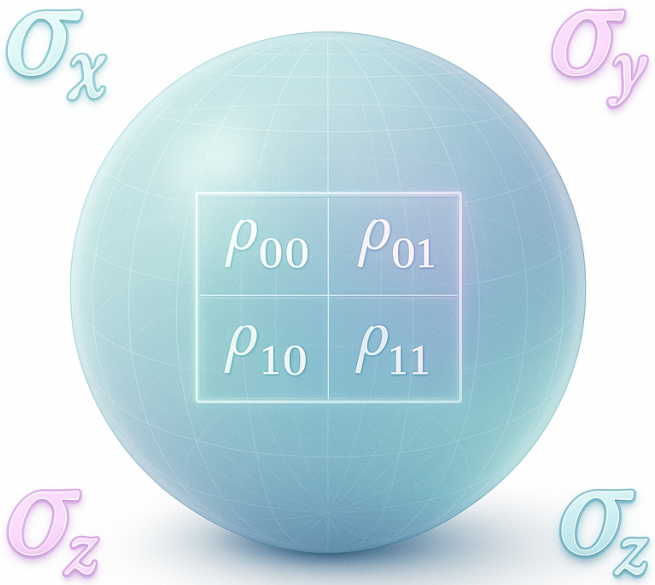
CHALMERS
UNIVERSITY OF TECHNOLOGY



Quantum State Tomography

Optimizing and Scheduling Quantum State Tomography Experiments with Graph Coloring Heuristics

Master's thesis in Nanotechnology (MPNAT)

Sumukh Suresh Moudghalya

DEPARTMENT OF MICROTECHNOLOGY AND NANOSCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2025
www.chalmers.se

Master's thesis 2025

Quantum State Tomography

Optimizing and Scheduling Quantum State Tomography Experiments with Graph Coloring Heuristics

Sumukh Suresh Moudghalya

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Microtechnology and Nanoscience
Applied Quantum Physics Lab
Chalmers University of Technology
Gothenburg, Sweden 2025

Optimizing and Scheduling Quantum State Tomography
Experiments with Graph Coloring Heuristics
Sumukh Suresh Moudghalya

Supervisor: Akshay Gaikwad and Anton Frisk Kockum, AQP
Examiner: Anton Frisk Kockum, AQP

Master's Thesis 2025
Department of Microtechnology and Nanoscience
Applied Quantum Physics (AQP)
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2025

Optimizing and Scheduling Quantum State Tomography Experiments with Graph Coloring Heuristics
Sumukh Suresh Moudghalya
Department of Microtechnology and Nanoscience
Chalmers University of Technology

Abstract

Quantum State Tomography (QST) is the standard tool for reconstructing an unknown quantum state-typically represented by a density matrix-through the measurement of expectation values of a set of observables (data acquisition step), followed by suitable post-processing. One of the key challenges in QST is to determine an optimal set of quantum circuits (unitary operations) to measure a tomographically complete set of observables-a task that becomes computationally hard due to the exponential scaling of the Hilbert space with the number of qubits. In this work, we recast the task of identifying an optimal set of quantum circuits in QST as a graph coloring problem. Within this framework, we explore a variety of algorithms and heuristic approaches to optimize QST experiments, including Degree of Saturation (DSATUR), Recursive Largest First (RLF), Integer Linear Programming (ILP), Graph Neural Networks (GNNs), Spectral Clustering (SC), and a range of hybrid methods that aim to balance solution optimality with computational efficiency. Our results demonstrate that the graph coloring heuristic not only determines and substantially reduces the number of required quantum circuits but also guides appropriate priority-based experiment scheduling that maximizes information gain per experimental setting for efficient QST. Furthermore, the optimization process completes within minutes for systems of up to five qubits on a standard laptop, offering a substantial speedup over brute-force methods, which can take several hours. This highlights the heuristic's potential as a practical and scalable solution for characterizing noisy intermediate-scale quantum (NISQ) devices.

Keywords: Quantum State Tomography, Graph Coloring, Graph Neural Network

Acknowledgments

I would first like to thank my supervisors, Akshay Gaikwad and Anton Frisk Kockum, for all the useful discussions and help I have received from them during this project. I would also like to thank them for introducing me to the field of quantum tomography which helped me to build a strong foundation in quantum information science. I would like to express my gratitude to my examiner, Anton Frisk Kockum, for letting me attend the group meetings, which made me feel more included. Finally, I would like to appreciate the Knut and Alice Wallenberg Foundation for its contributions to the Wallenberg Centre for Quantum Technology, which is a large part of why this thesis was possible here at Chalmers.

Sumukh Suresh Moudghalya, Gothenburg, May 2025

Contents

1

Introduction

Quantum State Tomography (QST) is a fundamental technique for characterizing quantum systems in quantum computing and quantum information science [1, 2]. It provides a standard method to completely determine the state of a quantum system without prior assumptions. This is crucial for tasks such as verifying the output of quantum algorithms, benchmarking quantum devices, and studying entanglement and decoherence in quantum experiments [3, 4]. In practice, QST involves preparing many identical copies of a quantum state and performing a series of measurements on different bases. From the collected data, one can reconstruct the density matrix (or state vector) that describes the quantum state. The importance of QST lies in its universality - it can, in principle, fully characterize any quantum state, enabling researchers to validate quantum gates, calibrate qubits, and gain insights into complex quantum phenomena. However, this power comes at a steep cost: QST is notoriously resource-intensive. The number of measurements and the classical post-processing effort scale exponentially with the number of constituents (e.g. qubits) [5]. For example, an n-qubit system has dimension 2^n, and a general (mixed) state is described by a $2^n \times 2^n$ density matrix with 4^n independent parameters. Consequently, the measurement settings and samples required for full tomography grow exponentially as the system size increases, making standard QST infeasible beyond only a handful of qubits [6, 7]. This introduction provides an overview of QST and its challenges, reviews recent advances toward more efficient tomography, and motivates a new approach using graph coloring heuristics to optimize and schedule QST experiments.

