W^\dagger$	5.95×10^5
	(d) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'2')} \Pi_{s_1}^{(36)}$	9.79×10^2		9.79×10^2		5.95×10^5
	(r) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;5'6')}$	1.18×10^4		1.18×10^4		7.16×10^6
	(l) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;5'6')}$	2.88×10^4		2.88×10^4		2.79×10^7
	(a) $\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;5'6')}$	4.25×10^3		4.25×10^3		2.90×10^6
Average	(u)	1.24×10^3		2.58×10^6		2.13×10^9
	(d)	1.45×10^3		2.58×10^6		2.13×10^9
	(r)	9.34×10^3		3.11×10^7		2.57×10^{10}
	(l)	2.28×10^4		7.57×10^7		1.00×10^{11}
	(a)	4.42×10^3		1.12×10^7		1.04×10^{10}

Table A.13: Minimal difficulty weight measurement sequences for Pauli cosets of controlled–Pauli two-qubit gates when using Majorana–Pauli tracking methods. The labels (u) , (d) , (r) , and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r) , and (l) . The average difficulty weight of the four directions is labeled by (a) . For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Gate	Tracked Measurement Sequence
$\left(Z^{\frac{1-s_2 s'_0}{2}} \otimes X^{\frac{1-s_3 s_1}{2}} \right) C(X)_u$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(2'5')} \Pi_{s_1}^{(12;4'5')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2 s_0}{2}} \otimes X^{\frac{1-s_3 s_1 s_0}{2}} \right) C(X)_d$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'6')} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1-s_2}{2}} \otimes Z^{\frac{1-s'_4 s'_0}{2}} X^{\frac{1-s_3 s_1 s_0}{2}} \right) C(X)_r$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'6')} \Pi_{s_2}^{(1'6')} \Pi_{s_1}^{(56;3'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2 s_0 s'_0}{2}} \otimes X^{\frac{1-s_3 s_1 s_0}{2}} \right) C(X)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;2'5')} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1-s_2 s'_0}{2}} \otimes Y^{\frac{1-s_3 s_1}{2}} \right) C(Y)_u$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(1'5')} \Pi_{s_1}^{(12;4'5')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2 s_0 s'_0}{2}} \otimes Y^{\frac{1-s_3 s_1}{2}} \right) C(Y)_d$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2}{2}} Y^{\frac{1-s_3 s_1 s_0}{2}} \otimes Z^{\frac{1-s'_4 s'_0}{2}} \right) C(Y)_r$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'6')} \Pi_{s_2}^{(2'6')} \Pi_{s_1}^{(56;3'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2 s_0 s'_0}{2}} \otimes Y^{\frac{1-s_3 s_1 s_0}{2}} \right) C(Y)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'5')} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1-s_3 s_1}{2}} \otimes Z^{\frac{1+s_2 s'_0}{2}} \right) C(Z)_u$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(12;5'6')} \Pi_{s_1}^{(3'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2 s_0}{2}} \otimes Z^{\frac{1-s_3 s_1}{2}} \right) C(Z)_d$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(45)} \Pi_{s_2}^{(56;1'2')} \Pi_{s_1}^{(36)} \Pi_{s_0}^{(\text{anc})}$
$\left(X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_3 s_1 s'_0}{2}} \otimes Z^{\frac{1-s_2}{2}} \right) C(Z)_r$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;5'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_1 s_0 s'_0}{2}} \otimes Z^{\frac{1-s_2}{2}} \right) C(Z)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(14;5'6')} \Pi_{s_0}^{(\text{anc})}$

Table A.14: Pauli gate corrections tracked for the corresponding controlled-Pauli gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X^{\frac{1-s_n s_0}{2}} \Pi_{s_0}^{(34)} \otimes X^{\frac{1-s'_n s'_0}{2}} \Pi_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
SWAP(u)	$\left[\begin{array}{c} \prod_{+}^{(3'4')} \prod_{s_2 s_1}^{(12;4'5')} \prod_{-s_2}^{(26)} \\ \prod_{+}^{(12;5'6')} \prod_{s_2}^{(1'5')} \prod_{s_1}^{(16;3'5')} \end{array} \right]$	4.27×10^{13}	$C(X)^3$	2.92×10^{57}	$C(X)^3$	3.44×10^{87}
(d)	$\left[\begin{array}{c} \prod_{+}^{(34)} \prod_{s_2 s_1}^{(45;1'2')} \prod_{-s_2}^{(2'6')} \\ \prod_{+}^{(56;1'2')} \prod_{s_2}^{(15)} \prod_{s_1}^{(35;1'6')} \end{array} \right]$	4.27×10^{13}		2.92×10^{57}		3.44×10^{87}
(r)	$\prod_{+}^{(3'4')} \prod_{s_1}^{(4'5')} \prod_{+}^{(56;5'6')} \prod_{s_1}^{(25;2'3')}$	7.86×10^{14}		9.00×10^{63}		1.05×10^{94}
(l)	$\prod_{+}^{(3'4')} \prod_{-s_1}^{(3'5')} \prod_{+}^{(15;1'5')} \prod_{s_1}^{(25;2'4')}$	5.25×10^{15}		1.91×10^{66}		1.84×10^{97}
(a)		2.95×10^{14}		1.96×10^{61}		3.89×10^{91}
$W(u)$	$\prod_{+}^{(3'4')} \prod_{-s_1}^{(3'6')} \prod_{s_1}^{(12;3'5')}$	1.75×10^5	$S_1 S_2 C(Z)$	1.17×10^{15}	W	1.75×10^5
(d)	$\prod_{+}^{(34)} \prod_{-s_1}^{(36)} \prod_{s_1}^{(35;1'2')}$	1.75×10^5		1.17×10^{15}		1.75×10^5
(r)	$\prod_{+}^{(34)} \prod_{s_1}^{(45)} \prod_{s_1}^{(46;5'6')}$	2.54×10^7		1.70×10^{17}		2.54×10^7
(l)	$\prod_{+}^{(34)} \prod_{s_1}^{(45)} \prod_{s_1}^{(46;5'6')}$	3.05×10^8		1.01×10^{18}		3.05×10^8
(a)		3.92×10^6		2.20×10^{16}		3.92×10^6
Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
[SWAP] (u)	$\left[\begin{array}{c} \prod_{s_6}^{(3'4')} \prod_{s_5}^{(12;4'5')} \prod_{s_4}^{(26)} \\ \prod_{s_3}^{(12;5'6')} \prod_{s_2}^{(1'5')} \prod_{s_1}^{(16;3'5')} \end{array} \right]$	9.06×10^5	$C(X)^3$	1.72×10^{19}	$C(X)^3$	2.81×10^{38}
(d)	$\left[\begin{array}{c} \prod_{s_6}^{(34)} \prod_{s_5}^{(45;1'2')} \prod_{s_4}^{(2'6')} \\ \prod_{s_3}^{(56;1'2')} \prod_{s_2}^{(15)} \prod_{s_1}^{(35;1'6')} \end{array} \right]$	9.06×10^5		1.72×10^{19}		2.81×10^{38}
(r)	$\prod_{s_4}^{(3'4')} \prod_{s_3}^{(4'5')} \prod_{s_2}^{(25;2'5')} \prod_{s_1}^{(56;3'6')}$	9.97×10^5		3.01×10^{22}		4.91×10^{41}
(l)	$\prod_{s_4}^{(3'4')} \prod_{s_3}^{(4'5')} \prod_{s_2}^{(15;1'5')} \prod_{s_1}^{(25;2'3')}$	2.34×10^6		4.36×10^{21}		2.89×10^{43}
(a)		1.18×10^6		1.40×10^{21}		3.25×10^{40}
[W] (u)	$\prod_{s_3}^{(3'4')} \prod_{s_2}^{(3'6')} \prod_{s_1}^{(12;3'5')}$	2.26×10^2	$S_1 S_2 C(Z)$	2.58×10^6	W	2.26×10^2
(d)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(36)} \prod_{s_1}^{(35;1'2')}$	2.26×10^2		2.58×10^6		2.26×10^2
(r)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(45)} \prod_{s_1}^{(46;5'6')}$	2.72×10^3		3.11×10^7		2.72×10^3
(l)	$\prod_{s_3}^{(34)} \prod_{s_2}^{(45)} \prod_{s_1}^{(46;5'6')}$	1.06×10^4		7.58×10^7		1.06×10^4
(a)		1.10×10^3		1.12×10^7		1.10×10^3

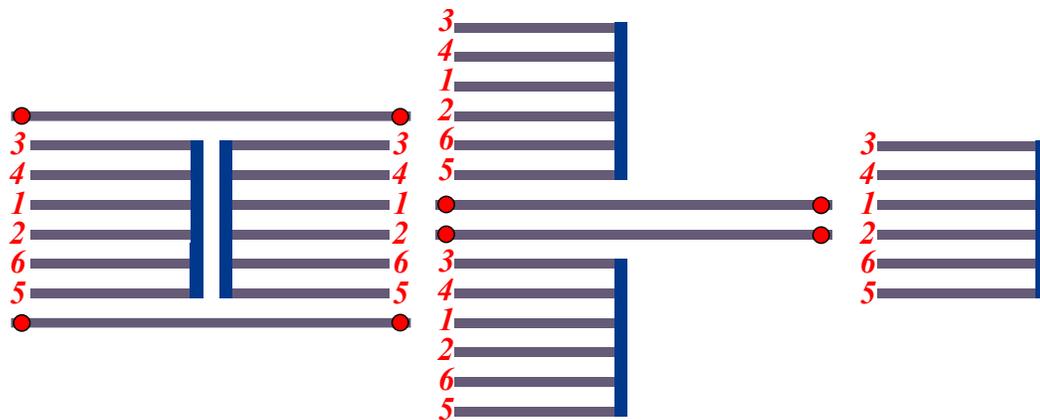
Table A.15: Minimal difficulty weight measurement sequences for the two-qubit SWAP and W gates when using forced-measurement methods and the Pauli cosets of SWAP and W when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using $\text{SWAP} = C(X)_{12}C(X)_{21}C(X)_{12}$ with $C(X)_{12}$ as given in Tables A.12 and A.13.

Gate	Tracked Measurement Sequence
$\left(X^{\frac{1+s_4}{2}} Y^{\frac{1-s_5 s_3 s_1}{2}} Z^{\frac{1-s'_6 s_2}{2}} \otimes X^{\frac{1-s'_0}{2}} Y^{\frac{1-s_3}{2}} Z^{\frac{1+s_4 s_2}{2}} \right) \text{SWAP}_u$	$\left[\begin{array}{l} \Pi_{s'_6}^{(3'4')} \Pi_{s_5}^{(12;4'5')} \Pi_{s_4}^{(26)} \Pi_{s_3}^{(12;5'6')} \\ \Pi_{s_2}^{(1'5')} \Pi_{s_1}^{(16;3'5')} \Pi_{s_0}^{(\text{anc})} \\ \Pi_{s_6}^{(34)} \Pi_{s_5}^{(45;1'2')} \Pi_{s_4}^{(2'6')} \Pi_{s_3}^{(56;1'2')} \\ \Pi_{s_2}^{(15)} \Pi_{s_1}^{(35;1'6')} \Pi_{s_0}^{(\text{anc})} \end{array} \right]$
$\left(X^{\frac{1-s'_0}{2}} Y^{\frac{1-s_3}{2}} Z^{\frac{1+s_4 s_2}{2}} \otimes X^{\frac{1+s_4}{2}} Y^{\frac{1-s_5 s_3 s_1}{2}} Z^{\frac{1-s_6 s_2}{2}} \right) \text{SWAP}_d$	
$(Z \otimes Z)^{\frac{1-s_2 s_0 s'_0}{2}} (X \otimes X)^{\frac{1-s_3 s_1 s_0}{2}} \text{SWAP}_r$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_1}^{(4'5')} \Pi_{s_2}^{(25;2'5')} \Pi_{s_3}^{(56;3'6')} \Pi_{s_0}^{(\text{anc})}$
$(X \otimes X)^{\frac{1-s_2 s_0 s'_0}{2}} (Y \otimes Y)^{\frac{1-s_3 s_1 s_0}{2}} \text{SWAP}_l$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(4'5')} \Pi_{s_2}^{(15;1'5')} \Pi_{s_1}^{(25;2'3')} \Pi_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Z)^{\frac{1-s'_3 s'_0}{2}} (Z \otimes Z)^{\frac{1+s_2 s_1 s'_0}{2}} W_u$	$\Pi_{s'_3}^{(3'4')} \Pi_{s_2}^{(3'6')} \Pi_{s_1}^{(12;3'5')} \Pi_{s_0}^{(\text{anc})}$
$(Z \otimes \mathbb{1})^{\frac{1-s_3 s_0}{2}} (Z \otimes Z)^{\frac{1+s_2 s_1 s_0}{2}} W_d$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(36)} \Pi_{s_1}^{(35;1'2')} \Pi_{s_0}^{(\text{anc})}$
$(Z \otimes Z)^{\frac{1-s_2 s_1 s_0 s'_0}{2}} W_r$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(45)} \Pi_{s_1}^{(46;5'6')} \Pi_{s_0}^{(\text{anc})}$
$(Z \otimes Z)^{\frac{1-s_2 s_1 s_0 s'_0}{2}} W_l$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(45)} \Pi_{s_1}^{(46;5'6')} \Pi_{s_0}^{(\text{anc})}$

Table A.16: Pauli gate corrections tracked for the corresponding SWAP and W gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X^{\frac{1-s_n s_0}{2}} \Pi_{s_0}^{(34)} \otimes X^{\frac{1-s'_n s'_0}{2}} \Pi_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

A.3.3 One-Sided Hexon with Configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$

The MZM labeling configuration $\langle 3, 4, 1, 2, 6, 5 \rangle$ is optimal for the one-sided hexon architecture, when using the forced-measurement methods for the gates: the controlled- X gate $C(X)$, and the geometric average of the controlled-Pauli gates $C(P)$. This configuration within an array looks like:



