ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

«ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ ΚΑΙ ΕΦΑΡΜΟΓΕΣ»

«MASTER OF SCIENCE IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS»

Theory and Design of AI-Optimized Quantum Error Correction Codes: Fundamental Limits and New Constructions

ΜΙΧΑΛΑΡΗ ΣΤΕΦΑΝΙΑ του ΔΗΜΗΤΕΡ

ΙΟΥΛΙΟΣ 2026

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ «ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ ΚΑΙ ΕΦΑΡΜΟΓΕΣ»

«Master of Science in Artificial Intelligence and Applications»

«Theory and Design of AI-Optimized Quantum Error Correction Codes: Fundamental Limits and New Constructions»

Διπλωματική Εργασία που υποβλήθηκε στο Τμήμα Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Θεσσαλίας ως μέρος των απαιτήσεων για την απόκτηση Διπλώματος Μεταπτυχιακών Σπουδών Τεχνητή Νοημοσύνη και Εφαρμογές από τον/την

ΜΙΧΑΛΑΡΗ ΣΤΕΦΑΝΙΑ του ΔΗΜΗΤΕΡ

Δήλωση Αυθεντικότητας, ζητήματα Copyright

Ο/Η Μεταπτυχιακός/ή Φοιτητής/τρια

Υπογραφή

ΙΟΥΛΙΟΣ 2026

Τριμελής Εξεταστική Επιτροπή

Η παρούσα Μ.Δ.Ε. εγκρίθηκε ομόφωνα από την τριμελή εξεταστική επιτροπή η οποία ορίστηκε από την Σ.Ε. / Συνέλευση του Τμήματος Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Θεσσαλίας, σύμφωνα με το νόμο και τον εγκεκριμένο Οδηγό Σπουδών του Π.Μ.Σ. «Τεχνητή Νοημοσύνη και Εφαρμογές». Τα μέλη της Επιτροπής ήταν:

1º Μέλος (Επιβλέπων):	ΚΟΛΟΜΒΑΤΣΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ
2º Μέλος:	
3º Μέλος:	

Η έγκριση της Μ.Δ.Ε. από το Τμήμα Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Θεσσαλίας, δεν υποδηλώνει αποδοχή των απόψεων του συγγραφέα.

Abstract

Quantum error correction is a central requirement for reliable large-scale quantum computation, since quantum information is highly vulnerable to decoherence, measurement disturbance, and physical noise. This thesis examines the theory and design of AI-optimized quantum error correction codes, with particular emphasis on fundamental limits, idealized decoding, and modern quantum low-density parity-check codes. The study approaches artificial intelligence not only as a practical tool for decoder implementation, but also as a theoretical lens through which the boundaries of quantum error correction can be examined more clearly.

The main argument of the thesis is that AI-based decoding can reduce algorithmic limitations, improve recovery strategies, and support the discovery of new code constructions, but it cannot overcome information-theoretic constraints such as quantum capacity, correctability conditions, and certain rate-distance trade-offs. To clarify this distinction, AI-optimized decoders are treated as idealized decoding oracles that allow the analysis to separate decoder inefficiency from deeper structural or fundamental barriers.

The thesis also discusses quantum LDPC codes as a promising route toward scalable quantum error correction. Their sparse structure, potential for constant rate, and modern construction techniques make them theoretically attractive, although their practical implementation remains constrained by locality, connectivity, and architectural requirements. Overall, the thesis concludes that AI does not replace quantum coding theory. Rather, it helps expose where present systems remain limited by algorithms, where new code structures are needed, and where quantum information theory itself draws the final boundary.

Table of Contents

List of Equations

List of Tables

Chapter 1. Introduction

1.1 General Background

Quantum computing is often presented as one of the most important technological directions of the coming decades. This is not only because quantum computers may become faster than classical machines in certain computational tasks, but also because they operate according to a very different logic. While classical computation is based on bits that take the values 0 or 1, quantum computation uses quantum bits, or qubits, which may exist in superposition states and may also become entangled with one another. In theory, this allows quantum systems to process information in ways that classical systems cannot easily reproduce.

However, this theoretical power comes with a serious weakness. Quantum states are extremely fragile. They interact with the environment, they lose coherence, and they are affected by many different types of noise. Even a very small disturbance may change the state of a qubit and cause an error in the computation. In classical systems, error correction is already important, but in quantum systems it becomes absolutely essential. Without quantum error correction, large-scale and reliable quantum computation would remain mostly theoretical.

Quantum error correction, usually abbreviated as QEC, is the field that studies how quantum information can be protected against noise. Its goal is not simply to detect that an error has occurred, but to preserve the logical information encoded in a quantum system even when some of its physical components are corrupted. This is a difficult problem because quantum information cannot be copied directly, due to the no-cloning principle, and measurement may disturb the state that one is trying to protect. Therefore, quantum error correction cannot be treated as a direct extension of classical error correction. It has its own rules, its own constraints, and, in a way, its own personality. It looks familiar at first, but then it resists the classical intuition.

Early theoretical work on quantum error correction showed that it is possible to encode one logical quantum state into a larger number of physical qubits so that the logical state can be recovered after certain errors occur. The theory of quantum error correction for general noise provides a formal basis for understanding which types of errors can be corrected and under what mathematical conditions this correction becomes possible [6].

Other approaches have connected quantum codes with geometry, algebra, and classical coding theory, showing that QEC is not only a practical engineering problem but also a deep theoretical structure [7], [8].

In recent years, artificial intelligence has entered this field in a visible way. Machine learning and reinforcement learning methods have been used to search for better codes, optimize decoding strategies, and improve the performance of quantum error-correcting systems [2], [10], [15]. Generative models and transformer-based methods have also been explored as decoders for quantum codes, suggesting that modern AI can learn complex error patterns that may be difficult to handle with traditional decoding algorithms [12], [17]. At first sight, this creates a strong impression that AI may be able to overcome some of the barriers of quantum error correction. But this impression needs to be examined carefully. Not every limitation is algorithmic. Some limits are fundamental.

The main idea of the thesis is to examine the theory and design of AI-optimized quantum error correction codes from a theoretical point of view. Instead of treating AI only as a practical tool for simulations or decoder implementation, the thesis considers AI as a conceptual instrument. More specifically, AI-based decoders can be modeled as idealized or nearly optimal decoding oracles. This allows one to ask a sharper question: if decoding were perfect, or at least extremely powerful, which limitations of quantum error correction would still remain?

This question matters because it separates two different kinds of obstacles. The first type consists of algorithmic limitations. These appear because real decoders are imperfect, computationally expensive, or unable to identify the most likely error pattern. AI may help with these limitations. The second type consists of fundamental limitations. These are imposed by quantum information theory, channel capacity, code structure, locality, distance bounds, and physical constraints. AI cannot simply remove these limits, no matter how powerful it becomes. It may help us approach them. It may expose them more clearly. But it cannot make them disappear.

1.2 Problem Statement

The central problem addressed in this thesis is the unclear boundary between decoding limitations and fundamental limitations in quantum error correction. In many cases, a quantum code may perform poorly not because the code itself is theoretically weak, but because the decoder is unable to identify the correct recovery operation. In other cases, even an optimal decoder cannot exceed a certain performance level because the noise channel itself imposes a capacity limit.

This distinction becomes more important as AI-based methods become more successful. Reinforcement learning, neural decoders, generative models, and hybrid quantum-assisted learning approaches have shown that AI can improve the construction and decoding of quantum error-correcting codes [10], [14], [15]. Yet, if one only observes improved performance, it is not always clear what has truly improved. Has AI discovered a better code? Has it improved the decoder? Has it reduced computational overhead? Or has it simply approximated a known theoretical optimum more efficiently?

The literature on AI for quantum error correction often emphasizes performance gains, but a deeper theoretical question remains. If AI is treated as an idealized optimizer, what does this reveal about the fundamental structure of QEC? A comprehensive review of artificial intelligence for quantum error correction shows that AI methods are increasingly relevant for decoding, code discovery, and control optimization, but it also indicates that theoretical interpretation is still needed [3]. This thesis therefore does not aim to produce a software system or experimental decoder. Instead, it aims to clarify the theoretical role of AI-optimized decoding.

A related problem concerns quantum low-density parity-check codes, or qLDPC codes. These codes are important because they promise scalable protection with sparse checks and potentially favorable asymptotic properties. The study of quantum codes from classical graphical models and graphical structures shows that there are meaningful connections between sparse classical models and quantum code construction [4]. Modern qLDPC research suggests that it may be possible to construct code families with constant rate and increasing distance, which is essential for scalable quantum memories and fault-tolerant computation. Still, one has to ask whether these codes can approach theoretical bounds under ideal decoding assumptions. The answer is not obvious. It sits somewhere between coding theory, quantum information, and computational complexity.

Thus, the problem is not simply how to design better quantum codes. The problem is how to understand what “better” means when AI optimization is introduced into the picture.

1.3 Aim and Objectives of the Thesis

The aim of this thesis is to study the theory and design of AI-optimized quantum error correction codes by examining fundamental limits, idealized decoding, and modern code constructions. The thesis is theoretical. It does not aim to implement a decoder or run numerical experiments. Instead, it analyzes the conceptual and mathematical relationship between AI-based optimization and the fundamental limits of quantum error correction.

The first objective is to study the basic information-theoretic limits that govern quantum error correction. These include the quantum capacity of noisy channels, the relationship between rate and distance, and the limits imposed by general noise models. Such limits define the outer boundary of what any code and any decoder can achieve [5], [6]. In this sense, they are not merely technical details. They are the walls of the room.

The second objective is to analyze whether AI-optimized decoding can overcome known limitations of quantum error-correcting codes. Here, the thesis makes an important distinction. AI may overcome limitations caused by inefficient decoding, poor heuristic choices, or limited search procedures. However, it cannot overcome limits that are truly information-theoretic. This point is important because discussions about AI sometimes become too optimistic. AI is powerful, but it is not magic. It cannot recover quantum information that has been irreversibly lost beyond the capacity of the channel.

The third objective is to investigate qLDPC codes and their structural properties. qLDPC codes are attractive because they use sparse parity checks, which may reduce overhead and support scalability. Their connection to graphical models, coherent parity-check constructions, and variational graphical approaches suggests that they form a rich class of codes for theoretical and practical study [4], [9], [19].

The fourth objective is to examine how close modern qLDPC constructions may come to theoretical capacity bounds under ideal decoding assumptions. This does not mean assuming that practical decoding is easy. Rather, it means temporarily removing decoder inefficiency from the analysis in order to see what remains. If a qLDPC code still fails to approach a bound under ideal decoding, the limitation may be structural or fundamental. If it performs well only under ideal decoding, the limitation may be algorithmic. This distinction is subtle, but it is useful.

The fifth objective is to clarify the role of AI in quantum coding theory. In this thesis, AI is not only understood as a practical method for training neural networks or reinforcement learning agents. It is also understood as a theoretical lens. AI-optimized decoding becomes a way of asking what would happen if the decoding problem were solved almost perfectly. In this form, AI becomes less like a tool and more like a thought experiment.

1.4 Research Questions

The thesis is guided by four main research questions.

The first research question asks whether fundamental bounds, such as quantum capacity, are independent of decoding strategies, including AI-based decoding. The expected answer is that true information-theoretic bounds are independent of the decoding algorithm. A better decoder can help a code approach the bound, but it cannot exceed the bound. This distinction is central to the thesis and is strongly connected to studies of fundamental limits in quantum error mitigation and correction [5], [6].

The second research question asks which known trade-offs in quantum error correction arise from physical constraints such as locality. This is particularly important because quantum systems are not abstract mathematical objects floating without cost. They have geometry. They require interactions. They are affected by locality constraints. Some limitations may therefore come not from decoding weakness, but from the physical architecture of the quantum system itself.

The third research question asks whether there exist families of qLDPC codes with constant rate and growing distance that approach the hashing bound under ideal decoding. This is a more technical question, but it is important for the long-term scalability of quantum error correction. Constant rate means that the number of logical qubits grows proportionally with the number of physical qubits. Growing distance means that the code becomes better at correcting errors as it increases in size. Achieving both properties is difficult, especially under practical constraints.

The fourth research question asks what theoretical significance AI-optimized decoders have in distinguishing fundamental limitations from algorithmic ones. This question connects the entire thesis. It does not assume that AI automatically solves QEC. Instead, it asks how AI changes the way researchers think about the boundary between possible and impossible.

1.5 Significance of the Study

The significance of this study lies in its attempt to clarify the theoretical meaning of AI optimization in quantum error correction. Many current studies focus on designing better decoders or discovering new codes through computational search. These are important directions. For example, reinforcement learning has been used to discover improved quantum error-correcting codes, and hybrid learning approaches have been proposed to support code optimization [10], [14], [15]. Generative decoding also suggests that modern AI architectures can model complex distributions of errors and recovery operations [12], [17].

However, the theoretical interpretation of these results remains equally important. If an AI decoder improves performance, one needs to understand whether the improvement reflects a better approximation to optimal decoding or a deeper change in the code design itself. Otherwise, the field may confuse practical progress with fundamental progress. They are related, but not the same.

This thesis is significant because it treats AI as a way to test the boundaries of QEC theory. By modeling AI decoders as idealized oracles, the thesis can separate limitations caused by decoding complexity from limitations imposed by quantum information theory. This is useful for future research because it helps identify where effort should be placed. If a limitation is algorithmic, then better AI decoders may help. If a limitation is structural, then new code constructions are needed. If a limitation is fundamental, then no decoder can remove it.