1.1 QST: Process and Challenges

In QST, one seeks to determine an unknown quantum state by performing a comprehensive set of measurements. The basic process of QST for an n-qubit system typically involves selecting a set of measurement bases (or observables) such that the outcomes are informationally complete, meaning they provide enough independent equations to reconstruct the 2^n-dimensional density matrix [8]. For a single qubit, a simple example is measuring spin along the X, Y, and Z axes; the resulting probabilities determine the Bloch vector coordinates of the qubit's state. For multi-qubit systems, a common approach is to measure tensor product bases such as combinations of single qubit Pauli operators (e.g. X, Y, Z on each qubit). In a naive scheme, one might perform 3^n distinct measurement settings (each setting being a choice of basis for each qubit) to collect statistics for all 3^n Pauli strings,

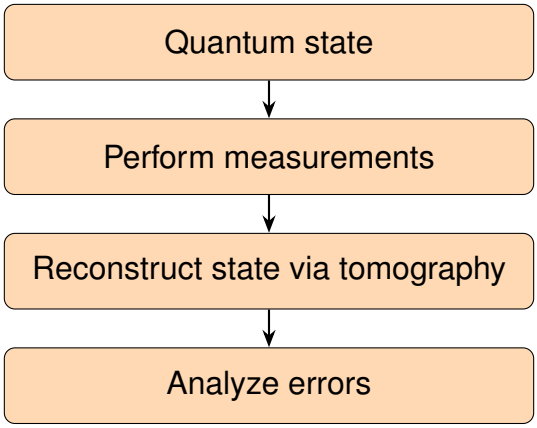


Figure 1.1: Flowchart of the quantum state tomography procedure.

which is sufficient to reconstruct an arbitrary n-qubit state [9, 10]. This number, however, grows exponentially with n for instance, for $n = 10$ qubits, $3^{10} = 59,049$ separate measurement configurations would be required in a brute-force approach. In general, the parameter space of an n qubit mixed state has dimension $4^n - 1$ (the number of free parameters in a $2^n \times 2^n$ density matrix minus one normalization constraint). The experimental effort to gather enough data for these parameters is enormous, as is the classical computational effort to invert the data to find the state. Indeed, quantum state tomography is an indispensable but costly part of many quantum experiments, typically requiring a large number of measurement settings on a fixed setup. Often the collected data is vastly overcomplete - many measurements are redundant, which points to significant inefficiencies in the naive approach [11]. As quantum processors grow in qubit count, this "curse of dimensionality" makes conventional QST impractical, motivating the search for more optimal strategies to reduce measurement overhead while still obtaining an accurate reconstruction of the state.

1.1.1 Recent Advances in QST

Researchers have developed a variety of methods to ease the computational burden of QST [8, 9, 10, 12, 13, 14, 15, 16, 11, 17, 18]. Broadly, these advances fall into two categories: (1) improved post-processing algorithms that reconstruct states from data more efficiently or with fewer measurements, and (2) optimized measurement strategies that reduce the number of measurements or settings needed in the first place. We review key developments in both areas below, highlighting their advantages and limitations.

1.1.1.1 Post-Processing (Reconstruction Algorithms)

A straightforward approach to QST is linear inversion by solving the linear equations from measurement outcomes but this can yield unphysical states due to statistical noise. To address this, Maximum Likelihood Estimation (MLE) [19] is often used to find the physical density matrix that best fits the data, imposing positivity and normalization constraints. MLE ensures a valid quantum state but still requires a

full data set and can be computationally intensive for large systems [20]. In recent years, more advanced algorithms leverage prior knowledge or assumptions about the state to reduce the number of measurements needed for reconstruction. One major development is compressed sensing tomography, which assumes the quantum state is of low rank (for example, nearly pure) and uses convex optimization to reconstruct the state from an incomplete set of measurements [21]. The rationale is that if the state has significantly fewer degrees of freedom than the 4^n maximum, one can infer the state from far fewer measurement outcomes than a full tomography would require. Gross et al. [22] and others demonstrated that for states that are nearly pure, the number of measurements can scale only linearly in the Hilbert space dimension (on the order of $O(2^n)$) rather than $O(4^n)$. Empirical studies [23] showed that compressed-sensing estimators can achieve higher fidelity reconstructions than standard MLE given the same data, and using fewer measurements leads to faster classical processing with negligible loss of accuracy.

Another promising direction is the use of machine learning in tomography. Neural network-based methods can represent quantum states in compressed forms (e.g. using restricted Boltzmann machines or variational autoencoders) and learn to reconstruct states from measurements. For instance, Torlai et al. [24] used a neural network to perform QST with far fewer measurement settings by training the network on measurement outcomes to predict the density matrix, effectively generalizing from sample data. Similarly, Bayesian approaches treat the state as a random variable and update a probability distribution (the belief over states) as measurements come in, often yielding an accurate estimate with fewer total measurements by adaptively focusing on the most uncertain parameters [25]. These post-processing techniques significantly extend the range of QST: they allow targeted or partial tomography, estimating only certain properties or assuming structure in the state to reduce resources. However, while such methods can improve the reconstruction efficiency, they often do not fully escape the exponential scaling in the worst case. Many compressed sensing and machine-learning approaches become inefficient or even more complex than standard tomography when extended to arbitrary mixed states. For example, algorithms tailored to low-rank states may fail or require many more measurements if the state is actually highly mixed (full rank). Thus, these methods excel in special scenarios (e.g. states that are pure or nearly pure), but a general solution for all states still demands further innovation.

1.1.1.2 Measurement Optimization Strategies

In parallel with better algorithms for reconstruction, significant research has focused on optimizing the choice of measurements to maximize information gain and minimize redundancy. Rather than using a fixed, brute-force set of measurements, these approaches ask: which measurements should we perform to get the most information about the state with the least effort? A notable example is measuring in Mutually Unbiased Bases (MUBs), which are pairs of bases where each basis state is an equal superposition of the states in the other basis [26]. It is known that performing projective measurements in a complete set of MUBs (when they exist for a given Hilbert space dimension) is very efficient for state estimation, as each

measurement outcome yields a maximal amount of new information [27]. In fact, for a d-dimensional system (such as a single qudit of dimension d), there exist $d+1$ MUBs (when d is a prime power) that together provide an informationally complete measurement set. Measuring in all $d+1$ MUBs gives a highly symmetric data set that minimizes overlap between measurements and thus maximizes information gain per measurement. This approach has strong theoretical justification and has been experimentally demonstrated for small systems (for instance, tomography of photonic qubits and qutrits using MUB settings). However, a limitation is that complete sets of MUBs are known to exist only for Hilbert spaces of prime or prime-power dimension [28, 29]. Multi-qubit systems have dimension 2^n (which is a prime power), so in principle, MUBs can be constructed; but implementing the joint measurements corresponding to MUBs for multiple qubits can be challenging in practice.

Another theoretically optimal scheme uses a symmetric informationally-complete positive operator-valued measure (SIC-POVM), which for a d-dimensional system consists of d^2 non-orthogonal measurement outcomes that are spread "as evenly as possible" in the state space [30]. A SIC-POVM is informationally complete and is minimal in the sense of having exactly d^2 outcomes (equal to the number of parameters of a density matrix). Recent work by Stricker et al. [31] demonstrated single-setting QST on a trapped-ion quantum processor using a SIC-POVM, effectively performing full tomography with one complex measurement configuration. In their experiment, they mapped the SIC-POVM to a sequence of standard operations and readouts, obtaining tomographically complete data from a single measurement basis and using a classical shadows approach to efficiently compute many properties of the state. This is a remarkable illustration that by clever choice of measurements, one can dramatically reduce the number of distinct settings required.