Gate	Forced Measurement Sequence	Weight	$\langle S, H \rangle$ Decomp.	Weight	$\langle S, B \rangle$ Decomp.	Weight
X	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_2}^{(16)} \hat{\Pi}_{s_1}^{(14)}$	3.10×10^4	$HSSH$	5.12×10^{19}	BB	1.30×10^8
Y	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_2}^{(26)} \hat{\Pi}_{s_1}^{(24)}$	4.95×10^4	$HSSHSS$	6.62×10^{27}	$BBSS$	6.66×10^{15}
Z	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_2}^{(12)} \hat{\Pi}_{s_1}^{(14)}$	1.95×10^4	SS	5.13×10^7	SS	5.13×10^7
S	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)}$	7.16×10^3	S	7.16×10^3	S	7.16×10^3
XS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_2}^{(16)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)}$	1.59×10^6	$HSSH S$	3.66×10^{23}	BBS	9.31×10^{11}
YS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_2}^{(16)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)}$	1.59×10^6	$SHSSH$	3.66×10^{23}	SBB	9.31×10^{11}
ZS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)}$	7.16×10^3	S^\dagger	7.16×10^3	S^\dagger	7.16×10^3
H	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_2}^{(12)} \hat{\Pi}_{s_2}^{(16)} \hat{\Pi}_{s_1}^{(14)}$	9.99×10^5	H	9.99×10^5	SBS	5.84×10^{11}
XH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(45)}$	1.82×10^4	HSS	5.12×10^{13}	$S^\dagger B^\dagger S$	5.84×10^{11}
YH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_2}^{(12)} \hat{\Pi}_{s_2}^{(16)} \hat{\Pi}_{s_1}^{(14)}$	9.99×10^5	$SSHSS$	2.63×10^{21}	$SB^\dagger S$	5.84×10^{11}
ZH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_1}^{(45)}$	1.82×10^4	SSH	5.12×10^{13}	$S^\dagger BS$	5.84×10^{11}
SH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_1}^{(46)}$	5.85×10^5	SH	7.15×10^9	$B^\dagger S^\dagger$	8.16×10^7
XSH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)} \hat{\Pi}_{s_1}^{(46)}$	5.85×10^5	$S^\dagger HSS$	3.67×10^{17}	BS^\dagger	8.16×10^7
YSH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_1}^{(46)}$	5.85×10^5	$SHSS$	3.67×10^{17}	$B^\dagger S$	8.16×10^7
ZSH	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_1}^{(46)}$	5.85×10^5	$S^\dagger H$	7.15×10^9	BS	8.16×10^7
HS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)} \hat{\Pi}_{s_1}^{(45)}$	9.32×10^5	HS	7.15×10^9	$S^\dagger B^\dagger$	8.16×10^7
XHS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_1}^{(45)}$	9.32×10^5	HS^\dagger	7.15×10^9	SB	8.16×10^7
YHS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{-s_1}^{(24)} \hat{\Pi}_{s_1}^{(45)}$	9.32×10^5	$SSH S^\dagger$	3.67×10^{17}	$S^\dagger B$	8.16×10^7
ZHS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_1}^{(24)} \hat{\Pi}_{s_1}^{(45)}$	9.32×10^5	$SSH S$	3.67×10^{17}	SB^\dagger	8.16×10^7
SHS	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_1}^{(14)} \hat{\Pi}_{s_1}^{(46)}$	1.14×10^4	SHS	5.12×10^{13}	B^\dagger	1.14×10^4
$XSHS$	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_1}^{(14)} \hat{\Pi}_{s_1}^{(46)}$	1.14×10^4	$S^\dagger HS^\dagger$	5.12×10^{13}	B	1.14×10^4
$YSHS$	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{-s_2 s_1}^{(14)} \hat{\Pi}_{-s_2}^{(12)} \hat{\Pi}_{s_2}^{(26)} \hat{\Pi}_{s_1}^{(23)}$	1.26×10^6	SHS^\dagger	5.12×10^{13}	$B^\dagger SS$	5.84×10^{11}
$ZSHS$	$\hat{\Pi}_+^{(34)} \hat{\Pi}_{s_2 s_1}^{(14)} \hat{\Pi}_{s_2}^{(12)} \hat{\Pi}_{s_2}^{(26)} \hat{\Pi}_{s_1}^{(23)}$	1.26×10^6	$S^\dagger HS$	5.12×10^{13}	BSS	5.84×10^{11}
Average		1.89×10^5		2.02×10^{14}		8.44×10^8

Table A.17: Minimal difficulty weight measurement sequences for each single-qubit Clifford gate when using forced-measurement methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Pauli Class	Unforced Measurement Sequence	Weight	$\langle S, H \rangle$ Decomp.	Weight	$\langle S, B \rangle$ Decomp.	Weight
[S]	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(24)} \Pi_{s_1}^{(14)}$	5.13×10^1	S	5.13×10^1	S	5.13×10^1
[H]	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(45)} \Pi_{s_1}^{(14)}$	8.17×10^1	H	8.17×10^1	SBS	1.70×10^5
[SH]	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(46)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(24)}$	2.81×10^2	SH	4.19×10^3	BS	3.32×10^3
[HS]	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(46)} \Pi_{s_2}^{(24)} \Pi_{s_1}^{(14)}$	2.81×10^2	HS	4.19×10^3	SB	3.32×10^3
[SHS]	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(46)} \Pi_{s_1}^{(14)}$	6.47×10^1	SHS	2.15×10^5	B	6.47×10^1
Average		1.16×10^2		1.74×10^3		1.44×10^3

Table A.18: Minimal difficulty weight measurement sequences for each Pauli coset of single-qubit Clifford gates when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H \rangle$ and $\langle S, B \rangle$.

Gate	Tracked Measurement Sequence
$Y^{\frac{1-s_3 s_0}{2}} Z^{\frac{1-s_2 s_1}{2}} S$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(24)} \Pi_{s_0}^{(34)}$
$Y^{\frac{1-s_3 s_0}{2}} Y^{\frac{1-s_2 s_1 s_0}{2}} ZH$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(45)} \Pi_{s_0}^{(34)}$
$Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_3 s_2}{2}} X^{\frac{1-s_2 s_1}{2}} YSH$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(24)} \Pi_{s_1}^{(46)} \Pi_{s_0}^{(34)}$
$Z^{\frac{1-s_4 s_0}{2}} Y^{\frac{1-s_3 s_2}{2}} X^{\frac{1-s_2 s_1}{2}} YHS$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(46)} \Pi_{s_2}^{(24)} \Pi_{s_1}^{(14)} \Pi_{s_0}^{(34)}$
$Y^{\frac{1-s_3 s_0}{2}} X^{\frac{1-s_2 s_1}{2}} SHS$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(46)} \Pi_{s_0}^{(34)}$

Table A.19: Pauli gate corrections tracked for the corresponding single-qubit gates implemented using Majorana-Pauli tracking methods. The implicit action on the ancillary qubit is $X^{\frac{1-s_n s_0}{2}} \Pi_{s_0}^{(34)}$ for sequences of length n .

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
$C(X)(u)$	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{-}{\Pi}_+(12) \overset{-}{\Pi}_{s_1}^{(23;2'5')}$	1.43×10^7	${}^H C(Z)$	8.94×10^{18}	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	1.51×10^{29}
(d)	$\overset{-}{\Pi}_+(3'4') \overset{-}{\Pi}_{s_1}^{(1'3')} \overset{+}{\Pi}_+(1'6') \overset{+}{\Pi}_{s_1}^{(56;1'3')}$	2.28×10^7		8.94×10^{18}		1.51×10^{29}
(r)	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{+}{\Pi}_+(12) \overset{+}{\Pi}_{s_1}^{(23;1'6')}$	1.03×10^9		6.46×10^{20}		2.19×10^{31}
(l)	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{+}{\Pi}_+(12) \overset{+}{\Pi}_{s_1}^{(23;1'6')}$	1.56×10^{10}		2.49×10^{22}		2.64×10^{32}
(a)		2.69×10^8		1.89×10^{20}		3.39×10^{30}
$C(Y)(u)$	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{-}{\Pi}_+(12) \overset{-}{\Pi}_{s_1}^{(23;1'5')}$	2.28×10^7	${}^{SH} C(Z)$	4.59×10^{26}	$B_2^\dagger W^\dagger S_2 B_2 S_1$	1.85×10^{21}
(d)	$\overset{+}{\Pi}_+(3'4') \overset{+}{\Pi}_{-s_1}^{(2'4')} \overset{-}{\Pi}_+(2'6') \overset{-}{\Pi}_{s_1}^{(56;3'6')}$	2.88×10^7		4.59×10^{26}		1.85×10^{21}
(r)	$\overset{+}{\Pi}_+(3'4') \overset{+}{\Pi}_{s_1}^{(2'4')} \overset{+}{\Pi}_+(2'6') \overset{+}{\Pi}_{s_1}^{(12;2'4')}$	8.17×10^8		3.31×10^{28}		2.69×10^{23}
(l)	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{+}{\Pi}_+(12) \overset{+}{\Pi}_{s_1}^{(23;1'5')}$	9.80×10^9		1.27×10^{30}		3.23×10^{24}
(a)		2.69×10^8		9.70×10^{27}		4.16×10^{22}
$C(Z)(u)$	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{-}{\Pi}_+(12) \overset{-}{\Pi}_{s_1}^{(23;5'6')}$	8.96×10^6	$C(Z)$	8.96×10^6	$S_1 S_2 W^\dagger$	1.43×10^{13}
(d)	$\overset{+}{\Pi}_+(3'4') \overset{+}{\Pi}_{-s_1}^{(1'4')} \overset{-}{\Pi}_+(1'2') \overset{-}{\Pi}_{s_1}^{(56;2'3')}$	8.96×10^6		8.96×10^6		1.43×10^{13}
(r)	$\overset{-}{\Pi}_+(34) \overset{-}{\Pi}_{s_1}^{(13)} \overset{-}{\Pi}_+(12) \overset{-}{\Pi}_{s_1}^{(24;1'2')}$	6.47×10^8		6.47×10^8		2.07×10^{15}
(l)	$\overset{+}{\Pi}_+(34) \overset{+}{\Pi}_{-s_1}^{(14)} \overset{+}{\Pi}_+(12) \overset{+}{\Pi}_{s_1}^{(23;1'2')}$	2.49×10^{10}		2.49×10^{10}		2.49×10^{16}
(a)		1.90×10^8		1.90×10^8		3.20×10^{14}
Average(u)		1.43×10^7		3.33×10^{17}		1.56×10^{21}
(d)		1.80×10^7		3.33×10^{17}		1.56×10^{21}
(r)		8.17×10^8		2.40×10^{19}		2.30×10^{23}
(l)		1.56×10^{10}		9.23×10^{20}		2.77×10^{24}
(a)		2.39×10^8		7.04×10^{18}		3.56×10^{22}

Table A.20: Minimal difficulty weight measurement sequences for controlled-Pauli two-qubit gates when using forced-measurement methods. The labels (u) , (d) , (r) , and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r) , and (l) . The average difficulty weight of the four directions is labeled by (a) . For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, C(Z) \rangle$	Weight			
[C(X)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(24)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(13;2'5')}$	${}^H C(Z)$	6.53×10^6	$S_2 B_2 S_1 W^\dagger S_2^\dagger B_2 S_2$	1.04×10^{13}			
	(d)	$\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(1'3')} \Pi_{s_2}^{(1'6')} \Pi_{s_1}^{(56;1'3')}$					1.24×10^3	6.53×10^6	1.04×10^{13}
	(r)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'6')}$					9.34×10^3	4.94×10^7	4.96×10^{13}
	(l)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'6')}$					3.64×10^4	3.06×10^8	4.90×10^{14}
	(a)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'6')}$					4.78×10^3	2.83×10^7	4.03×10^{13}
[C(Y)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(24)} \Pi_{s_2}^{(34;1'5')} \Pi_{s_1}^{(13)}$	${}^{SH} C(Z)$	1.72×10^{10}	$B_2^\dagger W^\dagger S_2 B_2 S_1$	3.97×10^9			
	(d)	$\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(3'5')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(1'4')}$					1.56×10^3	1.72×10^{10}	3.97×10^9
	(r)	$\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(2'4')} \Pi_{s_2}^{(2'6')} \Pi_{s_1}^{(12;2'4')}$					7.40×10^3	1.30×10^{11}	1.88×10^{10}
	(l)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'5')}$					2.88×10^4	8.06×10^{11}	1.86×10^{11}
	(a)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'5')}$					4.77×10^3	7.46×10^{10}	1.53×10^{10}
[C(Z)]	(u)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;5'6')}$	$C(Z)$	9.79×10^2	$S_1 S_2 W^\dagger$	9.47×10^5			
	(d)	$\Pi_{s_4}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(1'2')} \Pi_{s_1}^{(56;2'3')}$					9.79×10^2	9.79×10^2	9.47×10^5
	(r)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12;1'2')} \Pi_{s_1}^{(14)}$					7.40×10^3	7.40×10^3	4.50×10^6
	(l)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'2')}$					4.59×10^4	4.59×10^4	4.45×10^7
	(a)	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'2')}$					4.25×10^3	4.25×10^3	3.66×10^6
Average	(u)			4.79×10^6		3.40×10^9			
	(d)			4.79×10^6		3.40×10^9			
	(r)			3.62×10^7		1.61×10^{10}			
	(l)			2.25×10^8		1.59×10^{11}			
	(a)			2.07×10^7		1.31×10^{10}			

Table A.21: Minimal difficulty weight measurement sequences for Pauli cosets of controlled–Pauli two-qubit gates when using Majorana–Pauli tracking methods. The labels (u) , (d) , (r) , and (l) indicates that for a hexon acting as the control qubit of the $C(P)$ gate, the corresponding target qubit is the nearest neighbor hexon in the up, down, right, and left direction, respectively. Notice that the choice of control and target qubit is arbitrary for $C(Z)$, so (u) and (d) are related by symmetry, as are (r) , and (l) . The average difficulty weight of the four directions is labeled by (a) . For comparison, we also present the corresponding realization of the gates formed by using the generating gate sets $\langle S, H, C(Z) \rangle$ and $\langle S, B, W \rangle$. ${}^A C(Z) = A_2 C(Z) A_2^\dagger$ denotes conjugation of $C(Z)$ by A on the target qubit.

Gate	Tracked Measurement Sequence
$\left(X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_3 s_2 s_1 s_0 s'_0}{2}} \otimes X^{\frac{1-s_2}{2}} \right) C(X)_u$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(24)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(13;2'5')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1-s_2}{2}} \otimes Y^{\frac{1-s'_4 s'_0}{2}} X^{\frac{1-s_3 s_1 s_0}{2}} \right) C(X)_d$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(1'3')} \Pi_{s_2}^{(1'6')} \Pi_{s_1}^{(56;1'3')} \Pi_{s_0}^{(\text{anc})}$
$\left(Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_1 s_2 s_0}{2}} \otimes X^{\frac{1-s_2}{2}} \right) C(X)_r$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0}{2}} \otimes X^{\frac{1-s_2}{2}} \right) C(X)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1-s_2 s_0 s'_0}{2}} \otimes Y^{\frac{1+s_3 s_1 s_0}{2}} \right) C(Y)_u$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(24)} \Pi_{s_2}^{(34;1'5')} \Pi_{s_1}^{(13)} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_3 s_1}{2}} \otimes Y^{\frac{1-s_2 s_0 s'_0}{2}} \right) C(Y)_d$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(3'5')} \Pi_{s_2}^{(56;3'4')} \Pi_{s_1}^{(1'4')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1+s_2}{2}} \otimes X^{\frac{1-s'_4 s'_0}{2}} Y^{\frac{1-s_3 s_1}{2}} \right) C(Y)_r$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(2'4')} \Pi_{s_2}^{(2'6')} \Pi_{s_1}^{(12;2'4')} \Pi_{s_0}^{(\text{anc})}$
$\left(Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0 s'_0}{2}} \otimes Y^{\frac{1-s_2}{2}} \right) C(Y)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'5')} \Pi_{s_0}^{(\text{anc})}$
$\left(Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0 s'_0}{2}} \otimes Z^{\frac{1-s_2}{2}} \right) C(Z)_u$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;5'6')} \Pi_{s_0}^{(\text{anc})}$
$\left(Z^{\frac{1-s_2}{2}} \otimes Y^{\frac{1-s'_4 s'_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0 s'_0}{2}} \right) C(Z)_d$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(1'2')} \Pi_{s_1}^{(56;2'3')} \Pi_{s_0}^{(\text{anc})}$
$\left(X^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0}{2}} \otimes Z^{\frac{1-s_3 s_1 s_0}{2}} \right) C(Z)_r$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(23)} \Pi_{s_2}^{(12;1'2')} \Pi_{s_1}^{(14)} \Pi_{s_0}^{(\text{anc})}$
$\left(Y^{\frac{1-s_4 s_0}{2}} Z^{\frac{1+s_3 s_2 s_1 s_0}{2}} \otimes Z^{\frac{1-s_2}{2}} \right) C(Z)_l$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(14)} \Pi_{s_2}^{(12)} \Pi_{s_1}^{(23;1'2')} \Pi_{s_0}^{(\text{anc})}$