The study also contributes to the discussion of qLDPC codes. These codes are among the most promising families for scalable quantum error correction, but their performance depends on both construction and decoding. A theoretical analysis of their behavior under ideal decoding can clarify whether they are limited mainly by decoder complexity or by deeper structural constraints.

Finally, the thesis has broader relevance for fault-tolerant quantum computation. Reliable quantum computers will require not only good hardware, but also strong error correction. The design of quantum codes, the selection of decoding strategies, and the understanding of fundamental limits all contribute to this goal. In a sense, QEC is the quiet infrastructure of quantum computing. It does not always appear in popular descriptions, but without it the larger project cannot stand.

1.6 Scope and Limitations

The scope of this thesis is theoretical. It focuses on quantum error correction codes, fundamental limits, qLDPC constructions, and idealized AI decoding. It does not include software implementation, simulation results, experimental quantum hardware, or benchmarking of specific neural network architectures.

This limitation is intentional. A practical implementation would require many engineering decisions, such as the selection of datasets, noise models, training procedures, neural architectures, and evaluation metrics. These are valuable, but they would shift the thesis away from its main theoretical purpose. Here, the emphasis is on conceptual clarity.

The thesis also does not claim that AI can solve all difficulties in quantum error correction. On the contrary, one of its main arguments is that AI must be placed inside the theoretical boundaries of quantum information. AI-based decoders may be powerful, but they still operate on syndrome information and within the constraints of the code and channel. The decoder cannot create information that the physical process has destroyed.

At the same time, the thesis does not underestimate AI. It recognizes that AI can be useful for code discovery, decoding, optimization, and theoretical exploration [2], [3], [10], [12], [15], [17]. The point is not to reduce AI to a simple tool. The point is to understand exactly what kind of tool it is.

Chapter 2. Background on Quantum Error Correction and Quantum Information Theory

2.1. Quantum Information and the Nature of Noise

Quantum information theory studies how information is represented, processed, transmitted, and protected when the underlying physical system follows the laws of quantum mechanics. Its basic unit is the qubit, which differs from the classical bit in a fundamental way. A classical bit can take the value 0 or 1, while a qubit can exist in a superposition of the basis states $|0\rangle$ and $|1\rangle$. This property gives quantum computation much of its theoretical power, because quantum systems can represent and manipulate information in a richer state space than classical systems. At the same time, this same richness makes quantum information extremely fragile.

The fragility of quantum information is one of the main reasons why quantum error correction is necessary. In a real physical device, qubits are not isolated from the outside world. They interact with their surrounding environment, with control fields, with neighboring qubits, and sometimes with unwanted physical degrees of freedom that are difficult to model exactly. These interactions produce noise. A bit-flip error may change $|0\rangle$ into $|1\rangle$, while a phase-flip error may alter the relative phase of a quantum state. More generally, quantum noise is described through quantum channels, which represent physical transformations from input quantum states to output quantum states. Such channels may include depolarizing noise, dephasing noise, amplitude damping, leakage, correlated errors, and other forms of general noise that are not always reducible to a single simple model [6].

Noise in quantum computation should not be viewed as a rare disturbance. It is almost part of the system's natural condition. Quantum states tend to decohere because the environment continuously interacts with them. In a sense, the environment behaves like an unwanted observer. It does not "understand" the quantum state, of course, but it extracts information from it through physical interaction, and this process gradually destroys the coherence that quantum computation requires. The result is that the very features that make quantum computation powerful, such as superposition and entanglement, are also the features most vulnerable to disturbance.

For small quantum circuits, a limited number of errors may sometimes be tolerated. The computation may finish before the accumulated noise becomes too damaging. This is one reason why current noisy intermediate-scale quantum devices can still perform certain demonstrations. However, large-scale quantum computation requires much more. If a quantum algorithm involves many qubits and many gates, then even very small physical error rates may accumulate into a large probability of failure. Without a protection mechanism, quantum computation cannot scale reliably.

Quantum error correction provides this protection by encoding logical quantum information into a larger system of physical qubits. The idea is not to store the information directly in one fragile qubit, but to distribute it across several qubits in such a way that certain errors can be detected and corrected. This is similar in spirit to classical error correction, but the quantum case is more delicate. In classical computing, information can be copied and inspected directly. In quantum computing, the no-cloning principle prevents the creation of an arbitrary identical copy of an unknown quantum state, and direct measurement may destroy the information one is trying to protect.

For this reason, quantum error correction does not usually measure the logical quantum information itself. Instead, it measures indirect information about the error. These measurements are known as syndrome measurements. A syndrome reveals which stabilizer conditions have been violated, or more generally which error signature is present, without revealing the encoded logical state. This is the central trick of quantum error correction. The code must allow the system to learn enough about the error while keeping the logical quantum information hidden from measurement.

The mathematical theory of quantum error correction defines when a set of errors can be corrected. In general terms, the corrupted subspaces associated with different correctable errors must remain distinguishable in the appropriate mathematical sense. If two errors produce identical syndrome information but have different logical effects, then no decoder can always choose the correct recovery operation. If, however, two different physical errors have the same logical effect, then this ambiguity may be harmless. This is one of the areas where quantum error correction becomes more subtle than its classical counterpart [6].

Stabilizer codes are one of the most important frameworks for constructing and analyzing quantum error-correcting codes. In stabilizer codes, the codespace is defined as the common eigenspace of a set of commuting Pauli operators. These operators stabilize the valid encoded states, and errors are detected when they change the eigenvalues of some stabilizers. This algebraic formulation has become central in quantum coding theory because it provides a structured and relatively manageable way to describe many families of quantum codes. The connection between quantum codes, orthogonal geometry, and classical coding theory also shows that QEC is not only a physical engineering problem, but also a mathematical theory with deep algebraic roots [7]. It sits, rather interestingly, between physics, information theory, and discrete mathematics.

2.2. Quantum Code Parameters, Channels, and Fundamental Capacity Concepts

Quantum error-correcting codes are often described using parameters of the form [[n,k,d]]. In this notation, n is the number of physical qubits used by the code, k is the number of logical qubits encoded, and d is the distance of the code. These three parameters give a compact description of the code's basic efficiency and protection strength. They do not tell the whole story, but they provide a useful starting point.

The code rate is defined as k/n. A higher rate means that more logical information is stored per physical qubit. This is desirable because quantum hardware is difficult to build, maintain, and scale. Every additional physical qubit has a cost. It requires control, calibration, connectivity, and protection from noise. Therefore, a code that can encode many logical qubits using relatively fewer physical qubits is attractive from a resource perspective.

Distance, however, pushes the problem in another direction. The distance d of a quantum code is related to the minimum weight of an error that can cause a nontrivial logical failure. A larger distance generally means stronger protection, since more physical errors are required before the encoded logical information becomes unrecoverable. In simple terms, rate measures efficiency, while distance measures robustness. One wants both. The difficulty is that quantum coding theory does not usually allow both to be increased freely.

This creates the well-known rate-distance tension. A code with very high rate may not have enough redundancy to provide strong protection. A code with very large distance may require many physical qubits and thus have a lower rate. These trade-offs are not always due to poor engineering or weak decoding. Some are structural. Some arise from the algebraic constraints of quantum codes. Others are connected to locality, geometry, and the physical architecture on which the code must be implemented.

Overhead is closely related to this discussion. In practical quantum computing, overhead refers to the number of physical qubits required to protect a smaller number of logical qubits. If the overhead is too large, even a theoretically valid error correction scheme may become impractical. A code may look elegant on paper and still be very expensive in hardware. This is one of the reasons why modern research is interested not only in whether quantum error correction is possible, but also in whether it can be made efficient enough for large-scale quantum computation.

Quantum channels provide the broader information-theoretic setting for these questions. A quantum channel describes the noisy physical process through which quantum information is transmitted or stored. One of the most important quantities associated with a quantum channel is quantum capacity. Quantum capacity measures the maximum asymptotic rate at which quantum information can be transmitted reliably through a noisy channel. It is not merely a performance metric. It is a fundamental limit.

If the communication rate is below the quantum capacity of the channel, then reliable transmission may be possible with suitable coding and decoding in the asymptotic limit. If the rate is above capacity, reliable quantum communication is impossible, regardless of how clever the decoder is. This point is especially important for a thesis about AI-optimized quantum error correction. An AI decoder may improve practical performance. It may help a code approach a bound. It may uncover structure that a traditional decoder misses. But it cannot exceed the quantum capacity of the channel. The limitation is not computational; it is informational.

This is where the distinction between algorithmic limitations and fundamental limitations begins to appear. Suppose an AI-based decoder improves the logical error rate of a quantum code under a given noise model. This improvement may mean that the previous decoder was inefficient. It may also mean that the code had useful structure that was not being exploited. But if the code attempts to operate above a true capacity bound, then no decoding strategy can make it reliable in the asymptotic sense. In this case, the missing information is not hidden in a complex pattern. It has been lost through the noisy channel.

Related ideas appear in the theory of quantum error mitigation. Error mitigation differs from quantum error correction because it tries to reduce the effect of noise without fully encoding and correcting quantum information. Nevertheless, studies on the fundamental limits of quantum error mitigation show that noise cannot be removed arbitrarily without cost [5]. Although correction and mitigation are different techniques, they share an important lesson: quantum noise is constrained by physical and information-theoretic boundaries. These boundaries may be approached, but not simply ignored.

In QEC, capacity bounds, hashing bounds, no-go results, and rate-distance trade-offs help define the possible region of code performance. They do not always provide explicit constructions. Sometimes they only tell us what cannot be done. Yet this is useful knowledge. In fact, negative results often guide research by showing where effort should not be wasted. A good theory does not only open doors; it also marks the walls.

AI optimization becomes relevant in this context because it may reduce practical overhead or improve decoder efficiency. AI may help search for better code constructions, identify useful structures in large code spaces, and improve decoding for specific noise models [2], [10], [15]. However, its role must be interpreted carefully. AI can move a system closer to the best achievable performance, but if the limit is truly fundamental, AI does not move the limit itself.

The informal description of stabilizer codes given above can be stated more precisely, and doing so makes the later distinction between fundamental and algorithmic limitations sharper. Let the single-qubit Pauli operators be I, X, Y, Z, and let the n-qubit Pauli group be

EQUATION 1

$$P_n = \left\{ \alpha\, P_1 \otimes \cdots \otimes P_n \;:\; \alpha \in \{\pm 1, \pm i\}, \qquad P_j \in \{I, X, Y, Z\} \right\}.$$

A stabilizer code is defined by an abelian subgroup $\mathcal{S} \subseteq P_n$ that does not contain −I. The protected codespace is the simultaneous +1 eigenspace of every element of the group:

EQUATION 2

$$C(\mathcal{S}) = \{\, |\psi\rangle : S|\psi\rangle = |\psi\rangle \;\; for\ all\ \ S \in \mathcal{S} \,\}.$$

If $\mathcal{S}$ is generated by r independent commuting generators acting on n physical qubits, the code encodes

EQUATION 3

$$k = n - r$$

logical qubits, and one writes the code parameters as [[n, k, d]] with rate R = k/n. The distance d is the minimum weight of a non-trivial logical operator, that is, the minimum weight over the elements of the normalizer that lie outside the stabilizer,

EQUATION 4

$$d = min_{g \in N(\mathcal{S}) \setminus \mathcal{S}} \; wt(g).$$

Rate measures efficiency and distance measures protection; the tension between them, discussed informally earlier, is precisely the tension between making R large and making d large for fixed n.

2.3.1. Correctability

Whether a set of errors can be corrected is not a property of the decoder but of the code and the errors themselves. For a set of error operators {Ea} and the projector P onto the codespace C($\mathcal{S}$), the Knill–Laflamme condition states that the set is correctable if and only if

EQUATION 5

$$P\, E_a^{\dagger} E_b\, P = \alpha_{ab}\, P,$$

where (α_{ab}) is a Hermitian matrix of scalars. The condition has a direct interpretation. When the matrix is non-singular, distinct correctable errors map the codespace to mutually orthogonal subspaces and can be distinguished by syndrome measurement. When it is singular, distinct errors act identically on the codespace: this is degeneracy, and it is the formal reason a quantum decoder should target equivalence classes of errors rather than individual error patterns. No decoder, however powerful, can correct a set of errors that violates equation 5; the limitation is structural, written into the geometry of the code and the noise.

2.3.2 CSS codes and the qLDPC condition

A large and important family follows the Calderbank–Shor–Steane construction, which builds a quantum code from two classical binary codes with parity-check matrices HX and HZ. The requirement that the corresponding stabilizer generators commute reduces to the single algebraic condition

EQUATION 6

$$H_X H_Z^{\dagger} = 0 \ (mod\ 2).$$

This is the constraint that has no classical analogue and that makes quantum code design genuinely harder: an arbitrary sparse parity-check matrix is, in general, not a valid quantum code. A quantum LDPC code is then a stabilizer (typically CSS) code in which both H_X and H_Z are sparse every row has bounded weight and every column meets a bounded number of checks while still satisfying (Equation 6).

2.3 Decoding, Degeneracy, and the Role of Artificial Intelligence

A decoder is the component of a quantum error correction system that interprets the measured syndrome and chooses a recovery operation. This task is often difficult because the syndrome gives only partial information. It does not directly reveal the complete physical error, and it certainly does not reveal the logical state. The decoder must infer the most appropriate correction from incomplete and noisy evidence.