Apart from using special measurement bases, there are strategies that adapt or restrict the measurement process to reduce effort. Adaptive tomography techniques dynamically choose the next measurement based on previous measurement outcomes [32]. The idea is that once a rough estimate of the state is obtained from some data, subsequent measurements can be focused on the parts of the state space with the highest uncertainty, thereby extracting maximal information. Adaptive schemes (often Bayesian or machine-learning-assisted) have shown improvements in the number of measurements needed for a given estimation accuracy. There is also the concept of selective or partial tomography, where instead of fully characterizing the state, one targets specific features (for example, measuring only certain few-qubit reduced density matrices). The overlapping tomography approach is an example: one can reconstruct an N-qubit state by performing tomography on overlapping subsets of qubits (e.g. all k-qubit groups for some small k) and then combining these marginals to infer the global state. This reduces the complexity if the state has structure such as limited k-body correlations [33]. Experiments using overlapping tomography have demonstrated reconstruction of all two-qubit marginals of a multi-qubit state with fewer measurements than full tomography, focusing on local properties of the state.

Another notable measurement reduction strategy focuses on the fact that many quantum states of interest are nearly pure. Studies have investigated the minimal number of measurement settings needed to identify a pure state uniquely. In theory, a pure state in a d-dimensional Hilbert space is specified by $2d-2$ real parameters (much fewer than d^2-1 for a general mixed state). Protocols have been proposed to use as few as $2d-1$ or $2d-2$ measurements (instead of $d+1$ bases) to determine a pure state. In practice, however, such schemes may require adaptive measurements or entangled measurements that are difficult to implement, and they break down if the state is not exactly pure. In summary, recent methods in QST have made great strides by either improving the reconstruction algorithms or by optimizing the measurement procedure. Despite these advances, each method has limitations. Techniques that assume special structure (pure states, low entropy, etc.) are not applicable to general mixed states. Adaptive and overlapping methods can reduce redundancy, but they often add experimental or computational complexity and may not scale well to very large systems. Even optimal basis choices like MUBs or SIC-POVMs face practical constraints in implementation and are limited to certain system dimensions. And while brute-force optimization (e.g. using integer linear programming) can find minimal measurement sets for small cases, it becomes intractable for larger n due to the combinatorial explosion of possibilities. Therefore, there is a pressing need for new approaches that can bridge the gap - approaches that generalize to arbitrary states, scale to moderately large qubit numbers, and leverage all these insights (informational completeness, adaptive selection, etc.) in a more tractable optimization framework.

1.1.2 Limitations and the Need for New Approaches

Given the state of the art summarized above, it is clear that no single existing strategy completely overcomes the exponential complexity of QST. Each innovation offers only a piece of the solution. Post-processing improvements like compressed sensing require assumptions (e.g. low rank) that may not hold, and they can struggle when noise causes the true state to deviate from the assumed model. Measurement optimizations like MUBs or adaptive schemes can indeed reduce the number of settings or samples, but for a general unknown state one often still faces an exponential scaling of effort, albeit with a smaller base or improved constant factors. In practice, for medium-scale quantum systems, full tomography remains extremely challenging: the number of measurements needed for statistically reliable reconstruction can be astronomically large, and experiments must balance limited time and qubit coherence against the thoroughness of tomography. Furthermore, some sophisticated methods trade one type of complexity for another: for example, using a single SIC-POVM setting requires very complex measurement apparatus and classical post-processing of high-dimensional outcomes [34], while adaptive methods demand real-time classical computation and feedback between measurements. To push QST toward larger systems and practical applications, we need new approaches that can optimally allocate measurement resources and exploit all possible structure in the problem.

One promising direction is to view the design of QST experiments as a combina-

torial optimization problem that can be tackled with tools from computer science and graph theory. The problem of choosing an optimal set of measurements (or the optimal grouping of observables into measurement settings) has a discrete, combinatorial character: one must pick subsets of observables to measure together such that overall information is maximized and the number of measurements is minimized. This resembles problems in other domains like scheduling, resource allocation, and code design, hinting that algorithms from those domains could be repurposed for QST. In particular, as we elaborate next, the task of grouping quantum measurements can be mapped to a graph coloring problem. This mapping provides a fresh perspective and access to a rich toolkit of graph-theoretic algorithms and heuristics that have been successful in many hard optimization problems.

2

Theory

The full quantum state of a system is represented by a density matrix from measured data. In practice, one performs a set of measurements on identical prepared states to estimate the expectation values of a tomographically complete set of observables, and then uses those measurements to infer the density matrix of the state. This section reviews the formalism of density matrices and measurements, the Pauli operator basis for multi-qubit systems, and the concept of informational completeness, which together form the theoretical foundation of QST.

2.1 Quantum States and Density Matrices

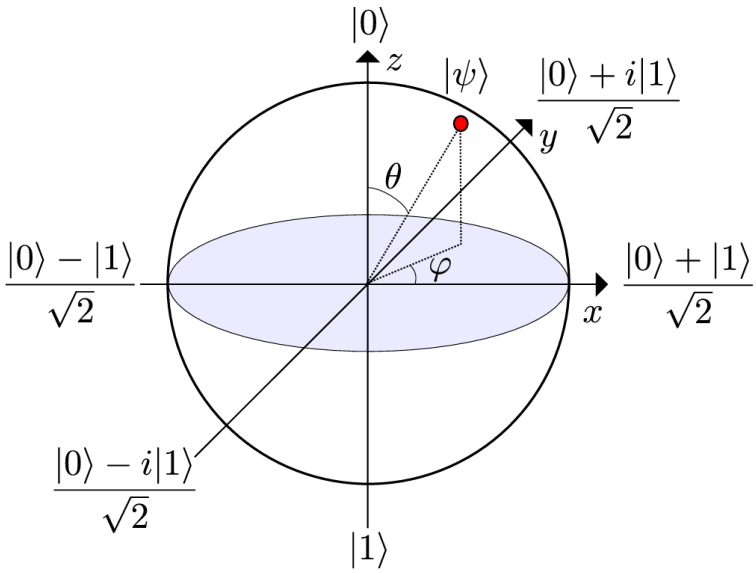


Figure 2.1: Bloch Sphere Representation of a Qubit state $|\psi\rangle$ [35].