Table A.22: Pauli gate corrections tracked for the corresponding controlled-Pauli gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X^{\frac{1-s_n s_0}{2}} \Pi_{s_0}^{(34)} \otimes X^{\frac{1-s'_n s'_0}{2}} \Pi_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

Gate	Forced Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
SWAP(u)	$\left[\begin{array}{c} \prod_{+}^{+(34)} \prod_{s_1}^{+(23;2'5')} \widehat{\prod}_{-s_2s_1}^{(26)} \\ \prod_{-s_1}^{+(35;5'6')} \prod_{s_2}^{(13)} \prod_{s_1}^{(23;2'5')} \end{array} \right]$	1.73×10^{14}	$C(Z)$	7.15×10^{56}	$C(Z)$	3.44×10^{87}
(d)	$\left[\begin{array}{c} \prod_{+}^{+(3'4')} \prod_{s_1}^{+(25;2'3')} \widehat{\prod}_{-s_2s_1}^{(2'6')} \\ \prod_{-s_1}^{+(56;3'6')} \prod_{s_2}^{(1'3')} \prod_{s_1}^{(25;2'3')} \end{array} \right]$	1.73×10^{14}		7.15×10^{56}		3.44×10^{87}
(r)	$\widehat{\prod}_{+}^{(34)} \prod_{s_1}^{+(13)} \prod_{+}^{+(12;1'2')} \prod_{s_1}^{(46;1'6')}$	6.05×10^{13}		2.70×10^{62}		1.05×10^{94}
(l)	$\prod_{+}^{+(34)} \widehat{\prod}_{-s_1}^{(14)} \prod_{+}^{+(15;1'5')} \prod_{s_1}^{(36;1'6')}$	6.82×10^{16}		1.54×10^{67}		1.84×10^{97}
(a)		5.93×10^{14}		6.79×10^{60}		3.89×10^{91}
$W(u)$	$\widehat{\prod}_{+}^{(34)} \prod_{s_1}^{+(13)} \prod_{s_1}^{(23;5'6')}$	4.44×10^5	$S_1 S_2 W$	4.59×10^{14}	W	4.44×10^5
(d)	$\widehat{\prod}_{+}^{(3'4')} \prod_{s_1}^{+(1'3')} \prod_{s_1}^{(56;2'3')}$	4.44×10^5		4.59×10^{14}		4.44×10^5
(r)	$\prod_{+}^{+(34)} \widehat{\prod}_{s_1}^{(14)} \prod_{s_1}^{(24;1'2')}$	7.92×10^6		3.32×10^{16}		7.92×10^6
(l)	$\widehat{\prod}_{+}^{(34)} \prod_{s_1}^{+(13)} \prod_{s_1}^{(23;1'2')}$	9.77×10^8		1.28×10^{18}		9.77×10^8
(a)		6.25×10^6		9.72×10^{15}		6.25×10^6
Pauli Class	Tracked Measurement Sequence	Weight	$\langle S, H, C(Z) \rangle$	Weight	$\langle S, B, W \rangle$	Weight
[SWAP] (u)	$\left[\begin{array}{c} \prod_{s_6}^{(34)} \prod_{s_5}^{(23;2'5')} \prod_{s_4}^{(26)} \\ \prod_{s_3}^{(35;5'6')} \prod_{s_2}^{(13)} \prod_{s_1}^{(23;2'5')} \end{array} \right]$	1.44×10^6	$C(X)^3$	2.78×10^{20}	$C(X)^3$	1.12×10^{39}
(d)	$\left[\begin{array}{c} \prod_{s_6}^{(3'4')} \prod_{s_5}^{(25;2'3')} \prod_{s_4}^{(2'6')} \\ \prod_{s_3}^{(56;3'6')} \prod_{s_2}^{(1'3')} \prod_{s_1}^{(25;2'3')} \end{array} \right]$	1.44×10^6		2.78×10^{20}		1.12×10^{39}
(r)	$\prod_{s_4}^{(34)} \prod_{s_3}^{(13)} \prod_{s_2}^{(12;1'2')} \prod_{s_1}^{(46;1'6')}$	3.92×10^5		1.21×10^{23}		1.22×10^{41}
(l)	$\prod_{s_4}^{(3'4')} \prod_{s_3}^{(1'4')} \prod_{s_2}^{(16;1'6')} \prod_{s_1}^{(15;3'5')}$	5.94×10^6		2.87×10^{25}		1.18×10^{44}
(a)		1.48×10^6		2.27×10^{22}		6.53×10^{40}
[W] (u)	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(23;5'6')}$	3.60×10^2	$S_1 S_2 C(Z)$	2.58×10^6	W	3.60×10^2
(d)	$\prod_{+}^{(3'4')} \prod_{s_1}^{(1'3')} \prod_{s_1}^{(56;2'3')}$	3.60×10^2		2.58×10^6		3.60×10^2
(r)	$\prod_{+}^{(34)} \prod_{s_1}^{(14)} \prod_{s_1}^{(24;1'2')}$	1.71×10^3		1.95×10^7		1.71×10^3
(l)	$\prod_{+}^{(34)} \prod_{s_1}^{(13)} \prod_{s_1}^{(23;1'2')}$	1.69×10^4		1.21×10^8		1.69×10^4
(a)		1.39×10^3		1.12×10^7		1.39×10^3

Table A.23: Minimal difficulty weight measurement sequences for the two-qubit SWAP and W gates when using forced-measurement methods and the Pauli cosets of SWAP and W when using Majorana-Pauli tracking methods. For comparison, we also present the corresponding realization of the gates formed by using $\text{SWAP} = C(X)_{12}C(X)_{21}C(X)_{12}$ with $C(X)_{12}$ as given in Tables A.20 and A.21.

Gate	Tracked Measurement Sequence
$\left(X \frac{1-s_6 s_3 s_1 s_0}{2} Y \frac{1-s_5 s_4 s_2 s'_0}{2} Z \otimes X \frac{1-s_6 s_3 s_0}{2} Y \frac{1-s_4 s_2 s'_0}{2} Z \frac{1+s_1}{2} \right) \text{SWAP}_u$	$\left[\begin{array}{c} \Pi_{s_6}^{(34)} \Pi_{s_5}^{(23;2'5')} \Pi_{s_4}^{(26)} \Pi_{s_3}^{(35;5'6')} \\ \Pi_{s_2}^{(13)} \Pi_{s_1}^{(23;2'5')} \Pi_{s_0}^{(\text{anc})} \\ \Pi_{s_6}^{(3'4')} \Pi_{s_5}^{(25;2'3')} \Pi_{s_4}^{(2'6')} \Pi_{s_3}^{(56;3'6')} \\ \Pi_{s_2}^{(1'3')} \Pi_{s_1}^{(25;2'3')} \Pi_{s_0}^{(\text{anc})} \end{array} \right]$
$\left(X \frac{1-s'_6 s_3 s'_0}{2} Y \frac{1-s_4 s_2 s_0}{2} Z \frac{1+s_1}{2} \otimes X \frac{1-s'_6 s_3 s_1 s'_0}{2} Y \frac{1-s_5 s_4 s_2 s_0}{2} Z \right) \text{SWAP}_d$	
$(Y \otimes \mathbb{1}) \frac{1-s_4 s_0}{2} (X \otimes X) \frac{1-s_2}{2} (Z \otimes Z) \frac{1-s_3 s_1 s_0}{2} \text{SWAP}_r$	$\Pi_{s_4}^{(34)} \Pi_{s_3}^{(13)} \Pi_{s_2}^{(12;1'2')} \Pi_{s_1}^{(46;1'6')} \Pi_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Y) \frac{1-s'_4 s'_0}{2} (Y \otimes Y) \frac{1-s_2}{2} (X \otimes X) \frac{1+s_3 s_1 s_0}{2} \text{SWAP}_l$	$\Pi_{s'_4}^{(3'4')} \Pi_{s_3}^{(1'4')} \Pi_{s_2}^{(16;1'6')} \Pi_{s_1}^{(15;3'5')} \Pi_{s_0}^{(\text{anc})}$
$(Y \otimes \mathbb{1}) \frac{1-s_3 s_0}{2} (Z \otimes Z) \frac{1-s_2 s_1 s'_0}{2} W_u$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(13)} \Pi_{s_1}^{(23;5'6')} \Pi_{s_0}^{(\text{anc})}$
$(\mathbb{1} \otimes Y) \frac{1-s'_3 s'_0}{2} (Z \otimes Z) \frac{1-s_2 s_1 s_0}{2} W_d$	$\Pi_{s'_3}^{(3'4')} \Pi_{s_2}^{(1'3')} \Pi_{s_1}^{(56;2'3')} \Pi_{s_0}^{(\text{anc})}$
$(Y \otimes \mathbb{1}) \frac{1-s_3 s_0}{2} (Z \otimes Z) \frac{1-s_2 s_1}{2} W_r$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(14)} \Pi_{s_1}^{(24;1'2')} \Pi_{s_0}^{(\text{anc})}$
$(Y \otimes \mathbb{1}) \frac{1-s_3 s_0}{2} (Z \otimes Z) \frac{1-s_2 s_1}{2} W_l$	$\Pi_{s_3}^{(34)} \Pi_{s_2}^{(13)} \Pi_{s_1}^{(23;1'2')} \Pi_{s_0}^{(\text{anc})}$

Table A.24: Pauli gate corrections tracked for the corresponding SWAP and W gates implemented using Majorana-Pauli tracking methods. The implicit action on the two ancillary qubits is $X \frac{1-s_n s_0}{2} \Pi_{s_0}^{(34)} \otimes X \frac{1-s'_n s'_0}{2} \Pi_{s'_0}^{(3'4')}$ for sequences of length n . (When there is not a final projector for one of the ancillary pairs, it is equivalent to there being a projector for that ancillary pair onto its initial projection channel, e.g. $s_n = s_0$ or $s'_n = s'_0$.)

Appendix B

Optimal surface code details

B.1 Measurement sequences for an all-hexon surface code

The following lists sequences implementing the target gate up to an overall-Pauli (determined by the measurement outcomes). It is for an all-hexon architecture on a square lattice where one sub-lattice contains ancilla hexons which are used to encode a single qubit (a $[[6, 1, 2]]$ stabilizer code stabilized by overall island parity $i^3\gamma_1\gamma_2\gamma_3\gamma_4\gamma_5\gamma_6$ and by $i\gamma_3\gamma_4$). The other sub-lattice contains data hexons which contain two qubits (a $[[6, 2, 2]]$ code stabilized only by overall island parity). Such a layout could be useful for a denser surface codes or for implementing color codes [33, 51, 56, 57]. The following sequences can be used to build out the necessary operations for performing stabilizer measurements or doing transversal gates.

For convenience we use the shorthand $ab|cd$ to mean jointly measure $i\gamma_a\gamma_b$ on the ancillary hexon (a) and $i\gamma_c\gamma_d$ on the data hexon ($1or2$), with $ab|oo$ meaning to just measure $i\gamma_a\gamma_b$ on

the ancillary hexon, analogously for $\circ\circ|cd$. The corresponding Pauli operators (up to sign) are also shown.

