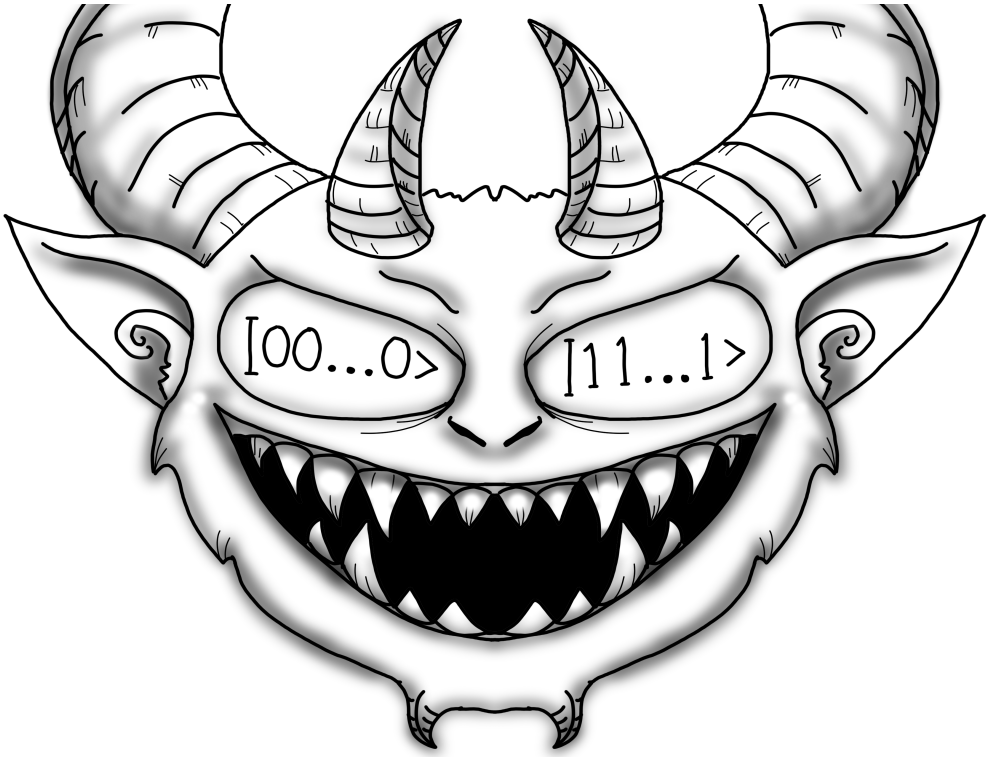
CHALMERS
UNIVERSITY OF TECHNOLOGY



Maxwell's Demon In Dynamic Quantum Circuits

Evaluation and Classical Simulation of Adaptive GHZ State Preparation Protocols

Master's thesis in Physics

OLOF HILDEBERG

DEPARTMENT OF MICROTECHNOLOGY AND NANOSCIENCE

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

MASTER'S THESIS 2026

Maxwell's Demon In Dynamic Quantum Circuits

Evaluation and Classical Simulation of Adaptive GHZ State Preparation Protocols

OLOF HILDEBERG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Microtechnology and Nanoscience
Division of Applied Quantum Physics
Chalmers university of technology
Gothenburg, Sweden 2026

Maxwell's Demon In Dynamic Quantum Circuits
Evaluation and Classical Simulation of GHZ State Preparation Protocols
OLOF HILDEBERG

Supervisor: Laura García-Álvarez
Examiner: Giulia Ferrini

Master's Thesis 2026
Department of Microtechnology and Nanoscience
Division of Applied Quantum Physics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 (0)31-772 1000

Cover: Maxwell's Demon observing the final GHZ state $|GHZ_N\rangle = \frac{1}{\sqrt{2}}(|00...0\rangle + |11...1\rangle)$, Saga Bokström Hildeberg, drawn digitally.

Typset in LaTeX.
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Maxwell's Demon In Dynamic Quantum Circuits
Evaluation and Classical Simulation of GHZ State Preparation Protocols
OLOF HILDEBERG
Department of Microtechnology and Nanoscience
Chalmers University of Technology

Abstract

Dynamic quantum circuits use intermediate measurements and classical feed-forward to change later operations during a computation. This makes them similar in spirit to Maxwell's demon, since measurement information is used to guide the evolution of the system. In this thesis, this idea is studied in the context of Greenberger–Horne–Zeilinger (GHZ) states, that is, entangled states of the form $(|0\rangle^{\otimes N} + |1\rangle^{\otimes N})/\sqrt{2}$, on noisy quantum devices. on noisy quantum devices. Three GHZ preparation protocols are compared: a non-adaptive protocol, a semi-adaptive protocol, and a fully adaptive protocol. The non-adaptive protocol uses only fixed unitary gates, while the adaptive protocols use ancilla measurements and conditional corrections. The protocols are implemented in a classical stabilizer simulation framework and compared using the final GHZ fidelity as the performance measure.

The simulations isolate four different error sources: CX gate errors, measurement errors, relaxation, and pure dephasing. This makes it possible to study not only which protocol performs best, but also which physical effects limit the performance of each protocol. In the CX gate-error regime, the results are determined by the number of CX gates. In the measurement-error regime, the adaptive protocols are limited by their reliance on intermediate measurements. For idle-time errors, the comparison is more subtle, since adaptive protocols reduce quantum depth but also introduce ancilla overhead and measurement and feed-forward delays.

For the noise models and timing assumptions used in this work, the non-adaptive protocol gives the highest fidelities in all isolated error regimes. The adaptive protocols therefore do not gain an advantage from their reduced depth under these conditions. The main bottleneck is found to be the measurement and feed-forward time, which exposes the data qubits to additional idle-time noise. This suggests that adaptive GHZ preparation could become more competitive on hardware with faster measurements, faster feed-forward, or lower-overhead adaptive constructions. The results show that reduced circuit depth alone is not sufficient to guarantee an advantage for dynamic circuits. Instead, the usefulness of adaptivity depends on the balance between gate count, circuit depth, measurement overhead, ancilla overhead, and hardware timing.

Keywords: dynamic quantum circuits, adaptive quantum circuits, GHZ state, state preparation, Maxwell's demon, quantum noise, stabilizer simulation

Acknowledgements

I would like to express my deepest gratitude to my supervisor Laura García-Álvarez for her invaluable feedback and expertise in quantum computing. Her persistent brainstorming which allowed me to continue with this work has also been vital.

I would also like to thank my sister Saga Bokström Hildeberg, for creating the cover image of this thesis, along with various other demons one may find in this work.

Olof Hildeberg, Gothenburg, May 2026

Use of AI and code availability

For transparency, AI tools were used as support during the preparation of this thesis, including for code generation, code revision, proofreading, improving readability, and finding relevant sources. AI-generated output was not used as an independent source of scientific information. All scientific claims, numerical results, and conclusions presented in the thesis are based on external sources, documented methods, or simulations performed by the author.

The code used in this work is available in a GitHub repository at `https://github.com/olofhildeberg/Masters-thesis-2026-OH-`.

Contents

1

Introduction

Quantum computing has the potential to solve certain problems more efficiently than classical methods, but current quantum devices, like superconducting circuits are limited by noise and decoherence [1]. Interactions with the environment, imperfect gate operations, and measurement errors all reduce the quality of quantum states during a computation. Because of this, the depth of a quantum circuit, meaning the number of sequential layers of operations, plays an important role in its performance, since longer circuits are more affected by errors that accumulate over time.

To reduce these effects, much effort has gone into improving quantum hardware, gate implementations, and circuit design in order to lower sensitivity to noise [1]. Another approach is to change how the circuit is run, and several works have explored what are known as adaptive quantum circuits [2]. In the standard circuit model, all operations are fixed in advance, and no information is used while the circuit is running [3]. An alternative approach is given by adaptive quantum circuits, where measurements are performed during the computation and used to guide later operations. By using this measurement information, these circuits introduce a different way of controlling how the quantum state evolves. Several known state-preparation protocols can be simplified using adaptive circuits. For example, Greenberger–Horne–Zeilinger (GHZ) states, W states, and Dicke states have been shown to be preparable in constant quantum depth using intermediate classical computations [2]. Constant depth means that the number of sequential quantum layers does not scale with the number of qubits in the circuit, instead of, for example, growing logarithmically with system size. This depth simplification is of great interest to reduce the errors that accumulate over time in quantum circuits, but it is still questionable whether that benefit outweighs the cost of the increased complexity caused by additional measurements, additional gates and corrective feed-forward operations which also introduce errors.

1.1 Maxwell's demon

The idea of Maxwell's demon was introduced as a thought experiment to explore the role of information in physical systems [4]. In this setup, a box of gas is divided into two compartments by a wall with a small door. A hypothetical demon controls this door and observes the motion of individual molecules. By allowing only fast molecules to pass in one direction and slow molecules in the other, the demon can create a temperature difference between the two sides without performing any apparent work. At first glance, this process seems to violate the second law of thermodynamics, since it appears to decrease entropy without any energy cost. The key point, however, is that the demon uses information about the system, namely the velocities of the molecules, to decide how to act. This thought experiment highlights that information itself can play an important role in physical processes, and that the act of measurement and decision-making must be taken into account when analyzing such systems. While the paradox is resolved when accounting for the physical cost of information processing, it remains a useful illustration of information-based control.

This relation between information and control can be connected to adaptive quantum circuits. In these circuits, measurements are used to extract information about the quantum state during the computation, and this information is then used to determine which operations should be applied next. In this sense, the classical feed-forward control acts in a way similar to Maxwell's demon, using measurement outcomes to guide the evolution of the system toward a desired final state.

1.2 Scope and outline

This thesis studies that idea of using measurement information to guide a quantum circuit in a concrete state preparation task. The target state chosen for this purpose is the GHZ state, of the form $(|0\rangle^{\otimes N} + |1\rangle^{\otimes N})/\sqrt{2}$, which is introduced in Chapter 3. GHZ states are useful for this comparison because they can be prepared using both fixed circuits and circuits with intermediate measurements and feed-forward.

The main question studied in this thesis is whether adaptivity improves GHZ state preparation on noisy quantum devices. Adaptive protocols can reduce quantum circuit depth, but they also require additional resources, including ancilla qubits, intermediate measurements, and classical feed-forward. These extra steps introduce new sources of error and additional idle time. We therefore also study how strongly different error sources affect the protocols, and which parts of the circuits are the main bottlenecks for performance. The goal is not only to determine which protocol gives the highest quality result, but also to explain why this happens in terms of gate errors, measurement errors, idle-time errors, ancilla overhead, and feed-forward delay.

To answer this question, three fixed GHZ preparation protocols are compared: one non-adaptive protocol, one semi-adaptive protocol, and one fully adaptive pro-

tocol. The comparison is performed using classical stabilizer simulations. The simulations study one noise source at a time, namely CX gate errors, measurement errors, relaxation, and pure dephasing. This makes it possible to see how each type of error affects the relative performance of the protocols.

The scope of the thesis is limited to this fixed-protocol comparison. The work does not search over all possible adaptive circuit constructions, and it does not include a complete hardware noise model where all error sources act at the same time. Final readout errors and error mitigation are also not included in the simulations. These simplifications make the comparison easier to interpret, but they also mean that the results should be understood within the chosen noise model and timing assumptions.

The thesis is organized as follows. Chapter 2 introduces the circuit model, adaptive circuits, the role of the Maxwell's demon, the noise mechanisms considered in this work, and fidelity as a performance measure. Chapter 3 introduces GHZ states and presents the three GHZ preparation protocols. Chapter 4 describes how the protocols and noise models are implemented in the classical simulations, how the fidelity is calculated within the stabilizer simulator and how we conduct the simulations. Chapter 5 presents the simulation results for the different error regimes. Chapter 6 discusses the results in relation to previous hardware work, summarizes the main conclusions, and gives an outlook on possible future directions.

2

Quantum circuits

This chapter introduces two general ways of building quantum circuits: non-adaptive circuits and adaptive circuits. In a non-adaptive circuit, all operations are fixed beforehand [3]. In an adaptive circuit, measurements can affect what operations are applied next. This adaptive behaviour can also be viewed from the perspective of a Maxwell's demon, as explained in section 1.1 [4]. In this interpretation, measurement outcomes provide information about the quantum system, and this information is then used to decide what corrective operations should be applied. From this point of view, adaptive circuits can be understood as protocols where information is used to improve the final state. Since real quantum devices are affected by noise, in section. 2.3 of this chapter, we also introduce the error models used in this work. These models will later be used to study how different circuit strategies perform under realistic conditions.

2.1 Non-adaptive quantum circuits

Non-adaptive quantum circuits are circuits where the sequence of operations is fixed before the computation starts [3]. The same gates are applied in the same order regardless of how the quantum state evolves during execution. In particular, no intermediate measurements are used to change the circuit. Such circuits are part of the standard quantum circuit model, where a computation is described as a sequence of unitary operations followed by measurements at the end [5].

In quantum algorithms, gates can often be parallelized, which determines the circuit depth. This makes depth a useful measure when comparing different circuit constructions, especially when studying the effect of noise, since longer circuits are typically more affected by noise. An example of a simple non-adaptive circuit is shown in Fig. 2.1, where a fixed sequence of single- and two-qubit gates is applied without any intermediate measurements.

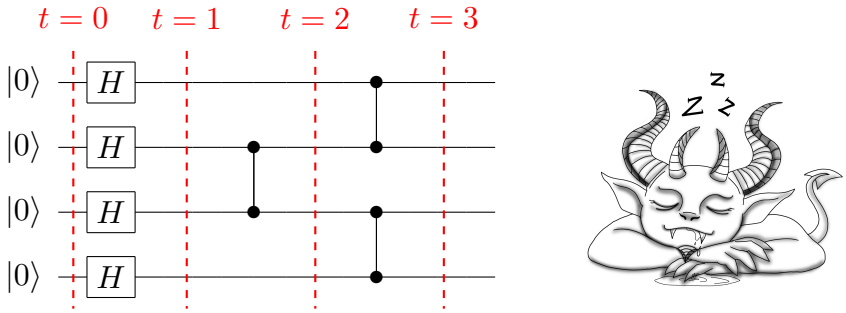


Figure 2.1: Example of a non-adaptive quantum circuit. Since no intermediate measurements are performed, Maxwell's demon remains idle. In the first layer, Hadamard gates are applied, followed by a CZ gate between qubits 2 and 3. The next layer contains CZ gates between qubits 1 and 2 and between qubits 3 and 4, which are performed simultaneously. The red dashed lines separate the circuit into sequential layers, so that operations between the same pair of dashed lines belong to the same layer.

For the remainder of this work, the red dashed lines will be omitted, and it will be assumed that gates within the same layer are applied simultaneously. In the absence of intermediate measurements, the Maxwell demon has no information to act upon and may therefore be considered unemployed.

2.2 Adaptive quantum circuits: Maxwell's demon in quantum computation

Adaptive quantum circuits differ from non-adaptive circuits in that the sequence of operations is not fully fixed in advance [3]. Instead, measurements performed during the computation can be used to decide which operations should be applied next. Such circuits are sometimes referred to as dynamic quantum circuits, since their structure may change depending on intermediate results. The new intermediate operations of measurement and classical feed-forward allow the circuit to react to the state of the system while the computation is running. Adaptive circuits are therefore useful in tasks such as quantum error correction, optimization of quantum algorithms, and measurement-based feedback protocols [3].

This behaviour can also be understood from the perspective of Maxwell's demon [4]. Measurements reveal information about the quantum system, allowing later operations to be chosen accordingly. From this viewpoint, adaptive circuits are protocols in which information is used to guide the evolution of the quantum state. An example of a simple adaptive circuit is shown in Fig. 2.2, where a measurement result determines which gate is applied in a later step.

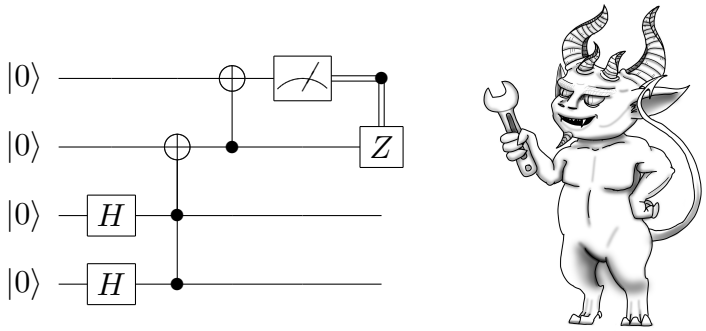


Figure 2.2: Example of an adaptive quantum circuit. With intermediate measurements, the Maxwell's demon decides the necessary corrections.