A density matrix (or density operator) ρ is the mathematical representation of a quantum state, generalizing both pure states and classical probabilistic mixtures. By definition, ρ is a Hermitian operator, positive semi-definite with no negative eigenvalues, and has trace one. Such an operator can always be expressed in an eigenbasis as

$$\rho = \sum_j p_j \left|\psi_j\right\rangle\!\left\langle\psi_j\right| \quad \text{with} \quad p_j \geq 0,\ \sum_j p_j = 1, \quad \langle\psi_j \mid \psi_k\rangle = \delta_{jk}. \tag{2.1}$$

Any Hermitian, unit-trace, positive operator meeting these criteria corresponds to a physically valid quantum state. Any 2-level Qubit State can also be represented as a vector on the Bloch sphere as shown in Figure 2.1

2.1.1 Properties of density matrices:

- Hermiticity: $\rho^\dagger = \rho$. This guarantees that ρ has real eigenvalues and is an observable operator in quantum mechanics.

- Unit trace: $\mathrm{Tr}(\rho) = 1$, reflecting total probability 1 for the state. This condition fixes the normalization of the state.

- Positivity: $\rho \geq 0$, meaning $\langle\phi|\rho|\phi\rangle \geq 0$ for all $|\phi\rangle$. Equivalently, all eigenvalues p_j satisfy $p_j \geq 0$. Positivity ensures ρ can be interpreted as a probability-weighted mixture of pure states.

Example 2.1 (Single-qubit state reconstruction):
Consider a qubit prepared in the state $|0\rangle$ (the computational basis state). The density operator is

$$\rho = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \tag{2.2}$$

. To characterize ρ, one can measure the expectation values of the Pauli X, Y, and Z operators. In this state, $\mathrm{Tr}(\rho Z) = +1$ (since $|0\rangle$ is an eigenstate of Z with eigenvalue $+1$), and $\mathrm{Tr}(\rho X) = \mathrm{Tr}(\rho Y) = 0$ (since $|0\rangle$ is an equal superposition of ± 1 eigenstates of X or Y). From these three measurements, we determine the Bloch vector $\mathbf{r} = (0, 0, +1)$, implying $\rho = \frac{1}{2}(I + 1 \cdot Z)$ as expected. In general, a minimum of three independent measurements (in the X, Y, and Z bases) are needed to fully determine an arbitrary single-qubit density matrix, consistent with the fact that a 2×2 Hermitian matrix has three free real parameters.

2.1.2 Pauli Operators and the Multi-Qubit Pauli Basis

The Pauli matrices are a fundamental set of single-qubit operators that, together with the identity, form a basis for 2×2 Hermitian operators. They are typically denoted:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.3}$$

Each Pauli matrix is Hermitian and unitary, with eigenvalues ± 1. For a single qubit, I, X, Y, Z forms an orthogonal operator basis (with I typically excluded when considering the space of traceless deviations). This idea generalizes to multi-qubit systems: the N qubit Pauli group P_N is defined as the set of all N-fold tensor products of single-qubit Pauli matrices (including the identity on any subsystem). For example, for $N = 2$ qubits one element is $X \otimes Y$ (often written XY for short), and in general an element of P_N can be written

$$P_{i_1} \otimes P_{i_2} \otimes \cdots \otimes P_{i_N} \tag{2.4}$$

where each $P_{i_j} \in \{I, X, Y, Z\}$. For example, in a two-qubit system, a typical observable such as $X \otimes Z$ is expressed explicitly as seen in Equation 2.5

$$X \otimes Z = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \tag{2.5}$$

There are 4^N such tensor-product Pauli operators in total, which indeed form a basis for all $2^N \times 2^N$ matrices. Restricting to the subspace of traceless Hermitian operators (relevant for state reconstruction since the identity has fixed trace), there are $4^N - 1$ independent Pauli observables.

In particular, Pauli operators either *commute* or do not commute with each other. For instance, ZZ and ZI commute (they act on different qubits or identically on the qubit in common), whereas XZ and ZZ do not commute (since X and Z anticommute on the same qubit). Commutation relations of Pauli operators play a central role in designing efficient tomography schemes, as discussed later.

The Pauli basis is very convenient in QST because expectation values of Pauli operators correspond to various spin projections or correlation measurements of the quantum system. Any N-qubit density matrix ρ can be expanded in the Pauli basis as shown in Equation 2.6, with real coefficients c_P.

$$\rho \;=\; \frac{1}{2^N}\Big(I^{\otimes N} \;+\; \sum_{P \in \mathcal{P}_N} c_P \, P\Big), \quad c_P = \mathrm{Tr}(\rho P). \tag{2.6}$$

Thus, if one can experimentally determine $\mathrm{Tr}(\rho P)$ for each of a spanning set of Pauli operators, one can reconstruct ρ by inverting this linear expansion. In practice, measuring all $4^N - 1$ Pauli observables individually is inefficient for large N, but the Pauli basis remains a conceptual foundation for many tomography and state estimation techniques.

2.2 Informationally Complete Measurements

A set of measurement observables is said to be informationally complete (IC) for a quantum system if the data collected from measuring those observables is sufficient to represent the state in all its degrees of freedom. In other words, an IC set of observables provides a complete linearly independent set of equations for the unknown state. Mathematically, if we treat the unknown density matrix ρ as a vector in a $(d^2 - 1)$-dimensional space (for a d-dimensional Hilbert space) and likewise vectorize the measurement outcomes, informational completeness means the measurement operator vectors span that space. One way to verify informational completeness is by checking the rank of the sensing matrix (or design matrix) that relates the state parameters to the measured expectations: for an IC measurement set, this matrix has full column rank (equal to $d^2 - 1$ for a single state parameter space). If the rank is deficient, there is at least one direction in state space that the measurements

cannot distinguish, meaning the measurements are not fully informative.

For example, measuring only the Z observable on a single qubit is not informationally complete, because it provides no information about the X or Y components of the Bloch vector (many different ρ yield the same Z expectation). By contrast, measuring X, Y, and Z for a qubit is an IC set, since together these three measurements determine the full Bloch vector. In general, for N qubits one simple IC scheme is to measure each qubit in the X, Y, and Z bases (i.e., perform all 3^N combinations of local Pauli measurements). This yields 3^N distinct measurement settings, which is sufficient to determine all 4^N-1 parameters of an N-qubit density matrix. Although a general N-qubit density matrix has 4^N-1 real parameters (after enforcing unit trace), each local measurement basis of the form $\{I, R_x, R_y\}^{\otimes N}$ yields 2^N outcome probabilities but only 2^N-1 of these are independent. Since there are 3^N such bases, the total number of independent data points is $3^N \times (2^N-1) = 6^N - 3^N$, which for all $N \geq 1$ exceeds 4^N-1. Thus the resulting linear system relating measurement outcomes to density-matrix entries is overdetermined (and generically full rank), guaranteeing a unique reconstruction of all 4^N-1 parameters from only 3^N measurement settings. As this scales exponentially (represented in Figure 2.2), it is necessary to find a way to perform tomographically complete measurements in an optimal way. This is possible when multi-qubit (global) measurements are allowed, as we will see.