$C(X)_{a,1}$	46	45	YZ	XI	<table border="1"> <tbody> <tr> <td rowspan="3">H_1</td> <td>45</td> <td>35</td> <td>XI</td> <td>YI</td> </tr> <tr> <td>35</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="3">H_2</td> <td>45</td> <td>26</td> <td>XI</td> <td>IY</td> </tr> <tr> <td>35</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="3">H_1H_2</td> <td>45</td> <td>35</td> <td>XI</td> <td>YI</td> </tr> <tr> <td>35</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>45</td> <td>26</td> <td>XI</td> <td>IY</td> </tr> <tr> <td rowspan="3">S_1</td> <td>35</td> <td>45</td> <td>YI</td> <td>XI</td> </tr> <tr> <td>35</td> <td>35</td> <td>YI</td> <td>YI</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="3">S_2</td> <td>35</td> <td>16</td> <td>YI</td> <td>IX</td> </tr> <tr> <td>35</td> <td>26</td> <td>YI</td> <td>IY</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">S_1S_2</td> <td>35</td> <td>45</td> <td>YI</td> <td>XI</td> </tr> <tr> <td>35</td> <td>35</td> <td>YI</td> <td>YI</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td>35</td> <td>16</td> <td>YI</td> <td>IX</td> </tr> <tr> <td rowspan="4">$C(Z)_{a,1}$</td> <td>46</td> <td>34</td> <td>YZ</td> <td>ZI</td> <td rowspan="4">$C(Z)_{1,2}$</td> <td>35</td> <td>12</td> <td>YI</td> <td>IZ</td> </tr> <tr> <td>56</td> <td>$\circ\circ$</td> <td>ZZ</td> <td>II</td> <td>34</td> <td>34</td> <td>ZI</td> <td>ZI</td> </tr> <tr> <td>46</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> <td>35</td> <td>$\circ\circ$</td> <td>YI</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(Z)_{a,2}$</td> <td>46</td> <td>12</td> <td>YZ</td> <td>IZ</td> <td rowspan="12"> <table border="1"> <tbody> <tr> <td rowspan="4">$C(X)_{1,a}$</td> <td>14</td> <td>34</td> <td>YY</td> <td>ZI</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{2,a}$</td> <td>14</td> <td>12</td> <td>YY</td> <td>IZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{a,1}C(X)_{a,2}$</td> <td>46</td> <td>23</td> <td>YZ</td> <td>XX</td> </tr> <tr> <td>56</td> <td>$\circ\circ$</td> <td>ZZ</td> <td>II</td> </tr> <tr> <td>46</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{1,a}C(X)_{2,a}$</td> <td>14</td> <td>56</td> <td>YY</td> <td>ZZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	H_1	45	35	XI	YI	35	$\circ\circ$	YZ	II	34	$\circ\circ$	ZI	II	H_2	45	26	XI	IY	35	$\circ\circ$	YZ	II	34	$\circ\circ$	ZI	II	H_1H_2	45	35	XI	YI	35	$\circ\circ$	YZ	II	45	26	XI	IY	S_1	35	45	YI	XI	35	35	YI	YI	34	$\circ\circ$	ZI	II	S_2	35	16	YI	IX	35	26	YI	IY	34	$\circ\circ$	ZI	II	S_1S_2	35	45	YI	XI	35	35	YI	YI	34	$\circ\circ$	ZI	II	35	16	YI	IX	$C(Z)_{a,1}$	46	34	YZ	ZI	$C(Z)_{1,2}$	35	12	YI	IZ	56	$\circ\circ$	ZZ	II	34	34	ZI	ZI	46	$\circ\circ$	YZ	II	35	$\circ\circ$	YI	II	34	$\circ\circ$	ZI	II	34	$\circ\circ$	ZI	II	$C(Z)_{a,2}$	46	12	YZ	IZ	<table border="1"> <tbody> <tr> <td rowspan="4">$C(X)_{1,a}$</td> <td>14</td> <td>34</td> <td>YY</td> <td>ZI</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{2,a}$</td> <td>14</td> <td>12</td> <td>YY</td> <td>IZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{a,1}C(X)_{a,2}$</td> <td>46</td> <td>23</td> <td>YZ</td> <td>XX</td> </tr> <tr> <td>56</td> <td>$\circ\circ$</td> <td>ZZ</td> <td>II</td> </tr> <tr> <td>46</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{1,a}C(X)_{2,a}$</td> <td>14</td> <td>56</td> <td>YY</td> <td>ZZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> </tbody> </table>	$C(X)_{1,a}$	14	34	YY	ZI	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II	$C(X)_{2,a}$	14	12	YY	IZ	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II	$C(X)_{a,1}C(X)_{a,2}$	46	23	YZ	XX	56	$\circ\circ$	ZZ	II	46	$\circ\circ$	YZ	II	34	$\circ\circ$	ZI	II	$C(X)_{1,a}C(X)_{2,a}$	14	56	YY	ZZ	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II
	H_1	45	35	XI			YI																																																																																																																																																																																												
		35	$\circ\circ$	YZ			II																																																																																																																																																																																												
		34	$\circ\circ$	ZI		II																																																																																																																																																																																													
H_2	45	26	XI	IY																																																																																																																																																																																															
	35	$\circ\circ$	YZ	II																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
H_1H_2	45	35	XI	YI																																																																																																																																																																																															
	35	$\circ\circ$	YZ	II																																																																																																																																																																																															
	45	26	XI	IY																																																																																																																																																																																															
S_1	35	45	YI	XI																																																																																																																																																																																															
	35	35	YI	YI																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
S_2	35	16	YI	IX																																																																																																																																																																																															
	35	26	YI	IY																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
S_1S_2	35	45	YI	XI																																																																																																																																																																																															
	35	35	YI	YI																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
	35	16	YI	IX																																																																																																																																																																																															
$C(Z)_{a,1}$	46	34	YZ	ZI	$C(Z)_{1,2}$	35	12	YI	IZ																																																																																																																																																																																										
	56	$\circ\circ$	ZZ	II		34	34	ZI	ZI																																																																																																																																																																																										
	46	$\circ\circ$	YZ	II		35	$\circ\circ$	YI	II																																																																																																																																																																																										
	34	$\circ\circ$	ZI	II		34	$\circ\circ$	ZI	II																																																																																																																																																																																										
$C(Z)_{a,2}$	46	12	YZ	IZ	<table border="1"> <tbody> <tr> <td rowspan="4">$C(X)_{1,a}$</td> <td>14</td> <td>34</td> <td>YY</td> <td>ZI</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{2,a}$</td> <td>14</td> <td>12</td> <td>YY</td> <td>IZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{a,1}C(X)_{a,2}$</td> <td>46</td> <td>23</td> <td>YZ</td> <td>XX</td> </tr> <tr> <td>56</td> <td>$\circ\circ$</td> <td>ZZ</td> <td>II</td> </tr> <tr> <td>46</td> <td>$\circ\circ$</td> <td>YZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> <tr> <td rowspan="4">$C(X)_{1,a}C(X)_{2,a}$</td> <td>14</td> <td>56</td> <td>YY</td> <td>ZZ</td> </tr> <tr> <td>16</td> <td>$\circ\circ$</td> <td>IX</td> <td>II</td> </tr> <tr> <td>36</td> <td>$\circ\circ$</td> <td>XZ</td> <td>II</td> </tr> <tr> <td>34</td> <td>$\circ\circ$</td> <td>ZI</td> <td>II</td> </tr> </tbody> </table>	$C(X)_{1,a}$	14	34	YY	ZI	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II	$C(X)_{2,a}$	14	12	YY	IZ	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II	$C(X)_{a,1}C(X)_{a,2}$	46	23	YZ	XX	56	$\circ\circ$	ZZ	II	46	$\circ\circ$	YZ	II	34	$\circ\circ$	ZI	II	$C(X)_{1,a}C(X)_{2,a}$	14	56	YY	ZZ	16	$\circ\circ$	IX	II	36	$\circ\circ$	XZ	II	34	$\circ\circ$	ZI	II																																																																																																																										
	$C(X)_{1,a}$	14	34	YY			ZI																																																																																																																																																																																												
		16	$\circ\circ$	IX			II																																																																																																																																																																																												
		36	$\circ\circ$	XZ			II																																																																																																																																																																																												
34		$\circ\circ$	ZI	II																																																																																																																																																																																															
$C(X)_{2,a}$	14	12	YY	IZ																																																																																																																																																																																															
	16	$\circ\circ$	IX	II																																																																																																																																																																																															
	36	$\circ\circ$	XZ	II																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
$C(X)_{a,1}C(X)_{a,2}$	46	23	YZ	XX																																																																																																																																																																																															
	56	$\circ\circ$	ZZ	II																																																																																																																																																																																															
	46	$\circ\circ$	YZ	II																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															
$C(X)_{1,a}C(X)_{2,a}$	14	56	YY	ZZ																																																																																																																																																																																															
	16	$\circ\circ$	IX	II																																																																																																																																																																																															
	36	$\circ\circ$	XZ	II																																																																																																																																																																																															
	34	$\circ\circ$	ZI	II																																																																																																																																																																																															

B.2 Optimal stabilizer measurement sequence

In this appendix we show one full cycle of the optimized 14 step measurement sequence given in Ch. 3 for the layout in Fig. B.0.

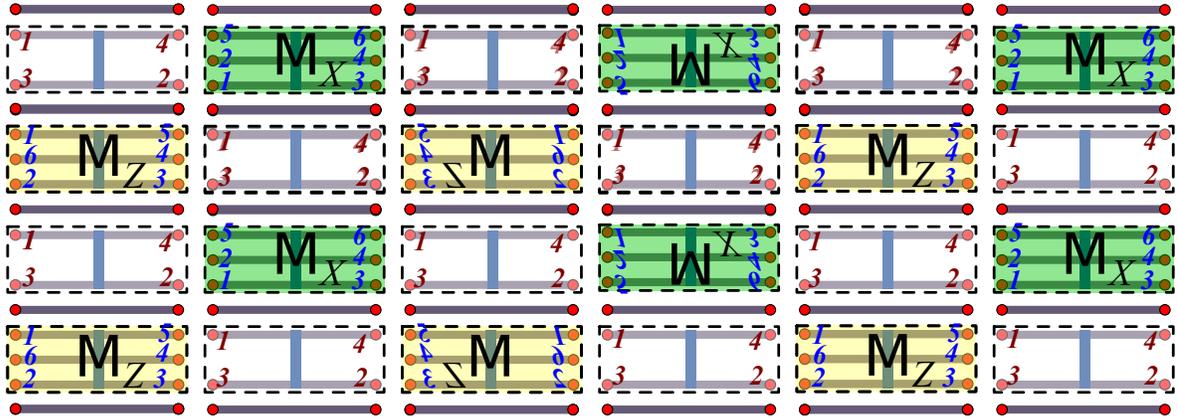


Figure B.0: Every other column of M_Z hexons is flipped vertically and every even column of M_X hexons is flipped horizontally

Step	M_X	M'_X	M_Z	M'_Z
1)	13 23 _l	13 14 _l	34 oo	34 oo
2)	34 oo	34 oo	12 oo	12 oo
3)	25 oo	25 oo	13 34 _r	13 12 _r
4)	34 oo	34 oo	16 oo	16 oo
5)	25 oo	25 oo	13 34 _l	13 12 _l
6)	13 14 _d	13 14 _d	34 oo	34 oo
7)	12 oo	12 oo	36 34 _d	14 12 _u
8)	24 23 _u	24 23 _u	16 oo	16 oo
9)	34 oo	34 oo	14 34 _u	36 12 _d
10)	13 23 _r	13 14 _r	34 oo	34 oo
11)	12 oo	12 oo	12 oo	12 oo
12)	13 23 _l	13 14 _l	34 oo	34 oo
13)	34 oo	34 oo	12 oo	12 oo
14)	25 oo	25 oo	13 34 _r	13 12 _r

Table B.1: An optimal 14-step interleaved measurement sequence for measuring stabilizer operators in the surface code layout in Fig. B.0. M'_X and M'_Z are the vertically and horizontally flipped M_X and M_Z hexons respectively. The notation $xx|yy_d$ means measure jointly xx on the hexon and yy on the tetron in the d direction from the hexon. The grayed out entries correspond to continuations of the previous measurement cycle measurements and are not performed for first and final cycles.

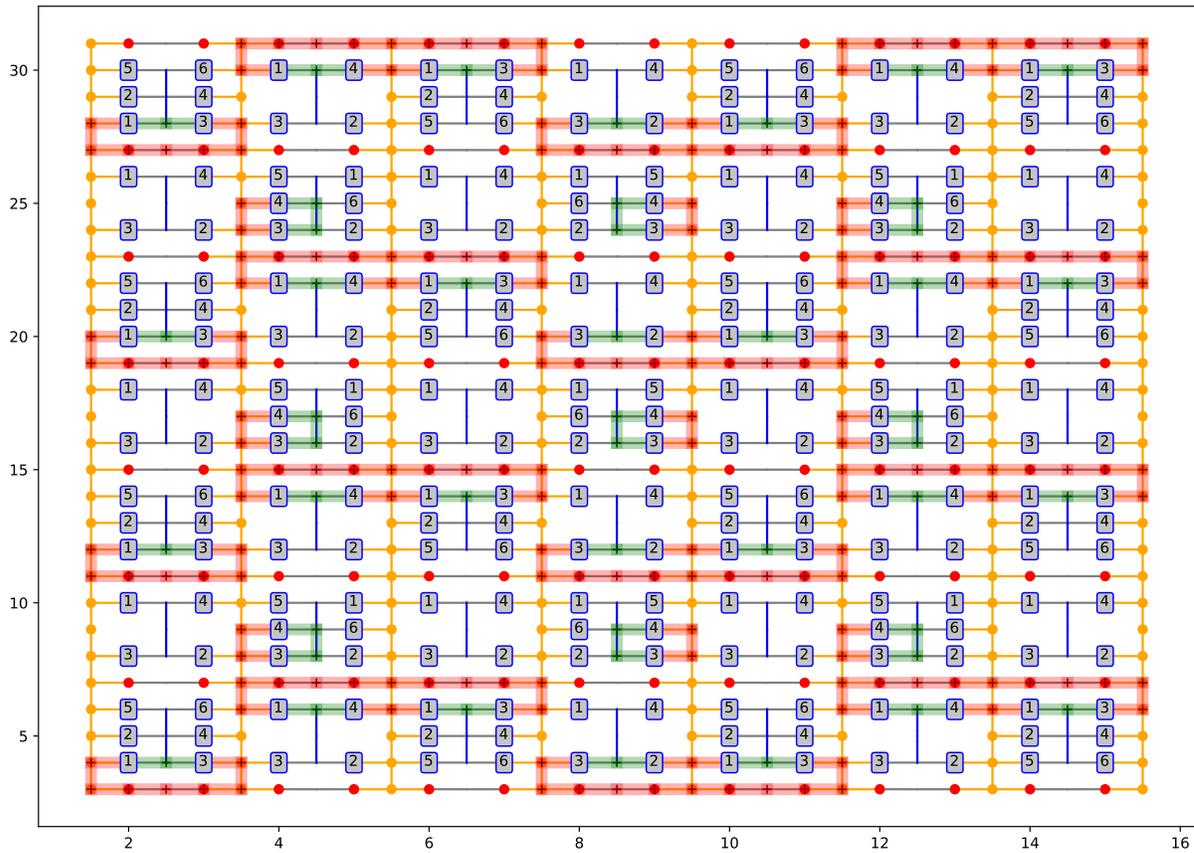


Figure B.1: Step 1 of Table B.1: measure $13|23_l$ on M_X , $13|14_l$ on M'_X , $34| \circ \circ$ on M_Z , and $34| \circ \circ$ on M'_Z . This is the first step for the M_Z and M'_Z hexons and prepares them into the logical code space of $i\gamma_3\gamma_4$. The red paths indicate cutter gates that should be turned on while the green are the induced paths within the qubit device.

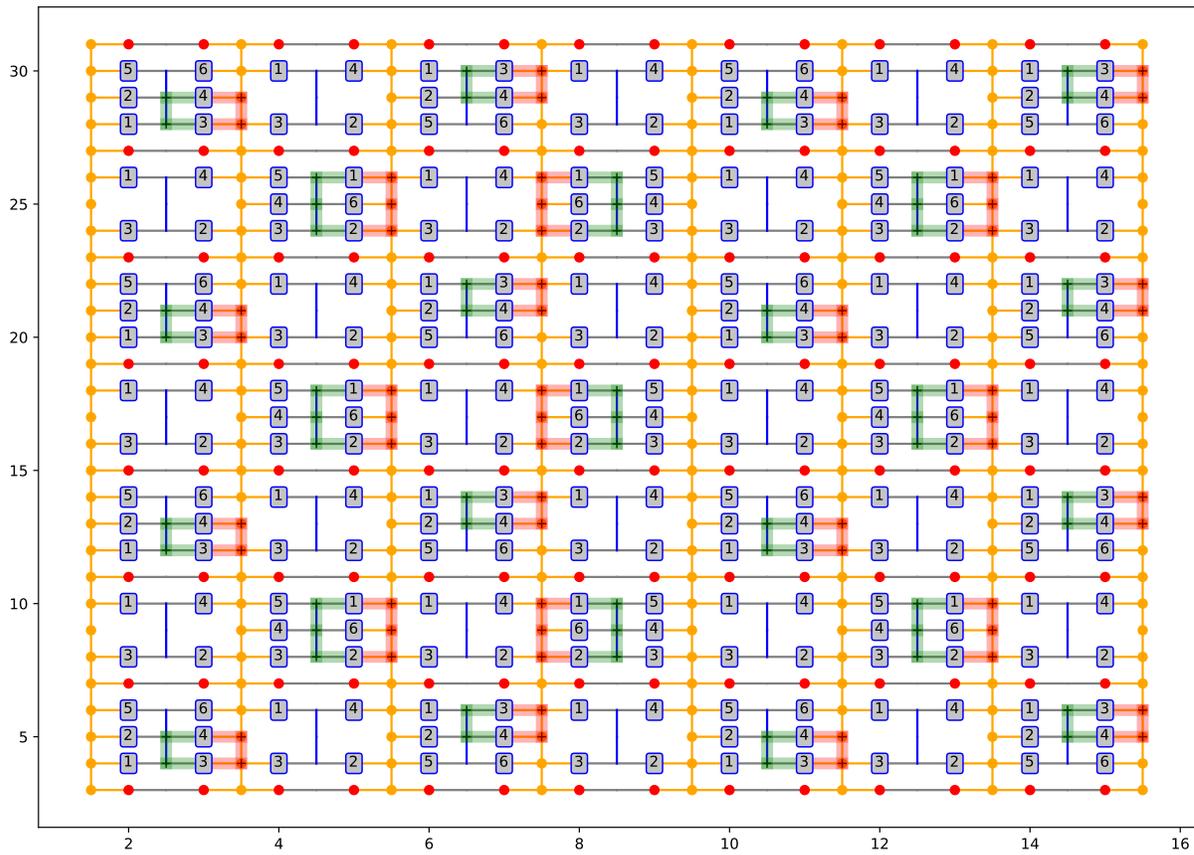


Figure B.2: Step 2 of Table B.1: measure $34|\circ\circ$ on M_X , $34|\circ\circ$ on M'_X , $12|\circ\circ$ on M_Z , and $12|\circ\circ$ on M'_Z . This step prepares the M_Z and M'_Z hexons into a logical $Z = i\gamma_1\gamma_2$ eigenstate.