In many QEC settings, decoding becomes computationally demanding as the system size increases. The number of possible error configurations can grow very quickly with the number of physical qubits. The noise may be biased, correlated, or time-dependent. Measurement errors may corrupt the syndrome itself. Even if the code is well designed, the decoder may fail to identify the best recovery operation within the required time. This creates a practical bottleneck.

Traditional decoding methods include minimum-weight perfect matching, belief propagation, renormalization group methods, tensor-network-inspired approaches, and other specialized algorithms. These methods are important and often effective, but they are not universally optimal. A decoder that works well for one code family may perform poorly for another. A decoder designed for independent Pauli noise may not adapt well to correlated noise. A decoder that ignores quantum degeneracy may make decisions that look reasonable classically but are not optimal quantum mechanically.

Quantum degeneracy is one of the main reasons why quantum decoding differs from classical decoding. In a classical code, different error patterns usually correspond to different corrupted codewords. In a quantum stabilizer code, different physical errors may have the same effect on the logical state if they differ by a stabilizer. This means that the decoder should not necessarily search for the single most likely physical error. Instead, it may need to identify the most likely equivalence class of errors. That detail sounds technical, but it changes the logic of decoding. The most probable individual error is not always the best recovery target.

Artificial intelligence enters quantum error correction at precisely this point. AI-based decoding methods attempt to learn the relationship between syndromes and recovery operations from data, from simulations, or from interaction with the decoding environment. A comprehensive review of AI for quantum error correction shows that machine learning can support decoding, code design, noise characterization, and adaptive error correction [3]. This makes AI relevant not only as a decoder, but also as a broader optimization tool for the QEC pipeline.

Reinforcement learning is particularly useful when quantum error correction is treated as a decision-making problem. An agent may choose code operations, recovery actions, or construction steps, receive feedback, and gradually improve its policy. Research on the discovery of optimal quantum error-correcting codes through reinforcement learning shows that AI can explore code spaces in ways that may be difficult for human researchers to perform manually [10]. Similar work on optimizing QEC codes with reinforcement learning suggests that learning agents can identify useful structures in the space of possible quantum codes [15].

Generative AI methods offer another important direction. Instead of mapping each syndrome to a fixed correction, generative decoding attempts to model the probability distribution of errors and recovery operations [12]. Transformer-based approaches, such as qecGPT, suggest that generative pre-trained architectures can be adapted to quantum decoding tasks [17]. The analogy is not perfect, because quantum syndromes are not natural language. Still, both involve structured sequences and hidden dependencies. In that sense, treating decoding as a kind of structured generation is not as strange as it may first appear.

Variational approaches also contribute to the same movement. Variational quantum error correction treats code and recovery design as an optimization problem over parametrized structures [13]. Variational graphical quantum error correction codes connect this optimization perspective with graph-based representations of quantum codes [19]. These approaches show that modern QEC is increasingly shaped by learning, optimization, and adaptive design.

The advantage of AI decoders is that they may learn patterns that are hard to express manually. They may adapt to specific noise models, exploit degeneracy more effectively, or approximate optimal decoding where exact decoding is computationally infeasible. However, there is a limit to this advantage. The decoder only has access to the information contained in the syndrome and the assumed noise model. If the syndrome cannot distinguish between harmful logical alternatives, then no AI system can fully resolve the ambiguity. It may guess better. It may choose the statistically optimal recovery. But it cannot recover information that is not present.

This point is central for the thesis. AI should not be understood as a way to escape the theory of quantum error correction. Rather, it should be understood as a way to probe the boundary between practical decoding limitations and fundamental limits. If an AI decoder improves performance, the improvement may show that previous decoding methods were suboptimal. If performance remains bounded even under idealized AI decoding, the limitation is probably deeper. In this way, AI becomes not only an engineering tool, but also a theoretical lens.

2.4 Quantum LDPC Codes and Chapter Synthesis

Quantum low-density parity-check codes, or qLDPC codes, are among the most important code families for scalable quantum error correction. They are inspired by classical LDPC codes, which are known for their sparse parity-check structure and strong performance in classical communication. In the quantum setting, qLDPC codes are defined by stabilizer checks that have low weight and sparse connectivity. Ideally, each stabilizer should involve only a small number of qubits, and each qubit should participate in only a small number of checks.

The appeal of qLDPC codes is clear. Sparse checks may reduce the complexity of syndrome extraction and make the code more suitable for large systems. If a qLDPC family can achieve constant rate and growing distance, then it may protect quantum information more efficiently than code families with large overhead. This is why qLDPC codes are frequently discussed in the context of scalable fault-tolerant quantum computation.

However, the construction of qLDPC codes is not straightforward. In classical LDPC codes, parity-check matrices can be designed with considerable freedom. In quantum stabilizer codes, the checks must commute. This imposes additional algebraic constraints. A sparse matrix is not automatically a valid quantum code. The stabilizer structure must satisfy compatibility conditions so that the corresponding checks can be measured consistently without destroying the encoded quantum information [7].

Research has shown that classical graphical models can inspire quantum code constructions [4]. Graphs are useful because they provide a natural representation of sparse constraints. Qubits, checks, and their relationships can be studied through graphical structures, and this makes qLDPC codes especially suitable for methods such as message passing, belief propagation, graph-based optimization, and possibly graph neural networks. Coherent parity-check construction also provides a framework for thinking about how quantum parity checks can be organized and implemented in a way that respects the structure of quantum error correction [9].

The long-term goal of qLDPC research is to construct code families with strong asymptotic properties. Ideally, one wants constant rate, growing distance, sparse stabilizer checks, efficient syndrome extraction, and efficient decoding. Having all of these properties at once is difficult. Some codes may have good distance but poor rate. Others may be sparse but hard to decode. Others may be theoretically attractive but physically difficult to implement because their connectivity requirements do not match realistic hardware.

This is where AI may become useful. Machine learning can search over graph structures, parity-check matrices, code parameters, and decoder strategies. Reinforcement learning can treat code construction as a sequential optimization problem [10], [15]. Generative models can learn complex syndrome-error distributions [12], [17]. Variational methods can optimize code and recovery structures together [13], [19]. These methods may help researchers explore qLDPC constructions that would be difficult to design manually.

Nevertheless, the same caution remains. AI can help discover, decode, or optimize qLDPC codes, but it does not remove the mathematical constraints that define them. The stabilizers must still commute. The code must still have sufficient distance. The channel still has a capacity. The syndrome still contains only limited information. In this sense, qLDPC codes provide an ideal setting for the main question of the thesis: which limitations are caused by inefficient decoding, and which are fundamental?

This chapter has presented the theoretical background required for the later analysis. It began with the nature of quantum information and the problem of noise. Qubits are powerful because they can exist in superposition and become entangled, but these same features make them vulnerable to decoherence and environmental disturbance. Quantum error correction responds to this problem by encoding logical information into larger physical systems and using syndrome measurements to detect errors indirectly.

The chapter then discussed the main code parameters, especially rate, distance, and overhead. These quantities express the tension between efficiency and protection. A code should store logical information efficiently, but it must also protect that information strongly enough to survive noise. Quantum channels and quantum capacity were introduced as fundamental concepts that define the limits of reliable quantum information transmission and storage. These limits are essential because they show that not every performance barrier can be overcome by better decoding.

The discussion then turned to decoding and AI. Decoding is often a major bottleneck because the decoder must infer the best recovery operation from incomplete syndrome information. AI methods may improve this process by learning patterns, adapting to noise, exploiting degeneracy, and approximating optimal recovery strategies [3], [12], [17]. At the same time, AI remains bounded by the information available in the syndrome and by the fundamental limits of the channel.

Finally, qLDPC codes were introduced as a promising direction for scalable quantum error correction. Their sparse structure makes them attractive, but their construction and decoding remain challenging because of the algebraic constraints of quantum stabilizer codes. AI may assist in exploring this space, but the theoretical question remains open: can qLDPC code families approach capacity-related bounds under ideal decoding assumptions, or do structural constraints prevent this? This question leads directly to the next chapter, where fundamental limits and idealized AI decoding are examined more directly.

Chapter 3. Fundamental Limits and Idealized AI Decoding

3.1 Fundamental Limits in Quantum Error Correction

The previous chapter introduced the basic concepts of quantum error correction, quantum noise, decoding, quantum capacity, and qLDPC codes. This chapter moves closer to the central theoretical question of the thesis: what remains impossible when the decoder is assumed to be extremely powerful? In other words, if decoding were no longer the main weakness of a quantum error correction scheme, which limitations would still remain?

This question is important because artificial intelligence is often presented as a promising tool for improving quantum error correction. AI-based methods can be used to train decoders, optimize code constructions, search through large combinatorial spaces, adapt to noise models, and identify recovery strategies that are difficult to express through traditional algorithms [2], [3]. However, this does not mean that AI can overcome every limitation. Some barriers in quantum error correction are practical or algorithmic, but others are fundamental. The distinction is not always obvious. It is easy to say that a decoder failed. It is much harder to know whether it failed because the decoder was weak, because the code was poorly constructed, or because the noisy quantum channel had already destroyed too much information.

A fundamental limit is a boundary that cannot be crossed simply by changing the decoding algorithm. In quantum error correction, such limits may arise from quantum capacity, correctability conditions, rate-distance trade-offs, the no-cloning principle, measurement disturbance, locality constraints, and the physical structure of noise. These limits are not all of the same type. Some are information-theoretic. Others are architectural or physical. Yet they share a common feature: they remain relevant even when decoding is improved.

Quantum capacity is one of the clearest examples. It describes the maximum asymptotic rate at which quantum information can be reliably transmitted through a noisy quantum channel. If a code attempts to transmit quantum information above this capacity, reliable recovery is impossible in the asymptotic sense. This failure is not caused by an inefficient decoder. It occurs because the channel itself does not preserve enough quantum information for reliable reconstruction. A better decoder may help a code approach the channel capacity, but it cannot allow the code to exceed it.

This idea is especially important when AI enters the discussion. In many machine learning applications, better models appear to overcome previous performance barriers. Sometimes they do, because those barriers were only practical. But quantum information theory contains limits that are not merely practical. If the channel has lost information beyond recovery, an AI decoder cannot recreate it. It may choose the most likely recovery. It may approximate the optimal decoding rule. It may even outperform traditional decoders by a large margin. But if the required information is absent from the syndrome and from the remaining quantum state, then the decoder has no real path to success.

Correctability conditions provide another way of seeing the same issue. The theory of quantum error correction for general noise shows that the possibility of correcting a set of errors depends on the relationship between the code space and the error operators [6]. If different errors cannot be distinguished in a way that preserves the logical quantum information, then no recovery operation can correct them perfectly. The limitation is therefore built into the geometry of the code and the structure of the noise. It is not simply a matter of finding a smarter algorithm.

Rate-distance trade-offs also define important constraints. A quantum code with high rate and high distance would be highly desirable, because it would store many logical qubits while still providing strong protection. However, not all combinations of rate and distance are achievable under a given family of codes or under specific locality assumptions. A code may be mathematically valid in an abstract model but difficult, or even unrealistic, to implement if it requires highly nonlocal stabilizer checks. Here the limitation is not purely information-theoretic, but it is still fundamental in a physical or architectural sense.

Quantum error mitigation offers a related lesson. Although mitigation is different from full quantum error correction, work on the fundamental limits of quantum error mitigation shows that reducing the effects of noise has unavoidable costs [5]. This reinforces a broader principle: quantum noise cannot be removed freely. Whether one uses correction, mitigation, or adaptive learning, the process remains bounded by the structure of quantum mechanics and information theory.

Therefore, the study of fundamental limits should not be seen as pessimistic. It does not claim that progress is impossible. Rather, it clarifies what kind of progress is possible. If a limitation is algorithmic, then AI optimization may be very valuable. If a limitation is structural, then new code constructions may be needed. If a limitation is truly information-theoretic, then no decoder, classical or AI-based, can remove it. This distinction is one of the main ideas of the thesis.

The claim that an idealized decoder can approach but never exceed a capacity bound can now be made precise. A noisy quantum process is modeled as a quantum channel, a completely positive trace-preserving map with a Kraus representation

EQUATION 7

$$\mathcal{N}(\rho) = \sum_k K_k \, \rho \, K_k^\dagger, \quad \sum_k K_k^\dagger K_k = I.$$

The relevant figure of merit is the coherent information. Writing $S(\rho) = -\mathrm{Tr}(\rho \log \rho)$ for the von Neumann entropy and $\mathcal{N}^c$ for the complementary channel to the environment, the coherent information of a state ρ through $\mathcal{N}$ is

EQUATION 8

$$I_c(\rho, \mathcal{N}) = S(\mathcal{N}(\rho)) - S(\mathcal{N}^c(\rho)).$$

It quantifies how much quantum information survives passage through the channel: the entropy that ends up in the output minus the entropy that leaks to the environment. The quantum capacity is then given by the regularized maximum of this quantity (the LSD theorem),

EQUATION 9

$$Q(\mathcal{N}) = lim_{n\to\infty} \frac{1}{n} \, max_\rho \, I_c(\rho, \mathcal{N}^{\otimes n}).$$

The regularization of the limit over n channel uses ,is not a technicality that a cleverer decoder can remove; it reflects the fact that quantum capacity can be superadditive. Equation 9 is the formal content of the central claim of the thesis: $Q(\mathcal{N})$ depends only on the channel. A decoder, classical or AI-based, optimal or heuristic, appears nowhere in it. Transmission at a rate below $Q(\mathcal{N})$ is achievable in principle; transmission above it is impossible, and no decoding strategy changes that.