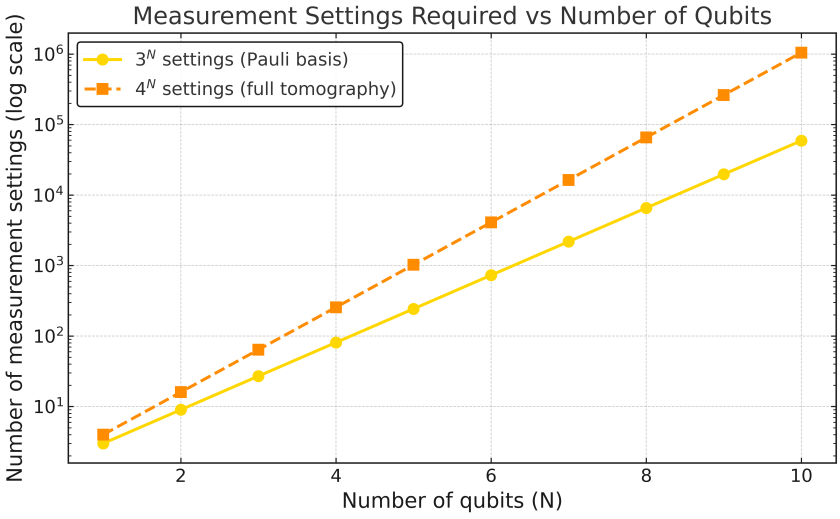


Figure 2.2: Exponential Growth of Measurement Settings in QST.

Example 2.2 (Informational completeness check):
Suppose we have a two-qubit system and we choose as our observables the set $X \otimes X, ; Y \otimes Y, ; Z \otimes Z$. We have to check if this measurement is informationally complete. Each of these observables is entangled across the two qubits (global Pauli operators), and they certainly provide some information about two-qubit correlations. However, this set is not IC, because it consists of only 3 measurements, whereas a general two-qubit state has 15 independent parameters. Indeed, one can

confirm that the $[3 \times 15]$ sensing matrix (mapping the 15-dimensional Bloch vector of ρ to these 3 measured expectations) has rank at most 3, far from full rank 15. By contrast, the complete set of 15 Pauli observables $[X_1, Y_1, Z_1, X_1X_2, X_1Y_2, \ldots, Z_1Z_2]$ is informationally complete (in fact over-complete). The central challenge for practical QST is to choose a minimal such IC set and an efficient way to measure it.

2.2.1 Problem Formulation

The goal of optimized tomography is to find the optimal (minimum) number of measurement settings or quantum circuits that still yield full information. Since many of these observables in the IC set can be measured simultaneously on a suitable basis (by making use of the commutative property of operators), the number of measurement settings required can be reduced. We therefore translate our motivation to find a smarter way to perform QST into a nontrivial partitioning problem, which results in a *combinatorial optimization* challenge. In Chapter 3 and beyond, we recast this task of grouping commuting Pauli operators into common measurement settings as an equivalent graph coloring problem to find an optimal (minimal) set of measurements.

2.3 Graph Theory and Graph Coloring

The task of optimizing QST measurements can be translated into a problem in graph theory. In this section, we introduce basic graph-theoretic concepts [36] and the graph coloring problem, then explain how the commutation relationships of Pauli operators can be mapped to a graph. We also overview several graph coloring algorithms and heuristics that will be used to solve the measurement optimization problem in later chapters.

2.3.1 Graph Basics and Terminology

In discrete mathematics, a graph $G = (V, E)$ consists of a set of vertices V (also called nodes) and a set of edges E, where each edge is an unordered pair of vertices in V. If an edge $E = v_1, v_2$ connects vertices v_1 and v_2, we say v_1 and v_2 are adjacent (neighbors) and that the edge is incident on those vertices. The degree of a vertex is the number of edges incident to it (i.e. the count of its neighbors). A graph is often represented visually by dots (vertices) connected by lines (edges) when adjacency exists, as seen in Figure 2.3.

2.3.2 Graph Coloring and Chromatic Number

A graph coloring is an assignment of labels (traditionally called "colors") to the vertices of a graph such that no two adjacent vertices share the same color. This is also known as vertex coloring [37]. The fundamental optimization question is how to minimize the number of colors used while maintaining a proper coloring. The smallest number of colors needed to color a graph G is called the chromatic number of G, denoted $\chi(G)$. Determining $\chi(G)$ for an arbitrary graph is a famous NP-hard

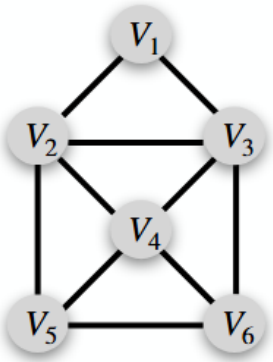


Figure 2.3: Representation of an Arbitrary Graph Construction

problem – in fact, the decision version ("can G be colored with k colors?") is NP-complete (one of Karp's original 21 NP-complete problems). This means that no polynomial-time algorithm is known for graph coloring in general, and the problem is believed to be intractable for large graphs. Consequently, practical approaches rely on clever algorithms and heuristics to find colorings that are optimal or near-optimal for a given graph.

Graph coloring has numerous applications in scheduling, register allocation, frequency assignment, and, as in this thesis, in grouping compatible quantum measurements [38]. A coloring of a graph can be viewed as a partition of the vertex set into disjoint color classes, where each color class is an independent set (no edges within a class). In the context of QST, a color class will correspond to a set of observables that can be measured together without conflict. We next make this correspondence precise by constructing the Pauli graph for a set of operators.

3

Methods

In this chapter, we detail the methodology and implementation of our approach to optimize and schedule Quantum State Tomography (QST) experiments using graph coloring heuristics. We first describe how the problem is formulated as a graph problem by constructing a Pauli graph from a given set of Pauli operators. We then present the graph coloring methods explored Degree of Saturation (DSATUR), Recursive Largest First (RLF), Graph Neural Network (GNN), Spectral Clustering, and Integer Linear Programming (ILP) based exact coloring, each in its own section with a description of the algorithm, pseudocode, implementation details, and complexity analysis. Finally, we summarize the overall implementation architecture, describing how experiments are organized. The results and comparative performance of these methods are deferred to Chapter 4.

3.1 Constructing the Pauli Graph for QST

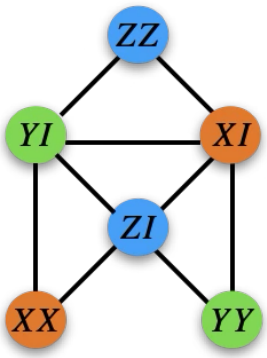


Figure 3.1: Graph Coloring Representation for an arbitrary Graph.