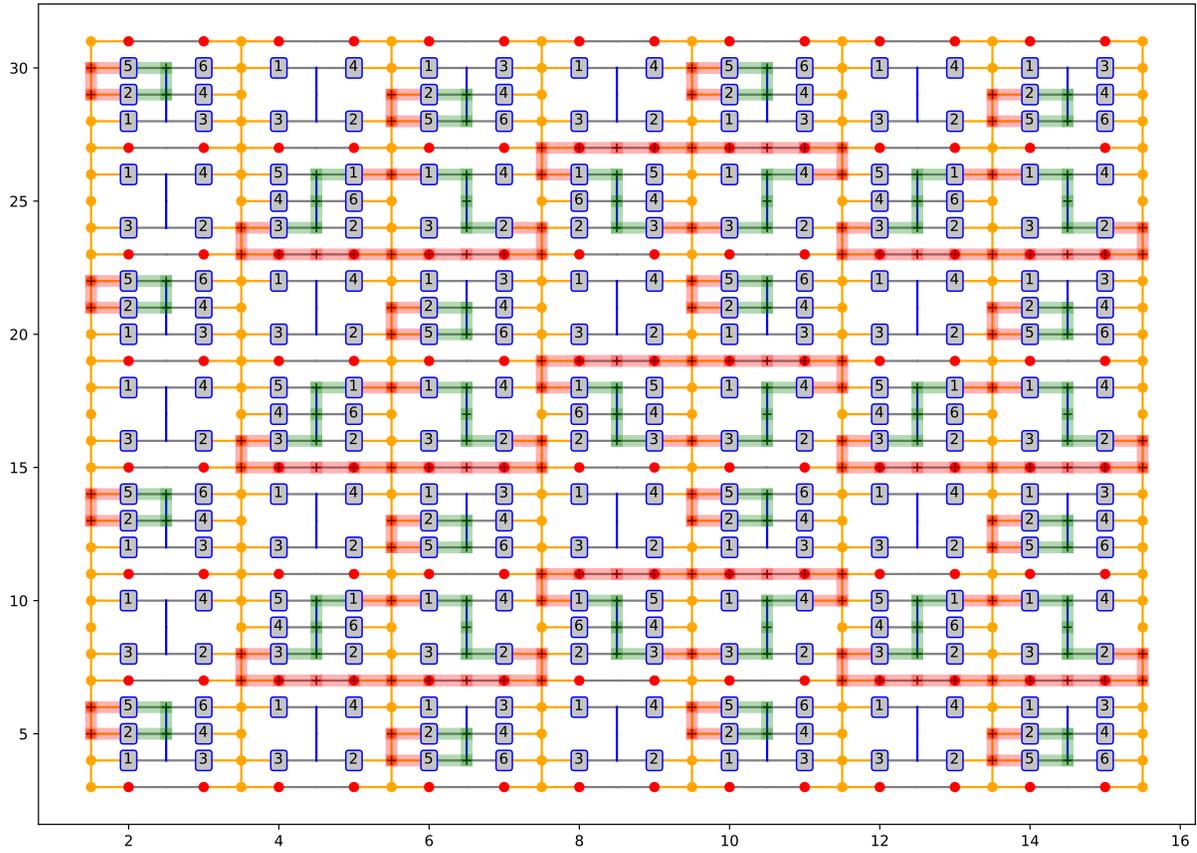


Figure B.3: Step 3 of Table B.1: measure $25|\circ\circ$ on M_X , $25|\circ\circ$ on M'_X , $13|34_r$ on M_Z , and $13|12_r$ on M'_Z . This step measures out the value of logical $X = i\gamma_2\gamma_5$ on the M_X and M'_X hexons. It is the final step and the ancillary M_X and M'_X hexon should be refreshed here.

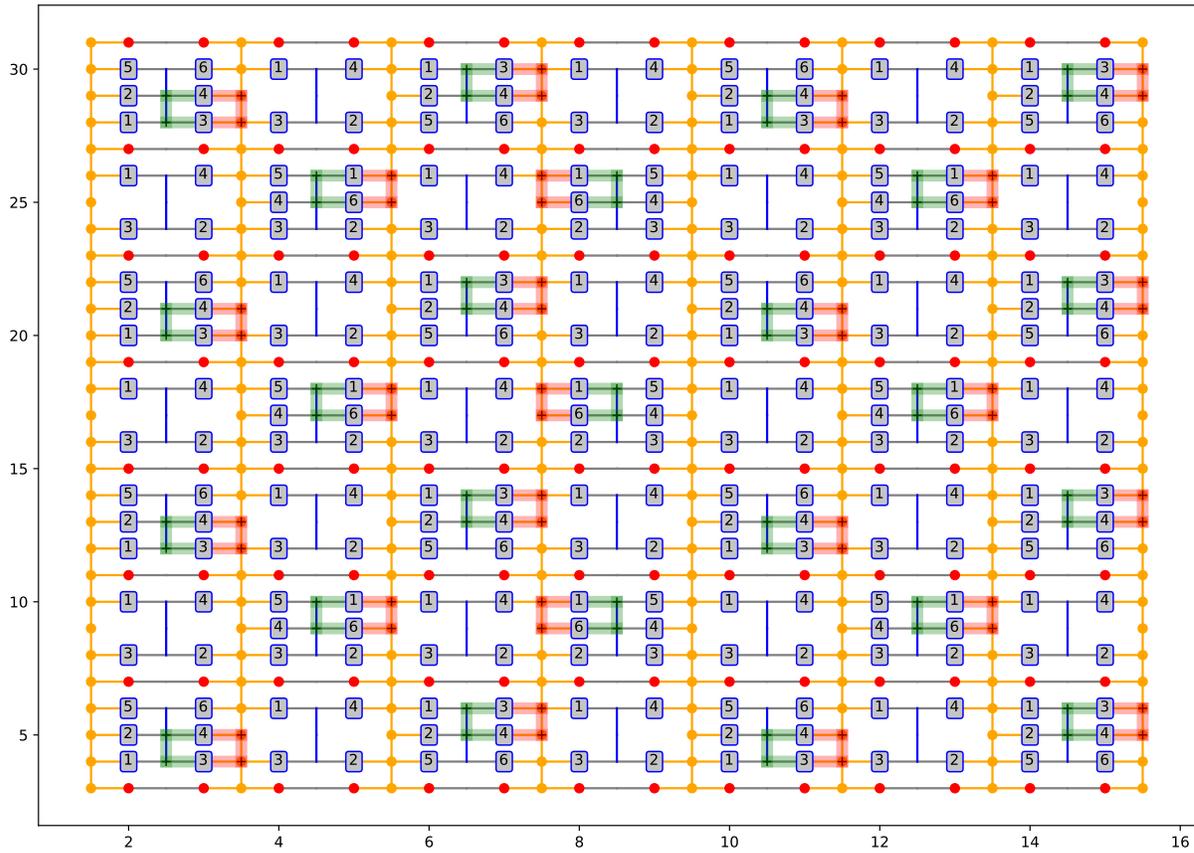


Figure B.4: Step 4 of Table B.1: measure $34|\circ\circ$ on M_X , $34|\circ\circ$ on M'_X , $16|\circ\circ$ on M_Z , and $16|\circ\circ$ on M'_Z . This is the first step for the M_X and M'_X hexons and prepares them into the logical code space of $i\gamma_3\gamma_4$.

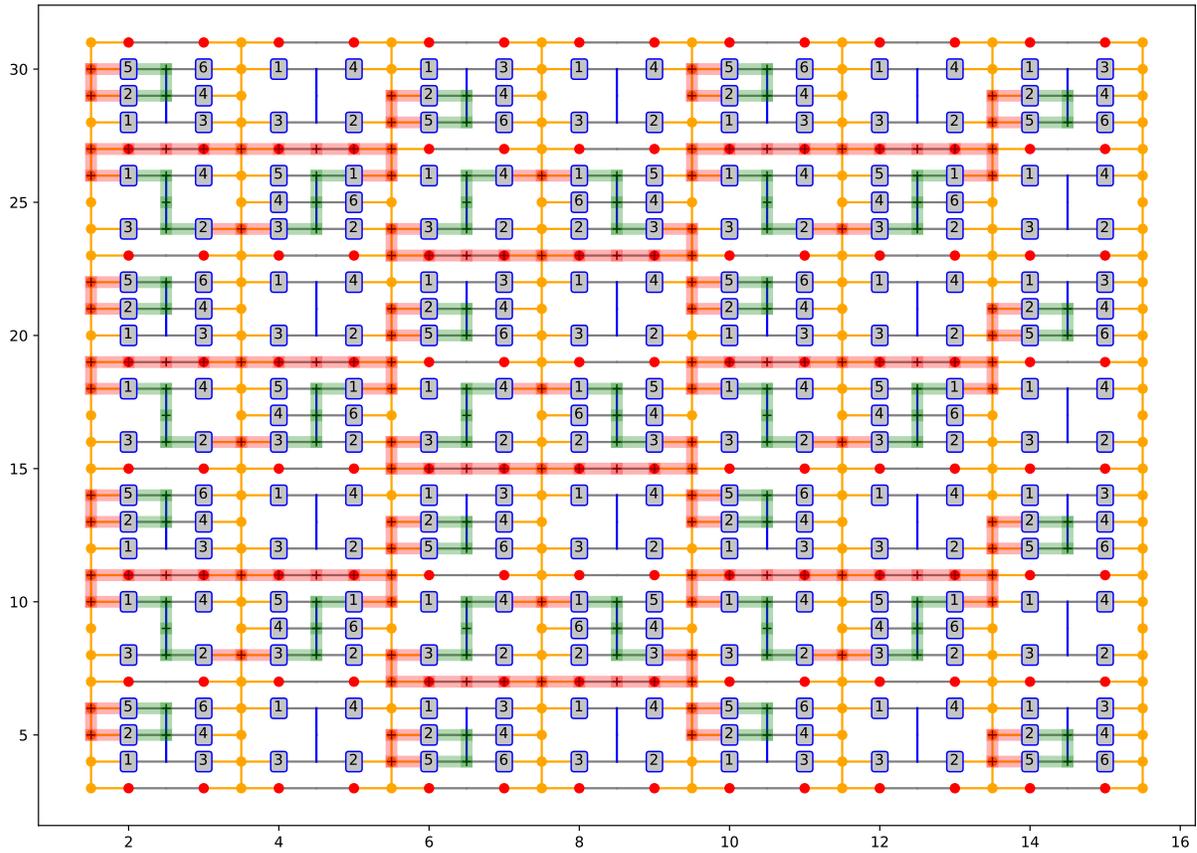


Figure B.5: Step 5 of Table B.1: measure $25|\circ\circ\rangle$ on M_X , $25|\circ\circ\rangle$ on M'_X , $13|34_l\rangle$ on M_Z , and $13|12_l\rangle$ on M'_Z . This step prepares the M_X and M'_X hexons into a logical $X = i\gamma_1\gamma_2$ eigenstate.

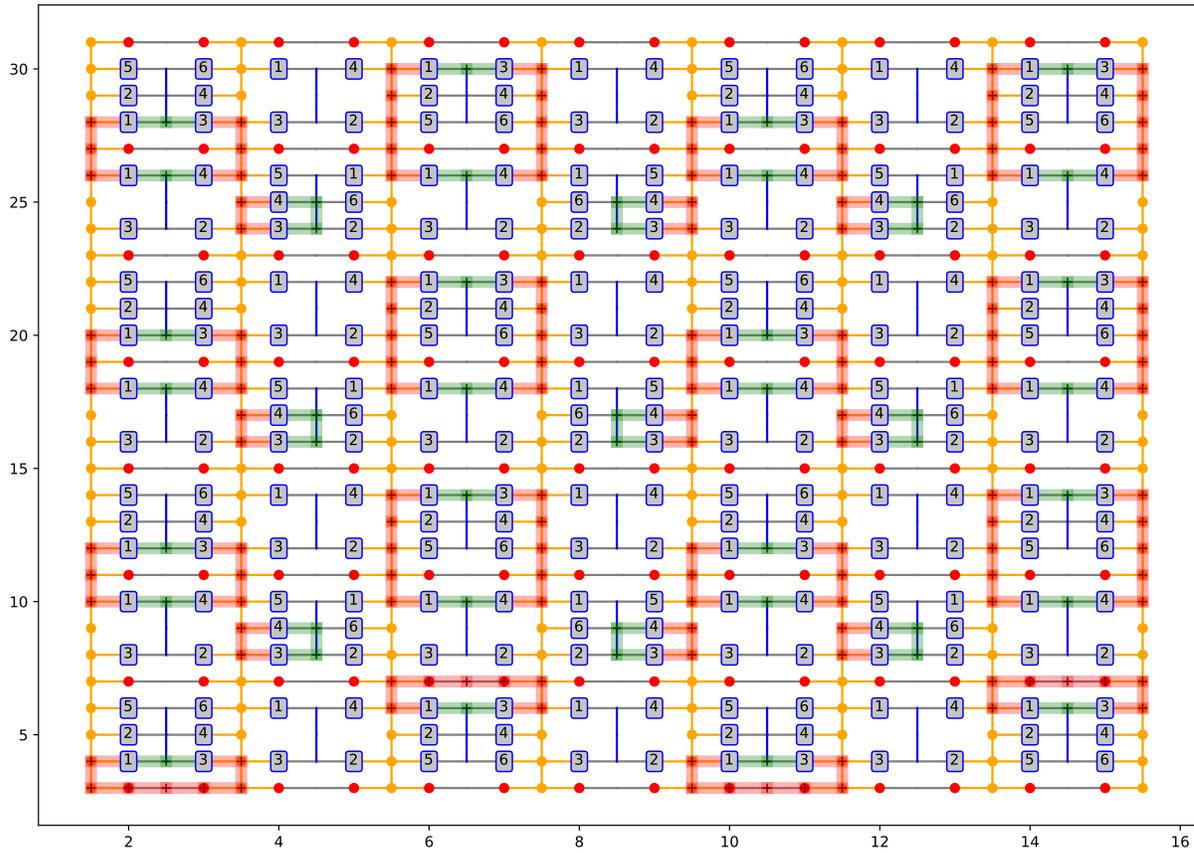


Figure B.6: Step 6 of Table B.1: measure $13|14_d$ on M_X , $13|14_d$ on M'_X , $34|\circ\circ$ on M_Z , and $34|\circ\circ$ on M'_Z .