The hashing bound. For a Pauli channel that applies I, X, Y, Z with probabilities $\{p_I, p_X, p_Y, p_Z\}$, random stabilizer coding achieves the rate

EQUATION 10

$$R_{hash} = 1 - H(p_I, p_X, p_Y, p_Z),$$

where H is the Shannon entropy in qubits; for the depolarizing channel of strength p the probabilities are (1−p, p/3, p/3, p/3). The hashing bound is the benchmark against which sparse families are measured throughout this thesis. Under the idealized-decoding assumption, the gap between the rate of an explicit qLDPC family and R_{hash} at a given noise level becomes a property of the codes alone, with no decoder inefficiency left to blame which is exactly what makes the open question of Section 4.3 well posed.

3.2 Algorithmic Limitations and Idealized AI Decoding

One of the most important distinctions in this thesis is the distinction between algorithmic limitations and information-theoretic limitations. Algorithmic limitations arise when a problem is too difficult to solve efficiently with available methods. In quantum error correction, decoding is often algorithmically difficult because the decoder must infer the most appropriate recovery operation from incomplete syndrome information. The number of possible error configurations may grow rapidly with the number of physical qubits. Noise may be correlated. The code may be degenerate. The decoder may also need to operate within strict time constraints if it is used in a fault-tolerant quantum architecture.

These are serious difficulties, but they are not always fundamental. A weak decoder may fail even when a better decoder would succeed. A heuristic may ignore useful structure in the syndrome. A conventional algorithm may perform well for one noise model but poorly for another. In such cases, AI can be useful because it may approximate a more powerful decoding rule. Neural networks can learn mappings from syndromes to corrections. Reinforcement learning can search for code constructions or adaptive decoding policies [10], [15]. Generative models can learn probability distributions over error patterns and recovery operations [12]. Transformer-based decoders can treat the syndrome as structured input and the correction as a generated output [17].

Information-theoretic limitations are different. They arise when the available information is insufficient even for an optimal decoder. If two logically different error processes produce indistinguishable syndrome information, then no decoder can always choose correctly. If a channel destroys quantum information beyond its capacity, no decoding strategy can reconstruct it reliably. In this case, the missing information is not hidden inside a complicated pattern waiting for AI to discover it. It is simply unavailable.

This distinction can be expressed quite directly: AI may improve the use of available information, but it cannot create unavailable information. The sentence is simple, almost too simple, but it captures the central issue. A powerful AI decoder can extract correlations, approximate complex decision boundaries, and adapt to specific noise conditions. Yet it still depends on the syndrome, the code, and the channel. If these do not contain enough recoverable information, the AI decoder cannot fully solve the problem.

For this reason, this thesis models AI-optimized decoders as idealized decoding oracles. An oracle, in this context, is an abstract decoder that receives syndrome information and returns the recovery operation that minimizes the probability of logical failure under a specified noise model. This does not mean that real AI systems are perfect oracles. Real AI models require training data, architectures, optimization procedures, computational resources, and validation. They may overfit. They may fail under distribution shifts. They may also be difficult to interpret. The oracle model is therefore not a practical description of AI. It is a theoretical simplification.

The usefulness of this simplification is that it removes decoder inefficiency from the analysis. Once decoding is idealized, any remaining failure must be explained by the code, the channel, the noise model, or the physical assumptions. The analysis becomes cleaner, even if the idealization itself is somewhat artificial. Sometimes theoretical tools have to be artificial in order to reveal what is otherwise hidden.

For example, suppose a qLDPC code performs poorly under a conventional decoder. This poor performance may have two very different explanations. The code may be structurally weak, or the decoder may not be exploiting the code properly. If an idealized AI decoder would allow the code to perform much better, then the code has hidden potential. The main obstacle is algorithmic. If, however, even the idealized decoder cannot improve performance beyond a certain level, then the limitation is probably structural or fundamental.

This way of thinking is connected to the broader use of AI in quantum error correction. AI coding methods attempt to learn or construct better error correction codes [2]. Reinforcement learning methods explore large code spaces and may discover improved quantum codes [10], [15]. Generative decoding methods attempt to infer recovery operations by learning error distributions [12]. These methods can be interpreted as practical approximations to a broader ideal optimization process. The thesis abstracts from their implementation details and asks what their theoretical endpoint would mean.

There is, however, a danger in the oracle model. If it is used carelessly, it may hide practical constraints that are essential in real quantum devices. An ideal decoder may require exponential computational resources. It may assume exact knowledge of the noise model. It may ignore latency constraints, hardware connectivity, and noisy syndrome extraction. Therefore, the model should not be understood as a practical design proposal. It is better understood as a conceptual instrument. It helps separate decoder failure from code failure, but it does not replace the engineering problem of building a fast and reliable decoder.

3.3 AI-Optimized Decoding, Capacity, and qLDPC Codes

Quantum capacity provides a direct test for the theoretical limits of AI-optimized decoding. If AI decoding is treated as optimal, then it can help determine whether a given code family can approach the capacity of a noisy channel. However, it cannot allow communication above capacity. This point must remain central. A decoder, no matter how advanced, cannot make an impossible transmission rate possible.

In practical terms, AI can still be extremely useful near capacity. Many traditional decoders fail far below theoretical limits, especially when the code is complex, the noise is biased, or the syndrome structure contains correlations that are difficult to model manually. An AI decoder may close part of this gap by identifying better recovery operations, exploiting quantum degeneracy, or adapting to a more realistic noise model

[3]. In such cases, AI does not violate capacity. It helps the code move closer to the best achievable performance.

This is where the hashing bound becomes conceptually useful. The hashing bound is often used as a reference point in discussions of achievable rates for quantum codes under certain noise assumptions. If a family of qLDPC codes can approach such a bound under ideal decoding, then this suggests that the code structure is theoretically promising. If the same code family performs poorly only under practical decoders, then the limitation may be algorithmic. If it fails even under ideal decoding, then the limitation is more likely to be structural or fundamental.

Quantum low-density parity-check codes are especially relevant in this discussion because they are among the most promising candidates for scalable quantum error correction. qLDPC codes use sparse stabilizer checks, meaning that each check acts only on a limited number of qubits and each qubit participates in a limited number of checks. In principle, this sparsity can reduce the complexity of syndrome extraction and support scalable architectures. Their promise lies in the possibility of combining constant rate, increasing distance, and efficient syndrome measurement.

However, qLDPC codes are also constrained by the algebraic structure of quantum stabilizers. Unlike classical LDPC codes, quantum LDPC codes cannot be constructed by choosing arbitrary sparse parity-check matrices. The stabilizer checks must commute, and this imposes additional mathematical restrictions. This makes qLDPC construction a much more delicate problem than its classical analogue. Sparse structure is attractive, but sparsity alone does not guarantee a good quantum code.

Graphical approaches help clarify this structure. Quantum codes from classical graphical models show that graphical representations can support quantum code construction and analysis [4]. Coherent parity-check construction provides another framework for understanding how parity checks may be organized in quantum error correction [9]. Variational graphical QEC codes extend this direction by combining graph-based structures with optimization [19]. These approaches are important because they show that quantum code construction is not only algebraic but also structural and representational.

Under ideal AI decoding, qLDPC codes can be examined more clearly. If sparse quantum codes fail because traditional decoders are weak, then AI may reveal stronger performance. If they fail because sparse checks cannot support the required rate-distance behavior under certain assumptions, then the limitation is deeper. The distinction matters because it affects the direction of future research. A decoding limitation suggests that better AI or algorithmic methods may help. A structural limitation suggests that new constructions are needed.

This point is particularly important for fault-tolerant quantum computation. A code that is theoretically strong but impossible to decode efficiently may still be impractical. Conversely, a code that performs well under idealized AI decoding may become attractive if practical AI methods can approximate that ideal closely enough. The oracle model does not solve the implementation problem, but it helps decide whether implementation is worth pursuing.

AI-optimized decoding therefore becomes a diagnostic tool. It helps answer whether performance gaps are caused by suboptimal decoding, weak code construction, or fundamental constraints. This is especially relevant for modern qLDPC codes, where the asymptotic potential may be strong but decoding remains difficult. The code may be better than it looks under a weak decoder. Or it may not. The idealized decoder helps reveal which case is more plausible.

A similar logic appears in variational and hybrid learning-based approaches to quantum error correction. Variational quantum error correction treats code and recovery design as an optimization problem [13]. Hybrid quantum-assisted machine learning approaches attempt to improve error correction by combining learning methods with quantum structures [14]. These methods do not remove capacity limits, but they may help identify better points within the feasible region. The feasible region remains. AI searches inside it.

3.4 AI-Based Code Discovery and Generative Decoding

AI may contribute not only to decoding but also to the discovery of new quantum error-correcting codes. The search space of quantum codes is extremely large, and human-designed codes are necessarily guided by specific mathematical intuitions. AI methods may explore combinations that are difficult or unintuitive for researchers to examine manually. This does not mean that AI replaces theory. Actually, the opposite may happen. AI-generated codes may require even more theoretical interpretation.

Research on learning to construct error correction codes shows that AI can support the code design process rather than merely improve decoding [2]. Reinforcement learning has also been used to discover optimal or improved quantum error-correcting codes by treating code design as an optimization task [10]. Other work on reinforcement learning for QEC optimization suggests that learning agents can identify useful structures in complex code spaces [15]. These approaches show that AI can participate in the creative side of quantum coding theory, although its outputs still need to be understood mathematically.

There is a productive tension here. AI can discover patterns before humans fully understand them. Theory can then try to explain those patterns. Sometimes the explanation may be simple. Sometimes it may require new mathematics. This interaction between AI search and theoretical interpretation may become one of the most interesting directions in quantum error correction. It is not exactly a replacement of human reasoning. It is more like a strange collaboration, where the machine explores and the theory tries to make sense of what was found.

From the perspective of this thesis, AI-based code discovery is relevant because it may generate new constructions that approach known limits more closely. However, even new constructions must still be evaluated against fundamental bounds. A discovered code may be surprising, but it cannot violate quantum capacity or correct errors that are theoretically uncorrectable. If it appears to do so, then either the assumptions have been misunderstood or the bound being considered was not actually fundamental under those conditions.

Generative decoding represents another important development. Instead of using a fixed rule to map syndromes to corrections, generative models attempt to learn the probability distribution of errors and produce suitable recovery operations [12]. This is useful because quantum noise may be complex, correlated, biased, and difficult to describe manually. Generative models may capture relationships that are not easily expressed through traditional decoding algorithms.

Transformer-based approaches, such as qecGPT, extend this idea by applying generative pre-trained architectures to quantum error correction [17]. These approaches suggest that decoding can be treated as a structured prediction problem. The syndrome becomes an input sequence with hidden correlations, while the recovery operation becomes an output sequence. The comparison with language modeling is not perfect. A quantum syndrome is not a sentence. Still, both involve patterns, dependencies, and context. The analogy is imperfect but useful.

From a practical perspective, generative decoding may allow AI systems to learn recovery strategies that are difficult to design manually. It may also make decoding more adaptable to changing noise models. From a theoretical perspective, however, generative decoding remains bounded by the same limits as every other decoder. A transformer-based decoder may approximate the best possible recovery more effectively than a classical heuristic. But if the syndrome does not contain enough information to distinguish harmful logical alternatives, then the model cannot infer the correct logical recovery with certainty. It may generate a plausible recovery. It may generate the statistically optimal recovery. But probability is not the same as guarantee.

This is why generative decoding fits naturally into the idealized AI framework. It can be seen as a practical movement toward oracle-like decoding. The better the generative model becomes, the more clearly researchers can observe whether remaining failures come from the decoder, the code, or the channel. In this way, AI decoding methods are useful not only because they may improve performance, but also because they help reveal the origin of performance limitations.

3.5 Classification of QEC Limitations and Chapter Summary

The discussion in this chapter suggests that limitations in quantum error correction can be grouped into three broad categories. These categories are not always perfectly separate, but they are useful for analysis.

The first category includes fundamental information-theoretic limitations. These include quantum capacity, correctability conditions, and bounds related to the recoverability of quantum information. Such limits cannot be overcome by AI. They define what is possible in principle. A decoder may approach these limits, but it cannot pass them.

The second category includes structural and architectural limitations. These arise from code design, stabilizer structure, locality, check weight, geometry, and physical implementation constraints. AI may help discover better structures or optimize codes under given assumptions, but it cannot ignore the physical and mathematical requirements of a valid quantum code. For example, a code that requires highly nonlocal checks may be mathematically interesting but difficult to realize in a realistic quantum device.

The third category includes algorithmic limitations. These arise from decoding complexity, inefficient search, weak approximation, limited training data, poor adaptation to noise, or failure to exploit degeneracy. AI is most powerful in this category. It may improve decoding, optimize code search, learn noise-specific recovery strategies, and reduce practical performance gaps [3], [10], [12], [15], [17].

This classification prevents unrealistic expectations. AI is not equally effective against all kinds of limitations. It is strongest where the problem is computational, statistical, or pattern-based. It is weaker where the limitation comes from physics, information theory, or exact algebraic constraints. When the limitation is structural, AI may help indirectly by discovering better constructions, but even then it remains bound by the requirements of quantum coding theory.