The first step is to model the QST experiment optimization problem as a graph coloring problem. Let $P = \{I, X, Y, Z\}^{\otimes N}$ be the set of all 4^N Pauli strings on N qubits (including the identity operator $I^{\otimes N}$). In our context, we exclude the global identity, so the total number of non-trivial Pauli observables is $|V| = 4^N - 1$. Each vertex $v_i \in V$ of the graph $G_P = (V, E)$ represents a unique Pauli operator $P_i \in P$. We then add an edge between two vertices whenever the corresponding Pauli operators do not commute, as stated in Chapter 3. In other words, for vertices v_i, v_j representing Pauli operators $P_i, P_j \in P$, the edge $e_{ij} \in E$ is included if and only if $[P_i, P_j] \neq 0$. Equivalently, the graph has edges between every pair of non-commuting

Pauli operators. The adjacency matrix A of this graph can be defined as mention in Equation 3.1

$$A_{ij}(G_P) = \begin{cases} 1, & \text{if } [P_i, P_j] \neq 0 \ , \\ 0, & \text{if } [P_i, P_j] = 0 \ , \end{cases} \tag{3.1}$$

Here, a 1 entry indicates a pair of Pauli operators that do not commute (thus an edge in the graph), whereas 0 indicates a commuting pair with no edge. In this construction, each color class in the graph will correspond to a set of mutually commuting operators (since no two vertices in the same color class share an edge). By coloring the graph, we effectively partition the Pauli observables into disjoint commuting sets, each of which can be measured in a single experimental setting. In Figure 3.1 we represent $G = (V, E)$ where $\chi(G)$ is 3, meaning all nodes can be partitioned into 3 groups based on the imposed condition. For higher numbers of qubits, G_P grows exponentially ($|V| = 4^N - 1$ vertices), and finding $\chi(G_P)$ is computationally challenging (NP-hard in general).

In our implementation, we leverage the NetworkX library's 'Graph' data structure to store vertices and edges for the Pauli graph. Pseudocode for constructing the graph is straightforward: iterate over all pairs of Pauli operators and check the commutation relation for each pair, adding an edge if they are non-commuting. This approach runs in $O(n^2)$ time for $n = |V|$ vertices. The subsequent sections discuss the algorithms and heuristics we employ to color these graphs efficiently.

3.2 Heuristic Algorithms (DSATUR and RLF)

Efficient graph coloring algorithms are critical for solving combinatorial problems across various practical domains. Among these methods, the *Degree of Saturation* (DSATUR)[39, 40, 41] and *Recursive Largest First* (RLF)[42, 43, 44] algorithms stand out due to their simplicity, effectiveness, and polynomial runtime complexities. Specifically, DSATUR operates with a time complexity of $O(n^2)$, and RLF has a complexity of $O(n^3)$, where n denotes the number of vertices in the graph.

3.2.1 Degree of Saturation - DSATUR

DSATUR, introduced by Brélaz [39], strategically orders vertices by prioritizing those with the highest saturation degree, defined as the number of distinct colors assigned to a vertex's adjacent vertices. The method begins with all vertices uncolored. At each step, the algorithm selects the uncolored vertex with the highest saturation degree for coloring. If multiple vertices share the highest saturation degree, the algorithm selects the one with the highest conventional degree (the largest number of adjacent vertices). After vertex selection, the smallest available color not assigned to neighboring vertices is applied. This iterative selection strategy tends to reduce the number of colors required, making DSATUR particularly effective for achieving near-optimal solutions on moderately large or structured graphs.
The pseudocode for the DSATUR algorithm is presented in Algorithm 1.

Algorithm 1 DSATUR Graph Coloring Algorithm

```
Initialize all vertices as uncolored
while there are uncolored vertices do
    Select the uncolored vertex with the highest saturation degree
    if multiple vertices have the same saturation degree then
        Select the one with the highest degree (most neighbors)
    end if
    Assign the smallest available color not used by adjacent colored vertices
end while
```

3.2.2 Recursive Largest First - RLF

In contrast, the Recursive Largest First (RLF) algorithm, introduced by Leighton [42], builds color classes iteratively, focusing initially on vertices of maximal degree. The approach starts similarly by considering all vertices as uncolored. It selects the uncolored vertex with the highest degree and initiates a new color class. Vertices that do not conflict (i.e., are not adjacent to any vertex already in the current color class) are subsequently added to this color class. Once no more vertices can be safely added to the current class without violating the commutation condition, a new color is assigned, and these vertices are removed from consideration. This procedure continues recursively until every vertex is colored.
The pseudocode for the RLF algorithm is provided in Algorithm 2.

Algorithm 2 RLF Graph Coloring Algorithm

```
Initialize all vertices as uncolored
while there are uncolored vertices do
    Select the uncolored vertex with the highest degree
    Initialize a new color class with this vertex
    for all uncolored vertices not adjacent to any vertex in the current color class do
        Add them to the current color class
    end for
    Assign a new color to all vertices in the current color class
    Remove vertices in current color class from the graph
end while
```

3.2.3 Application and Illustration of DSATUR and RLF

To contextualize these algorithms within the domain of quantum computing, we consider their application to the commutation graph derived from a subset of 2-qubit Pauli operators. Specifically, we analyze the set $\mathcal{P}_1 = \{XI, YI, ZI, XX, YY, ZZ\}$. In this setting, each Pauli operator is represented by a vertex in the graph, and edges connect vertices representing non-commuting operators. This graphical representation helps visualize the combinatorial constraints inherent in quantum state tomography, highlighting operator compatibility and conflicts during measurement.