This example shows an adaptive circuit where single-qubit Hadamard gates are first applied to two of the qubits to create superposition states. These qubits are then entangled with the remaining qubits through a sequence of CCX and CX operations. After the entangling operations, one of the qubits is measured, producing a classical outcome of either 0 or 1. This measurement result is then used to determine whether a corrective Z gate should be applied to another qubit. Unlike the non-adaptive example, this circuit includes a measurement during execution, allowing later operations to depend on the observed outcome. In this case, the demon uses the measurement result to decide whether a corrective operation is necessary. Throughout this work, we will occasionally refer to this decision-making element as a demon, meaning the agent that performs measurements and decides which operations should follow based on the obtained outcomes.

2.3 Noise and Error models

Quantum circuits implemented on current hardware are affected by noise due to interactions with the environment, imperfect control, and measurement imperfections [1]. These noise processes limit the achievable circuit depth and reduce the quality of the prepared quantum states. Important error mechanisms in superconducting qubit platforms include energy relaxation (T_1), dephasing (T_2), gate errors, and measurement errors [1]. These processes arise from unavoidable coupling to the surroundings together with imperfections in control and readout. Since the goal of this work is to compare different GHZ preparation protocols under realistic conditions, we focus on these four error sources. Studying how they affect the circuits allows us to relate structural properties such as circuit depth, measurements, and gate count to noise sensitivity.

In this section we briefly describe the physical origin and qualitative effects of some of these noise sources that are considered in this work. The exact noise models used in the simulations are described later in section 4.3.

2.3.1 Measurement Errors

Measurement errors, also called readout errors, occur when the classical measurement result does not reflect the true quantum state of the qubit [6]. These errors typically originate from noise in the readout signal, which limits the ability to clearly distinguish between measurement outcomes, as well as relaxation processes that may occur during the measurement itself [1].

Measurement errors are particularly important in circuits that rely on intermediate measurements [7]. In adaptive circuits, measurement outcomes are used to determine which correction operations should be applied later in the circuit. An incorrect measurement result may therefore lead to an incorrect correction being applied, which can propagate errors and reduce the quality of the final state. This is a known challenge in quantum circuits that rely on feed-forward based on measurement results.

In the GHZ preparation protocols considered in this work, these kind of errors affects the adaptive and semi-adaptive circuits, where measurement results determine conditional corrections. The non-adaptive protocol does not rely on intermediate measurements and is therefore not affected by this error source.

2.3.2 Dephasing Noise (T_2/T_ϕ)

Dephasing describes the loss of phase coherence of a quantum state without energy exchange with the environment [5]. Physically, this typically arises from fluctuations in the qubit frequency caused by environmental noise such as charge noise or flux noise [1]. To illustrate the effect of dephasing, consider a qubit in a superposition state

$$|\psi\rangle = a|0\rangle + b|1\rangle.$$

The corresponding density matrix is

$$\rho = |\psi\rangle\langle\psi| = \begin{pmatrix} |a|^2 & ab^* \\ a^*b & |b|^2 \end{pmatrix} = \begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix}.$$

Under dephasing, the populations (the diagonal elements of the density matrix) remain unchanged, while the coherences (off-diagonal elements) decay with time [5]. A simple model gives

$$\rho(t) = \begin{pmatrix} \rho_{00} & \rho_{01}e^{-t/T_2} \\ \rho_{10}e^{-t/T_2} & \rho_{11} \end{pmatrix},$$

showing that dephasing leads to an exponential decay of the off-diagonal elements, which represent quantum coherence. In the absense of relaxation, T_2 corresponds to the pure dephasing time T_ϕ. The timescale T_2 is known as the coherence time, and characterizes how quickly this decay occurs. A shorter T_2 therefore corresponds to faster loss of coherence.

Physically, this corresponds to a gradual loss of information about the relative phase between the basis states [5]. As a result, superposition states lose their quantum character and approach mixed states that behave classically. This process is therefore often referred to as loss of coherence or decoherence. As an example, consider the superposition state

$$|+\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + |1\rangle \right).$$

The corresponding density matrix is

$$\rho = |+\rangle\langle+| = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}.$$

Under dephasing, the off-diagonal elements decay while the populations remain unchanged, giving

$$\rho(t) = \frac{1}{2} \begin{pmatrix} 1 & e^{-t/T_2} \\ e^{-t/T_2} & 1 \end{pmatrix}.$$

For long times, the off-diagonal elements vanish and the state approaches

$$\rho = \frac{1}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|.$$

This final state can no longer be written as $|\psi\rangle\langle\psi|$ for any state $|\psi\rangle$ [5], but instead represents a classical probabilistic mixture of the basis states. This illustrates how dephasing converts quantum superpositions into classical mixtures by destroying phase coherence.

2.3.3 Relaxation noise (T_1)

Relaxation noise, also known as amplitude damping describes energy relaxation of a qubit due to interactions with its environment [5]. Physically, this corresponds to spontaneous decay from the excited state $|1\rangle$ to the ground state $|0\rangle$. The ground state $|0\rangle$ is unaffected by amplitude damping, as it only causes $|1\rangle \to |0\rangle$ and not $|0\rangle \to |1\rangle$.

This process is characterized by the relaxation time T_1, which describes how quickly the excited state population decays [1]. Unlike dephasing noise, which leaves the populations unchanged, amplitude damping involves energy exchange with the environment and therefore affects both the populations and coherences of the density matrix [5]. Even in the absence of pure dephasing, relaxation causes decay of the off-diagonal elements of the density matrix. For this reason, the total decoherence time T_2 depends both on relaxation and pure dephasing processes.

The relation between these timescales is given by

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\phi},$$

where T_ϕ is the pure dephasing time. This shows that relaxation alone limits the maximum achievable coherence time, even in the absence of additional phase noise [1]. To illustrate the effect of amplitude damping, consider again a general qubit density matrix

$$\rho = \begin{pmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{pmatrix}.$$

Under amplitude damping, the excited state population decays while the ground state population increases [1]. A simple model gives

$$\rho(t) = \begin{pmatrix} 1 - \rho_{11} e^{-t/T_1} & \rho_{01} e^{-t/2T_1} \\ \rho_{10} e^{-t/2T_1} & \rho_{11} e^{-t/T_1} \end{pmatrix},$$

showing that relaxation affects both the populations and the coherences of the density matrix. In particular, the excited state population decays with timescale T_1, while the off-diagonal elements also decay, demonstrating that relaxation contributes to the overall loss of coherence.

2.3.4 Gate Errors

Two-qubit gates are typically a major source of errors in current quantum hardware [8]. Compared to single-qubit operations, two-qubit gates generally require longer execution times and more complex control, making them more susceptible to decoherence and other imperfections. In particular, it has been observed that two-qubit gate error rates increase with gate duration due to decoherence occurring during the interaction. For example, superconducting qubit platforms typically achieve single-qubit gate errors on the order of 10^{-4}–10^{-3}, while two-qubit gate errors are often an order of magnitude larger [9].

In this work, we therefore focus on errors associated with CX gates, also called controlled-NOT (CNOT) gates. A CX gate is a two-qubit operation that applies an X gate to a target qubit if the control qubit is in the state $|1\rangle$, and does nothing to the target qubit if the control qubit is in the state $|0\rangle$ [5]. The exact effect of two-qubit errors and how these errors are modeled are discussed in more detail in Chapter 4. GHZ state preparation circuits are dominated by two-qubit operations, while single-qubit gate errors are typically significantly smaller and therefore expected to have a reduced impact on the overall quality of the final state. Since the protocols compared in this work differ in their number of CX gates, modelling two-qubit errors provides a meaningful way to compare the final quality of the produced GHZ state. These errors can lead to imperfect entangling operations due to decoherence occurring during the gate interaction, thereby reducing the quality of the generated quantum states [8].

2.4 Fidelity as a performance measure

To compare how well different circuit constructions prepare the desired target state, we use the fidelity as a measure of state overlap. For two general quantum states described by density matrices ρ and σ, the fidelity is defined as [5]

$$\mathcal{F}(\rho, \sigma) = \mathrm{Tr}\left(\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right).$$

In this work, the target state is a pure GHZ state, which allows this expression to simplify. If the ideal target state is denoted by $|\psi_{\mathrm{tar}}\rangle$ and the state produced by the circuit is described by the density matrix ρ, then the fidelity becomes

$$\mathcal{F}(|\psi_{\mathrm{tar}}\rangle, \rho) = \sqrt{\langle\psi_{\mathrm{tar}}| \rho |\psi_{\mathrm{tar}}\rangle}.$$

Since the quantity inside the square root directly gives the overlap between the obtained state and the desired pure state, it is often convenient to use

$$\mathcal{F} = \langle\psi_{\mathrm{tar}}| \rho |\psi_{\mathrm{tar}}\rangle,$$

which takes values between 0 and 1 [10]. This is the convention more commonly used in the quantum computing community. A value of $\mathcal{F} = 1$ means that the target state has been prepared exactly, while smaller values indicate deviations due to noise, imperfect gates, decoherence, or measurement errors. This quantity will be used throughout the thesis to evaluate how successfully a given circuit prepares the desired GHZ state in the presence of noise.

3

GHZ state preparation

Several approaches exist for preparing GHZ states, which mainly differ in how measurements and classical information are used during the construction of the state. Some protocols use measurement outcomes to guide later operations, while others rely on fixed circuit structures without feedback. In this work, we compare adaptive, non-adaptive, and semi-adaptive preparation protocols, focusing on how the use of measurement information affects noise sensitivity and overall performance.

3.1 The Greenberger–Horne–Zeilinger State

The GHZ state is a fundamental example of multipartite (involving more than two qubits) entanglement and plays an important role in quantum information processing, quantum benchmarking [11], and quantum error correction [12]. In this section we introduce the definition of the GHZ state and briefly review the properties relevant for the preparation protocols studied in this work.

For three qubits, the GHZ state is defined as [11]

$$|GHZ_3\rangle = \frac{1}{\sqrt{2}} \left(|000\rangle + |111\rangle \right). \tag{3.1}$$

This state is a superposition of two basis states in which all qubits are either in the state $|0\rangle$ or $|1\rangle$. Because of this structure, measurements on one qubit are maximally correlated with measurements on the others. The GHZ state can be extended to N qubits as

$$|GHZ_N\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle^{\otimes N} + |1\rangle^{\otimes N} \right), \tag{3.2}$$

which describes the same type of correlation shared between all qubits in the system [11].

3.2 Non-adaptive GHZ Preparation

The non-adaptive GHZ preparation protocol described in this section follows the construction presented in Ref. [2]. In the non-adaptive protocol, the GHZ state is prepared using only unitary gates without intermediate measurements or classical feed-forward. The circuit begins by applying a Hadamard gate to one qubit to create a superposition, followed by a sequence of CX gates that distribute the entanglement across the remaining qubits. The Hadamard (H) gate, is a single-qubit gate that maps computational basis states into equal superpositions, according to [5]

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}. \tag{3.3}$$

In the ideal case, this preparation can be implemented with logarithmic circuit depth by arranging the CX gates in layers such that the number of entangled qubits doubles at each step. Since CX gates are two-qubit operations, a single qubit cannot participate in multiple CX operations simultaneously, which determines the required layering structure of the circuit. Since each layer doubles the number of entangled qubits, the number of layers k required to prepare an N-qubit GHZ state follows

$$2^k = N \quad \Rightarrow \quad k = \log_2(N). \tag{3.4}$$

This shows that, when only two-qubit gates are allowed, the circuit depth required to prepare an N-qubit GHZ state scales logarithmically with the system size in the ideal case. This remains true even with all-to-all qubit connectivity, since each layer of two-qubit gates can at most double the number of qubits entangled with the initially prepared qubit. Fig. 3.1 illustrates this structure for an 8-qubit GHZ state, where the number of entangled qubits grows from 1 to 2, then 4, then 8.

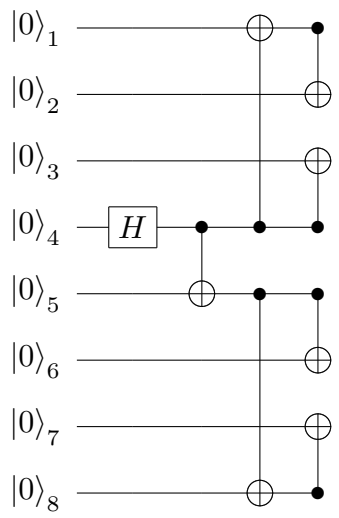


Figure 3.1: Non-adaptive circuit preparing an 8-qubit GHZ state. The CNOT gates are represented by a filled control dot connected to a circled plus symbol on the target qubit; the target qubit is flipped when the control qubit is in the state $|1\rangle$.

3.3 Adaptive GHZ Preparation

The adaptive GHZ preparation protocol described in this section is inspired by the construction presented in [2]. The adaptive protocol prepares the GHZ state by first creating local superpositions on the data qubits with Hadamard gates and then using ancilla qubits to fuse these states together. Each ancilla interacts with two neighboring data qubits through CX gates, storing parity information about the quantum state. In this way, the measurements reveal whether neighboring qubits are in the same or different computational basis states. After these interactions, the ancillas are measured and the outcomes determine which corrections must be applied to obtain the desired GHZ state. Figure 3.2 shows an example of the adaptive GHZ preparation circuit used in this work for four data qubits. The circuit illustrates how the ancilla qubits are used to fuse neighboring qubits through CX gates, followed by measurements and conditional corrections.

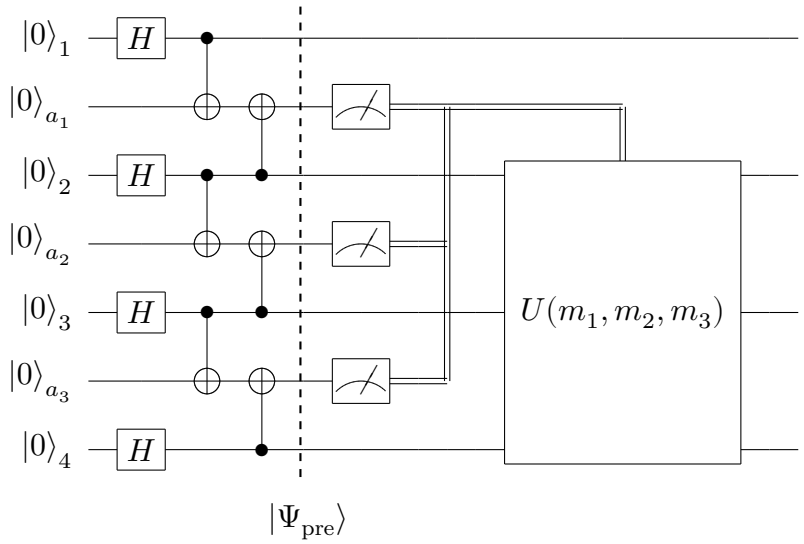


Figure 3.2: Four-qubit adaptive circuit preparing a GHZ state with ancilla fusion.

Compared to the non-adaptive protocol, this protocol increases the number of qubits required but reduces the required quantum circuit depth, since the entangling operations can be performed in a constant number of quantum layers independent of the number of data qubits. To understand how the measurement outcomes determine the required corrections, it is useful to examine the quantum state of the data qubits and ancillas immediately before the measurements are performed. At this stage, the system is in a superposition of all possible measurement branches, where each branch corresponds to a GHZ-type state differing only by known Pauli corrections. For the four-qubit example shown in Fig. 3.2, this state takes the explicit form

$$\begin{aligned}|\Psi_{\text{pre}}\rangle = \frac{1}{4} \quad & ((|0000\rangle + |1111\rangle)\,|000\rangle_a + (|0001\rangle + |1110\rangle)\,|001\rangle_a \\ & + (|0011\rangle + |1100\rangle)\,|010\rangle_a + (|0010\rangle + |1101\rangle)\,|011\rangle_a \\ & + (|0111\rangle + |1000\rangle)\,|100\rangle_a + (|0110\rangle + |1001\rangle)\,|101\rangle_a \\ & + (|0100\rangle + |1011\rangle)\,|110\rangle_a + (|0101\rangle + |1010\rangle)\,|111\rangle_a).\end{aligned}$$

A detailed derivation of this expression is provided in Appendix A.1. The derivation also shows that the required corrections never need to be applied to the first data qubit, as illustrated in Fig. 3.2. Measuring the ancilla qubits projects the system onto one of these branches. Each measurement outcome corresponds to a specific Pauli frame of the GHZ state, meaning that the resulting state differs from the target GHZ state only by known bit-flip operations. By applying conditional corrections based on the measurement outcomes, the protocol produces the desired GHZ state. As an example for this 4-qubit GHZ protocol, if the measurement outcome is $(m_1, m_2, m_3) = (1, 0, 0)$, the required correction is $U = X_2X_3X_4$, which yields

$$\frac{1}{\sqrt{2}}(|0000\rangle + |1111\rangle) = |\text{GHZ}_4\rangle\,.$$

3.4 Semi-adaptive GHZ Preparation

The semi-adaptive GHZ preparation protocol used in this work is inspired by fusion-based approaches, where smaller entangled states are combined to form larger GHZ states [13]. Unlike the fully adaptive protocol, this approach reduces the number of measurements by first preparing smaller GHZ states and then fusing them together using a single ancilla qubit.