The classification also helps organize the later parts of the thesis. qLDPC codes, for example, must be evaluated across all three categories. Their sparse stabilizer structure is a design feature. Their rate and distance are theoretical properties. Their physical implementation depends on architecture and locality. Their performance under noise depends strongly on decoding. AI interacts with each layer differently. It can search, optimize, and approximate, but it cannot erase the boundaries that define the problem.

The main argument of this chapter is therefore that AI-optimized decoders can overcome algorithmic limitations, but they cannot overcome information-theoretic limits. Quantum capacity, correctability conditions, and certain rate-distance trade-offs remain valid regardless of whether the decoder is classical, heuristic, neural, generative, or idealized.

This does not reduce the importance of AI. Rather, it places AI in the correct theoretical position.

The chapter also introduced the idea of modeling AI decoders as idealized oracles. This model is not meant to describe real AI systems exactly. Instead, it is a theoretical tool for separating decoder inefficiency from deeper limitations. If a code succeeds under ideal decoding but fails under practical decoding, then the limitation is likely algorithmic. If it fails even under ideal decoding, then the limitation is likely structural or fundamental.

Finally, the chapter discussed qLDPC codes, AI-based code discovery, reinforcement learning, variational methods, and generative decoding. These areas show that AI has genuine potential in quantum error correction, especially in decoding and construction search [2], [10], [12], [13], [15], [17], [19]. At the same time, they confirm the need for theoretical interpretation. AI can help researchers move closer to the boundary of what is possible, but it does not erase the boundary itself.

In this sense, AI-optimized quantum error correction should not be understood as a replacement for quantum coding theory. It should be understood as a new way of interrogating it. The theory remains present, sometimes strict, sometimes unexpectedly generous, but always active. AI may help reveal where the limits are. It may even help us approach them more efficiently. But when a limit is truly fundamental, the decoder must stop there.

The three-way classification used throughout this chapter is summarized in Table 1.

Table 1: Classification of limitations in quantum error correction and the scope of idealized AI-optimized decoding. AI is decisive only in the third row; in the second it can assist code or architecture search but cannot substitute for it; in the first it cannot help at all.

Category	**Origin / cause**	**Representative examples**	**Removable by idealized AI decoding?**
Fundamental (information-theoretic)	Quantum mechanics and the noisy channel	Quantum capacity $Q(N)$; Knill–Laflamme correctability; no-cloning; the hashing-bound rate ceiling at a given noise level	No ,independent of the decoder
Structural / architectural	Code construction, geometry, and locality	Vanishing surface-code rate $k/n \to 0$; rate–distance limits under 2D locality; the stabilizer commutation constraint; the nonlocal connectivity required by good qLDPC codes	No by decoding; needs new codes or architecture (AI may assist the search)

Algorithmic	Decoder efficiency	Suboptimality of MWPM / belief propagation; adaptation to correlated or biased noise; exploiting degeneracy; decoding latency	Yes ,AI’s primary domain

Chapter 4. Quantum LDPC Codes and Modern Constructions

4.1 From Classical LDPC Codes to Quantum LDPC Codes

The previous chapters introduced quantum low-density parity-check codes as a promising family for scalable quantum error correction, and they used the model of idealized AI decoding to separate algorithmic limitations from fundamental ones. This chapter turns to the codes themselves. It examines where qLDPC codes come from, why their construction is mathematically delicate, which structural barriers shaped their development for two decades, and how a series of modern constructions changed what is considered achievable. The discussion remains theoretical, but it now concerns concrete code families rather than abstract bounds. At the end of the chapter, these families are evaluated against the central question of the thesis: how close can sparse quantum codes come to fundamental limits when decoding is assumed to be ideal?

The starting point is classical. Low-density parity-check codes were introduced in classical coding theory as linear codes whose parity-check matrices contain very few nonzero entries. Each parity check involves only a small number of bits, and each bit participates in only a small number of checks. For several decades these codes received limited attention, mainly because the computational resources required to decode them effectively were not available. When they were rediscovered, the situation changed quickly. Classical LDPC codes, decoded with iterative message-passing algorithms, were shown to operate remarkably close to the Shannon capacity of classical channels, and today they appear in wireless standards, satellite links, and storage systems. The classical lesson is simple and rather encouraging. Sparse structure, combined with good decoding, can approach fundamental limits.

It is natural to ask whether the same lesson transfers to the quantum setting. A quantum LDPC code is, informally, a stabilizer code whose stabilizer generators have low weight and whose qubits participate in a bounded number of checks. The stabilizer formalism provides the algebraic language for this definition: the codespace is the common eigenspace of a set of commuting Pauli operators, and errors are detected through the syndrome pattern they leave on these operators [23]. Within this framework, sparsity acquires a direct physical meaning. Low-weight checks involve few qubits, which means simpler syndrome extraction circuits, fewer opportunities for errors to spread during measurement, and a structure that is, at least in principle, friendlier to large devices [21].

The transfer from the classical to the quantum world is, however, far from automatic. In a classical LDPC code, the designer can choose almost any sparse parity-check matrix. In a quantum stabilizer code, the checks must commute with one another, otherwise they cannot be measured consistently and the code is simply not valid. This requirement imposes algebraic constraints that classical coding theory does not face. The CSS construction expresses these constraints in a particularly clean form: a quantum code can be built from a pair of classical codes, provided the two codes satisfy a containment relation with each other. The connection between quantum codes and classical coding structures, including the correspondence with orthogonal geometry, was established already in the early years of the field [7], [22]. The good news from this line of work is that classical coding theory can be reused. The less good news is that it cannot be reused freely. The containment requirement couples the two classical codes together, and sparse classical codes that satisfy it tend to produce quantum codes with disappointing distance.

This coupling has a second, more practical consequence. The commutation constraint typically forces short cycles into the graphical representation of the quantum code, and short cycles are exactly the structures that damage belief propagation and other message-passing decoders. Quantum LDPC codes therefore inherited a double difficulty. It was hard to construct sparse codes with good parameters, and even when such codes existed, the iterative decoders that made classical LDPC codes successful did not perform nearly as well on them. Degeneracy complicates the picture further, since the decoder should target equivalence classes of errors rather than individual error patterns, a point discussed in Chapter 2. For a long time, this combination of construction problems and decoding problems kept qLDPC codes in a somewhat awkward position: theoretically attractive, practically frustrating.

One interesting way around part of the algebraic constraint is entanglement assistance. If the communicating parties share entanglement in advance, the commutation requirement can be relaxed, and classical codes that would not normally yield a valid quantum code can be imported into the quantum setting. Constructions of good entanglement-assisted quantum error-correcting codes show that this route can produce codes with strong parameters from classical ingredients [20]. Entanglement assistance does not solve the general construction problem, since pre-shared entanglement is itself a costly and fragile resource. It demonstrates, though, something conceptually useful: several of the difficulties of quantum code design come from the commutation structure, not from quantum information as such.

4.2 Locality, Overhead, and the Lessons of the Surface Code

Any honest discussion of qLDPC codes has to pass through the surface code. The surface code is a topological stabilizer code defined on a two-dimensional lattice of qubits, with stabilizer checks acting only on small neighborhoods of the lattice. In a strict sense it is itself a quantum LDPC code, since its checks have constant weight and each qubit participates in a constant number of checks. It is also the code that dominates current experimental efforts, and for understandable reasons. Its checks are geometrically local, which matches the planar connectivity of many hardware platforms. It has a high error threshold under standard noise models. Its decoding is well studied. The review of quantum error correction for quantum memories presents it as the central example of a practically motivated quantum memory architecture [21].

The weakness of the surface code lies in its rate. A standard surface code patch encodes one logical qubit, no matter how large the patch becomes. The distance grows with the linear size of the lattice, which is good, but the number of physical qubits grows with the area, which means the rate k/n tends to zero as the protection becomes stronger. Protecting many logical qubits therefore requires many patches, and the overhead becomes very large. Resource estimates for useful fault-tolerant computation routinely involve hundreds or thousands of physical qubits per logical qubit. In the current era of noisy intermediate-scale quantum devices, where machines offer a limited number of noisy qubits and no full error correction, this overhead is one of the main obstacles separating present hardware from large-scale fault tolerance [26].

This weakness is not an accident of the surface code's design. It reflects a deeper structural fact about local codes. When stabilizer checks are required to be geometrically local in two dimensions, the achievable combinations of rate and distance are severely constrained, and codes of this type cannot enjoy both constant rate and rapidly growing distance at the same time; the surface code essentially sits at the boundary of what two-dimensional locality permits [21]. In the classification proposed in Chapter 3, this is a structural and architectural limitation. It is not information-theoretic in the strict sense, because it disappears once the locality assumption is removed. But it also cannot be overcome by any decoder, classical or AI-based. No decoding intelligence can change the geometry of a code.

This point deserves emphasis because it illustrates the methodology of the thesis in a very concrete case. Suppose one observes that surface codes require enormous overhead and asks whether AI can reduce it. AI can certainly improve surface code decoding, and learned decoders have already shown competitive performance in this setting [3]. The rate of the code, however, is fixed by its definition. An idealized decoding oracle would change nothing about the overhead scaling. The limitation is therefore not algorithmic, and effort invested in decoders, however sophisticated, addresses the wrong layer of the problem. If lower overhead is the goal, the code itself must change. This is precisely the motivation behind modern qLDPC research: to keep the sparse, measurement-friendly structure that makes the surface code attractive, while abandoning strict geometric locality in order to escape its rate limitations.

4.3 Good Quantum LDPC Codes: Constant Rate, Growing Distance, and Efficient Decoding

In coding theory, a family of codes is called asymptotically good if both the rate and the relative distance remain bounded below by positive constants as the block length grows. For classical codes, the existence of good families has been known for a very long time, and good LDPC families in particular can be obtained through random or expander-based constructions. For quantum LDPC codes, the corresponding question remained open for roughly twenty years. It was not known whether sparse quantum codes with constant rate and linearly growing distance exist at all. The question was not decorative. A positive answer would mean that the overhead of quantum error correction can, in principle, remain constant as systems scale, which is exactly the property that surface-code architectures lack.

For most of this period, the best known constructions fell visibly short of the target. Product constructions, which combine two classical codes into a quantum code, became the main engine of progress. The hypergraph product gave qLDPC families with constant rate, but their distance grew only with the square root of the block length. This was already an important step, since it separated qLDPC codes from the surface code regime, yet the gap to linear distance remained. What followed was a rapid sequence of structural refinements, in which the product operation was twisted, balanced, and lifted in increasingly sophisticated ways, with expansion properties of the underlying graphs playing a central role. (The analogy with classical expander codes is helpful here, even if it cannot be pushed too far.) This line of work culminated in constructions that finally achieve constant rate and linear distance simultaneously, and subsequent work strengthened the result in an equally important direction by presenting good quantum LDPC codes equipped with decoders that run in linear time [24].

The theoretical significance of this result is difficult to overstate, and it matters for this thesis in two distinct ways. The first concerns the classification of limitations developed in Chapter 3. Before these constructions, one could reasonably suspect that the poor rate-distance behavior of sparse quantum codes was a fundamental feature of quantum mechanics, some unavoidable price imposed by the commutation constraints of stabilizer codes. The existence of good qLDPC codes shows that this suspicion was wrong. The barrier was a limitation of the known constructions, not of quantum coding theory itself. In the vocabulary of this thesis, what looked fundamental turned out to be structural, and the structural obstacle was eventually removed by better mathematics. This is a useful warning. The boundary between fundamental and structural limitations is not always where it appears to be, and it can move when the theory improves.

The second contribution concerns decoding, and it connects directly to the idealized oracle model. Throughout this thesis, AI-optimized decoding has been modeled as an oracle in order to remove decoder inefficiency from the analysis. One might object that this idealization is too generous, since real decoding is computationally hard in general. The linear-time decoders of [24] weaken this objection considerably, at least for specific code families. They show that for good qLDPC codes the gap between idealized decoding and efficient decoding can be closed in principle: there exist explicit algorithms that correct a constant fraction of adversarial errors in time proportional to the block length. An ideal decoder is no longer a pure fantasy for these codes. It is an asymptotic description of something that provably exists. It should be admitted, of course, that the asymptotic language hides many practical details, including constant factors, noise models, and measurement errors. Still, the conceptual point stands.

How do these codes relate to capacity-type bounds? The hashing bound provides the natural reference point. It originates in the study of mixed-state entanglement and quantum error correction, where random stabilizer-type protocols were shown to achieve a positive rate determined by the entropy of the noise [25]. For a Pauli channel, the hashing bound gives an achievable rate that decreases as the noise entropy increases, and it is often used as the benchmark against which practical code families are measured. Random codes achieve it asymptotically, but random codes are not sparse and are not equipped with efficient decoders. The question relevant to this thesis is therefore whether structured, sparse families can operate near the hashing bound when decoding is ideal.

The honest answer is that the situation is partly resolved and partly open. The existence of good qLDPC codes guarantees constant rate and linear distance, which concerns the regime of adversarial errors and asymptotic protection. Operating close to the hashing bound at a given noise level is a related but distinct property, since it concerns the trade-off between rate and the actual error probability of the channel rather than worst-case distance. Under the idealized decoding assumption of this thesis, the gap between the performance of explicit qLDPC families and the hashing bound becomes a property of the codes alone, with no decoder inefficiency to blame. Whether this gap can be made arbitrarily small by sparse codes at all noise levels does not appear to be settled, and the thesis does not pretend to settle it. What can be said is more modest. Nothing in quantum information theory forbids sparse codes from approaching the bound, the historical pattern suggests that construction barriers in this area tend to fall, and the burden of impossibility, if there is one, has not been demonstrated. The question remains open in the literature.