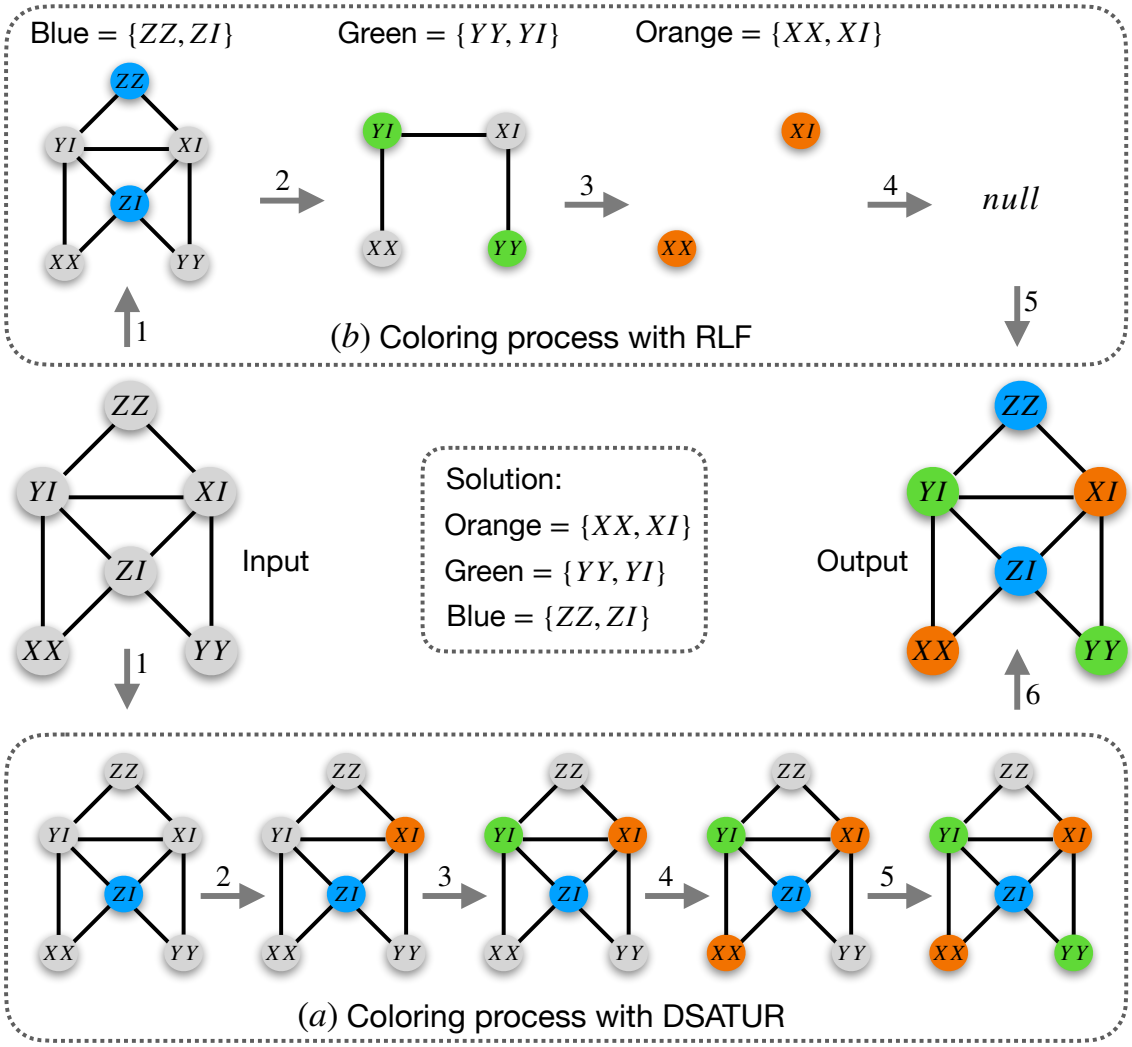


Figure 3.2: Step-by-step graph-coloring of the two-qubit Pauli commutation graph using (a) DSATUR and (b) RLF heuristics. Vertices represent Pauli operators, and edges link non-commuting pairs. Colors correspond to joint measurement groups—Orange = XX, XI, Green = YY, YI, Blue = ZZ, ZI. In (a), DSATUR iteratively selects the uncolored vertex with the highest saturation degree, breaking ties by vertex degree; in (b), RLF builds each color class by repeatedly adding vertices with no conflicts to the largest available class.

Figure 3.2 illustrates the step-by-step graph coloring process using both DSATUR [panel (a)] and RLF [panel (b)] algorithms for the set $\mathcal{P}_1$. Both algorithms iteratively assign colors based on their respective strategies, ultimately partitioning the graph into distinct commuting operator groups. The illustrative example clearly shows the iterative process and decision-making differences between the two heuristics. Notably, despite their methodological differences, both DSATUR and RLF yield identical optimal coloring solutions for this small example, resulting in three distinct commuting sets:

$$\begin{aligned} \textbf{Orange} &= \{XX, XI\}, \\ \textbf{Green} &= \{YY, YI\}, \\ \textbf{Blue} &= \{ZZ, ZI\}. \end{aligned}$$

This illustrative example underscores the effectiveness of these heuristic methods, especially in domains like quantum computing, where minimizing measurement resources through optimal operator grouping is critically important.

3.2.4 Complexity and Practical Implications

The polynomial complexities ($O(n^2)$ for DSATUR and $O(n^3)$ for RLF) make these algorithms suitable for moderate-size problems, frequently encountered in practical scenarios such as combinatorial scheduling, frequency assignment in wireless communication networks, and register allocation in compiler design. Despite their polynomial nature, heuristic decisions in both algorithms imply they do not guarantee global optimality. Nonetheless, extensive empirical evaluations suggest that DSATUR often provides close-to-optimal results quickly, while RLF, though computationally slightly more demanding, typically provides equally robust solutions for dense and structured graphs.

In conclusion, heuristic graph coloring algorithms like DSATUR and RLF offer robust, practical solutions for combinatorial optimization challenges. Their adaptability to diverse problem structures and proven effectiveness in quantum computing contexts demonstrate their broad applicability and continuing relevance in contemporary computational research.

3.3 Integer Linear Programming (ILP)

While heuristic methods such as DSATUR and RLF are widely used due to their computational efficiency, they do not guarantee optimal solutions. In contrast, Integer Linear Programming (ILP) explicitly formulates graph coloring as a combinatorial optimization problem, capable of yielding provably optimal solutions. Although ILP approaches have exponential complexity in the worst-case scenarios, making them computationally infeasible for large graphs, their ability to provide exact solutions is invaluable for verifying and benchmarking heuristic results on smaller-scale problems [45, 7].

3.3.1 ILP Formulation for Graph Coloring

The ILP formulation of the graph coloring problem encodes constraints and objectives explicitly into an integer linear program. The primary objective is to minimize the number of colors used, ensuring each vertex is assigned precisely one color, and no two adjacent vertices share the same color.

Formally, consider a graph $\mathcal{G}_\mathcal{P} = (\mathcal{V}, \mathcal{E})$, where vertices $\mathcal{V}$ represent the elements (in our case, Pauli operators), and edges $\mathcal{E}$ represent non-commutation constraints. The ILP algorithm explicitly introduces two sets of binary decision variables, $x_{v,j}$ and y_j, defined in Equation 3.2

$$x_{v,j} = \begin{cases} 1 & \text{if vertex } v \text{ is assigned color } c_j, \\ 0 & \text{otherwise,} \end{cases} \qquad y_j = \begin{cases} 1 & \text{if color } c_j \text{ is used,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.2}$$

We define the coloring function in Equation 3.3

$$\chi : \mathcal{V} \to \{c_1, c_2, \ldots, c_k\}, \tag{3.3}$$

where $\chi(v) = c_j$ denotes vertex v is assigned color c_j. Initially, we set $k = |\mathcal{V}|$, an upper bound representing the worst-case scenario where each vertex is assigned a unique color and minimize $\sum_{j=1}^{k} y_j$. We enforce $\sum_j x_{v,j} = 1$ so each vertex is in exactly one group, and $x_{v,j} \leq y_j$ so that activating any $x_{v,j}$ forces $y_j = 1$. For every non-commuting pair (u, v), the constraint $x_{u,j} + x_{v,j} \leq 1$ forbids them sharing group j. Algorithm 3 outlines the step-by-step ILP-based graph coloring procedure.