This protocol can be seen as a compromise between the non-adaptive and fully adaptive approaches. Compared to the non-adaptive protocol, it introduces a measurement step that allows conditional corrections. At the same time, it requires fewer measurements than the fully adaptive protocol, but instead uses additional CX layers, which slightly increases the circuit depth.

Figure 3.3 shows the semi-adaptive GHZ preparation circuit used in this work for eight data qubits. First, two four-qubit GHZ states are prepared separately. These states are then fused together using an ancilla qubit that interacts with one qubit from each GHZ block through CX gates. Measuring the ancilla determines whether a correction is needed to obtain the final eight-qubit GHZ state.

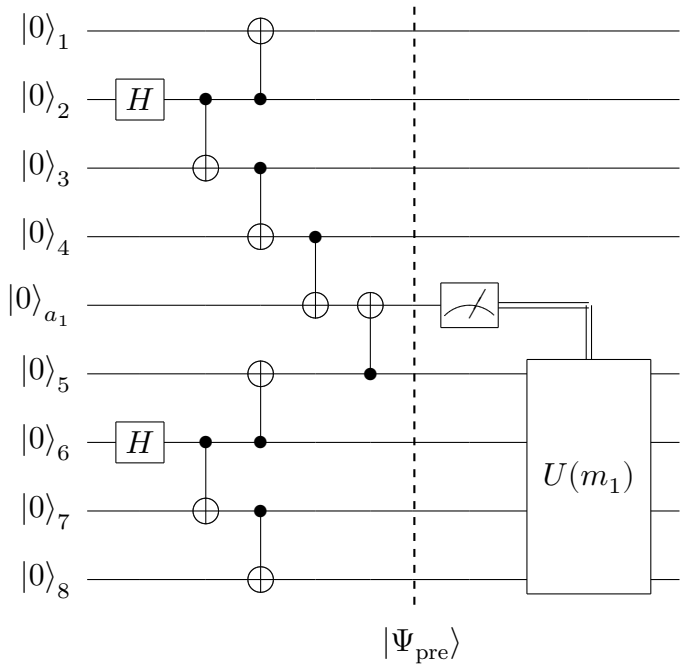


Figure 3.3: 8-qubit semi-adaptive circuit preparing a GHZ state with ancilla fusion.

Similar to the fully adaptive approach, the semi-adaptive protocol can also be implemented with constant quantum circuit depth, independent of the total number of data qubits. As shown in Fig. 3.3, only a single qubit from each GHZ block needs to interact with the ancilla in order to fuse the states, which keeps the number of required fusion operations small. This circuit can therefore be expanded to any number of qubits by connecting $|GHZ_4\rangle$ blocks and ancillas in parallel.

As in the adaptive protocol, it is useful to examine the quantum state immediately before the ancilla measurement to understand how the required corrections arise. For this eight-qubit example, this state takes the form

$$|\Psi_{\text{pre}}\rangle = \frac{1}{2}((|00000000\rangle + |11111111\rangle)\,|0\rangle_a + (|00001111\rangle + |11110000\rangle)\,|1\rangle_a).$$

The derivation follows analogously to the adaptive case, with the difference that the initial $|GHZ_4\rangle$ states before fusion are prepared using the non-adaptive protocol. Measuring the ancilla projects the system onto one of the two branches. If the measurement outcome is 0, the system is already in the desired GHZ state. If the outcome is 1, the state differs from the target GHZ state by known bit-flip operations on the second GHZ block, which can be corrected by $U = X_5X_6X_7X_8$ to yield $|GHZ_8\rangle$. And like the adaptive protocol, U needs to be applied to all GHZ-blocks except the first.

For the adaptive protocols, the use of measurement outcomes to determine conditional corrections can be interpreted as a Maxwell's demon–type mechanism. The measurement provides information about the relative alignment of the two GHZ blocks, which is then used to apply corrections that ensure successful GHZ state preparation.

4

Classical simulation of quantum circuits

This chapter describes how the GHZ preparation protocols introduced in chapter 3 were implemented in a classical simulation environment in order to compare their performance under realistic noise conditions. The focus is placed on the modeling choices required to translate the theoretical error models into a numerical simulation.

Classical simulation of quantum circuits can quickly become computationally expensive, since the required computational resources can grow exponentially with the number of qubits like in statevector simulation [14]. To make simulations of quantum systems feasible, it is therefore often necessary to introduce approximations that reduce the computational cost, especially with many qubits involved. In this work, this mainly involves choosing an efficient simulation framework and using noise models that capture the most important physical effects while still allowing many circuit simulations to be performed. Although such approximations may reduce physical accuracy, they make it possible to compare the relative performance of different protocols under consistent assumptions.

All simulations were performed in Python using the Qiskit framework. The full simulation code developed for this work is available in an online repository[1]. The following sections describe the chosen simulation framework, the implementation of the GHZ protocols, the applied noise models, and the procedure used to evaluate circuit performance.

4.1 Simulation framework

Classical simulation of quantum circuits can in general be computationally expensive, since the size of the quantum state grows exponentially with the number of

[1] `https://github.com/olofhildeberg/Masters-thesis-2026-OH-`

qubits [14]. For a system of n qubits, a general pure state requires storing 2^n complex amplitudes. When noise is included, a density matrix description is typically needed, which instead requires storing $2^n \times 2^n = 4^n$ complex numbers. This rapid growth quickly makes exact simulation impractical even for relatively small systems. To make the simulations feasible, it is therefore necessary to use more efficient simulation methods. In this work, this is done using the stabilizer simulator available in Qiskit, which is based on the stabilizer formalism. The stabilizer formalism provides an efficient way to describe a special class of quantum states known as stabilizer states [5]. These include many important states used in quantum information, including GHZ states. Instead of storing the full quantum state, these states can be described using a set of operators, called stabilizers, that leave the state unchanged. This allows the state to be tracked using a number of parameters that grows much more slowly with the number of qubits.

The reason this can be done efficiently is given by the Gottesman–Knill theorem [5]. This theorem states that quantum circuits consisting only of Clifford gates, Pauli measurements, and classical feed-forward operations can be simulated efficiently on a classical computer. The GHZ preparation protocols studied in this work only use Clifford operations such as Hadamard (H) and CX gates, together with measurements and conditional Pauli corrections, meaning they fall within this class.

This makes the stabilizer simulator well suited for this work, since it allows efficient simulation of the circuits considered. However, the restrictions imposed by the stabilizer formalism mean that not all noise processes can be implemented directly. In such cases, suitable approximations must be introduced to ensure compatibility with the simulator. Although this limits the exact physical description of the noise, it allows efficient and consistent comparisons between the different GHZ preparation protocols.

4.2 GHZ protocol modeling

In this section, we describe how the different GHZ preparation protocols were modeled in the simulations. Since the circuits have different structures, they also require different modelling assumptions. In particular, the non-adaptive circuit is modeled with limited qubit connectivity to better reflect hardware constraints, while the adaptive circuits do not have the same connectivity limitations since no qubit needs to connect to more than two other qubits.

4.2.1 Non-adaptive GHZ protocol

The non-adaptive GHZ preparation protocol scales logarithmically with the system size in the ideal case. However, this scaling usually assumes perfect connectivity between qubits, meaning that entangling gates can be applied between any pair of qubits. In practice, quantum devices usually have limited connectivity, meaning that each qubit can interact directly with only a small number of neighbouring

qubits [15]. This is especially relevant for the non-adaptive protocol, since preparing large GHZ states with logarithmic depth may require some qubits to interact with many different neighbours. By contrast, the adaptive protocols considered here only require each qubit to interact with at most two other qubits. Hardware connectivity limitations can therefore introduce additional routing operations for the non-adaptive protocol, increasing the circuit depth and reducing its practical advantage.

Fig. 4.1 shows the heavy-hex connectivity structure used in this work, which is based on IBM's quantum processor architecture [16]. This layout is designed to limit the number of connections per qubit, resulting in qubits having either two or three nearest neighbours instead of full connectivity. To account for this, the non-adaptive circuit was modeled using this hardware connectivity structure.

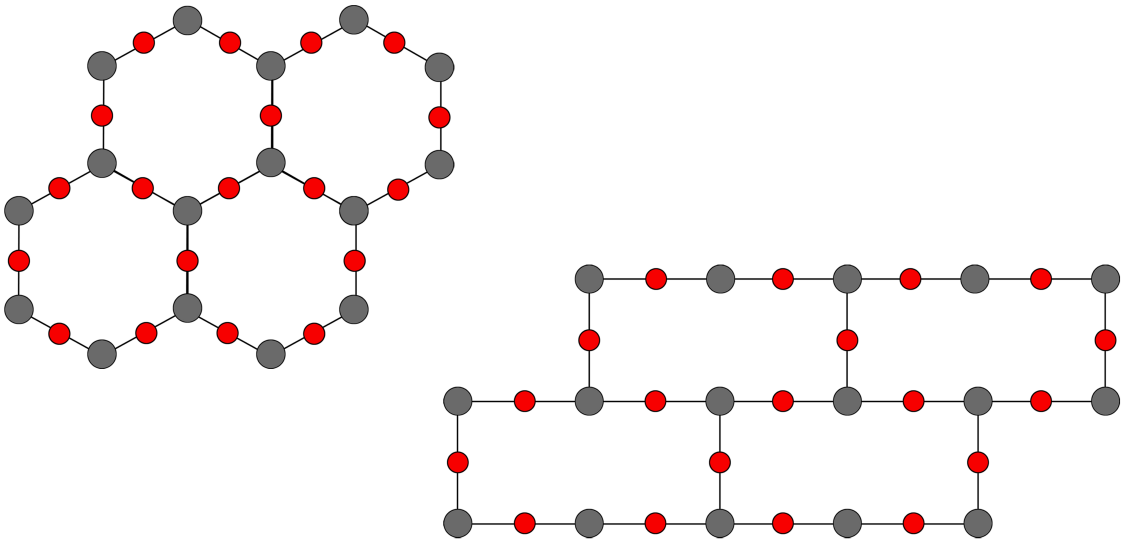


Figure 4.1: Two schematics of the heavy-hex qubit lattice, illustrating how qubits alternate between having two (red) and three (gray) nearest neighbours.

To model a routing procedure, the simulation first constructs a connectivity graph following this heavy-hex pattern. The GHZ state is then generated by expanding an entanglement tree along the available connections. To allow the entanglement to spread as efficiently as possible across the device, the procedure begins from a qubit near the center of the connectivity graph rather than at the edge. This helps minimize the circuit depth required to reach all qubits. This approach allows the protocol to be evaluated under more realistic connectivity constraints, while still keeping the modelling procedure simple enough to allow systematic comparisons with the adaptive protocols. The exact routing procedure used in our simulations is explained in Appendix A.2. Part of the code logic and structure was inspired from Ref. [17].

4.2.2 Semi-adaptive and adaptive GHZ protocols

For the semi-adaptive and adaptive GHZ preparation protocols, the main difference compared to the non-adaptive protocol lies in how measurement outcomes determine the required correction operations. Another important difference is that in these protocols each qubit only needs to interact with at most two neighbouring qubits during the fusion process. Because of this limited connectivity requirement, these protocols do not face the same hardware mapping challenges as the non-adaptive protocol. We therefore assume perfect connectivity for these circuits and do not model any specific hardware architecture.

During the fusion process, measurements introduce randomness into the state. However, the measurement outcomes also contain the information needed to determine which corrections must be applied in order to recover the standard GHZ form. Although the fusion operations in these protocols are implemented in parallel, it is helpful to understand the correction procedure by viewing the process sequentially. In this picture, each fusion step connects a previously prepared GHZ block to a new qubit through a measurement involving an ancilla. The measurement result then determines how the superposition branches of the two systems are connected.

To understand this intuitively, recall that a GHZ state consists of two computational basis branches,

$$|GHZ_n\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle^{\otimes n} + |1\rangle^{\otimes n}\right),$$

meaning that all qubits must agree on whether they are in the 0 or 1 branch. When a new qubit is fused with an existing GHZ block, both systems already contain such superpositions. The fusion measurement effectively determines whether the matching branches (0–0 and 1–1) or the crossed branches (0–1 and 1–0) become connected. If the matching branches are connected, the GHZ structure is obtained directly. However, if the crossed branches are connected, the previously constructed GHZ block becomes attached to the opposite branch of the new qubit. In this case the state still has GHZ structure, but with some qubits effectively flipped relative to the standard form. This can be corrected by applying X operations to all the previously connected qubits, which swaps their computational basis branches and restores the usual GHZ ordering.

Essentially, each fusion measurement determines whether the previously constructed GHZ block has been connected to the correct or opposite branch of the newly added qubit. A measurement outcome of 1 indicates that the previous GHZ block must be flipped in order to recover the standard GHZ structure. In contrast, a measurement outcome of 0 means that no correction is required at that fusion step. Since several fusion steps may affect the same qubits, the final correction cannot be determined from a single measurement alone. Instead, the required correction depends on how many times a given qubit has effectively been flipped during the fusion process. Because two flips cancel each other, only whether this number is even or odd matters. This means that the final correction can be determined by tracking the measurement outcomes and counting how many times each qubit is

affected. This operation corresponds to computing the parity of the qubits that the measurement results affect [5], which can be implemented using classical XOR operations. In this work, this tracking procedure plays the role of the demon in the protocol. Based on the measurement outcomes, it keeps track of which corrections are required.

As a concrete example, consider the 4-qubit fusion protocol discussed previously in Section 3.3 and illustrated in Appendix A.1. For the measurement branch $(m_1, m_2, m_3) = (1, 0, 1)$, the correction rule can be read directly from the fusion process. The first measurement outcome $m_1 = 1$ indicates that the first qubit must be flipped, while $m_2 = 0$ gives no additional flip. The third outcome $m_3 = 1$ indicates that the previously connected 3-qubit block must be flipped. As a result, qubit 1 is flipped twice and therefore unchanged, while qubits 2 and 3 are flipped once. The total correction pattern is therefore 0110, corresponding to the application of X on qubits 2 and 3. Eq. 4.1 shows this procedure.

$$\begin{aligned}\frac{1}{\sqrt{2}}\left(|0000\rangle + |1111\rangle\right) &\xrightarrow{m_1=1:X_4} \frac{1}{\sqrt{2}}\left(|0001\rangle + |1110\rangle\right) \xrightarrow{m_2=0:I} \frac{1}{\sqrt{2}}\left(|0001\rangle + |1110\rangle\right) \\ &\xrightarrow{m_3=1:X_2X_3X_4} \frac{1}{\sqrt{2}}\left(|0110\rangle + |1001\rangle\right).\end{aligned} \tag{4.1}$$

which matches the measurement branch shown in the circuit example. This illustrates how the measurement outcomes are used to keep track of which GHZ branch we are in, and therefore which correction is required.

Rather than explicitly applying all correction gates in the simulation, these corrections can equivalently be handled by updating the Pauli frame [18]. In this approach, the required X corrections are tracked classically instead of being physically applied as gates. This produces the same final state while avoiding additional circuit operations and reflects how feed-forward corrections are typically handled in stabilizer-based simulations and quantum hardware.