There is also a cost hidden inside these constructions, and it should be stated clearly. The expansion properties that make good qLDPC codes possible require connectivity that is not geometrically local in two dimensions. Qubits must interact with distant partners, and the interaction graph does not embed nicely into a planar architecture. This is unavoidable in a precise sense, since two-dimensional locality is exactly the constraint that forbids good parameters [21]. The practical implication is that good qLDPC codes are best suited to platforms with flexible, long-range connectivity, and that architectures built around them will look quite different from surface-code architectures. Whether such connectivity can be engineered at scale in near-term platforms is, honestly, unclear. This is a hardware question, and the thesis leaves it open, but the theoretical conclusion survives: the overhead problem of quantum error correction is not fundamental, it is architectural.

4.4 Graphical, Variational, and AI-Assisted Constructions

Alongside the asymptotic results, a second line of modern work approaches qLDPC construction from a representational angle, treating codes as graphical or variational objects that can be analyzed, manipulated, and optimized. This direction matters for the thesis because it is exactly where AI methods enter code design in practice.

Graphical representations are a natural fit for sparse codes. The study of quantum codes derived from classical graphical models shows that structures originally developed for classical statistical inference can guide the construction of quantum codes, and that the mapping between graphical models and code properties is informative in both directions [4]. The coherent parity check framework offers a complementary perspective, organizing quantum parity checks within an operational formalism that clarifies how checks can be composed and implemented [9]. More recently, variational graphical quantum error correction codes extend the graphical viewpoint by attaching tunable parameters to the graph structure and optimizing the code numerically against a target noise model [19]. Variational quantum error correction follows the same philosophy at the level of the full encoding and recovery scheme, treating code design as a continuous optimization problem rather than a purely combinatorial one [13].

Once code design is phrased as optimization, learning methods become applicable almost by definition. Reinforcement learning agents have been used to discover quantum error-correcting codes by treating construction as a sequential decision problem, and they have produced codes that match or improve upon hand-designed ones in specific settings [10], [15]. Hybrid quantum-assisted machine learning has been proposed as a way to learn better codes by combining classical learning with quantum resources [14]. On the decoding side, generative models learn the joint distribution of errors and syndromes [12], and transformer-based decoders treat decoding as structured sequence prediction [17]. There are even attempts to bring more abstract algebraic machinery, such as quantum superalgebras, into contact with AI-optimized code design [1], alongside continuing efforts to consolidate the general theory of quantum error-correcting codes [8]. The breadth of this activity is documented in the comprehensive review of AI for quantum error correction, which treats decoding, code discovery, and control as three faces of the same optimization landscape [3].

From the standpoint of this thesis, these approaches share one important property. They search inside the feasible region defined by quantum coding theory; they do not redraw its boundary. A reinforcement learning agent that discovers a surprising code has found a point that always existed in the space of valid stabilizer codes. A variational optimizer that adapts a graphical code to biased noise has exploited structure that the noise model already contained. This is not a criticism. Finding good points in an enormous space is precisely where learning methods are strong, and the space of sparse quantum codes is enormous by any standard. The asymptotically good constructions of the previous section were found through deep structural mathematics, but there is no reason to believe that every useful code at practical block lengths will be found the same way. At finite sizes, where asymptotic arguments are silent, AI search may be the most effective instrument available. The division of labor suggests itself: theory establishes what is possible, and learning finds where it is.

4.5 Noise-Adapted Constructions, Alternative Encodings, and Chapter Summary

A final family of modern directions deserves brief mention, because it modifies not the code structure but the assumptions underneath it. Noise-adapted quantum error correction abandons the goal of correcting arbitrary errors and instead tailors the code and the recovery to the dominant noise of a specific device, which can reduce overhead substantially when the noise is structured [18]. Continuous-time formulations optimize error correction for arbitrary noise without assuming the discrete cycle of syndrome measurement and correction [16]. At the level of physical encodings, theoretical work on molecular qudits shows that carefully designed multilevel systems can host error correction natively, shifting part of the burden from the code to the physical platform [11]. These directions do not compete with qLDPC codes so much as they remind us that the noise model and the physical layer are design variables too. Even here, the lesson of fundamental limits applies: noise can be reshaped and exploited, but its suppression always carries a cost, in correction as in mitigation [5].

This chapter examined quantum LDPC codes as concrete objects rather than abstract promises. The classical LDPC paradigm motivates the search for sparse quantum codes, but the commutation constraints of stabilizer codes make the quantum construction problem genuinely harder [7], [22], [23]. The surface code shows what two-dimensional locality can offer and what it costs, and its vanishing rate is a structural limitation that no decoder can repair [21], [26]. The modern breakthrough results prove that sparse quantum codes with constant rate and linear distance exist and can be decoded in linear time [24], which moves the overhead problem from the fundamental column to the architectural one and gives the idealized decoding oracle of this thesis a concrete mathematical counterpart. The relationship between these families and the hashing bound remains an open and interesting question [25]. Graphical, variational, and AI-assisted methods, finally, supply the practical machinery for exploring this design space at finite sizes [4], [9], [10], [13], [15], [19].

Two conclusions carry forward. The first is encouraging: the most discouraging limitation of practical quantum error correction, its overhead, has turned out not to be fundamental. The second is sobering: the constructions that remove it demand long-range connectivity that present hardware does not naturally provide. The next chapter takes up exactly this tension, and asks what AI-optimized codes and good qLDPC families imply for the design of realistic quantum architectures.

Chapter 5. Discussion and Implications for Quantum Architectures

5.1 The Architectural Meaning of AI-Optimized Quantum Error Correction

The previous chapters examined quantum error correction mainly from the viewpoint of theory: first through the basic principles of quantum information, then through fundamental limits, idealized decoding, and modern qLDPC constructions. This chapter turns the same discussion toward quantum architecture. This shift is necessary because a quantum error-correcting code does not operate in an abstract mathematical space alone. It must eventually be placed inside a physical machine, measured through real circuits, decoded under time restrictions, and supported by hardware that has its own geometry, noise, and connectivity. The theory may open the door, but the architecture decides how wide that door can actually become.

Quantum architecture is therefore not a secondary detail. It is part of the error-correction problem itself. A stabilizer code may be formally valid because its stabilizer generators commute and define a protected codespace, but these generators still have to be measured on real qubits [22], [23]. If a check involves qubits that are physically far apart, then the architecture must provide a way for those qubits to interact, either directly or through additional routing operations. If the decoder assumes clean and repeated syndrome information, the hardware must produce that information reliably. If a code has excellent asymptotic parameters but requires an interaction graph that is incompatible with a realistic device, then its practical value becomes less direct. It is not lost, exactly. But it becomes conditional.

This is where the central distinction of the thesis becomes architecturally meaningful. Some limitations in quantum error correction are information-theoretic. These include quantum capacity, correctability conditions, and the basic fact that no decoder can reconstruct quantum information that has become unrecoverable through the noise channel [5], [6]. Other limitations are structural. They arise from code construction, distance, rate, stabilizer weight, locality, and the geometry of checks [7], [21]. A third category is algorithmic. It concerns whether a decoder can infer a good recovery operation from syndrome information efficiently enough. AI methods are most relevant in this third category, although they can also help explore the second category by searching through code constructions [2], [3], [10], [15].

The architectural implication is simple, but it should not be underestimated. AI can make decoding more powerful, but it cannot make hardware constraints disappear. If the weakness of a system comes from an inefficient decoder, then AI may improve the system substantially. If the weakness comes from the structure of the code, AI may help discover a better construction. If the weakness comes from a fundamental bound, AI cannot overcome it. And if the weakness comes from physical architecture, such as limited qubit connectivity or noisy syndrome extraction, then AI may assist but cannot replace architectural design. The decoder is clever, perhaps very clever, but it still works inside a machine.

Surface-code architectures show this point clearly. The surface code remains one of the most influential models for fault-tolerant quantum computation because its checks are local and its layout is naturally compatible with two-dimensional hardware [21]. This makes it attractive for platforms where qubits are arranged in planar arrays and interact mainly with nearest neighbors. Its logic is conservative. It asks the hardware to do something difficult, but not something geometrically unrealistic. This is why the surface code continues to dominate many discussions of near-term fault tolerance.

At the same time, this architectural convenience comes with a cost. Surface codes have high overhead because their rate tends to zero as the system grows [21]. In practical terms, many physical qubits are required to protect a much smaller number of logical qubits. This is not simply a failure of decoding. A better decoder may reduce the logical error rate or improve performance under biased noise, but it cannot change the basic rate behavior of the code family. The surface code is therefore a useful example of a structural limitation that remains even when decoding improves. It protects well, but it protects expensively.

qLDPC codes offer a different architectural possibility. Their appeal lies in the fact that they combine sparse stabilizer checks with the possibility of constant rate and growing distance [24]. This is exactly what scalable quantum computation needs: protection that does not require the physical-to-logical qubit ratio to grow without control. The result that good qLDPC codes can exist is therefore highly significant. It suggests that the large overhead usually associated with practical QEC is not a fundamental law of quantum information. It is, at least partly, a consequence of the code families and architectures that have been easiest to implement.

However, qLDPC codes also make a demand. They often require connectivity that is not naturally planar or local. A code may be sparse in the sense that each stabilizer check touches only a bounded number of qubits, but those qubits may not be close to one another in a physical layout. Sparse is not the same as geometrically local. This small distinction carries a large architectural weight. It means that qLDPC codes may be better suited to hardware platforms with flexible or long-range connectivity, such as modular quantum processors, ion traps, neutral atom systems, or architectures supported by photonic links. These platforms are not automatically superior, of course. They have their own difficulties. But they may align more naturally with the connectivity patterns required by good qLDPC constructions.

The architectural question therefore changes. It is no longer only: which code has the best parameters? It becomes: which code has the best parameters under the interaction constraints of a particular architecture? A code with excellent asymptotic rate and distance may become less attractive if its checks require expensive routing. A code with weaker asymptotic properties may be more practical if its stabilizers are easy to measure. This may sound like an engineering compromise, but it is also a theoretical lesson. Code performance cannot be separated cleanly from the physical assumptions under which the code is realized.

AI-optimized decoding enters this discussion as a diagnostic layer. It helps separate what is caused by weak decoding from what is caused by the code or the architecture. If a qLDPC code performs poorly with a conventional decoder but well under an idealized AI decoder, then the code may have hidden potential. The limitation is mostly algorithmic. If, however, the same code remains limited even when the decoder is assumed to be nearly optimal, then the problem is likely structural or architectural. The architecture, in this sense, becomes a kind of witness. It reveals whether the beautiful object produced by theory can survive contact with physical constraints.

There is something slightly paradoxical here. AI is often associated with flexibility, adaptation, and generalization. Quantum architecture, by contrast, is full of fixed constraints: layout, gate fidelity, measurement errors, crosstalk, coherence times, and connectivity. The promise of AI-optimized QEC lies in bringing these two worlds into contact. AI can learn from the imperfections of hardware, adapt to noise, and improve decoding decisions. But it cannot float above the architecture. It must become part of it.

5.2 Hardware-Aware Codes, AI Decoding, and the Problem of Realistic Implementation

The practical success of quantum error correction depends on the co-design of codes, decoders, and hardware. This is one of the most important implications of the thesis. A code cannot be evaluated only by its rate and distance. A decoder cannot be evaluated only by its theoretical accuracy. Hardware cannot be evaluated only by the number of physical qubits. These quantities matter, but they matter together. A fault-tolerant quantum computer is not built from isolated achievements. It is built from their coordination.

The first part of this coordination concerns noise. Real quantum noise is rarely as simple as the noise models used in theoretical analysis. It may be biased, correlated, time-dependent, or affected by leakage and control errors. Measurement operations may introduce their own faults. Crosstalk may create dependencies between qubits that are not captured by independent Pauli error models. For this reason, noise-adapted quantum error correction is an important direction. Rather than designing a code for a generic channel, one may design or select a code that fits the dominant error mechanisms of a particular device [18]. This is not as elegant as a universal theory, but it may be more useful in practice.

AI is especially relevant here because learning methods can adapt to data. A decoder trained under realistic noise may identify patterns that are difficult to express analytically [3]. It may learn that certain syndrome patterns occur together, that certain qubits are more vulnerable, or that particular measurement histories indicate a higher probability of logical failure. Generative decoding approaches attempt to model the distribution of errors and recoveries rather than apply a fixed rule [12]. Transformer-based approaches treat syndrome information as structured input, allowing temporal or spatial dependencies to be represented more flexibly [17]. These methods do not remove the need for theory, but they soften the rigid boundary between theoretical noise models and device behavior.

Reinforcement learning adds another dimension. Instead of treating decoding as a static mapping from syndromes to corrections, reinforcement learning can frame QEC as a sequential decision problem [10], [15]. This is relevant because quantum error correction is not a single act. Syndrome extraction happens repeatedly. Recovery decisions interact with future errors. Measurement schedules and correction strategies may influence later states of the system. In this setting, an AI agent can, at least in principle, learn policies that account for long-term consequences rather than immediate syndrome correction alone.