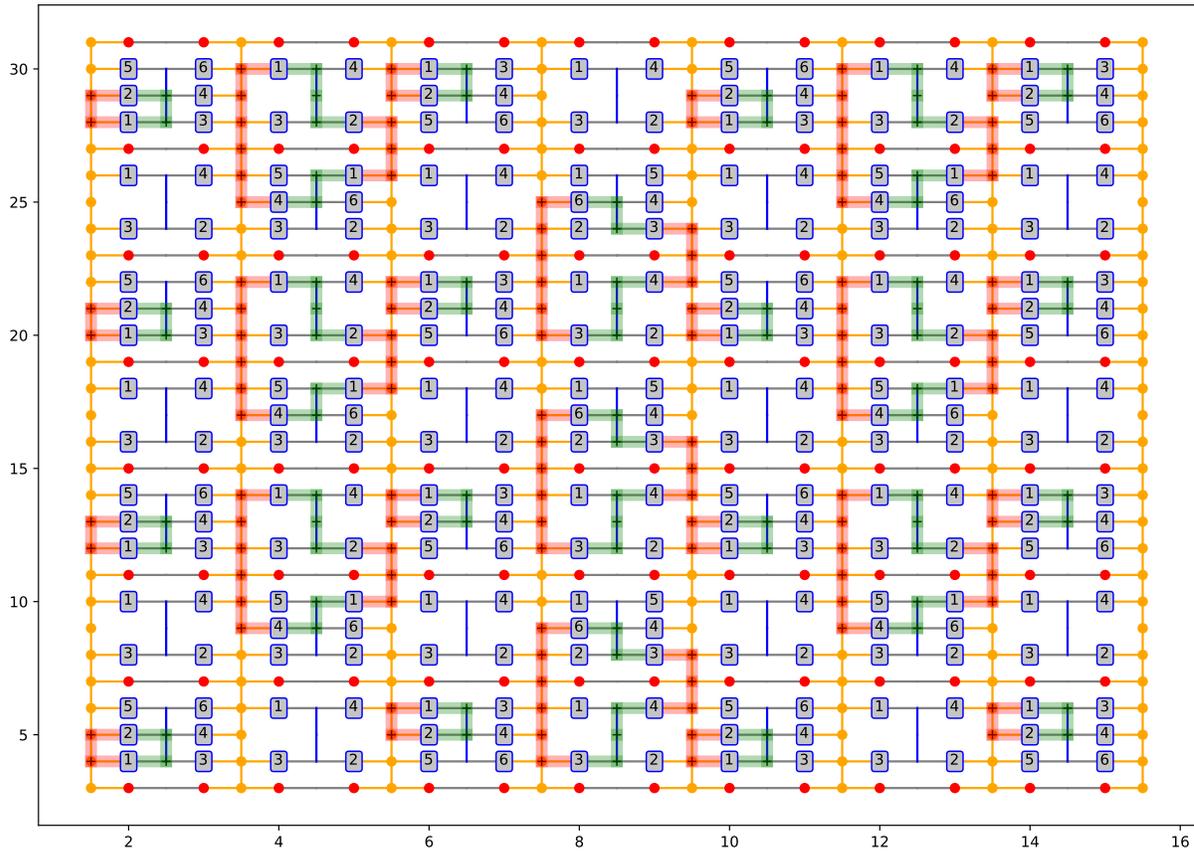


Figure B.7: Step 7 of Table B.1: measure $12|\circ\circ$ on M_X , $12|\circ\circ$ on M'_X , $36|34_d$ on M_Z , and $14|12_u$ on M'_Z .

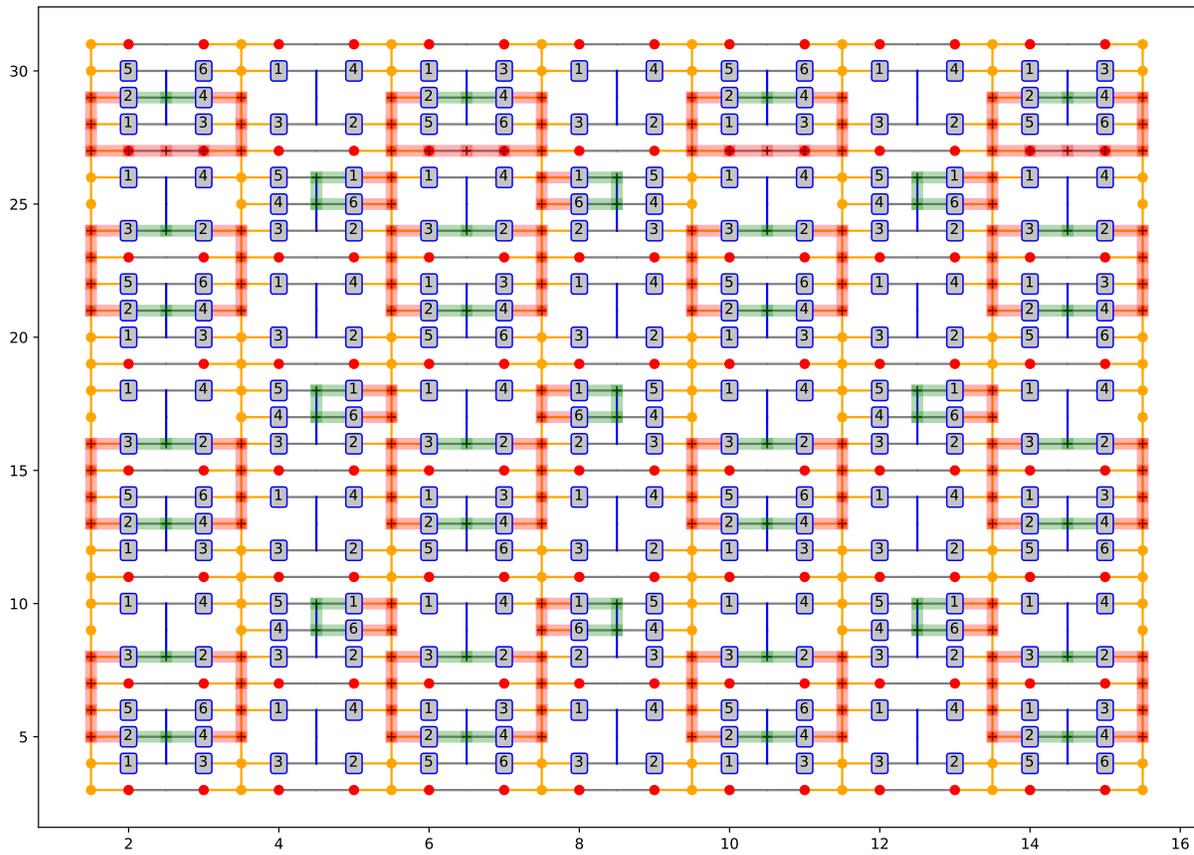


Figure B.8: Step 8 of Table B.1: measure $24|23_u$ on M_X , $24|23_u$ on M'_X , $16|\circ\circ$ on M_Z , and $16|\circ\circ$ on M'_Z .

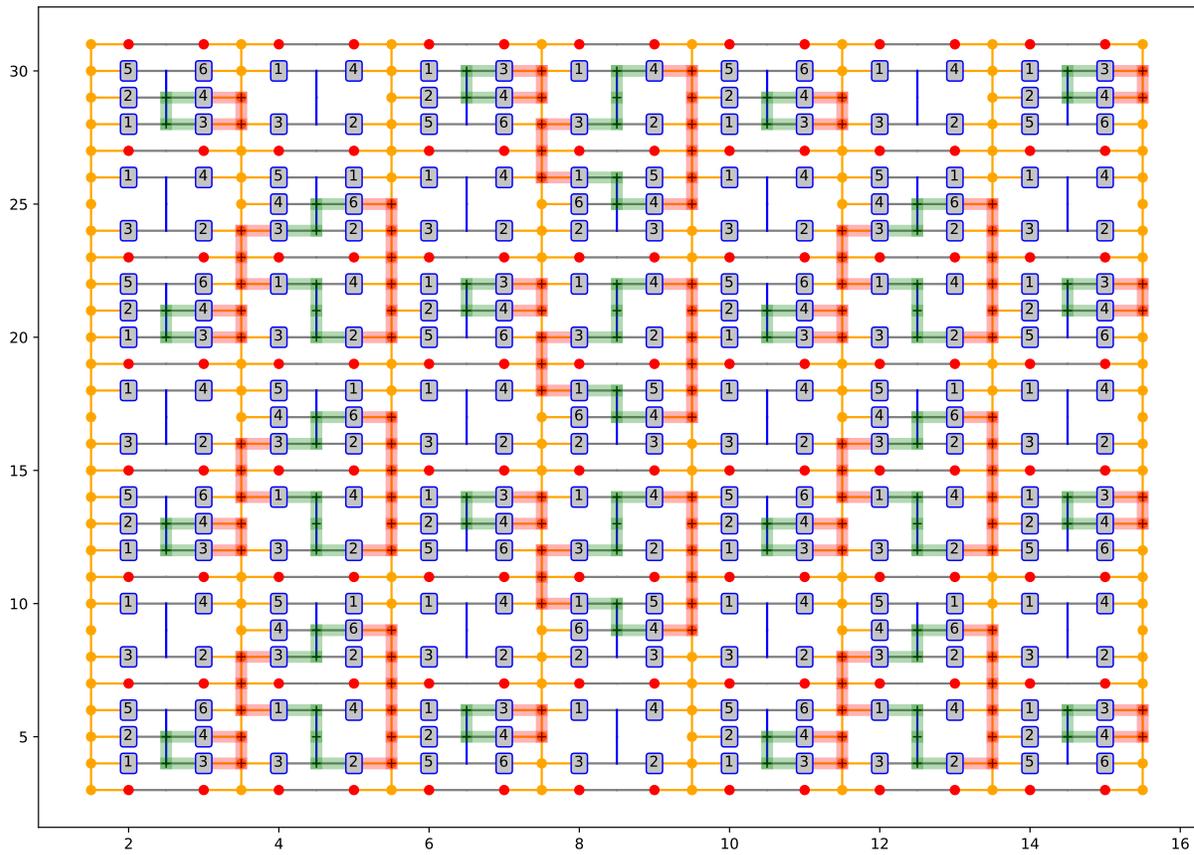


Figure B.9: Step 9 of Table B.1: measure $34|\circ\circ\rangle$ on M_X , $34|\circ\circ\rangle$ on M'_X , $14|34_u\rangle$ on M_Z , and $36|12_d\rangle$ on M'_Z .

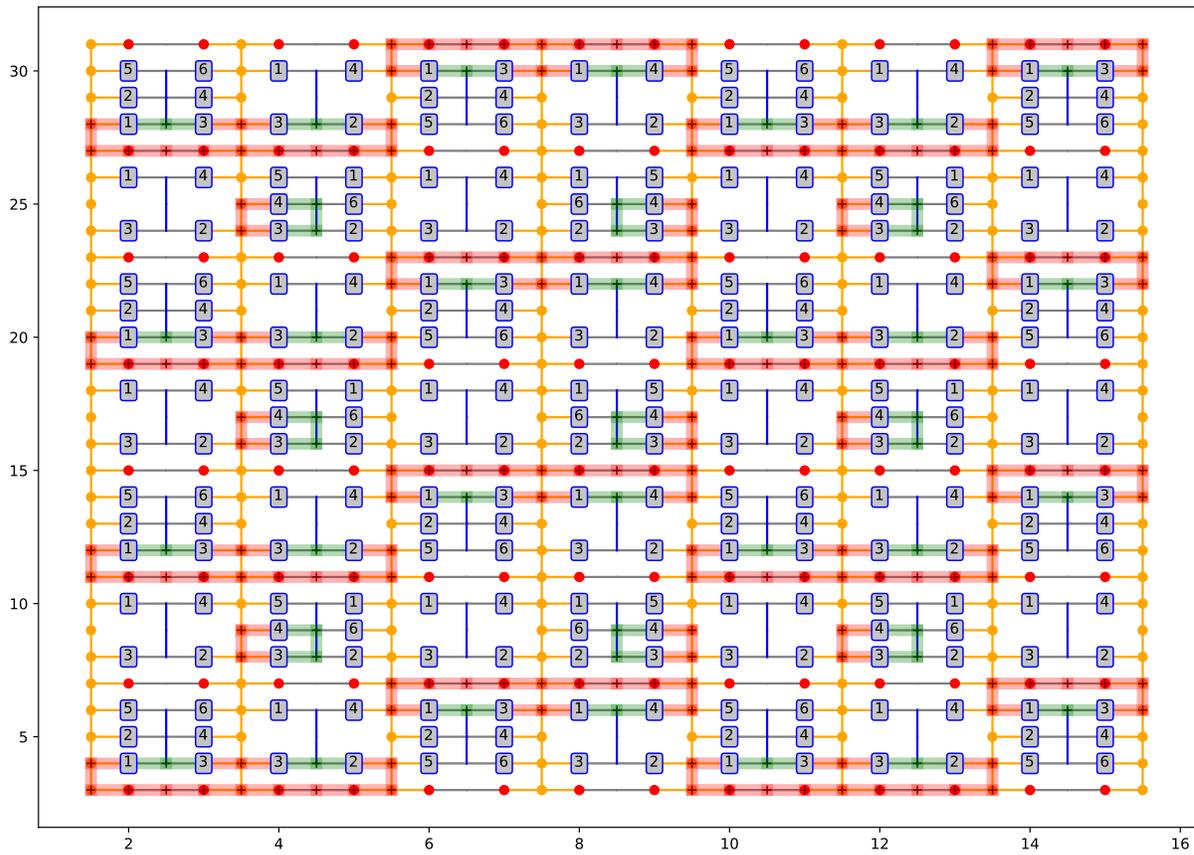


Figure B.10: Step 10 of Table B.1: measure $13|23_r$ on M_X , $13|14_r$ on M'_X , $34|\circ\circ$ on M_Z , and $34|\circ\circ$ on M'_Z .

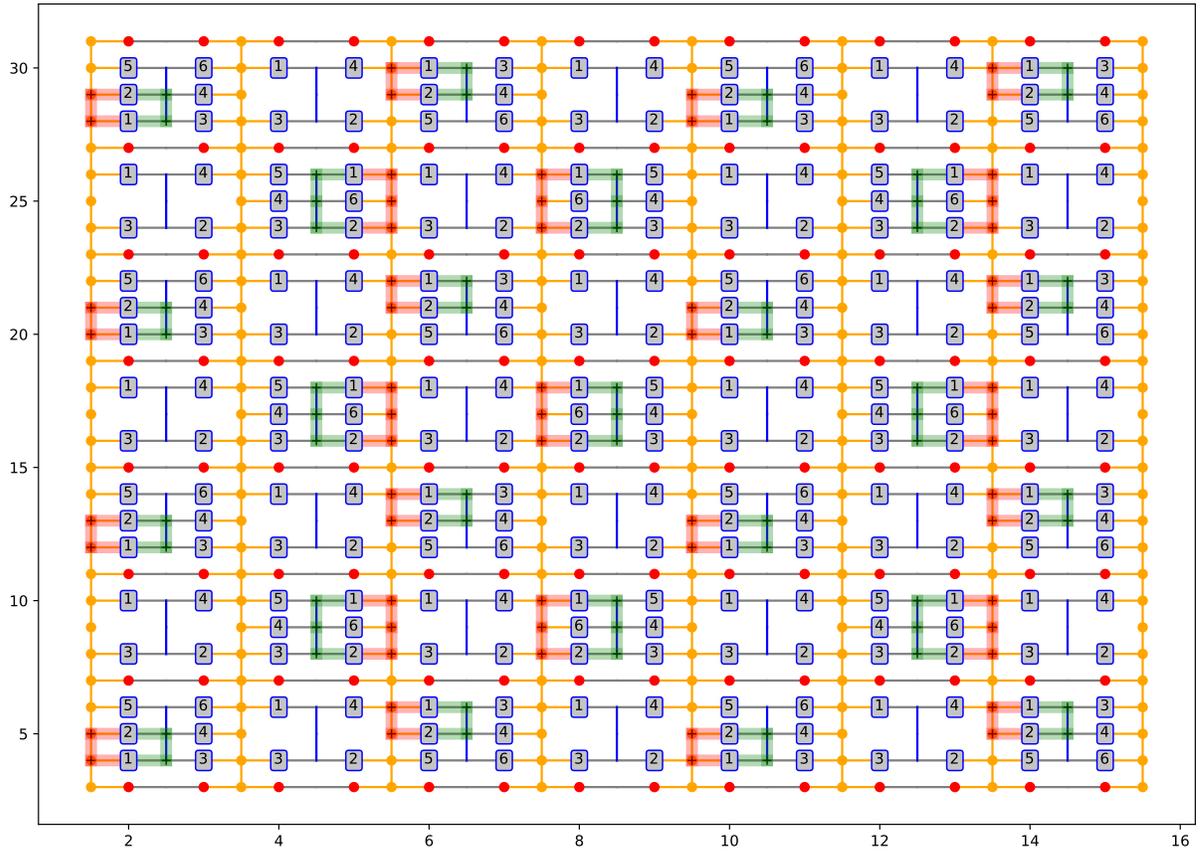


Figure B.11: Step 11 of Table B.1: measure $12|\circ\circ$ on M_X , $12|\circ\circ$ on M'_X , $12|\circ\circ$ on M_Z , and $12|\circ\circ$ on M'_Z . This step measures out the value of logical $Z = i\gamma_1\gamma_2$ on the M_Z and M'_Z hexons. It is the final step and the ancillary M_Z and M'_Z hexon should be refreshed here.