4.3 Noise model implementation

To study how errors affect the GHZ state preparation protocols, noise was added to the simulations through a simple stochastic error model compatible with efficient classical stabilizer simulation. The model considers four main error sources, which are implemented separately rather than simultaneously: relaxation during idle time, decoherence during idle time, two-qubit gate errors, and measurement errors affecting the readout of ancilla qubits. Each error source is studied in a separate error regime, where it is isolated from the other sources of error. Since the circuits in this work only use Clifford operations (Hadamard and CX gates) and Pauli measurements, the noise was modeled using stochastic Pauli errors.

The goal of this model is not to reproduce the exact noise of a specific quantum

device, but to include physically motivated error mechanisms that capture the main effects of decoherence, relaxation and imperfect operations. For the sake of comparing the different protocols, exact models of noises become less relevant than for exact quantitative results. However, we will make some comparisons to real quantum hardware, and provide some realistic choices of gate times which will be used to model the time delays between the layers of the circuit.

4.3.1 Relaxation and dephasing errors

Decoherence between circuit layers was modeled by inserting explicit delay steps after each layer of the circuit. During each such time step, qubits were subjected to a stochastic Pauli error channel representing relaxation and dephasing. As discussed in Section 2.3, in physical quantum devices, relaxation is typically described by the amplitude damping channel, which models energy loss from the excited state. Dephasing noise describes the loss of phase coherence between quantum states without necessarily involving energy loss. Relaxation and dephasing are usually parameterized by the coherence times T_1 and T_2. Here, T_1 describes energy relaxation, while T_2 describes the decay of phase coherence. The total dephasing time is related to the pure dephasing time T_ϕ by

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\phi},$$

where the first term represents the contribution from energy relaxation and T_ϕ denotes pure dephasing.

Simulating these channels exactly quickly becomes inefficient for multi-qubit systems [19]. Instead, the combined effect of relaxation and dephasing was approximated using a Pauli channel obtained through the Pauli twirling approximation. This replaces the physical decoherence process with stochastic X, Y, and Z errors with probabilities derived from the coherence times.

In the implementation, the probabilities of X, Y, and Z errors are calculated from the chosen values of T_1, T_ϕ (pure dephasing time), and the duration of one time step. These errors are then applied independently to each qubit during every delay step. This provides a simple way to include relaxation and dephasing while keeping the simulation compatible with stabilizer methods. In the simulation, relaxation and dephasing during an idle time step of duration dt are modeled using a Pauli-twirled single-qubit channel. The probabilities are calculated like they do in Ref. [19]

$$p_X = p_Y = \frac{1 - e^{-dt/T_1}}{4},$$

$$p_Z = \frac{1 - e^{-dt/T_2}}{2} - \frac{1 - e^{-dt/T_1}}{4} = \frac{1 - e^{-dt/T_2}}{2} - p_X,$$

where dt is whichever time length one may want, and we calculate T_2 using the relation $\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\phi}$. When it comes to the choice of dt, we need to take into

consideration the circuit structure, and which gates are used in the specific layers. All three circuits use the first layer to prepare some number of qubits in the $|+\rangle$ state using an H gate. This is a single qubit gate, and should have much less of time delay than two-qubit gates such as the CX gate [20]. Recent works show that current superconducting quantum hardware from Google and IBM supports single-qubit gates which have around 10-50ns gate duration, and two-qubit gates which can range from 40-400ns. Another work [21], suggests that choosing 20ns for an Rx-gate, which is an arbitrary rotation around the X-axis on the bloch sphere, and 300ns for a CZ gate, which is a control-Z gate, are realistic choices. With this evidence in mind, choosing a gate time of 20ns and 300ns for our H and CX gates seems reasonable.

We could therefore define $dt = 20$ns and insert a single time delay after the first Hadamard layer corresponding to its physical duration, and insert 15 time delays for the remaining CX layers. However, this would seriously slow down the simulations, due to the amount of instructions of the circuit. And considering that there are many more CX layers, the 20ns time delay caused by the Hadamard layer, would be negligible in comparison to the time delay contribution of the CX layers. We therefore neglect the time delay of the Hadamard layer of all circuits, and define $dt = 300$ns. Meaning that we apply a single time delay step after each CX layer defined to be 300ns long. However, both the adaptive and semi-adaptive protocols also have layers which require measurement of the ancilla qubits, as well as the classical post-processing to adjust the Pauli frame. And importantly, the data qubits have to remain idle during this time in order to be used, and will therefore be subjected to idle noise during this time.

According to IBM, they are able to do measurements, classical post-processing and corrective quantum gates in between 3-5μs when performing active qubit resets [22]. To avoid overly pessimistic assumptions, we will assume a total time delay of 3μs caused in total by the measurement of the ancillas, and the classical Pauli-reframing. We choose the lower bound of the reported data, since this corresponds to approximately ten CX gate durations, which already puts the non-adaptive protocol at a great advantage compared to the fully adaptive and semi-adaptive protocols. And since the current works suggests that the non-adaptive protocol out-performs the adaptive version [2], we investigate whether there are any error regimes in which the adaptive or semi-adaptive protocols can nevertheless create a GHZ state with higher fidelity. However, we also need to take into consideration when we apply these delay errors. If we apply them after the measurement has taken place, the ancilla qubits will be unaffected by them. Instead, we need to split up the measurement process and classical post processing, where the ancillas will be affected by the delays caused during the measurement, but not by the classical postprocessing, as we will have projected the ancilla states by then, and no longer need them.

To estimate a realistic measurement time, we checked the readout length from IBM backend calibration data using Qiskit. The value depends on the specific backend: for example, we found readout lengths of 2.584μs for ibm_marrakesh, 2.28μs for ibm_kingston, and 1.56μs for ibm_fez. This confirms that measurement

time is not fixed, but hardware-dependent. We choose 2.1 μs so it is divisible by 300ns and choose the feed-forward time to be 900ns to make the total measurement and classical feed-forward time to be 3μs. This means that 7 time delays are applied before the measurements, and 3 after, which will not affect the ancillas.

4.3.2 Gate errors

Gate errors were included only for the two-qubit CX gates. After each CX operation, each of the two involved qubits was given an independent probability p_{CX} of suffering a single-qubit Pauli error (X, Y, or Z). Each Pauli error was chosen with equal probability $p_{\mathrm{CX}}/3$, while the identity I was applied with probability $1 - p_{\mathrm{CX}}$. Since the errors on the two qubits are applied independently, this means that single-qubit gate faults occur with probability p_{CX}, while simultaneous errors on both qubits occur with probability p_{CX}^2.

This corresponds to a local depolarizing error model [5]. Since two-qubit gate errors depend on the hardware architecture and the specific gate implementation [19], we choose to model these two-qubit gate errors as simple Pauli error models to study general circuit behaviour rather than device-specific noise mechanisms. Depolarizing channels in particular provide a standard way of representing noise as random Pauli errors and are widely used because they are simple to analyze and capture the basic effect of gate imperfections.

No additional errors were included for single-qubit Hadamard gates. Since two-qubit gate errors dominate the overall circuit errors, as discussed in Subsection 2.3.4, restricting the gate noise model to CX errors captures the main contribution of gate imperfections while keeping the model simple.

4.3.3 Measurement errors

Measurement errors were modeled as classical readout errors on the ancilla qubits. After each measurement, the reported measurement outcome was flipped with probability p_m. This corresponds to a symmetric readout error model where both false positives and false negatives occur with equal probability. This simple model was chosen since the purpose of the simulations is to study how measurement imperfections affect protocol fidelity rather than reproduce specific hardware readout mechanisms.

4.4 Simulation procedure

The simulations were carried out by comparing the three GHZ preparation protocols under separate noise regimes. For 32, 64 and 128 data qubits, the corresponding non-adaptive, semi-adaptive, and fully adaptive circuits were constructed. Noise was then added according to the models described above, and the GHZ fidelity was estimated using a method that will be described in the next section.

To isolate the effect of each noise source, one noise parameter was varied at a

time while the remaining error sources were set to zero. Four separate parameter sweeps were performed. First, the CX gate error probability p_{CX} was varied in order to study sensitivity to two-qubit gate errors. Second, the measurement error probability p_m was varied for the adaptive and semi-adaptive protocols, since the non-adaptive protocol contains no intermediate measurements. Third, the relaxation time T_1 was varied while pure dephasing was neglected, corresponding to $T_\phi \to \infty$. Finally, the pure dephasing time T_ϕ was varied in order to study loss of phase coherence independently. Note that in the pure dephasing regime, $T_1 \to \infty$, so that $T_\phi = T_2$ by the relation

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\phi}.$$

For each point in a parameter sweep, the noisy circuit was sampled over many stochastic trajectories (shots). Each shot corresponds to one realization of the errors in the noise model. The final GHZ fidelity was then obtained by averaging over these simulation shots. This procedure was repeated for each protocol and for each number of data qubits, allowing the noise sensitivity of the different circuit constructions to be compared under identical assumptions. However, since two of these results are somewhat trivial by inspection of the circuits, only the simulation of 64 data qubits will be shown for the CX and measurement errors.

The resulting fidelities are presented in Chapter 5 as functions of the swept noise parameter. This makes it possible to identify which protocol is most sensitive to each error source, and to relate the observed behaviour to circuit properties such as CX count, number of measurements, circuit depth, and total idle time.

4.5 Fidelity evaluation

The fidelity defined in Section 2.4 as $\mathcal{F} = \langle\psi_{\mathrm{tar}}|\,\rho\,|\psi_{\mathrm{tar}}\rangle$ was evaluated using stabilizer simulations of the noisy circuits. Since the noise model introduces stochastic Pauli errors, each simulation shot produces one possible state corresponding to one realization of the noise. The physical noisy state is therefore described by a statistical mixture of these possible outcomes [5],

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|,$$

where p_i is the probability of state $|\psi_i\rangle$ occurring in a single simulation. The density matrix is therefore estimated by averaging observables over many simulation shots. Starting from the fidelity definition with $|\psi_{\mathrm{tar}}\rangle = |GHZ\rangle$

$$\mathcal{F} = \langle GHZ|\rho|GHZ\rangle,$$

and writing the N-qubit GHZ state as

$$|GHZ\rangle = \frac{1}{\sqrt{2}}(|0^N\rangle + |1^N\rangle),$$

the fidelity can be expanded as

$$\mathcal{F} = \frac{1}{2}(\langle 0^N|\rho|0^N\rangle + \langle 1^N|\rho|1^N\rangle + \langle 0^N|\rho|1^N\rangle + \langle 1^N|\rho|0^N\rangle).$$

The first two terms correspond to the populations of the GHZ basis states, while the last two terms describe the coherence between them. Using tensor products of Pauli operators [5], the operator $X^{\otimes N}$ takes the form

$$X^{\otimes N} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \ddots & \iddots & 0 & 0 \\ \vdots & \vdots & \iddots & \ddots & \vdots & \vdots \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} = |0^N\rangle\langle 1^N| + |1^N\rangle\langle 0^N| + \text{remaining off-diagonal elements},$$

meaning it flips all bits in a computational basis state. This allows the coherence terms to be written as the expectation value [5]

$$\langle X^{\otimes N}\rangle = \mathrm{Tr}(\rho X^{\otimes N}) = \langle 0^N|\rho|1^N\rangle + \langle 1^N|\rho|0^N\rangle.$$

Since the GHZ density matrix only contains two off-diagonal elements,

$$\rho = |\mathrm{GHZ}\rangle\langle\mathrm{GHZ}| = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \ddots & \iddots & 0 & 0 \\ \vdots & \vdots & \iddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix},$$

all remaining off-diagonal contributions vanish. The fidelity can therefore be written as

$$\mathcal{F} = \frac{1}{2}(P_{0^N} + P_{1^N}) + \frac{1}{2}\langle X^{\otimes N}\rangle, \tag{4.2}$$

where P_{0^N} and P_{1^N} are obtained from measurement statistics, while $\langle X^{\otimes N}\rangle$ captures the coherence between the GHZ components. The expectation value $\langle X^{\otimes N}\rangle$ measures how well the relative phase between the GHZ components is preserved. The GHZ state satisfies

$$X^{\otimes N}|GHZ\rangle = |GHZ\rangle = |GHZ^+\rangle\,,$$

with an eigenvalue of +1. While a phase flip (Z error) changes the state to

$$Z_i|GHZ^+\rangle = \frac{1}{\sqrt{2}}(|0^N\rangle - |1^N\rangle) = |GHZ^-\rangle\,,$$

which instead gives

$$X^{\otimes N}|GHZ^-\rangle = -\,|GHZ^-\rangle\,,$$

with an eigenvalue of -1. Thus each Z error flips the eigenvalue of $X^{\otimes N}$, meaning even numbers of phase errors preserve the phase while odd numbers invert it. In the simulations, the expectation value $\langle X^{\otimes N}\rangle$ is obtained by measuring all qubits in the X basis. This is implemented by applying Hadamard gates before measurement [23]. Using the Hadamard transformation [5]

$$H^{\otimes N}|0^N\rangle = \frac{1}{\sqrt{2^N}}\sum_x |x\rangle,$$

and using the general Hadamard transform identity [5]

$$H^{\otimes N} = \frac{1}{\sqrt{2^N}}\sum_{x,y}(-1)^{x\cdot y}\,|x\rangle\,\langle y|\,,$$

only $y = 1^N$ survives when applying to $|1^N\rangle$ $\quad\Rightarrow\quad$ $H^{\otimes N}|1^N\rangle = \frac{1}{\sqrt{2^N}}\sum_x(-1)^{x\cdot 1^N}|x\rangle,$

where the dot product becomes (x and y are bitstring vectors)

$$x\cdot 1^N = x_1 + x_2 + ... + x_N = \mathrm{wt}(x),$$

giving

$$H^{\otimes N}|1^N\rangle = \frac{1}{\sqrt{2^N}}\sum_x(-1)^{\mathrm{wt}(x)}|x\rangle,$$

where $\mathrm{wt}(x)$ denotes the Hamming weight, defined as the number of non-zero elements in a bitstring [24], in this case all 1s.

Combining these results gives

$$H^{\otimes N}|GHZ^+\rangle = \frac{1}{\sqrt{2^{N+1}}}\sum_x\left(1+(-1)^{\mathrm{wt}(x)}\right)|x\rangle.$$

This shows that only bitstrings with even Hamming weight remain, since

$$1+(-1)^{\mathrm{wt}(x)} = \begin{cases} 2, & \text{if wt}(x) \text{ is even} \\ 0, & \text{if wt}(x) \text{ is odd} \end{cases}$$

while a phase-flipped GHZ state instead gives

$$H^{\otimes N}|GHZ^{-}\rangle = \frac{1}{\sqrt{2^{N+1}}} \sum_{x} \left(1 - (-1)^{\mathrm{wt}(x)}\right) |x\rangle,$$

which only lets odd-weight bitstrings survive. The parity of the number of 1 outcomes therefore directly reflects whether an even or odd number of phase errors occurred. For this reason $\langle X^{\otimes N} \rangle$ is obtained by assigning $+1$ to bitstrings with even Hamming weight and -1 to those with odd Hamming weight, corresponding to the respective eigenvalues of $X^{\otimes N}$, and averaging over all shots. Since dephasing introduces random phase flips, this averaging reduces the expectation value toward zero as the dephasing probability increases, corresponding to a loss of coherence.

To give an intuitive illustration of this loss of coherence, considering a statistical mixture where half of the runs produce $|GHZ^{+}\rangle$ and half produce $|GHZ^{-}\rangle$,

$$\rho = \frac{1}{2}|GHZ^{+}\rangle\langle GHZ^{+}| + \frac{1}{2}|GHZ^{-}\rangle\langle GHZ^{-}|.$$

Expanding the two density matrices gives

$$|GHZ^{+}\rangle\langle GHZ^{+}| = \frac{1}{2}(|0^N\rangle\langle 0^N| + |1^N\rangle\langle 1^N| + |0^N\rangle\langle 1^N| + |1^N\rangle\langle 0^N|),$$
$$|GHZ^{-}\rangle\langle GHZ^{-}| = \frac{1}{2}(|0^N\rangle\langle 0^N| + |1^N\rangle\langle 1^N| - |0^N\rangle\langle 1^N| - |1^N\rangle\langle 0^N|).$$

Adding these together shows that the coherence terms cancel,

$$\rho = \frac{1}{2}(|0^N\rangle\langle 0^N| + |1^N\rangle\langle 1^N|),$$

leaving only the classical populations. This illustrates how random phase flips destroy coherence while leaving measurement probabilities unchanged. The expression for the fidelity given in Eq. 4.2 therefore shows how this decomposition allows the GHZ fidelity to be estimated efficiently without reconstructing the full density matrix.