This is attractive, but it should be treated carefully. A reinforcement learning agent may perform well in simulation but fail when the noise distribution changes. A neural decoder may learn correlations that are present in training data but absent in another hardware regime. A generative model may produce plausible recoveries that are statistically strong but difficult to certify. In classical machine learning, such failures may be inconvenient. In fault-tolerant quantum computation, they may be disastrous. The decoder protects information that cannot be inspected directly without disturbance. Its errors may remain hidden until a logical failure has already occurred. This gives reliability a different weight.

For this reason, AI decoders must be not only accurate but also fast, stable, and verifiable. Speed is essential because syndrome data are produced continuously. A decoder that requires too much time may allow errors to accumulate before recovery decisions are applied. Stability is necessary because quantum hardware may drift. Calibration changes, temperature variations, and hardware aging can all affect the noise process. Verifiability is important because a black-box decoder may be difficult to trust in a fault-tolerant stack. The architecture cannot simply accept an answer because a neural network produced it. It needs confidence, bounds, or at least reliable empirical validation.

The same issue appears in AI-assisted code discovery. Machine learning methods can search through large spaces of possible codes, including constructions that may not be obvious to human researchers [2], [10], [15]. Variational quantum error correction treats encoding and recovery as an optimization problem [13]. Variational graphical QEC codes extend this idea by representing codes through graph-based structures with tunable parameters [19]. Coherent parity-check constructions and graphical models also show how sparse constraints can be organized in ways that connect coding theory with representational structure [4], [9]. This is useful because qLDPC codes are naturally graph-like objects. Their checks, qubits, and connectivity relations can be treated as a sparse network.

Still, AI-generated codes must pass the same tests as human-designed codes. The stabilizers must commute. The code must have meaningful distance. The syndrome extraction circuit must be physically plausible. The decoder must run under realistic constraints. A code discovered by AI is not exempt from quantum coding theory. It is not a shortcut around mathematics. It is a candidate that must be interpreted mathematically and architecturally. This is where theory and AI begin to look less like competitors and more like uneven collaborators. AI explores. Theory judges. Architecture decides whether the result can be built.

The issue of locality remains central. In strictly two-dimensional local architectures, there are limits on how efficiently quantum information can be protected. Surface-code-like constructions fit such architectures well, but their overhead remains high [21]. Good qLDPC codes improve rate-distance behavior but often require nonlocal connectivity [24]. This creates a trade-off between physical simplicity and coding efficiency. A two-dimensional nearest-neighbor machine may favor local codes with high overhead. A modular or networked machine may support qLDPC-style connectivity, but then it must manage link errors, synchronization, routing, and additional control complexity.

This does not mean that qLDPC codes are impractical. It means that their practicality depends on architectural evolution. If future hardware remains mostly planar and nearest-neighbor, then surface-code architectures may remain dominant for a long time. If hardware develops reliable long-range couplers, modular connections, or photonic interconnects, then qLDPC codes may become more attractive. The theory has already shown that better asymptotic parameters are possible. The remaining question is whether architecture can provide the physical graph on which those parameters become useful.

There is also a finite-size issue. Many theoretical results are asymptotic. They tell us what happens as the number of qubits becomes very large. Real quantum computers, especially early fault-tolerant machines, will operate at finite sizes. At these sizes, asymptotic superiority may not immediately translate into practical superiority. A qLDPC family may have excellent scaling but poor constants. A surface code may be inefficient asymptotically but robust and predictable at moderate sizes. AI search may be especially useful here because it can explore finite-size designs where clean asymptotic theory gives less guidance. In this region, practical code discovery becomes almost experimental, even when no physical experiment is performed.

Continuous-time quantum error correction also expands the architectural discussion. Instead of treating QEC as a sequence of discrete syndrome measurements and corrections, continuous-time approaches view protection as a dynamical process that can be optimized against arbitrary noise [16]. This connects QEC with feedback control and suggests that future architectures may blur the line between decoding and physical control. The decoder may not simply sit outside the quantum device, waiting for syndrome bits. It may participate in an ongoing control loop. That possibility changes the role of AI again. AI may help optimize not only recovery decisions, but also the timing, strength, and structure of error-correction dynamics.

Other physical encodings also show that architecture can reshape the QEC problem. Theoretical work on molecular qudits suggests that the internal structure of physical systems may be designed with error correction in mind [11]. Entanglement-assisted quantum error-correcting codes show that additional entanglement resources can relax certain coding constraints and produce useful constructions in communication settings [20]. These examples are not the main focus of the thesis, but they matter because they show that architecture can supply resources that change what a code can do. The code is not merely imposed on hardware. Sometimes the hardware itself becomes part of the code design.

Nevertheless, the same boundary remains. Noise-adapted codes, AI decoders, continuous-time correction, and alternative encodings may improve performance, sometimes significantly. But they do not erase fundamental limits. Studies on the limits of quantum error mitigation show that noise reduction cannot be achieved without cost [5]. The same general attitude applies to QEC. One may distribute the cost differently: more qubits, more connectivity, more computation, more control, or more training. But the cost does not vanish. It simply changes its shape.

5.3 Implications for Fault-Tolerant Quantum Computing and Chapter Synthesis

The main implication of this chapter is that scalable fault-tolerant quantum computing requires co-design. This term is sometimes used casually, but here it has a precise meaning. The code, the decoder, and the architecture must be designed with each other in mind. A theoretically strong code may fail architecturally if its stabilizer checks are too difficult to measure. A powerful decoder may fail practically if it cannot run in real time.

A flexible architecture may still fail if it does not support a code with sufficient distance and low enough logical error rate. Fault tolerance emerges only when these layers work together.

The comparison between surface codes and qLDPC codes illustrates the problem clearly. Surface codes are local, robust, and well matched to two-dimensional layouts [21]. Their main disadvantage is overhead. qLDPC codes promise better asymptotic efficiency, including constant rate and growing distance [24]. Their main disadvantage is architectural complexity, especially the need for connectivity patterns that may not be naturally available in planar hardware. Neither approach is simply better in all contexts. The surface code is easier to imagine in current hardware. qLDPC codes are more attractive when thinking about long-term scalability. The field may need both.

A likely direction is hierarchical architecture. At lower levels, local error correction may handle frequent small-scale errors close to the hardware. At higher levels, more global codes, perhaps qLDPC-inspired, may provide better logical density if long-range connectivity is available. AI decoders could operate across these layers, using fast local inference where immediate decisions are needed and more complex global inference where timing constraints are less severe. This is speculative, but not unreasonable. The architecture of a mature quantum computer may not resemble a single uniform code repeated everywhere. It may be layered, adaptive, and somewhat untidy.

AI fits naturally into this layered picture because it can operate at different levels. It can support low-level decoding by learning device-specific syndrome patterns [3]. It can support code design by searching over graph structures and stabilizer arrangements [2], [10], [15]. It can support generative decoding by modeling the distribution of likely errors [12], [17]. It can support variational optimization of codes and recoveries [13], [19]. It may even help select between architectural strategies by estimating whether a performance gap is mainly algorithmic, structural, or physical. This diagnostic role is perhaps one of the most important contributions of AI in the theoretical framework of the thesis.

At the same time, AI should not be placed outside the limits of quantum information theory. The decoder receives syndrome information. It does not receive the unknown logical state directly. It cannot copy quantum information, and it cannot reverse a noise process once the relevant recoverable information has been lost [6], [22]. This is why the idealized AI oracle used in the thesis is useful but also limited. It removes decoder inefficiency from the analysis, allowing the remaining limitations to become visible. But it does not describe a real deployable system. A real AI decoder must be trained, executed, maintained, validated, and integrated into a hardware control stack.

This distinction has consequences for how future research should be organized. If the goal is to reduce overhead, then qLDPC codes and other high-rate constructions deserve serious attention. If the goal is to improve near-term fault tolerance on local hardware, then surface-code improvements, noise-adapted decoding, and better syndrome processing may be more immediate. If the goal is to understand fundamental limits, then idealized decoding remains useful because it separates code limitations from decoder limitations. These goals overlap, but they are not identical. Sometimes research discussions blur them. The result is confusion: a practical improvement is mistaken for a fundamental breakthrough, or a fundamental result is dismissed because it is not yet easy to implement.

The most careful interpretation is this: AI-optimized quantum error correction does not replace quantum coding theory or quantum architecture. It connects them. It searches through spaces that are too large for manual exploration. It learns patterns in noise and syndrome data. It may approximate optimal decoding where exact decoding is infeasible. It may reveal that a code performs poorly only because the decoder is weak. But once a limitation is caused by quantum capacity, correctability conditions, locality, or hardware connectivity, AI alone cannot remove it. The work must then shift to code construction, architecture, or physical implementation.

This chapter therefore supports the broader argument of the thesis. Fundamental limits define the outer boundary of what is possible. qLDPC constructions show that some previously discouraging overhead barriers are not fundamental in the strongest sense [24]. Surface-code architectures show that physical locality remains a powerful and restrictive design principle [21]. AI methods show that decoding and code discovery can be improved, sometimes dramatically, when learning is used to explore complex spaces [3], [10], [12], [15], [17]. But none of these elements is sufficient by itself.

The future of fault-tolerant quantum computing will probably depend on how well these elements are combined. A good code without a feasible architecture is a theory waiting for a machine. A good architecture without efficient error correction is fragile hardware.

A good decoder without enough syndrome information is only making the best possible guess. The quiet lesson is that quantum error correction is not one problem. It is a layered problem, and each layer speaks to the others.

In conclusion, the implications of AI-optimized QEC for quantum architectures are both optimistic and restrained. They are optimistic because AI can help approach the best performance allowed by a code and may assist in discovering new finite-size constructions that are difficult to find manually. They are restrained because AI cannot override quantum information theory or physical implementation constraints. This balance is important. It prevents unrealistic claims, but it also leaves room for genuine progress. AI may not move the fundamental boundary, but it can help identify where that boundary actually is, and perhaps more importantly, where current systems are still far from it.

Chapter 6. Conclusions

6.1 Summary of the Thesis

This thesis examined the theory and design of AI-optimized quantum error correction codes, with emphasis on fundamental limits, idealized decoding, and modern quantum LDPC constructions. Its central argument was that artificial intelligence should not be understood as a force that removes the boundaries of quantum information theory. Rather, it should be treated as a powerful method for approaching those boundaries more clearly, and sometimes more efficiently. This distinction may seem small at first. It is not. Much of the discussion around AI and quantum error correction becomes overly optimistic when decoder improvement is confused with fundamental improvement.

Quantum error correction is necessary because quantum information is intrinsically fragile. Qubits are affected by decoherence, unwanted interaction with the environment, measurement disturbance, and different forms of physical noise. Unlike classical information, quantum information cannot simply be copied and checked directly, because the no-cloning principle and the nature of measurement impose deep restrictions on what can be observed and recovered [22], [23]. The mathematical theory of quantum error correction shows that correctability depends on the relationship between the code space and the error operators, not only on the cleverness of the recovery procedure [6]. This means that even before AI enters the discussion, QEC already operates within a strict theoretical frame.

The thesis first established the background of quantum information theory, stabilizer codes, quantum channels, code parameters, and decoding. The parameters of a quantum code, usually expressed through the number of physical qubits, logical qubits, and distance, reveal a persistent tension between efficiency and protection. A code with high rate is attractive because it uses fewer physical qubits per logical qubit. A code with high distance is attractive because it provides stronger protection against errors. But these goals cannot always be increased together. This rate-distance tension is not merely a technical inconvenience; in many cases it reflects structural or physical constraints of quantum coding theory [7], [21], [23].

A major conclusion of the thesis is that AI-optimized decoding can help overcome algorithmic limitations, but it cannot overcome information-theoretic limits. This is perhaps the most important point of the whole study. If a decoder fails because it is inefficient, poorly adapted to the noise model, or unable to exploit code degeneracy, then AI may provide a real advantage. Machine learning, reinforcement learning, generative decoding, and transformer-based methods can learn complex syndrome-error relations and approximate stronger recovery strategies [2], [3], [10], [12], [15], [17]. However, if the quantum channel has destroyed the relevant recoverable information, or if the syndrome does not contain enough information to distinguish logically different errors, then no decoder can fully repair the loss. The missing information is not hidden. It is gone.

For this reason, the thesis used the idea of the idealized AI decoder as a theoretical oracle. This model is not meant to describe a real deployable AI system. Real AI decoders require training data, computational resources, validation, hardware integration, and robustness against changes in noise. The idealized oracle is useful for another reason: it removes decoder inefficiency from the analysis. Once decoding is assumed to be optimal, any remaining failure must come from the code, the channel, the noise structure, or the architecture. In this sense, AI becomes a diagnostic instrument. It helps reveal whether a limitation is algorithmic, structural, architectural, or truly fundamental.

6.2 Main Findings

The first main finding is that fundamental bounds such as quantum capacity are independent of the particular decoding strategy. A decoder can help a code approach a capacity-related limit, but it cannot allow reliable communication above that limit. This conclusion follows directly from the information-theoretic nature of quantum capacity and from the general theory of quantum error correction for noisy channels [5], [6], [25]. In practical terms, this means that an AI decoder may improve logical error rates and may outperform conventional decoding algorithms, but it does not change the capacity of the physical channel. The boundary remains. AI may walk closer to it, but it does not move the wall.