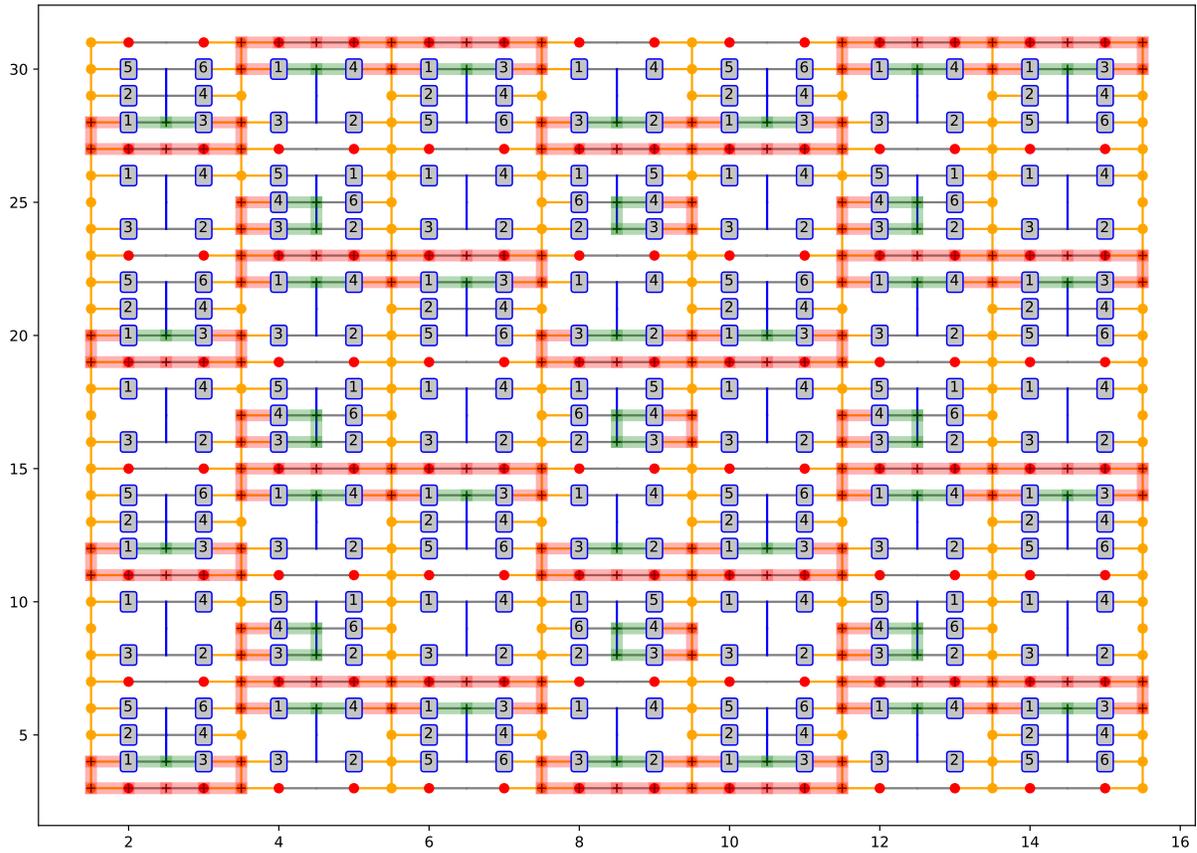


Figure B.12: Step 12 of Table B.1: measure $13|23_l$ on M_X , $13|14_l$ on M'_X , $34|\circ\circ$ on M_Z , and $34|\circ\circ$ on M'_Z . This is the first step for the M_Z and M'_Z hexons and prepares them into the logical code space of $i\gamma_3\gamma_4$.

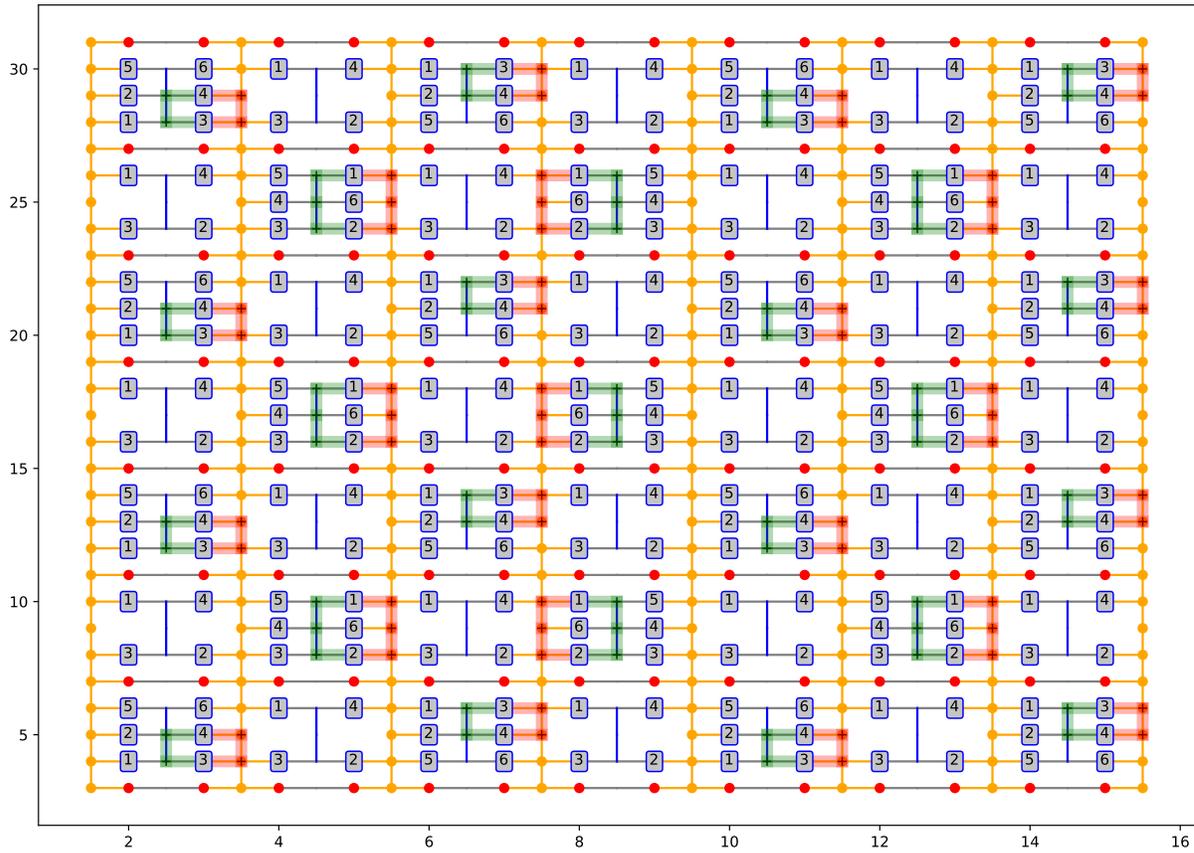


Figure B.13: Step 13 of Table B.1: measure $34|_{\circ\circ}$ on M_X , $34|_{\circ\circ}$ on M'_X , $12|_{\circ\circ}$ on M_Z , and $12|_{\circ\circ}$ on M'_Z . This step prepares the M_Z and M'_Z hexons into a logical $Z = i\gamma_1\gamma_2$ eigenstate.

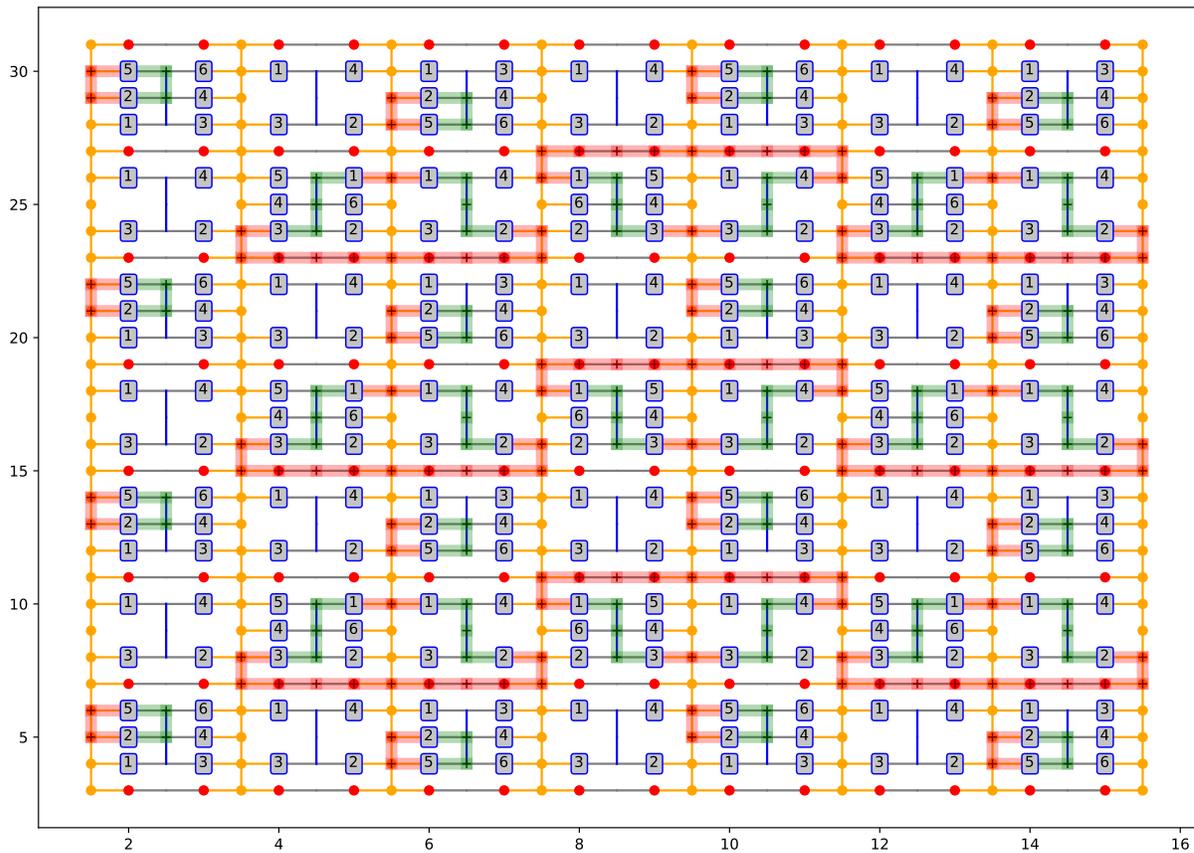


Figure B.14: Step 14 of Table B.1: measure $25| \circ \circ$ on M_X , $25| \circ \circ$ on M'_X , $13|34_r$ on M_Z , and $13|12_r$ on M'_Z . This step measures out the value of logical $X = i\gamma_2\gamma_5$ on the M_X and M'_X hexons. It is the final step and the ancillary M_X and M'_X hexon should be refreshed here.

Bibliography

- [1] Aasen, D., M. Hell, R. V. Mishmash, A. Higginbotham, J. Danon, M. Leijnse, T. S. Jaspersen, J. A. Folk, C. M. Marcus, K. Flensberg, and J. Alicea “Milestones Toward Majorana-Based Quantum Computing,” *Phys. Rev. X* **6**, 031016 (2016).
- [2] Alexander, R. N., P. S. Turner, and S. D. Bartlett, “Randomized benchmarking in measurement-based quantum computing,” *Physical Review A* **94** (3), 10.1103/PhysRevA.94.032303 (2016).
- [3] Alicea, J., “New directions in the pursuit of majorana fermions in solid state systems,” *Reports on Progress in Physics* **75** (7), 076501 (2012).
- [4] Arrazola, J. M., A. Delgado, B. R. Bardhan, and S. Lloyd, “Quantum-inspired algorithms in practice,” arXiv:1905.10415 [quant-ph] (2019).
- [5] Bagrets, D., and A. Altland, “Class d spectral peak in majorana quantum wires,” *Phys. Rev. Lett.* **109**, 227005 (2012).
- [6] Benioff, P., “The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines,” *J Stat Phys* **22**, 563 (1980).
- [7] Bennett, C. H., E. Bernstein, G. Brassard, and U. Vazirani, “Strengths and weaknesses of quantum computing,” *SIAM Journal on Computing* **26** (5), 1510, <https://doi.org/10.1137/S0097539796300933> (1997).
- [8] Bennett, C. H., and G. Brassard, “Quantum cryptography: Public key distribution and coin tossing,” *Theoretical Computer Science* **560**, 7 (2014).
- [9] Bennett, C. H., and G. Brassard, “Post-quantum cryptography,” *Nature* **549**, 10.1038/nature23461 (2017).
- [10] Bernstein, D. J., J. Buchmann, and E. Dahmen, Eds. (2009), *Post-Quantum Cryptography* (Springer Berlin Heidelberg, Berlin, Heidelberg).
- [11] Bocharov, A., and K. M. Svore, “Resource-Optimal Single-Qubit Quantum Circuits,” *Phys. Rev. Lett.* **109**, 190501 (2012).