5

Circuit choice

This chapter presents the simulation results for the three GHZ state preparation protocols. The protocols are compared by plotting the estimated GHZ fidelity as a function of the noise parameters introduced in Chapter 4. The simulations are performed for different numbers of data qubits, where data qubits refer to the qubits belonging to the final GHZ state and exclude any ancilla qubits used during the protocol.

The chapter is organized as follows. First, the relevant circuit metrics are summarized in order to establish the expected scaling of each protocol. The results are then separated into two types of noise regimes: errors associated with discrete circuit operations: CX gates and measurements, and errors associated with idle time: relaxation and pure dephasing. Across the parameter regimes studied here, the non-adaptive protocol gives the highest fidelities, although the difference in performance between the protocols depends strongly on the type of noise considered.

5.1 Circuit metrics

We summarize the metrics of the circuits in table. 5.1, to see predictable trends and to compare advantages and disadvantages between the protocols. By inspecting the circuits in Chapter 3, we can see how various metrics such as number of CX gates, CX layers and more are directly dependent on the number of data qubits N. Note that a measurement delay correspond here to the same amount of idle time as a CX-gate layer.

Table 5.1: Circuit properties, 10 measurement delays is equal to 10 CX delay layers.

Metric	Non-adaptive	Semi-adaptive	Fully adaptive
Number of qubits	N	$\frac{5}{4}N-1$	$2N-1$
Number of mid-measurements	0	$\frac{1}{4}N-1$	$N-1$
Number of CX gates	$N-1$	$\frac{5}{4}N-2$	$2N-2$
Number of CX layers	Depends on N*	4	2
Measurement delays	0	10	10

* For the routed non-adaptive protocol, the number of CX layers is obtained from the routing algorithm rather than from a fixed expression.

As seen from the scaling of the number of measurements and CX gates, the fidelity trends in the measurement-error and CX-error regimes can be understood directly from the number of error locations in each protocol. In these regimes, the non-adaptive protocol is expected to perform best, since it uses no intermediate measurements and requires fewer total operations than the adaptive protocols. The idle-time error regimes, corresponding to T_1 relaxation and T_ϕ dephasing, depend on a more subtle trade-off. The adaptive protocols use more qubits but have constant circuit depth, whereas the non-adaptive protocol uses fewer qubits but has a depth that increases with system size. To investigate this trade-off, idle-time errors were simulated for 32, 64, and 128 data qubits. The routed non-adaptive protocol required 7, 10, and 13 CX layers, respectively, compared with 5, 6, and 7 layers in the ideal all-to-all connectivity case. Thus, the necessary routing due to the heavy-hex connectivity shown in Fig. 4.1 increases the circuit depth, but the routed implementation still remains close to the ideal logarithmic-depth construction.

5.2 CX & Measurement errors

We first consider error sources that are associated with individual circuit operations rather than with the total time spent by the circuit. In this category, the relevant quantities are the number of CX gates and the number of intermediate measurements.

Fig. 5.1 shows the GHZ fidelity as a function of the CX error probability p_{CX} for the three protocols. The ordering of the protocols follows the expected CX gate counts. The non-adaptive protocol gives the highest fidelity, followed by the semi-adaptive protocol, while the fully adaptive protocol gives the lowest fidelity. This is because the non-adaptive protocol uses the fewest CX gates, whereas the fully adaptive protocol uses the largest number.

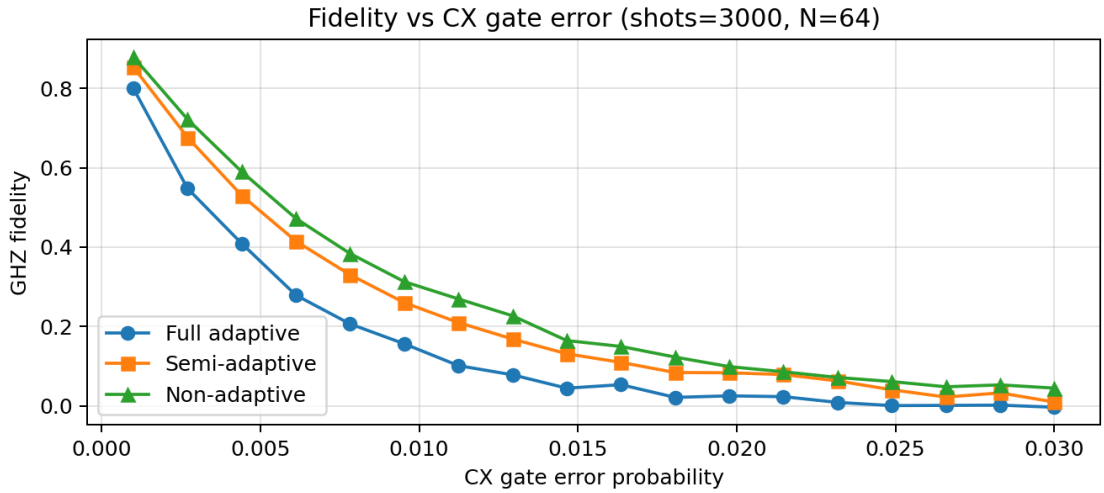


Figure 5.1: GHZ fidelity is plotted against CX error probabilities for GHZ protocols for 64 data qubits.

Fig. 5.2 shows the corresponding sweep over the measurement error probability p_m. In this case, only the adaptive and semi-adaptive protocols are shown, since the non-adaptive protocol does not use intermediate measurements and is therefore unaffected by this error source in the present model. The semi-adaptive protocol is less sensitive to measurement errors than the fully adaptive protocol. This follows directly from the smaller number of ancilla measurements required in the semi-adaptive construction. In the fully adaptive protocol, each additional fusion measurement introduces another opportunity for an incorrect classical outcome, which may lead to an incorrect Pauli-frame update and reduce the final GHZ fidelity.

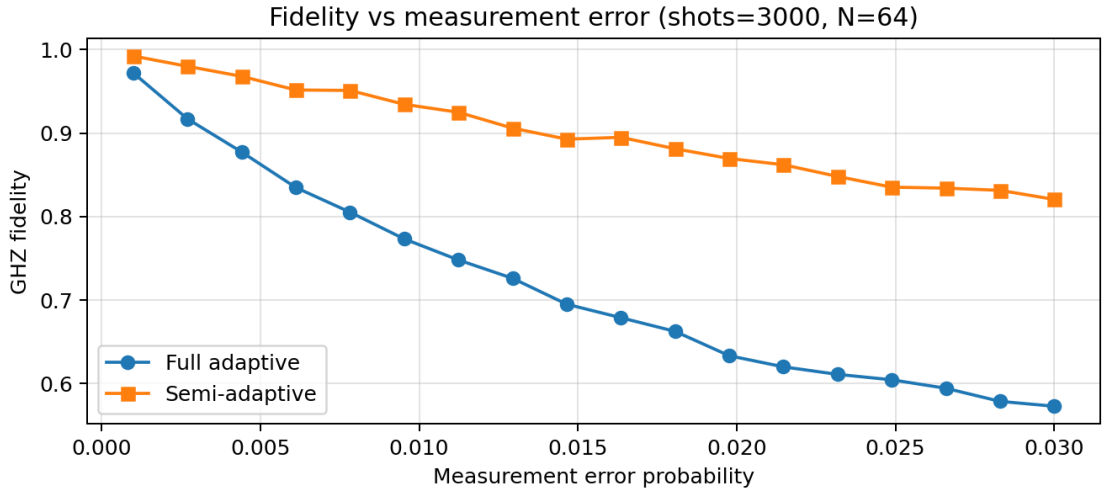


Figure 5.2: GHZ fidelity is plotted against measurement error probabilities for GHZ protocols for 64 data qubits.

These two error regimes therefore behave as expected from the circuit metrics alone. In the isolated CX-error regime, the protocol with the fewest CX gates performs best. In the isolated measurement-error regime, the protocol with fewer intermediate measurements performs best. The time error regimes can partly be predicted by the properties in Table. 5.1, but due to the scaling of the CX layers of the non-adaptive protocol, there is not an easy formula to describe for which number of data qubits N one protocol may be better than another.

5.3 Idle-time errors

The remaining two noise regimes are associated with errors that accumulate while qubits remain idle during the circuit. Unlike CX and measurement errors, these errors are not determined by the number of operations, but by the number of qubits, the circuit depth, and the additional delay introduced by measurement and feed-forward in the adaptive protocols. We consider relaxation (T_1) and pure dephasing (T_ϕ) separately.

5.3.1 Relaxation T_1

Relaxation errors are studied by varying the relaxation time T_1 while neglecting pure dephasing, corresponding to $T_\phi \to \infty$. Since this noise source acts during the delay steps inserted between circuit layers, the relevant comparison depends on both the number of qubits and the number of delay layers in each protocol. The resulting fidelities are shown in Fig. 5.3.

As Fig. 5.3 illustrates, the fidelity against the varying relaxation time T_1 scales differently between the number of data qubits depending on the GHZ protocol. For all three plots in the figure, the fidelity for the semi-adaptive and fully adaptive protocol are roughly the same, with the semi-adaptive showing slightly better fidelities. The non-adaptive protocol consistently achieves the highest fidelity over all numbers of data qubits. For $N = 32$, the gap in fidelity between the protocols reaches a maximum of $\Delta\mathcal{F} \approx 0.45$ which is a substantial performance gap. However, this gap gets reduced to approximately $\Delta\mathcal{F} \approx 0.08$ at $T_1 = 400\mu s$. The maximum gap in fidelity performance is also reduced for increasing number of data qubits. This reduced advantage of the non-adaptive protocol can be understood by comparing the total number of qubit delays in Table 5.2. The semi-adaptive and fully adaptive protocols contain more qubit delays than the non-adaptive protocol, especially for smaller system sizes. They are therefore more strongly affected when T_1 is short, but also show a larger improvement as T_1 is increased.

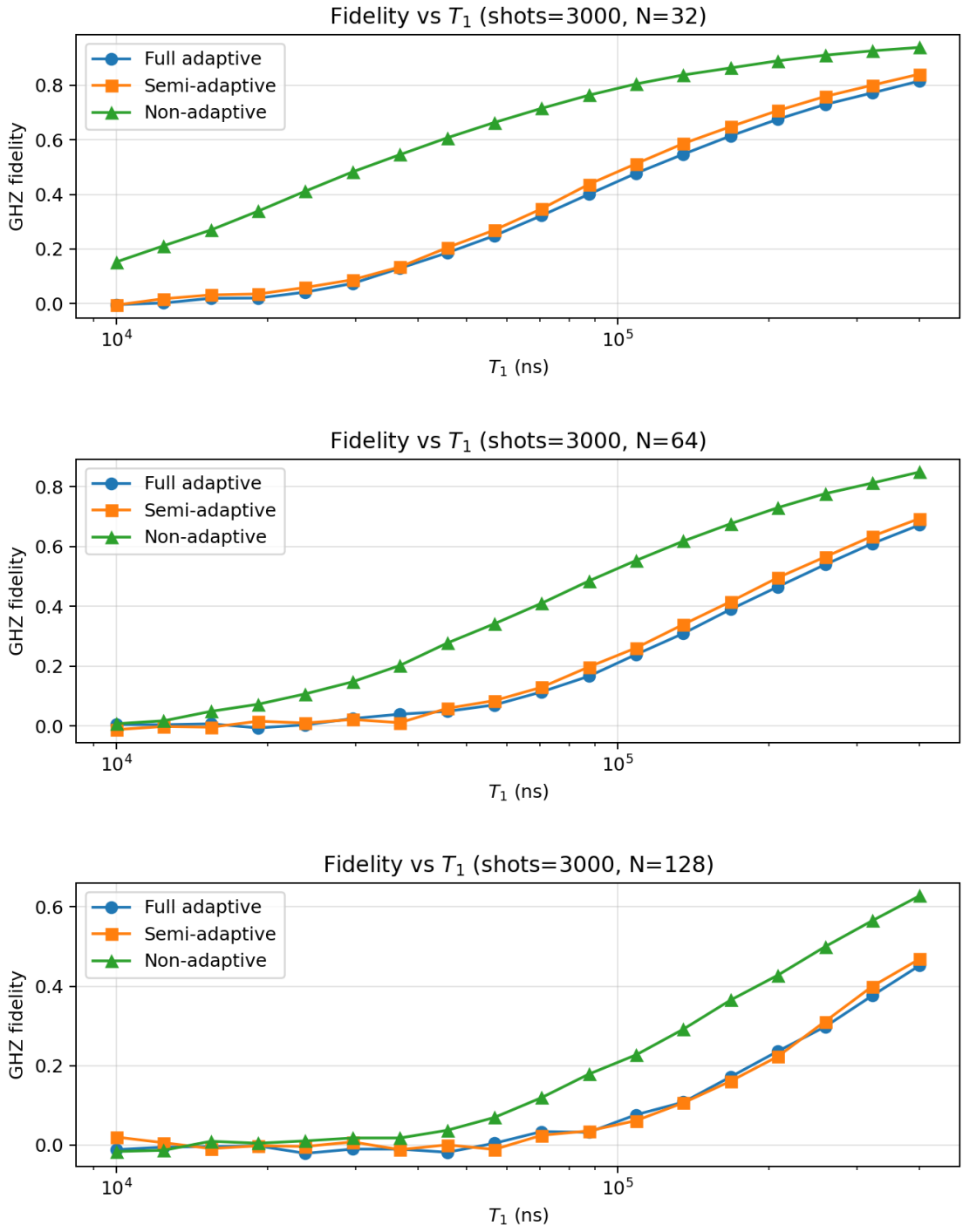


Figure 5.3: GHZ fidelity is plotted against Relaxation times T_1 (10-400μs) for GHZ protocols for 32, 64 and 128 data qubits.

Table. 5.2 shows how we did the calculations for the number of qubit delays of each protocol for each of the three number data qubits from our simulations. By simply taking the number of qubits times the number of delay layers.

Table 5.2: Comparison of total qubit delays in terms of CX gate durations for different N between the protocols

Metric	Non-adaptive	Semi-adaptive	Fully adaptive
Measurement delays	0	10	10
CX layers	7/10/13 for $N = 32/64/128$	4	2
Total delay layers	7/10/13 for $N = 32/64/128$	14	12
32 Data qubits			
Number of qubits	32	39	63
Number of qubit delays	224	546	756
64 Data qubits			
Number of qubits	64	79	127
Number of qubit delays	640	1106	1524
128 Data qubits			
Number of qubits	128	159	255
Number of qubit delays	1664	2226	3060

These calculations explain the expected ordering of the protocols in the idle-time error simulations. The non-adaptive protocol has the lowest total number of qubit delays and therefore achieves the highest fidelity. However, the relative advantage of the non-adaptive protocol decreases as the number of data qubits increases. This can be understood from how the total number of qubit delays scales with system size. For the non-adaptive protocol, the number of qubit delays increases from 224 at 32 data qubits to 1664 at 128 data qubits, corresponding to an increase by a factor of approximately 7.4. For the semi-adaptive protocol, the corresponding increase is from 546 to 2226 qubit delays, which is a factor of approximately 4.1. Thus, although the semi-adaptive protocol has more qubit delays overall due to measurement delays and additional ancilla qubits, its delay count grows more slowly over this range. This is because the non-adaptive protocol requires an increasing number of CX layers, and each additional layer exposes all qubits in the circuit to idle-time errors.

The idle-time noise is implemented using the Pauli-twirled approximation. In this model, each delay layer of duration $dt = 300$ns is associated with a probability of applying an X, Y, or Z error. As discussed in Subsection 4.3.1, these probabilities

are

$$p_X = p_Y = \frac{1 - e^{-dt/T_1}}{4},$$
$$p_Z = \frac{1 - e^{-dt/T_2}}{2} - \frac{1 - e^{-dt/T_1}}{4} = \frac{1 - e^{-dt/T_2}}{2} - p_X.$$

Moreover, as T_1 grows, the exponent $-dt/T_1$ decreases, which is why we see the improvement in fidelity diminishing in the figure for larger values of T_1.

At first sight, Tab. 5.2 suggests that the difference between the semi-adaptive and fully adaptive protocols should be larger than what is observed in Fig. 5.3. This missmatch is because the table counts every qubit during every delay layer, but not every error occurring at these locations has the same effect on the final GHZ fidelity.