The second finding is that some important limitations in QEC arise not from decoding weakness, but from structure and locality. Surface codes illustrate this point clearly. They are robust, local, and well suited to two-dimensional quantum architectures, which makes them highly attractive for realistic hardware [21]. At the same time, their asymptotic rate is poor, and this creates large overhead when scaling toward fault-tolerant quantum computation. An ideal decoder can improve decoding performance, but it cannot change the basic code parameters of the surface code. Therefore, the overhead problem is not only a decoding problem. It is also a structural problem. This is why future progress cannot depend only on better decoders; it must also involve better code families and better architectural support.

The third finding concerns quantum LDPC codes. qLDPC codes are important because they promise sparse checks, scalable syndrome extraction, and more favorable asymptotic behavior than many traditional constructions. Their sparse structure resembles classical LDPC codes, but the quantum case is more constrained because stabilizer checks must commute [7], [22], [23]. This makes qLDPC design mathematically delicate. Still, modern results have shown that good quantum LDPC codes with constant rate and growing distance do exist, and even that such codes can be equipped with linear-time decoders [24]. This is a major theoretical development. It shows that the poor overhead scaling of earlier sparse quantum codes was not a fundamental law of quantum mechanics. It was, at least partly, a limitation of known constructions.

At the same time, this result does not solve the entire problem. Good qLDPC codes may require connectivity patterns that are difficult to realize in present quantum hardware. This shifts the challenge from the purely coding-theoretic level to the architectural level. The limitation is no longer simply "we do not know whether good sparse quantum codes exist." They do exist [24]. The new question is whether their interaction structure, syndrome extraction requirements, and decoding demands can be embedded into a realistic fault-tolerant quantum computer. The problem has not disappeared. It has moved.

The fourth finding is that AI has theoretical significance because it helps distinguish different kinds of limitations. Reinforcement learning can search for code constructions that are difficult to identify manually [10], [15]. Generative decoders can model complex error distributions and produce recovery operations in a more flexible way than fixed heuristic rules [12], [17]. Variational and graphical approaches can connect code design with optimization over structured representations [4], [13], [19]. These methods are valuable because they expand the searchable design space of QEC. Yet they remain bounded by stabilizer constraints, channel capacity, syndrome information, and physical implementation costs.

This leads to a more careful interpretation of AI in quantum coding theory. AI is not only a practical decoder. It is also a way of asking: if decoding were nearly optimal, what would still prevent the code from succeeding? Sometimes the answer will be poor architecture. Sometimes it will be weak code structure. Sometimes it will be insufficient physical qubit quality. Sometimes, more quietly but more decisively, it will be an information-theoretic limit. This layered interpretation is one of the central contributions of the thesis.

6.3 Contributions of the Thesis

The first contribution of the thesis is the classification of QEC limitations into three broad categories: fundamental, structural or architectural, and algorithmic. Fundamental limitations include quantum capacity, correctability conditions, and unavoidable restrictions imposed by quantum information theory [5], [6], [25]. Structural and architectural limitations include locality, check weight, connectivity, stabilizer commutation, and hardware layout [7], [21], [23]. Algorithmic limitations include inefficient decoding, poor heuristic search, inability to exploit degeneracy, limited adaptation to noise, and computational complexity. AI is most powerful in the third category. It may assist the second category by helping discover new constructions. It cannot remove the first category.

The second contribution is the interpretation of AI-optimized decoding as an idealized theoretical lens. This lens is useful because it clarifies what remains difficult after decoder inefficiency is removed. If a code performs well only under ideal decoding, then the main practical obstacle is likely algorithmic. If it performs poorly even under ideal decoding, then the limitation is probably deeper. This framework does not produce a new decoder by itself, but it gives a clearer way to interpret decoder performance. That matters, because in QEC it is easy to mistake a poor decoder for a poor code, or a poor code for a fundamental impossibility.

The third contribution is the discussion of qLDPC codes in relation to AI and fundamental limits. Modern qLDPC constructions show that sparse quantum codes can achieve strong asymptotic properties [24]. This changes the theoretical landscape. It suggests that scalable QEC may not require accepting the very large overheads associated with strictly local two-dimensional codes. However, the thesis also emphasized that qLDPC codes introduce serious architectural questions. Sparse does not automatically mean easy to implement. A sparse graph can still be nonlocal. A low-weight check can still connect qubits that are physically far apart. The code may look elegant on paper while the hardware resists it.

The fourth contribution is the connection between AI-based code discovery and future architecture-aware QEC. AI methods may be useful not only for decoding existing codes, but also for exploring new code families, noise-adapted strategies, and hybrid constructions [2], [3], [10], [15], [18]. Continuous-time QEC and alternative physical encodings also suggest that the boundary between code, decoder, and device may become less rigid in future systems [11], [16]. This is an important direction. Quantum error correction may not develop as a single universal code applied everywhere. It may become more heterogeneous, more layered, and more dependent on the physical platform. A little less neat, perhaps, but more realistic.

6.4 Limitations and Future Work

The main limitation of this thesis is that it remained theoretical. It did not implement AI decoders, simulate qLDPC codes, compare neural architectures, or benchmark logical error rates under specific noise models. This limitation was intentional, because the purpose of the thesis was to clarify the conceptual relationship between AI optimization and fundamental QEC limits. However, it also means that the conclusions should be understood as theoretical rather than experimental. They explain how AI should be interpreted within QEC theory, but they do not prove that a specific AI model will perform well on a specific hardware platform.

A second limitation is that the thesis treated the idealized AI decoder as an oracle. This is useful for theoretical reasoning, but real AI systems are not oracles. They can overfit, fail under distribution shift, require large training datasets, and produce decisions that are difficult to interpret. In a fault-tolerant quantum computer, the decoder must also operate under strict latency constraints. A decoder that is accurate but too slow may be practically useless. This is especially important because quantum error correction must often run continuously, not as an offline analysis performed after the computation has ended.

A third limitation concerns the complexity of physical noise. Much of quantum coding theory relies on simplified noise models, such as independent Pauli noise. Real devices may experience correlated noise, leakage, crosstalk, calibration drift, measurement error, and non-Markovian effects. Noise-adapted QEC attempts to exploit the structure of specific noise models [18], while continuous-time approaches treat correction as an ongoing dynamical process [16]. These directions deserve further study because they may make AI more useful in practice. AI systems are often strongest when patterns exist but are difficult to model explicitly. Real quantum hardware may provide exactly that kind of problem.

Future work should therefore move in several directions. One direction is the development of architecture-aware AI decoders for qLDPC codes. Such decoders should not only minimize logical error rates in abstract simulations, but also account for connectivity, syndrome extraction cost, timing, and hardware noise. Another direction is AI-assisted code discovery under explicit architectural constraints. Instead of searching for the best abstract code, AI could search for the best code that a specific device can actually implement. This would make the design process less idealized and more useful for fault-tolerant engineering.

A further direction is the study of hybrid QEC architectures. Surface codes may remain attractive at lower hardware levels because of their locality and robustness [21]. qLDPC-inspired codes may become more useful at higher layers where better connectivity or logical-level operations are available [24]. AI decoders could operate across these layers, selecting different inference strategies depending on time constraints and syndrome structure. Such a system would not be perfectly uniform, but mature technologies often are not. They accumulate layers. They become practical by becoming slightly untidy.

Finally, future research should continue to clarify the relationship between qLDPC constructions and capacity-related bounds. The existence of good qLDPC codes is a major step, but the question of how close practical and idealized qLDPC families can come to hashing-type or capacity-related limits remains open in important settings [24], [25]. AI may help explore this question numerically and constructively, but the final interpretation must still be theoretical. Otherwise, performance improvement may be mistaken for a proof of optimality.

6.5 Final Conclusion

The overall conclusion of this thesis is that AI-optimized quantum error correction should be understood neither as a miracle solution nor as a minor technical detail. It occupies a more interesting position. AI can improve decoding, assist code discovery, adapt to noise, and reveal hidden structure in complex QEC systems [2], [3], [10], [12], [15], [17]. It can also act as a theoretical lens by approximating the idea of optimal decoding and helping researchers distinguish algorithmic failure from deeper limitations. In that sense, AI changes how QEC can be studied.

However, AI does not stand outside quantum information theory. It cannot copy unknown quantum states. It cannot extract information that the syndrome does not contain. It cannot exceed quantum capacity. It cannot make nonlocal checks local by decoding more intelligently. These restrictions are not signs of failure; they are the conditions under which QEC exists [5], [6], [22], [23]. A serious theory of AI-optimized QEC must therefore be both ambitious and restrained.

The most promising path forward is co-design. Codes, decoders, and architectures must be developed together. qLDPC codes show that better asymptotic efficiency is possible [24]. Surface codes show that locality and architectural compatibility remain powerful advantages [21]. AI shows that complex decoding and design spaces can be explored more effectively than before [3], [10], [15], [17]. None of these elements is sufficient alone. Together, they suggest a realistic direction for fault-tolerant quantum computing.

In the end, the thesis argues that the value of AI in QEC is not that it abolishes limits. Its value is that it helps locate them. Sometimes it may reveal that a supposed limit was only a weakness of the decoder. Sometimes it may show that a code has more potential than expected. And sometimes it will confirm that the boundary is genuinely fundamental. That result may feel less spectacular than a claim of unlimited optimization, but it is more useful. Quantum error correction needs progress, but it also needs clarity. AI can contribute to both, provided it remains inside the theory rather than pretending to float above it.

References

[1] M. K. Pasupuleti, “Harnessing Quantum Superalgebras for AI-Optimized Quantum Error Correction,” 2024.

[2] L. Huang, H. Zhang, R. Li, Y. Ge, and J. Wang, “AI Coding: Learning to Construct Error Correction Codes,” 2024.

[3] Z. Wang and H. Tang, “Artificial Intelligence for Quantum Error Correction: A Comprehensive Review,” 2024.

[4] J. Roffe, S. Zohren, D. Horsman, and N. Chancellor, “Quantum Codes From Classical Graphical Models,” 2020.

[5] R. Takagi, S. Endo, S. Minagawa, and M. Gu, “Fundamental Limits of Quantum Error Mitigation,” npj Quantum Information, 2022.

[6] E. Knill, R. Laflamme, and L. Viola, “Theory of Quantum Error Correction for General Noise,” Physical Review Letters, vol. 84, no. 11, pp. 2525–2528, 2000.

[7] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, “Quantum Error Correction and Orthogonal Geometry,” Physical Review Letters, vol. 78, no. 3, pp. 405–408, 1997.

[8] L. Fang, “A Theory of the Quantum Error-Correcting Codes,” 2020.

[9] N. Chancellor, A. Kissinger, S. Zohren, and D. Horsman, “Coherent Parity Check Construction for Quantum Error Correction,” 2016.

[10] V. P. Su, C. Cao, H.-Y. Hu, Y. Yanay, and C. Tahan, “Discovery of Optimal Quantum Error Correcting Codes via Reinforcement Learning,” 2023.

[11] A. Chiesa, F. Petiziol, M. Chizzini, P. Santini, and S. Carretta, “Theoretical Design of Optimal Molecular Qudits for Quantum Error Correction,” Physical Review Letters, vol. 128, no. 13, 2022.

[12] H.-Y. Cao, F. Pan, D. Feng, Y. Wang, and P. Zhang, “Generative Decoding for Quantum Error-Correcting Codes,” 2023.

[13] N. Meyer, C. Mutschler, A. Maier, and D. D. Scherer, “Variational Quantum Error Correction,” 2022.

[14] Y. Yanay, “Learning Better Error Correction Codes with Hybrid Quantum-Assisted Machine Learning,” 2022.

[15] H. P. Nautrup, N. Delfosse, V. Dunjko, H. J. Briegel, and N. Friis, “Optimizing Quantum Error Correction Codes with Reinforcement Learning,” Quantum, vol. 3, p. 215, 2019.

[16] A. Lanka, S. Hegde, and T. A. Brun, “Optimizing Continuous-Time Quantum Error Correction for Arbitrary Noise,” 2023.
[17] H.-Y. Cao, F. Pan, Y. Wang, and P. Zhang, “qecGPT: Decoding Quantum Error-Correcting Codes with Generative Pre-trained Transformers,” 2024.
[18] A. Jayashankar and P. Mandayam, “Quantum Error Correction: Noise-Adapted Techniques and Applications,” 2023.
[19] Y. Shao, Y.-C. Li, F. Wei, H. Zhan, and B. Wang, “Variational Graphical Quantum Error Correction Codes,” 2024.
[20] K. Guenda, S. Jitman, and T. A. Gulliver, “Constructions of Good Entanglement-Assisted Quantum Error Correcting Codes,” Designs, Codes and Cryptography, vol. 86, pp. 121–136, 2018.
[21] B. M. Terhal, “Quantum Error Correction for Quantum Memories,” Reviews of Modern Physics, vol. 87, no. 2, pp. 307–346, 2015.
[22] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, 10th Anniversary ed. Cambridge, U.K.: Cambridge University Press, 2010.
[23] D. Gottesman, “Stabilizer Codes and Quantum Error Correction,” Ph.D. dissertation, California Institute of Technology, Pasadena, CA, USA, 1997.
[24] I. Dinur, M.-H. Hsieh, T.-C. Lin, and T. Vidick, “Good Quantum LDPC Codes with Linear Time Decoders,” in Proc. 55th Annual ACM Symposium on Theory of Computing, 2023, pp. 905–918.
[25] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, “Mixed-State Entanglement and Quantum Error Correction,” Physical Review A, vol. 54, no. 5, pp. 3824–3851, 1996.
[26] J. Preskill, “Quantum Computing in the NISQ Era and Beyond,” Quantum, vol. 2, p. 79, 2018.