- [12] Bombin, H., and M. A. Martin-Delgado, “Quantum measurements and gates by code deformation,” *Journal of Physics A: Mathematical and Theoretical* **42** (9), 095302 (2009).
- [13] Bonderson, P., “Measurement-only topological quantum computation via tunable interactions,” *Phys. Rev. B* **87**, 035113 (2013).
- [14] Bonderson, P., M. Freedman, and C. Nayak, “Measurement-Only Topological Quantum Computation,” *Phys. Rev. Lett.* **101**, 010501 (2008).
- [15] Bonderson, P., M. Freedman, and C. Nayak, “Measurement-only topological quantum computation via anyonic interferometry,” *Annals of Physics* **324** (4), 787 (2009).
- [16] Bonderson, P. H. (2007), *Non-Abelian Anyons and Interferometry*, Ph.D. thesis (California Institute of Technology), <https://thesis.library.caltech.edu/2447/2/thesis.pdf> .
- [17] Bravyi, S., and A. Kitaev, “Universal quantum computation with ideal Clifford gates and noisy ancillas,” *Phys. Rev. A* **71**, 022316 (2005).
- [18] Bravyi, S., B. M. Terhal, and B. Leemhuis, “Majorana fermion codes,” *New Journal of Physics* **12** (8), 083039 (2010).
- [19] Chen, J., P. Yu, J. Stenger, M. Hocevar, D. Car, S. R. Plissard, E. P. A. M. Bakkers, T. D. Stanescu, and S. M. Frolov, “Experimental phase diagram of zero-bias conductance peaks in superconductor/semiconductor nanowire devices,” *Science Advances* **3** (9), e1701476 (2017).
- [20] Churchill, H. O. H., V. Fatemi, K. Grove-Rasmussen, M. T. Deng, P. Caroff, H. Q. Xu, and C. M. Marcus, “Superconductor-nanowire devices from tunneling to the multichannel regime: Zero-bias oscillations and magnetoconductance crossover,” *Phys. Rev. B* **87**, 241401 (2013).
- [21] Cleve, R., D. Leung, L. Liu, and C. Wang, “Near-linear constructions of exact unitary 2-designs,” arXiv:1501.04592 [quant-ph] (2015).
- [22] Dankert, C., R. Cleve, J. Emerson, and E. Livine, “Exact and approximate unitary 2-designs and their application to fidelity estimation,” *Phys. Rev. A* **80**, 012304 (2009).
- [23] Das, A., Y. Ronen, Y. Most, Y. Oreg, M. Heiblum, and H. Shtrikman, “Zero-bias peaks and splitting in an Al-InAs nanowire topological superconductor as a signature of majorana fermions,” *Nature Physics* **8** (12), 887 (2012).
- [24] Delfosse, N., and N. H. Nickerson, “Almost-linear time decoding algorithm for topological codes,” arXiv:1709.06218 [quant-ph] (2017).
- [25] Delfosse, N., and G. Zemor, “Linear-time maximum likelihood decoding of surface codes over the quantum erasure channel,” *Physical Review Research* **2** (3), 10.1103/PhysRevResearch.2.033042 (2020).

- [26] Deng, M. T., C. L. Yu, G. Y. Huang, M. Larsson, P. Caroff, and H. Q. Xu, “Anomalous zero-bias conductance peak in a nb-insb nanowire-nb hybrid device,” *Nano Letters* **12** (12), 6414, pMID: 23181691, <https://doi.org/10.1021/nl303758w> (2012).
- [27] Dennis, E., A. Kitaev, A. Landahl, and J. Preskill, “Topological quantum memory,” *Journal of Mathematical Physics* **43** (9), 4452, <https://doi.org/10.1063/1.1499754> (2002).
- [28] Deutsch, D., and R. Jozsa, “Rapid solution of problems by quantum computation,” *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* **439** (1907), 553 (1992).
- [29] Deutsch, D., and R. Penrose, “Quantum theory, the church-turing principle and the universal quantum computer,” *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* **400** (1818), 97 (1985).
- [30] Feynman, R. P., “Simulating physics with computers,” *Int J Theor Phys* **21**, 467 (1982).
- [31] Finck, A. D. K., D. J. Van Harlingen, P. K. Mohseni, K. Jung, and X. Li, “Anomalous modulation of a zero-bias peak in a hybrid nanowire-superconductor device,” *Phys. Rev. Lett.* **110**, 126406 (2013).
- [32] Flensberg, K., “Tunneling characteristics of a chain of majorana bound states,” *Phys. Rev. B* **82**, 180516 (2010).
- [33] Fowler, A. G., “Two-dimensional color-code quantum computation,” *Phys. Rev. A* **83**, 042310 (2011).
- [34] Fowler, A. G., M. Mariantoni, J. M. Martinis, and A. N. Cleland, “Surface codes: Towards practical large-scale quantum computation,” *Phys. Rev. A* **86**, 032324 (2012).
- [35] Fu, L., and C. L. Kane, “Superconducting proximity effect and majorana fermions at the surface of a topological insulator,” *Phys. Rev. Lett.* **100**, 096407 (2008).
- [36] Gottesman, D. (1997), *Stabilizer codes and quantum error correction*, Ph.D. thesis (California Institute of Technology).
- [37] Gottesman, D., “The Heisenberg Representation of Quantum Computers,” arXiv:quant-ph/9807006 [quant-ph] (1998).
- [38] Gross, D., K. Audenaert, and J. Eisert, “Evenly distributed unitaries: On the structure of unitary designs,” *Journal of Mathematical Physics* **48** (5), 052104 (2007).
- [39] Hastings, M. B., “Small Majorana Fermion Codes,” arXiv:1703.00612 [quant-ph] (2017).
- [40] Hyart, T., B. van Heck, I. C. Fulga, M. Burrello, A. R. Akhmerov, and C. W. J. Beenakker, “Flux-controlled quantum computation with Majorana fermions,” *Phys. Rev. B* **88**, 035121 (2013).

- [41] Karzig, T., C. Knapp, R. M. Lutchyn, P. Bonderson, M. B. Hastings, C. Nayak, J. Alicea, K. Flensberg, S. Plugge, Y. Oreg, C. M. Marcus, and M. H. Freedman, “Scalable designs for quasiparticle-poisoning-protected topological quantum computation with Majorana zero modes,” *Phys. Rev. B* **95**, 235305 (2017).
- [42] Kassal, I., J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, “Simulating chemistry using quantum computers,” *Annual Review of Physical Chemistry* **62** (1), 185 (2011).
- [43] Kitaev, A., “Fault-tolerant quantum computation by anyons,” *Annals of Physics* **303** (1), 2 (2003).
- [44] Kitaev, A., “Anyons in an exactly solved model and beyond,” *Annals of Physics* **321** (1), 2 (2006).
- [45] Kitaev, A., and C. Laumann, “Topological phases and quantum computation,” arXiv:0904.2771 [cond-mat.mes-hall] (2009).
- [46] Kitaev, A. Y., “Unpaired Majorana fermions in quantum wires,” *Physics-Uspekhi* **44** (10S), 131 (2001).
- [47] Kjaergaard, M., K. Wölms, and K. Flensberg, “Majorana fermions in superconducting nanowires without spin-orbit coupling,” *Phys. Rev. B* **85**, 020503 (2012).
- [48] Knapp, C., M. Beverland, D. I. Pikulin, and T. Karzig, “Modeling noise and error correction for majorana-based quantum computing,” *Quantum* **2**, 88 (2018a).
- [49] Knapp, C., T. Karzig, R. M. Lutchyn, and C. Nayak, “Dephasing of majorana-based qubits,” *Physical Review B* **97** (12), 10.1103/PhysRevB.97.125404 (2018b).
- [50] Knill, E., “Quantum computing with realistically noisy devices,” *Nature* **434** (7029), 39 (2005).
- [51] Kubica, A., B. Yoshida, and F. Pastawski, “Unfolding the color code,” *New Journal of Physics* **17** (8), 083026 (2015).
- [52] Landau, L. A., S. Plugge, E. Sela, A. Altland, S. M. Albrecht, and R. Egger, “Towards realistic implementations of a majorana surface code,” *Phys. Rev. Lett.* **116**, 050501 (2016).
- [53] Law, K. T., P. A. Lee, and T. K. Ng, “Majorana fermion induced resonant andreev reflection,” *Phys. Rev. Lett.* **103**, 237001 (2009).
- [54] Lee, E. J. H., X. Jiang, R. Aguado, G. Katsaros, C. M. Lieber, and S. De Franceschi, “Zero-bias anomaly in a nanowire quantum dot coupled to superconductors,” *Phys. Rev. Lett.* **109**, 186802 (2012).

- [55] Lee, E. J. H., X. Jiang, M. Houzet, R. Aguado, C. M. Lieber, and S. De Franceschi, “Spin-resolved andreev levels and parity crossings in hybrid superconductor-semiconductor nanostructures,” *Nature Nanotechnology* **9** (1), 79 (2013).
- [56] Litinski, D., M. S. Kesselring, J. Eisert, and F. von Oppen, “Combining topological hardware and topological software: Color-code quantum computing with topological superconductor networks,” *Phys. Rev. X* **7**, 031048 (2017).
- [57] Litinski, D., and F. von Oppen, “Quantum computing with majorana fermion codes,” *Physical Review B* **97** (20), 205404 (2018).
- [58] Liu, C.-X., J. D. Sau, T. D. Stanescu, and S. Das Sarma, “Andreev bound states versus majorana bound states in quantum dot-nanowire-superconductor hybrid structures: Trivial versus topological zero-bias conductance peaks,” *Phys. Rev. B* **96**, 075161 (2017).
- [59] Liu, J., A. C. Potter, K. T. Law, and P. A. Lee, “Zero-bias peaks in the tunneling conductance of spin-orbit-coupled superconducting wires with and without majorana end-states,” *Phys. Rev. Lett.* **109**, 267002 (2012).
- [60] Lutchyn, R. M., E. P. A. M. Bakkers, L. P. Kouwenhoven, P. Krogstrup, C. M. Marcus, and Y. Oreg, “Majorana zero modes in superconductor–semiconductor heterostructures,” *Nature Reviews Materials* **3** (5), 52 (2018).
- [61] Lutchyn, R. M., J. D. Sau, and S. Das Sarma, “Majorana fermions and a topological phase transition in semiconductor-superconductor heterostructures,” *Phys. Rev. Lett.* **105**, 077001 (2010).
- [62] Magesan, E., J. M. Gambetta, and J. Emerson, “Characterizing quantum gates via randomized benchmarking,” *Physical Review A* **85** (4), 10.1103/PhysRevA.85.042311 (2012).
- [63] Motrunich, O., K. Damle, and D. A. Huse, “Griffiths effects and quantum critical points in dirty superconductors without spin-rotation invariance: One-dimensional examples,” *Phys. Rev. B* **63**, 224204 (2001).
- [64] Mourik, V., K. Zuo, S. M. Frolov, S. R. Plissard, E. P. A. M. Bakkers, and L. P. Kouwenhoven, “Signatures of majorana fermions in hybrid superconductor-semiconductor nanowire devices,” *Science* **336** (6084), 1003 (2012).
- [65] Nayak, C., S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, “Non-Abelian anyons and topological quantum computation,” *Rev. Mod. Phys.* **80**, 1083 (2008).
- [66] Nielsen, M. A., “A simple formula for the average gate fidelity of a quantum dynamical operation,” *Physics Letters A* **303** (4), 249 (2002).
- [67] Nielsen, M. A., and I. L. Chuang (2010), *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press).

- [68] Oreg, Y., G. Refael, and F. von Oppen, “Helical liquids and majorana bound states in quantum wires,” *Phys. Rev. Lett.* **105**, 177002 (2010).
- [69] Pawlak, R., S. Hoffman, J. Klinovaja, D. Loss, and E. Meyer, “Majorana fermions in magnetic chains,” *Progress in Particle and Nuclear Physics* **107**, 1, 19 (2019).
- [70] Perdrix, S., “State transfer instead of teleportation in measurement-based quantum computation,” *International Journal of Quantum Information* **03** (01), 219, <https://doi.org/10.1142/S0219749905000785> (2005).
- [71] Pientka, F., A. Keselman, E. Berg, A. Yacoby, A. Stern, and B. I. Halperin, “Topological superconductivity in a planar Josephson junction,” *Phys. Rev. X* **7**, 021032 (2017).
- [72] Pikulin, D. I., J. P. Dahlhaus, M. Wimmer, H. Schomerus, and C. W. J. Beenakker, “A zero-voltage conductance peak from weak antilocalization in a majorana nanowire,” *New Journal of Physics* **14** (12), 125011 (2012).
- [73] Plugge, S., L. A. Landau, E. Sela, A. Altland, K. Flensberg, and R. Egger, “Roadmap to majorana surface codes,” *Phys. Rev. B* **94**, 174514 (2016).
- [74] Prada, E., P. San-Jose, M. W. A. de Moor, A. Geresdi, E. J. H. Lee, J. Klinovaja, D. Loss, J. Nygaard, R. Aguado, and L. P. Kouwenhoven, “From Andreev to Majorana bound states in hybrid superconductor-semiconductor nanowires,” arXiv:1911.04512 [cond-mat.supr-con] (2019).
- [75] Preskill, J., “Topological Quantum Computation,” Lecture notes, <http://www.theory.caltech.edu/preskill/ph219/topological.ps> (2004).
- [76] Proctor, T. J., A. Carignan-Dugas, K. Rudinger, E. Nielsen, R. Blume-Kohout, and K. Young, “Direct randomized benchmarking for multiqubit devices,” *Physical Review Letters* **123** (3), 10.1103/PhysRevLett.123.030503 (2019).
- [77] Raussendorf, R., D. E. Browne, and H. J. Briegel, “Measurement-based quantum computation on cluster states,” *Phys. Rev. A* **68**, 022312 (2003).
- [78] Read, N., and D. Green, “Paired states of fermions in two dimensions with breaking of parity and time-reversal symmetries and the fractional quantum Hall effect,” *Phys. Rev. B* **61**, 10267 (2000).
- [79] Reiher, M., N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, “Elucidating reaction mechanisms on quantum computers,” *Proceedings of the National Academy of Sciences* **114** (29), 7555 (2017).
- [80] Sau, J. D., R. M. Lutchyn, S. Tewari, and S. Das Sarma, “Generic new platform for topological quantum computation using semiconductor heterostructures,” *Phys. Rev. Lett.* **104**, 040502 (2010).

- [81] Selinger, P., “Efficient Clifford+T Approximation of Single-qubit Operators,” *Quantum Information and Computation* **15** (1-2), 159 (2015).
- [82] Shor, P. W. (1994), in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pp. 124–134.
- [83] Stanescu, T. D., R. M. Lutchyn, and S. Das Sarma, “Majorana fermions in semiconductor nanowires,” *Phys. Rev. B* **84**, 144522 (2011).
- [84] Takei, S., B. M. Fregoso, H.-Y. Hui, A. M. Lobos, and S. Das Sarma, “Soft superconducting gap in semiconductor majorana nanowires,” *Phys. Rev. Lett.* **110**, 186803 (2013).
- [85] Tang, E., “A quantum-inspired classical algorithm for recommendation systems,” *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing - STOC 2019* 10.1145/3313276.3316310 (2019).
- [86] Tran, A., A. Bocharov, B. Bauer, and P. Bonderson, “Optimizing Clifford gate generation for measurement-only topological quantum computation with Majorana zero modes,” *SciPost Phys.* **8**, 91 (2020).
- [87] Vijay, S., and L. Fu, “Teleportation-based quantum information processing with majorana zero modes,” *Phys. Rev. B* **94**, 235446 (2016).
- [88] Vijay, S., T. H. Hsieh, and L. Fu, “Majorana fermion surface code for universal quantum computation,” *Physical Review X* **5** (4), 10.1103/PhysRevX.5.041038 (2015).
- [89] Viyuela, O., S. Vijay, and L. Fu, “Scalable fermionic error correction in majorana surface codes,” *Physical Review B* **99** (20), 10.1103/PhysRevB.99.205114 (2019).
- [90] Webb, Z., “The clifford group forms a unitary 3-design,” arXiv:1510.02769 [quant-ph] (2015).
- [91] Yung, M.-H., J. D. Whitfield, S. Boixo, D. G. Tempel, and A. Aspuru-Guzik, “Introduction to quantum algorithms for physics and chemistry,” *Quantum Information and Computation for Chemistry*, 67 (2014).
- [92] Zhang, H., D. E. Liu, M. Wimmer, and L. P. Kouwenhoven, “Next steps of quantum transport in Majorana nanowire devices,” *Nature Communications* **10** (1), 10.1038/s41467-019-13133-1 (2019).
- [93] Zheng, H., A. Dua, and L. Jiang, “Measurement-only topological quantum computation without forced measurements,” *New Journal of Physics* **18** (12), 123027 (2017).