One important example is given by Z errors in the Pauli-twirled noise model. A Z error on an ancilla qubit does not directly change the relative phase of the final GHZ state, since the ancillas are measured and are not part of the final target state. Similarly, a Z error on a data qubit that has not yet been entangled into the GHZ state has no effect if the qubit is still in $|0\rangle$, since $Z|0\rangle = |0\rangle$. Therefore, some of the delay locations included in Tab. 5.2 do not contribute to fidelity loss in the same way as errors in active data qubits.

This effect is particularly relevant for the fully adaptive protocol, which contains more ancilla qubits than the semi-adaptive protocol. The raw number of qubit-delay locations therefore overestimates the effective noise exposure of the fully adaptive protocol. In addition, the delay count in Tab. 5.2 includes the delay layers associated with classical feed-forward after the ancilla measurements. These layers do not affect the ancillas, since they have already been measured and are no longer used. This further reduces the difference between the adaptive protocols compared with what the simple qubit-delay count suggests.

For this reason, Tab. 5.2 should be interpreted as a useful qualitative measure of idle-time exposure rather than as a direct quantitative predictor of the fidelity. A complete accounting would require tracking which error locations actually affect the final GHZ state, which depends on the circuit state at each time step. Since the Pauli-twirled relaxation model is already an approximation to physical relaxation, we do not attempt to assign an effective corrected delay count here. Instead, the table is used to explain the overall trends in Fig. 5.3.

5.3.2 Pure dephasing T_ϕ

Pure dephasing is studied by varying T_ϕ, or equivalently T_2, since $T_\phi = T_2$ while neglecting relaxation errors ($T_1 \to \infty$). This analysis isolates the loss of phase coherence in the GHZ state. Since pure dephasing is represented by stochastic Z-errors in the simulation, the effect of this noise source depends not only on the total idle time, but also on whether a qubit is already part of the coherent GHZ superposition. The resulting fidelities are shown in Fig. 5.4.

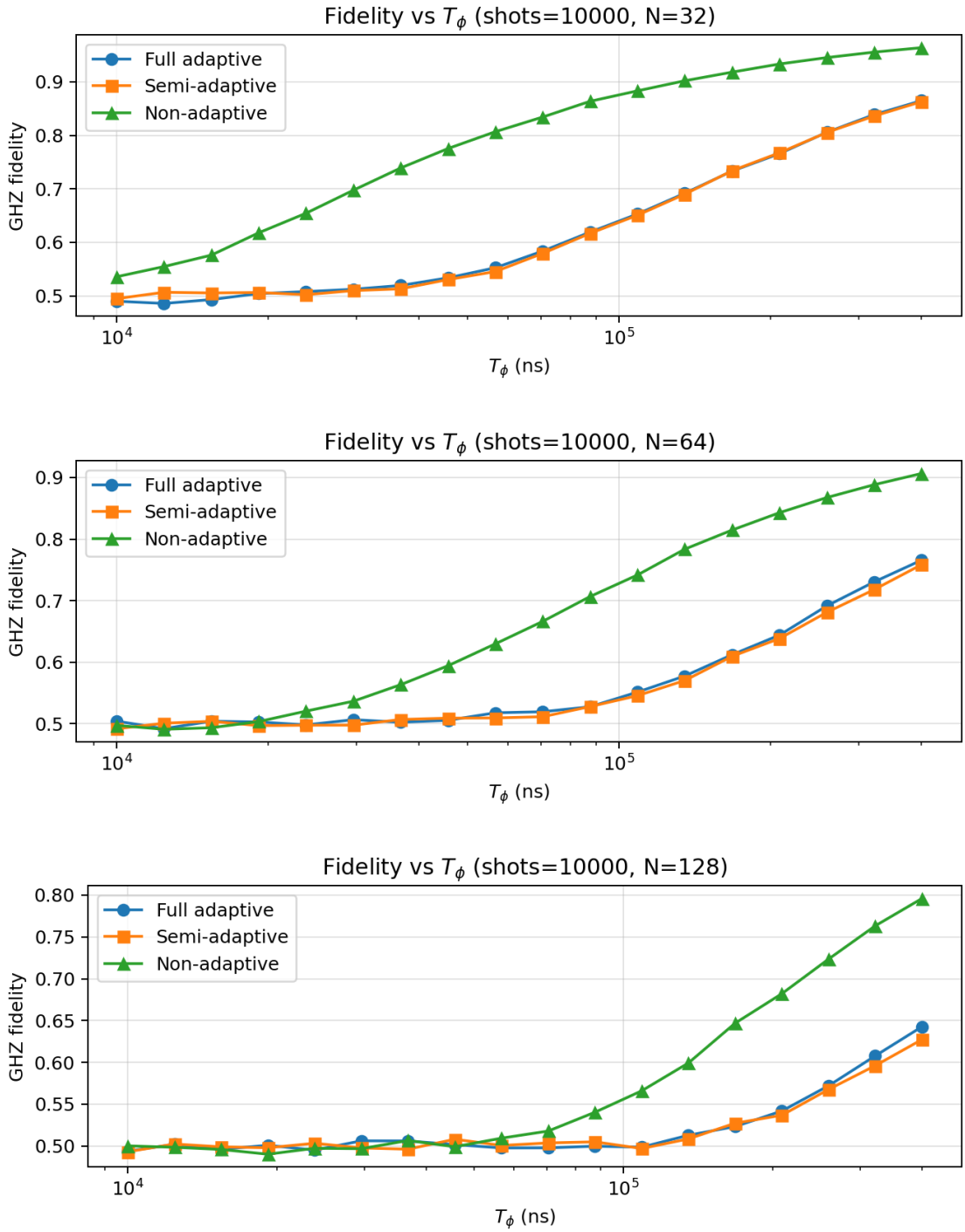


Figure 5.4: GHZ fidelity is plotted against Pure dephasing times T_ϕ (10-400μs) for GHZ protocols for 32, 64 and 128 data qubits.

Figure 5.4 shows the GHZ fidelity as a function of the pure dephasing time T_ϕ. The semi-adaptive and fully adaptive protocols give very similar fidelity curves, with the fully adaptive protocol performing slightly better over most of the parameter range. The non-adaptive protocol gives consistently higher fidelities for all system sizes considered.

This behaviour can be understood from how pure dephasing acts on inactive qubits. In the non-adaptive protocol, most data qubits remain in the state $|0\rangle$ during the early layers of the circuit and are only entangled into the GHZ state later. A Z error on such an inactive qubit has no effect, since $Z|0\rangle = |0\rangle$. The qubit only becomes sensitive to phase errors once it is part of the coherent GHZ superposition. In the ideal connectivity model, the number of active qubits doubles in each layer, meaning that the majority of the qubits are added only in the last few layers of the circuit. Although the heavy-hex connectivity used here increases the number of required CX layers, this delayed activation still greatly reduces the effective exposure of the non-adaptive protocol to pure dephasing.

For the adaptive protocols, the situation is different. The data qubits are prepared in superposition at the beginning of the circuit and are therefore exposed to dephasing during the entirety of the protocol. However, Z errors on ancilla qubits do not directly contribute to a phase error in the final GHZ state, since the ancillas are measured and are not part of the target state. As a result, the effective dephasing exposure of the adaptive protocols is comparable to the non-adaptive protocol with N data qubits. This explains why the semi-adaptive and fully adaptive curves are close to each other. Since the semi-adaptive protocol has two additional delay layers in this model, it is expected to perform slightly worse than the fully adaptive protocol in the pure dephasing regime.

The maximum fidelity gap between the non-adaptive and adaptive protocols is smaller than in the relaxation regime. For example, for $N = 32$ data qubits, the largest difference is approximately $\Delta\mathcal{F} \approx 0.28$ around $T_\phi = 30\,\mu$s. However, this difference is retained more clearly as the number of data qubits increases compared to the T_1 regime, where this gap decreased with increasing N. This trend is consistent with the fact that many qubits in the non-adaptive protocol remain effectively immune to Z errors until they are entangled into the GHZ state. This point will be discussed further in Chapter 6, where the simulation model is compared with other GHZ preparation protocols.

The fidelities in Fig. 5.4 approach a minimum value of approximately 0.5. This is expected because pure dephasing destroys the coherence between the two GHZ components but does not change the computational basis populations. In the limit of complete dephasing, the state approaches the classical mixture

$$\rho = \frac{1}{2}|0^N\rangle\langle 0^N| + \frac{1}{2}|1^N\rangle\langle 1^N|,$$

which still has overlap $1/2$ with the target GHZ state.

Some estimated fidelity values fluctuate slightly below 0.5 (or 0 in the T_1 regime). This is an effect of the finite sampling caused by the way the coherence term is estimated. As described in Section 4.5, the expectation value $\langle X^{\otimes N}\rangle$ is estimated

from the parity of the final measurement outcomes in the X basis. A shot is assigned the value $+1$ if the number of measured 1s is even, and -1 if the number of measured 1s is odd. For a GHZ state, this parity corresponds to whether the coherence has the sign expected from an even or odd number of effective Z errors. When the true coherence is close to zero, this estimator has a large relative variance, so the sampled value can fluctuate slightly above or below zero. This produces small fluctuations in the estimated fidelity, even when using 10000 shots.

For a Monte Carlo estimator obtained by averaging M independent samples, the standard error scales as $\sigma/\sqrt{M}$, where σ is the standard deviation of the samples [25]. As shown in Eq. 4.2, the fidelity was expressed as

$$\mathcal{F} = \frac{1}{2}(P_{0^N} + P_{1^N}) + \frac{1}{2}\langle X^{\otimes N}\rangle.$$

The variance and standard deviation of the observable $X^{\otimes N}$ are given by [5]

$$\mathrm{var}(X^{\otimes N}) = \langle (X^{\otimes N})^2\rangle - \langle X^{\otimes N}\rangle^2.$$
$$\text{Standard deviation: } \Delta(X^{\otimes N}) = \sqrt{\mathrm{var}(X^{\otimes N})}.$$

This makes the standard error scale as

$$\text{Error } \propto \sqrt{\frac{\mathrm{var}(X^{\otimes N})}{M}}$$

Since X is a Pauli operator, it has the property $(X^{\otimes N})^2 = I^{\otimes N}$ [5] and $\langle I^{\otimes N}\rangle = 1$ which gives

$$\mathrm{var}(X^{\otimes N}) = 1 - \langle X^{\otimes N}\rangle^2,$$
$$\text{and as } \langle X^{\otimes N}\rangle \to 0 \quad \mathrm{var}(X^{\otimes N}) \to 1.$$

This is the maximum variance for $\langle X^{\otimes N}\rangle$ as the coherence goes to zero, since the expectation value for $\langle X^{\otimes N}\rangle \in [-1, 1]$ and $\langle X^{\otimes N}\rangle^2 \in [0, 1]$.

In the relaxation (T_1) simulation, the noise includes errors that can affect the population of the GHZ state. This differs from the pure dephasing case, where the main sampling fluctuations come from estimating the coherence term $\langle X^{\otimes N}\rangle$ using the parity of X-basis measurement outcomes. For the population term, each shot instead contributes either 1 or 0 depending on whether the measured bit string belongs to the GHZ subspace. A single X or Y error typically produces a state with no overlap with the GHZ basis states $|0^N\rangle$ and $|1^N\rangle$. As a result, the corresponding computational-basis measurement outcome will not be 0^N or 1^N, and that shot contributes 0 to the population estimator. Conversely, if the outcome is 0^N or 1^N, the shot contributes 1. This defines a Bernoulli random variable O [26], where

$$O = \begin{cases} 1, & \text{if the measured outcome is } 0^N \text{ or } 1^N, \\ 0, & \text{otherwise.} \end{cases}$$

The average of this random variable over many shots estimates the population contribution

$$P_{0^N} + P_{1^N}.$$

If p denotes the probability that a shot gives either 0^N or 1^N, then the variance of this Bernoulli estimator is

$$\mathrm{Var}(O) = p(1-p),$$

where the random variable O takes the value 1 with probability p and 0 with probability $1-p$. Since $p \in [0,1]$, the maximum variance is $1/4$, which occurs at $p = 0.5$. This maximum is four times smaller than the maximum variance of the observable $X^{\otimes N}$, where each shot contributes a parity value of either $+1$ or -1. This helps explain why the fidelity curves are less smooth in simulations dominated by the coherence term, especially in the pure dephasing case. Although the T_1 model can also include Z errors, these make up a smaller part of the total error process than in the T_ϕ model. A similar argument applies to the measurement-error and CX-error regimes, since a single error can produce an outcome other than 0^N or 1^N and therefore cause a shot to contribute 0 to the population estimator. CX errors may also lead to somewhat larger sampling fluctuations than pure population errors, since they can include Z errors.

5.4 Summary

In summary, the non-adaptive protocol gives the highest GHZ fidelities in all isolated error regimes considered in this chapter. The size of this advantage, however, depends strongly on the type of noise. For errors associated with CX gates and measurements, the behaviour follows directly from the circuit metrics discussed above. In the CX error regime, the non-adaptive protocol performs best because it uses the fewest CX gates. In the measurement-error regime, the non-adaptive protocol is unaffected in the present model, since it contains no intermediate measurements. Among the adaptive protocols, the semi-adaptive protocol performs better than the fully adaptive protocol because it requires fewer ancilla measurements and CX gates.

For idle-time errors, the comparison is more subtle. Although the adaptive protocols have smaller quantum depth, this advantage is offset by the additional ancilla qubits and by the delay associated with measurement and classical feed-forward. For the parameter choices used in this work, these effects are large enough that the non-adaptive protocol still gives the highest fidelities in both the relaxation and pure dephasing regimes. The largest observed fidelity gaps are approximately $\Delta\mathcal{F} \approx 0.4$ in the T_1 sweep and $\Delta\mathcal{F} \approx 0.28$ in the T_ϕ sweep.

These results show that reducing circuit depth alone is not sufficient to guarantee an advantage for adaptive GHZ preparation for up to 128 data qubits. The performance depends on the balance between CX gate depth and idle time delays, number of measurements and number of ancillas. The next chapter discusses how these conclusions depend on the modeling assumptions and how they compare with previous hardware results.

6

Discussion and Conclusion

The results in Chapter 5 show that the non-adaptive GHZ preparation protocol gives the highest fidelities in the isolated noise regimes considered in this work. This chapter discusses how these results should be interpreted, with focus on how the modelling assumptions affect the comparison between adaptive and non-adaptive protocols.

6.1 Comparison with hardware results

A useful reference point for the simulation results is the hardware study in Ref. [27], where adaptive and non-adaptive GHZ preparation protocols were compared experimentally. Their results are qualitatively consistent with ours, in the sense that the non-adaptive protocol achieves higher fidelities for the system sizes considered. However, the fidelity gap between the protocols is smaller in their hardware results than in the idle-time simulations presented in Chapter 5.

This difference can mainly be explained by a few different modeling choices of the comparison. The routing of the non-adaptive circuit, the adaptive protocol used in the hardware experiment, and the hardware timing and error-mitigation assumptions all significantly impact the fidelity performance in various ways.

6.1.1 Non-adaptive Routing

As described in Appendix A.2, our routing procedure of the CX gates for the non-adaptive protocol was to minmize the CX layers used for constructing the GHZ state. The routing procedure was done within the limitations of the Heavy hex lattice structure described in subsection 4.2.1, and the work done in [27] was also done on the same qubit lattice structure. However, they use a simple 1D chain, where they start with the Hadamard gate in the middle of it, and expand their tree from both sides. Effectively limiting the number of CX gates per layer to 2. This produces a depth which scales with the number of qubits like $N/2-1$ excluding the

first Hadamard layer. This scales much worse compared to our routing procedure, which for example has 13 CX layers for 128 qubits, compared to theirs which would have 63 CX layers. Their plots are limited to 21 qubits, since they run on a small quantum device supporting only 27 qubits. For 21 qubits, they required 10 CX layers, and when we tested 21 qubits, we got 6 CX layers, a 40 % reduction. This routing difference for the non-adaptive protocol is one important reason why their protocols have a closer performance in fidelity, and is very impactful for larger qubit sizes.

6.1.2 Adaptive protocols

The adaptive protocol used in Ref. [27] differs from the adaptive protocols studied in this work. It can be viewed as a semi-adaptive variant of the fully adaptive GHZ preparation protocol. After the measurement and feed-forward step, the previously used ancilla qubits are reused as data qubits through an additional CX layer. This final layer can effectively double the size of the GHZ state without introducing additional connectivity constraints, since the ancillas are already neighbouring qubits to the data qubits from the fusion step.

As a result, the total number of physical qubits required to prepare an N-qubit GHZ state scales as N, rather than requiring additional ancilla qubits beyond the final GHZ size. Furthermore, during the measurement stage, roughly half of the qubits are still ancillas and are therefore not yet part of the final GHZ state. As discussed earlier, this reduces the effective exposure to dephasing idle-time errors. This makes the protocol more resistant to idle-time errors than the adaptive protocols simulated in this work. Figure 6.1 illustrates the construction for a 7-qubit GHZ state using their adaptive protocol.

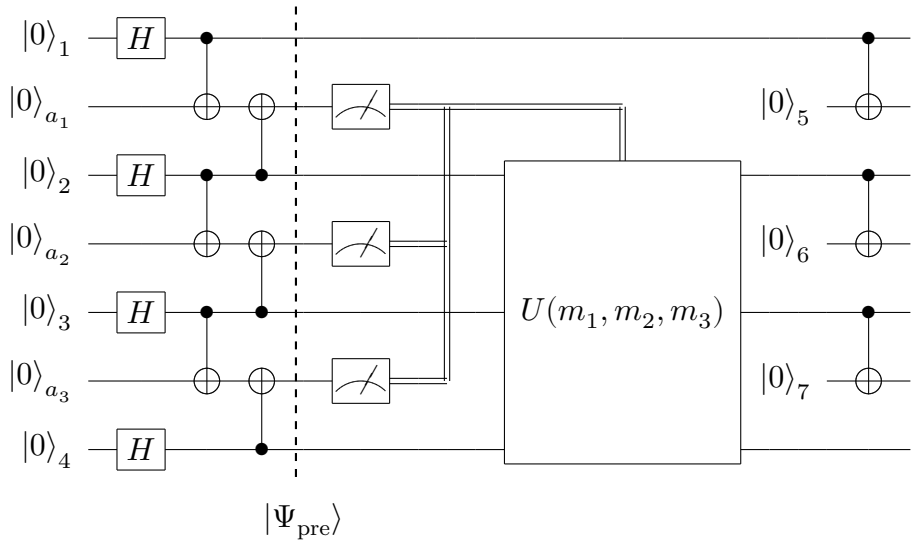


Figure 6.1: 7-qubit semi-adaptive circuit preparing a GHZ state with ancilla fusion from Ref. [27].

However, this protocol also has drawbacks compared with the semi-adaptive protocol used in this work. It requires more CX gates and more intermediate measurements, which means that it would be expected to perform worse in isolated CX-error and measurement-error regimes. However, compared with the fully adaptive protocol, it uses fewer CX gates, fewer measurements, and fewer qubits, and would therefore be expected to perform better under all the noise models considered here. Table 6.1 summarizes the main differences between the two semi-adaptive constructions. The table also shows the number of qubit delays that their protocol would have for $N = 32$ if simulated using the same timing assumptions as in this work.

Table 6.1: Comparison of total qubit delays for different N between the protocols

Metric	Our Semi-adaptive	Their Semi-adaptive
Number of qubits	$\frac{5N}{4} - 1$	N
Number of CX gates	$\frac{5N}{4} - 2$	$\frac{3N}{2} - \frac{3}{2}$
Number of measurements	$\frac{N}{4} - 1$	$\frac{N}{2} - 1$
Measurement delays	10	10*
CX layers	4	3
Total delay layers	14	13
32 Data qubits		
Number of qubits	39	32
Number of qubit delays	546	416

* They do not have 10 measurement delays, but their protocol would have in our simulations. Note that one measurement delay is taken to be equivalent to the duration of one CX layer.

The qubit-delay count in Table 6.1 still overestimates the effective idle-error exposure of the protocol from Ref. [27]. During the measurement delay, several of the qubits are ancillas and are not part of the final GHZ state. As we discussed earlier, the ancillas are not affected by Z-errors with respect to the final GHZ phase. Therefore, the protocol is even more favorable in the idle-error regimes, and especially in the pure dephasing regime than the raw qubit-delay count suggests.

6.1.3 Hardware timing and model assumptions

Another important difference between the two comparisons is the set of hardware timing parameters. These parameters affect the balance between adaptive and non-adaptive protocols, since adaptive circuits are penalized by measurement and feed-forward delays, while non-adaptive circuits are penalized by the number of CX layers. Table 6.2 summarizes the relevant timing parameters.

Table 6.2: Comparison of timing parameters used in this work and in Ref. [27].

Metric	Our model	Their hardware
Readout time	2.1 μs	0.9 μs
Feed-forward time	0.9 μs	0.7 μs
Total measurement delay	3 μs	1.6 μs
CX gate time	0.3 μs	0.6 μs
Measurement delay layers	3/0.3=10	$1.6/0.6 \approx 2.67$

The CX gate time in Ref. [27] is twice as long as the CX delay used in this work. This penalizes the non-adaptive protocol more strongly, especially for larger GHZ states, since its depth is determined by the number of CX layers. At the same time, their total measurement and feed-forward delay is shorter. Using their measurement delay with the CX gate time assumed in this work would correspond to approximately $1.6/0.3 \approx 5.33$ delay layers, rather than the 10 delay layers used in our simulations. This would significantly reduce the idle-time cost of the adaptive protocols.

A further limitation is the modelling of relaxation. In the physical relaxation process, it causes $|1\rangle \rightarrow |0\rangle$ and therefore does not affect qubits that remain idle in $|0\rangle$. In this work, relaxation was instead approximated using a Pauli-twirled channel, where stochastic Pauli errors are applied during delay steps. This approximation makes the simulations compatible with stabilizer methods, but it also means that inactive qubits in $|0\rangle$ can be assigned errors in the relaxation regime. This mainly works against the non-adaptive protocol, since many of its data qubits remain inactive during the early layers of the circuit. The protection of inactive $|0\rangle$ qubits is instead more clearly represented in the pure dephasing regime, where $Z|0\rangle = |0\rangle$.

Error mitigation also affects the comparison. No error mitigation was included in the simulations in this work, in order to keep the comparison between protocols as simple as possible. In Ref. [27], however, the authors include final readout-error mitigation and also compare results with and without dynamical decoupling. The readout-error mitigation applies to the final measurements rather than the mid-circuit measurements considered in our measurement-error regime. In our simulations, the final readout was assumed to be perfect, which should affect all protocols in the same way. Dynamical decoupling, on the other hand, can change the relative performance of the protocols by reducing idle-time decoherence and crosstalk effects [28]. In their results, dynamical decoupling improves the non-adaptive protocol enough that it consistently outperforms the adaptive protocol across the qubit-number sweep. While in their hardware run without dynamical decoupling, the non-adaptive protocol performs roughly the same as the adaptive one.

Overall, the smaller fidelity gap in Ref. [27] can be explained by several differences between the two setups. Their non-adaptive circuit uses a less optimized routing, which increases its CX depth and makes it more sensitive to idle-time errors.

At the same time, their adaptive protocol uses fewer qubits and is less exposed to idle-time errors than the adaptive protocols studied in this work. Their shorter measurement and feed-forward delay also reduces the cost of using adaptive operations. These effects make the adaptive protocol more competitive in their experiment than in our simulations. This helps explain why their hardware results show a smaller difference between the adaptive and non-adaptive protocols, even though the non-adaptive protocol still performs best in both comparisons.

6.2 Conclusion

In this thesis, we compared non-adaptive, semi-adaptive, and fully adaptive protocols for preparing GHZ states. The goal was to test whether the reduced depth of adaptive circuits can compensate for the extra measurements, feed-forward operations, and ancilla qubits that they require.

In the simulations performed here, the non-adaptive protocol gave the highest fidelities in all error regimes considered. For CX gate errors, this is because the non-adaptive protocol uses the fewest CX gates. For measurement errors, the non-adaptive protocol is unaffected, since it does not contain intermediate measurements. In these two regimes, the results follow directly from the number of gates and measurements in each protocol.

The results of the simulations in the idle-time error regimes are more subtle. The adaptive protocols have lower quantum depth, but they also require measurement and feed-forward steps. During these steps, the data qubits must remain idle and are therefore exposed to noise. With the timing assumptions used in this work, this idle time is large enough that the adaptive protocols lose their depth advantage. The main bottleneck therefore appears to be the measurement and feed-forward time, rather than the adaptive idea itself. This is also consistent with Ref. [27], where feed-forward time is identified as an important limitation for dynamic GHZ circuits.

This interpretation is further supported by earlier versions of the simulations. When the measurement delay was set equal to one CX-layer time, the adaptive protocols could outperform the non-adaptive protocol in the idle-time regime for large systems, for example at $N = 128$ data qubits. This suggests that adaptive GHZ preparation can become competitive if the measurement and feed-forward delay is sufficiently reduced.

Overall, the results show that adaptive GHZ preparation is not automatically better just because it has lower circuit depth. Its performance depends on whether the information gained from intermediate measurements is worth the extra time and noise introduced by measuring and feeding forward. For the parameters used in this work, the non-adaptive protocol is the most robust choice. However, the comparison could change on hardware with faster measurements, faster feed-forward, and using other versions of the adaptive protocol with fewer ancillas and better resistance to idling-errors.

6.3 Outlook

The original motivation for this project was broader than the comparison of fixed GHZ preparation protocols studied in this thesis. The initial idea was to use reinforcement learning to train an agent that could act as a Maxwell's demon for dynamic quantum circuits. Instead of deciding in advance whether the circuit should be non-adaptive, semi-adaptive, or fully adaptive, the agent would learn which choice works best. Given hardware parameters such as gate errors, measurement errors, relaxation times, and feed-forward delays, the agent would choose a circuit structure that maximizes the final GHZ fidelity.

This idea was inspired by Ref. [29], where a reinforcement learning agent is interpreted as a quantum Maxwell's demon. In that work, the agent learns when to measure, when to thermalize, and when to apply unitary feedback to an open quantum system. The goal is to find a good balance between cooling the system and paying the cost of acquiring measurement information. In this sense, the agent does not only act as an optimizer. It also plays the role of the demon, since it uses information from measurements to decide what to do next.

A similar idea could be used for dynamic quantum circuits. For GHZ state preparation, the agent would not choose between thermalization, measurement, and unitary feedback. Instead, it would choose how to build the circuit. For example, it could choose where to apply entangling gates, where to place intermediate measurements, and how the measurement outcomes should determine later corrections. The learned demon could then produce non-adaptive, fully adaptive, or semi-adaptive circuits depending on the hardware parameters.

This direction was not completed in the present work because the reward function was difficult to define in a useful way. A reward based only on the final GHZ fidelity would be expensive to evaluate, since the full noisy circuit has to be simulated first. It would also give the agent little information during the circuit construction itself. On the other hand, adding penalties for circuit depth, number of measurements, number of ancillas, or feed-forward time could easily bias the agent toward one type of circuit. The reward would therefore need to balance the final fidelity against the physical cost of using measurements and feed-forward, without building the desired answer into the reward by hand. For the parameter regime used in this thesis, this learning problem would probably not be very useful yet. The non-adaptive protocol gives the highest fidelities in all the simulated error regimes, so an agent would likely learn to choose the non-adaptive circuit for almost every input. In that case, reinforcement learning would not add much beyond the direct comparison of the fixed protocols.

However, this could change in a regime where the adaptive protocols are more competitive. As discussed above, shorter measurement and feed-forward times could make adaptive protocols advantageous for sufficiently large GHZ states. One possible future direction would therefore be to study more optimistic timing assumptions, either based on expected hardware improvements or on the best reported values from current devices. In such a regime, a Maxwell's demon learning model could become

more meaningful. The agent could learn that, for some hardware parameters, a non-adaptive circuit is optimal, while for others a semi-adaptive or fully adaptive circuit gives the highest fidelity.

The protocol comparison in this thesis can therefore be seen as a first step toward a learning-based approach. By studying the non-adaptive, semi-adaptive, and fully adaptive protocols separately, we identified which features matter most: CX count, measurement count, circuit depth, ancilla overhead, and measurement and feed-forward delay. These results could guide future reinforcement learning studies by showing which physical costs should be included when evaluating a candidate circuit. A natural continuation would be to train an agent in a parameter regime where both adaptive and non-adaptive circuits can perform well. The agent could take hardware parameters as input and output a circuit adapted to those parameters. This would address a more general version of the question studied in this thesis, namely when the use of measurements and information is useful for a certain quantum device. Although this learning-based approach was outside the final scope of the thesis, it is the direction most directly connected to the Maxwell's demon perspective in the title.

Bibliography

[1] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Applied Physics Reviews*, vol. 6, pp. 1, 10–18, 2019.

[2] N. M. P. Neumann, "Theoretical and experimental analysis of adaptive quantum computers," pp. 1, 6–10, 2025. DOI: 10.48550/arXiv.2509.06455. arXiv: 2509.06455 [quant-ph].

[3] W. Cambiucci, R. M. Silveira, and W. V. Ruggiero, "Hypergraphic representation for adaptive quantum circuits," pp. 1–2, 2025. DOI: 10.48550/arXiv.2504.09318. arXiv: 2504.09318 [quant-ph].

[4] K. Maruyama, F. Nori, and V. Vedral, "The physics of maxwell's demon and information," pp. 1–2, 2007. DOI: 10.48550/arXiv.0707.3400. arXiv: 0707.3400 [physics.hist-ph].

[5] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th Anniversary Edition. Cambridge University Press, 2010.

[6] IBM Quantum, *Readout error mitigation for the sampler primitive using M3*, IBM Quantum Documentation, Accessed: 2026-03-23. [Online]. Available: https://quantum.cloud.ibm.com/docs/tutorials/readout-error-mitigation-sampler.

[7] J. M. Koh, D. E. Koh, and J. Thompson, "Readout error mitigation for mid-circuit measurements and feedforward," p. 1, 2026. DOI: 10.1103/cj89-4h5t. arXiv: 2406.07611 [quant-ph].

[8] Z. Chen et al., "Efficient implementation of arbitrary two-qubit gates via unified control," pp. 1–2, 2025. DOI: 10.48550/arXiv.2502.03612. arXiv: 2502.03612 [quant-ph].

[9] Z. Li et al., "Error per single-qubit gate below 10^{-4} in a superconducting qubit," pp. 1–2, 2023. DOI: 10.48550/arXiv.2302.08690. arXiv: 2302.08690 [quant-ph].

[10] A. Gilchrist, N. K. Langford, and M. A. Nielsen, "Distance measures to compare real and ideal quantum processes," pp. 3–4, 2004. DOI: 10.48550/arXiv.quant-ph/0408063. arXiv: quant-ph/0408063 [quant-ph].

[11] W. McCutcheon et al., "Experimental verification of multipartite entanglement in quantum networks," *Nature Communications*, vol. 7, pp. 1–2, 2016. DOI: 10.1038/ncomms13251.

[12] L. DiCarlo et al., "Preparation and measurement of three-qubit entanglement in a superconducting circuit," *Nature*, vol. 467, pp. 574–577, 2010. DOI: 10.1038/nature09416.

[13] S. de Bone, R. Ouyang, K. Goodenough, and D. Elkouss, "Protocols for creating and distilling multipartite ghz states with bell pairs," *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–4, 2021. DOI: 10.1109/TQE.2020.3044179.

[14] O. T. Brown and J. Richings, *Quantum computing research: Energy efficiency of statevector simulation at scale*, EPCC, Accessed: 2026-04-01. [Online]. Available: https://www.epcc.ed.ac.uk/whats-happening/articles/quantum-computing-research-energy-efficiency-statevector-simulation-scale.

[15] J. Kattemölle and S. Hariharan, "Line-graph qubit routing: From kagome to heavy-hex and more," p. 1, 2023. DOI: 10.48550/arXiv.2306.05385. arXiv: 2306.05385 [quant-ph].

[16] P. D. Nation, H. Paik, A. W. Cross, and Z. Nazario, *The IBM quantum heavy hex lattice*, IBM Quantum Blog, Accessed: 2026-04-08, 2021. [Online]. Available: https://www.ibm.com/quantum/blog/heavy-hex-lattice.

[17] Qiskit Community, *IBM Quantum Challenge 2024: Bonus Lab Solutions*, GitHub repository notebook, Accessed: 2026-05-10, 2024. [Online]. Available: https://github.com/qiskit-community/ibm-quantum-challenge-2024/blob/main/solutions/lab_bonus/bonus-lab-solutions.ipynb.

[18] A. Paler, S. J. Devitt, K. Nemoto, and I. Polian, "Software pauli tracking for quantum computation," pp. 3–4, 2014. DOI: 10.48550/arXiv.1401.5872. arXiv: 1401.5872 [quant-ph].

[19] J. Ghosh, A. G. Fowler, and M. R. Geller, "Surface code with decoherence: An analysis of three superconducting architectures," pp. 3–4, 2012. DOI: 10.48550/arXiv.1210.5799. arXiv: 1210.5799 [quant-ph].

[20] N.-H. Kaneko, "Metrology for quantum hardware standardization – charting a pathway: A strategic review," p. 4, 2026. DOI: 10.48550/arXiv.2603.09098. arXiv: 2603.09098 [quant-ph].

[21] Z. Zeidan, "Recursive quantum approximate optimization algorithm with noise: Performance of noisy rqaoa for the frustrated ising model and maxcut and analysis of qaoa with broken z2 symmetry," Unpublished/internal manuscript, Master's thesis, Chalmers University of Technology, Gothenburg, Sweden, 2026, p. 29.

[22] Qiskit Development Team, *Restless measurements*, Qiskit Experiments Documentation, Accessed: 2026-04-23, 2024. [Online]. Available: https://qiskit-community.github.io/qiskit-experiments/stable/0.6/manuals/measurement/restless_measurements.html.

[23] IBM Quantum, *Specify observables in the pauli basis*, IBM Quantum Documentation, Accessed: 2026-04-09. [Online]. Available: https://quantum.cloud.ibm.com/docs/en/guides/specify-observables-pauli.

[24] S. Ling and C. Xing, *4.3: Hamming weight*, Excerpt from *Coding Theory: A First Course*, hosted by GlobalSpec, Accessed: 2026-04-09, 2004. [Online]. Available: https://www.globalspec.com/reference/56260/203279/4-3-hamming-weight.

[25] A. B. Owen, *Monte Carlo Theory, Methods and Examples*. Stanford University, 2013, pp. 104–105, Accessed: 2026-04-21. [Online]. Available: https://artowen.su.domains/mc/.

[26] W. Monroe, *Bernoulli and binomial random variables*, CS109 Lecture Notes #7, Stanford University, Based on a chapter by Chris Piech; accessed: 2026-04-22, 2017. [Online]. Available: https://web.stanford.edu/class/archive/cs/cs109/cs109.1178/lectureHandouts/070-bernoulli-binomial.pdf.

[27] E. Bäumer et al., "Efficient long-range entanglement using dynamic circuits," *PRX Quantum*, vol. 5, no. 4–18, p. 030 339, 2024. DOI: 10.1103/PRXQuantum.5.030339.

[28] V. Tripathi, H. Chen, M. Khezri, K.-W. Yip, E. M. Levenson-Falk, and D. A. Lidar, "Suppression of crosstalk in superconducting qubits using dynamical decoupling," p. 1, 2021. DOI: 10.48550/arXiv.2108.04530. arXiv: 2108.04530 [quant-ph].

[29] P. A. Erdman et al., *Artificially intelligent maxwell's demon for optimal control of open quantum systems*, 2024. DOI: 10.48550/arXiv.2408.15328. arXiv: 2408.15328 [quant-ph].

A

Appendix

This appendix contains two supplementary parts. Appendix A.1 gives the derivation of the pre-measurement state for the 4-qubit adaptive GHZ protocol. Appendix A.2 then describes the exact routing procedure used for the non-adaptive protocol implemented in Qiskit.

A.1 Derivation of the pre-measurement state for the 4-qubit adaptive GHZ protocol

We consider four data qubits, labeled q, and three ancillas, labeled a. The initial state is

$$|\Psi_0\rangle = |0000\rangle_q |000\rangle_a .$$

After applying Hadamard gates to the four data qubits, the state becomes

$$\begin{aligned}|\Psi_1\rangle = \frac{1}{4}(&|0000\rangle_q |000\rangle_a + |0001\rangle_q |000\rangle_a + |0010\rangle_q |000\rangle_a + |0011\rangle_q |000\rangle_a \\ &+ |0100\rangle_q |000\rangle_a + |0101\rangle_q |000\rangle_a + |0110\rangle_q |000\rangle_a + |0111\rangle_q |000\rangle_a \\ &+ |1000\rangle_q |000\rangle_a + |1001\rangle_q |000\rangle_a + |1010\rangle_q |000\rangle_a + |1011\rangle_q |000\rangle_a \\ &+ |1100\rangle_q |000\rangle_a + |1101\rangle_q |000\rangle_a + |1110\rangle_q |000\rangle_a + |1111\rangle_q |000\rangle_a).\end{aligned}$$

Next, the first layer of CX gates is applied in parallel, with data qubits $1, 2, 3$ acting as controls on ancillas a_1, a_2, a_3, respectively. Dropping the $_q$ and $_a$ notation from here, this gives

$$\begin{aligned}|\Psi_2\rangle = \frac{1}{4} \; &(|0000\rangle |000\rangle + |0001\rangle |000\rangle + |0010\rangle |001\rangle + |0011\rangle |001\rangle \\ &+ |0100\rangle |010\rangle + |0101\rangle |010\rangle + |0110\rangle |011\rangle + |0111\rangle |011\rangle \\ &+ |1000\rangle |100\rangle + |1001\rangle |100\rangle + |1010\rangle |101\rangle + |1011\rangle |101\rangle \\ &+ |1100\rangle |110\rangle + |1101\rangle |110\rangle + |1110\rangle |111\rangle + |1111\rangle |111\rangle).\end{aligned}$$

Then the second layer of CX gates is applied in parallel, with data qubits $2, 3, 4$ acting as controls on ancillas a_1, a_2, a_3, respectively. The resulting pre-measurement state is

$$\begin{aligned}|\Psi_{\text{pre}}\rangle = \frac{1}{4}\ & (|0000\rangle|000\rangle + |0001\rangle|001\rangle + |0010\rangle|011\rangle + |0011\rangle|010\rangle \\ & + |0100\rangle|110\rangle + |0101\rangle|111\rangle + |0110\rangle|101\rangle + |0111\rangle|100\rangle \\ & + |1000\rangle|100\rangle + |1001\rangle|101\rangle + |1010\rangle|111\rangle + |1011\rangle|110\rangle \\ & + |1100\rangle|010\rangle + |1101\rangle|011\rangle + |1110\rangle|001\rangle + |1111\rangle|000\rangle).\end{aligned}$$

It is now convenient to group the terms according to the ancilla state. This gives

$$\begin{aligned}|\Psi_{\text{pre}}\rangle = \frac{1}{4}\ & ((|0000\rangle + |1111\rangle)|000\rangle + (|0001\rangle + |1110\rangle)|001\rangle \\ & + (|0011\rangle + |1100\rangle)|010\rangle + (|0010\rangle + |1101\rangle)|011\rangle \\ & + (|0111\rangle + |1000\rangle)|100\rangle + (|0110\rangle + |1001\rangle)|101\rangle \\ & + (|0100\rangle + |1011\rangle)|110\rangle + (|0101\rangle + |1010\rangle)|111\rangle).\end{aligned}$$

After measuring the ancillas, the data qubits collapse onto one of eight branches:

$$\begin{aligned} 000 &\to \frac{1}{\sqrt{2}}(|0000\rangle + |1111\rangle), & 001 &\to \frac{1}{\sqrt{2}}(|0001\rangle + |1110\rangle), \\ 010 &\to \frac{1}{\sqrt{2}}(|0011\rangle + |1100\rangle), & 011 &\to \frac{1}{\sqrt{2}}(|0010\rangle + |1101\rangle), \\ 100 &\to \frac{1}{\sqrt{2}}(|0111\rangle + |1000\rangle), & 101 &\to \frac{1}{\sqrt{2}}(|0110\rangle + |1001\rangle), \\ 110 &\to \frac{1}{\sqrt{2}}(|0100\rangle + |1011\rangle), & 111 &\to \frac{1}{\sqrt{2}}(|0101\rangle + |1010\rangle). \end{aligned}$$

Each branch differs from the target GHZ state only by a known Pauli-X correction. Writing

$$|\text{GHZ}_4\rangle = \frac{1}{\sqrt{2}}(|0000\rangle + |1111\rangle),$$

the required corrections are

$$\begin{aligned} 000 &\to I, & 001 &\to X_4, \\ 010 &\to X_3X_4, & 011 &\to X_3, \\ 100 &\to X_2X_3X_4, & 101 &\to X_2X_3, \\ 110 &\to X_2, & 111 &\to X_2X_4. \end{aligned}$$

Thus, for every ancilla outcome, an appropriate conditional correction maps the post-measurement state to $|\text{GHZ}_4\rangle$.

A.2 Non-adaptive circuit routing procedure

The code structure used for the routed non-adaptive circuit was partly inspired by the IBM Quantum Challenge 2024 bonus lab. In particular, that implementation uses explicit lists of CNOT sources, CNOT targets, CNOT layers, and a fixed physical layout to define a low-depth entangling circuit [17]. However, the routing procedure used in this work was not taken directly from that code. The challenge implementation uses hard-coded lists for a fixed-size example, whereas our algorithm needed to create these lists depending on an arbitrary qubit size.

The non-adaptive protocol requires the preparation of an N-qubit GHZ state using only two-qubit gates between connected qubits. Since the qubits are not assumed to have all-to-all connectivity, the circuit has to be embedded into a graph that approximates the connectivity of a heavy-hex quantum processor. The purpose of the routing procedure is therefore to find a low-depth way of spreading the GHZ state through this graph, while respecting the constraint that CX gates can only be applied between neighboring qubits.

The construction begins by generating a square grid of candidate qubit positions. This grid is not used directly as the connectivity graph. Instead, some of the grid points are removed in a staggered pattern in order to create a heavy-hex-like structure. In this way, the resulting graph resembles the restricted local connectivity of IBM heavy-hex devices, where most qubits have only a small number of nearest-neighbor connections. The graph is generated larger than the final number of data qubits required for the GHZ state. This is important, since it gives the routing procedure freedom to choose a suitable connected patch of the lattice rather than being forced to use a predetermined set of the first N qubits.

After the heavy-hex-like graph has been generated, the algorithm searches for a good way to grow a GHZ state on this graph. The goal is to reduce the number of CX layers. A CX layer is a set of CX gates that can be applied in parallel, meaning that no qubit is used in more than one CX gate in the same layer. Reducing the number of such layers reduces the total circuit duration in the simplified timing model used in the simulations.

The GHZ state is constructed as a broadcast process. One qubit is chosen as the root qubit and is first placed in a superposition by applying a Hadamard gate. This root qubit is then considered active, meaning that it is already part of the growing GHZ state. In each subsequent CX layer, active qubits may entangle neighboring inactive qubits. When an inactive qubit is targeted by a CX from an active qubit, it becomes part of the GHZ state and can itself be used as a source qubit in later layers. In this way, the GHZ state spreads through the connectivity graph.

At each layer, the algorithm considers all possible CX gates from active qubits to inactive neighboring qubits. These candidate gates are scored using local properties of the graph. In particular, the algorithm chooses inactive target qubits that still have many unused neighbors. The reason is that such qubits are more useful after they have been added to the GHZ state, since they can help spread the state further in later layers. The algorithm then selects as many high-scoring, non-overlapping

CX gates as possible for the current layer. The non-overlap condition ensures that no qubit participates in two CX gates during the same layer.

This algorithm makes locally favorable choices at each step rather than performing an exhaustive search over all possible routing patterns. To reduce the dependence on a single local choice, the search is repeated for several candidate root qubits and with several slightly different scoring rules. Each attempted construction produces a candidate GHZ schedule, consisting of a list of CX sources, CX targets, and the CX layer in which each gate is applied. The schedule with the smallest number of CX layers is then chosen. If several schedules have the same number of layers, the algorithm prefers schedules where the gates are more strongly parallelized within the layers.

Once a schedule has been selected, the qubits that were actually reached during the construction define the physical patch used for the non-adaptive circuit. The remaining qubits of the larger heavy-hex-like graph are discarded. Equivalently, the procedure can be viewed as first constructing a possible GHZ broadcast circuit on a larger candidate lattice and then cutting away the unused qubits. In this implementation, the final Qiskit circuit is built only on the selected N logical qubits, while the corresponding physical layout records where these qubits were located in the original synthetic heavy-hex-like graph.

Figure A.1 illustrates this routing procedure for a 32-qubit example. The numbers shown on the qubits indicate the CX layer in which each qubit is added to the GHZ state. The root qubit is active from the start (layer 0), while the remaining qubits are added layer by layer through neighboring CX operations. In the illustrated example, the final group of qubits is added in the seventh CX layer, where nine new qubits enter the GHZ state. Qubits shown with reduced opacity are unused qubits from the original heavy-hex-like graph. These qubits are available during the routing search, but are not part of the final selected GHZ circuit. The figure is meant to illustrate the logic of the construction rather than being geometrically exact, since the full 32-qubit lattice, including unused qubits, has a different layout.

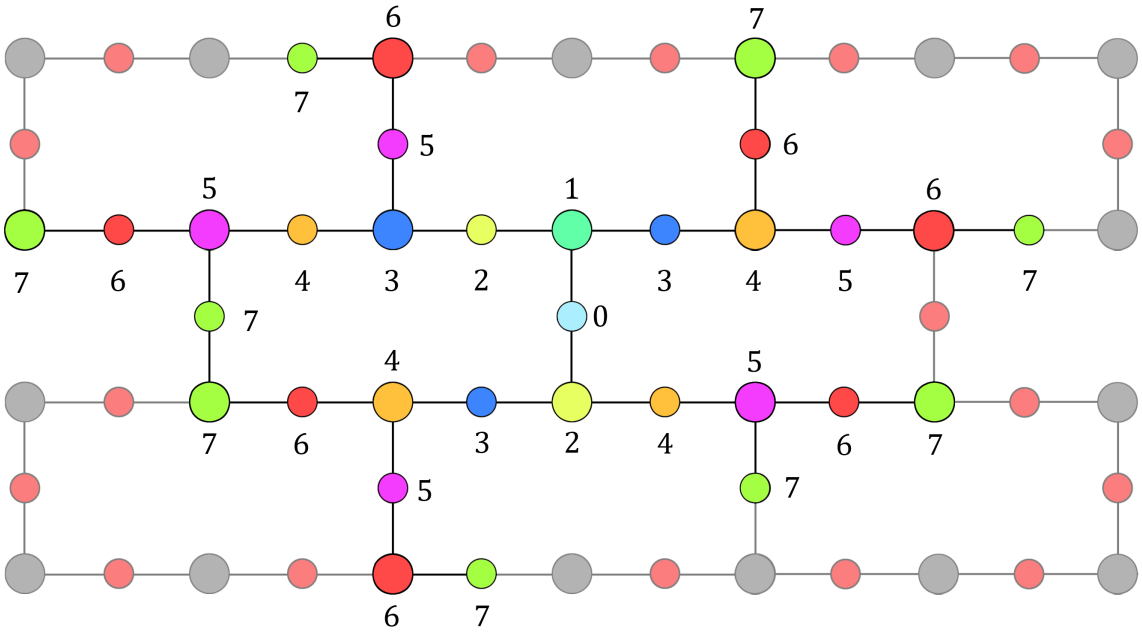


Figure A.1: Example of routing procedure of the non-adaptive GHZ protocol for 32 qubits. Qubits added to the state in the same layer share the same color, and unused qubits are faded out.

After the selected schedule has been obtained, the actual circuit is constructed by applying the CX gates layer by layer. A delay instruction is inserted after each entangling layer on all qubits. The circuit is then transpiled using the selected coupling map and physical layout, with additional routing disabled. Thus, Qiskit is not used to find a new SWAP-based route. Instead, the routing has already been determined by the algorithm, and the transpilation step is used only to express the already routed circuit in a form suitable for simulation.

This routing method should therefore be understood as a heuristic low-depth construction. It does not prove that the obtained schedule has the minimum possible depth on the chosen graph. However, by searching over a larger heavy-hex-like lattice, trying several root qubits, and selecting the best result among many greedy broadcast attempts, it provides a practical way to obtain compact non-adaptive GHZ circuits for the system sizes studied in this work.