Algorithm 3 ILP algorithm for Graph coloring

Require: Graph $\mathcal{G}_\mathcal{P} = (\mathcal{V}, \mathcal{E})$
Ensure: Optimal coloring $\chi : \mathcal{V} \to \{c_1, c_2, \ldots, c_k\}$
1: Set $k \leftarrow |\mathcal{V}|$
2: Define binary variables:

$$x_{v,j}, y_j \in \{0, 1\} \quad \forall v \in \mathcal{V},\ j = 1, \ldots, k$$

3: Objective: $\min \sum_{j=1}^{k} y_j$
4: Constraints:
- $\sum_{j=1}^{k} x_{v,j} = 1 \quad \forall v \in \mathcal{V}$
- $x_{v,j} \leq y_j \quad \forall v \in \mathcal{V},\ j = 1, \ldots, k$
- $x_{u,j} + x_{v,j} \leq 1 \quad \forall e_{uv} \in \mathcal{E},\ j = 1, \ldots, k$

5: Solve ILP using CBC solver
6: Assign $\chi(v) = c_j$ if $x_{v,j} = 1$
7: **return** χ

Practically, this ILP formulation is implemented using optimization solvers such as Coin-or Branch and Cut (CBC), Gurobi, or CPLEX. These solvers employ advanced branch-and-bound techniques, cuts, to efficiently explore the search space.

3.3.2 Computational Complexity and Optimality

Graph coloring is a classic NP-hard problem [46], indicating that finding exact solutions through ILP generally entails exponential complexity. The worst-case runtime arises from the exponential growth of the branch-and-bound search tree, theoretically up to $O(2^n)$, where n is the number of vertices. However, practical performance depends strongly on graph structure. Modern ILP solvers can efficiently solve graph coloring problems containing up to a few hundred vertices, provided the graphs are sparse or exhibit special structure (e.g., low maximum vertex degree, clustering, or symmetry). For dense graphs or those with arbitrary structure, exact solutions quickly become computationally intractable.

3.3.3 ILP in Quantum Computing Context

Within quantum state tomography (QST), graph coloring methods are pivotal for partitioning Pauli measurement operators into minimal commuting groups, directly reducing measurement resources. ILP's capability to deliver guaranteed optimal partitions for small qubit systems makes it invaluable for benchmarking heuristic approaches. By confirming optimal partitions for modest qubit systems [7], ILP provides rigorous confidence that heuristic methods (like DSATUR and RLF) produce solutions close to optimal in larger-scale, practically relevant quantum problems. Additionally, ILP solutions often serve as valuable reference points to calibrate heuristic performance, thus enhancing confidence in heuristic results for problem sizes where ILP becomes computationally infeasible.

3.4 Graph Neural Network (GNN)

Heuristic methods like DSATUR and RLF, as well as exact methods like ILP, are well-established classical approaches to the graph coloring problem. While heuristics are computationally efficient, they don't always guarantee optimal solutions, and exact methods, though precise, can quickly become computationally infeasible. Graph Neural Networks (GNNs), a modern class of machine learning models, offer an alternative, leveraging learning-based methods to predict graph colorings efficiently by recognizing structural patterns within graph data [47, 48, 49]. Here, we explore the application of a GNN to graph coloring, specifically framing the problem as a supervised node classification task.

3.4.1 Formulation of the GNN Approach

In our GNN-based framework, the coloring task is treated as a supervised learning problem. Given a graph $\mathcal{G}_\mathcal{P} = (\mathcal{V}, \mathcal{E})$, each vertex $v \in \mathcal{V}$ is considered as a data point whose "label" corresponds to its assigned color. The graph structure itself provides rich relational information through adjacency (commutation relations in the quantum tomography setting). By training the neural network on graphs with known optimal or near-optimal colorings, the GNN learns generalized patterns that capture both local and global graph structures. Once trained, the model can rapidly

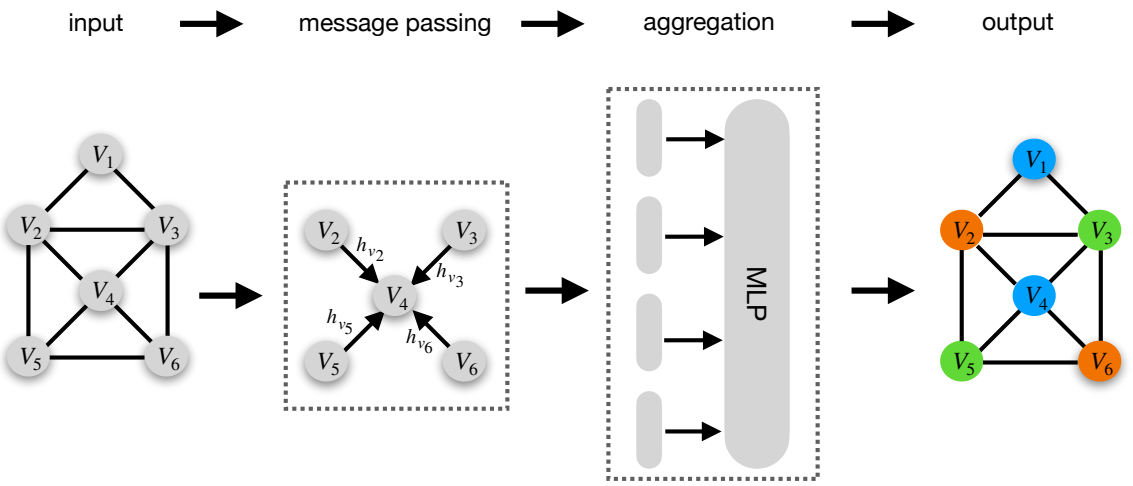


Figure 3.3: Graph Neural Network Process Flow Chart: The uncolored, structured graph is the input for the GNN. The central node gathers Feature Vectors and is then aggregated to update the node's current feature. The logits from the final layer of GNN are transformed into scores to color the input graph.

infer colors for new graph instances via a single forward pass, as represented in Figure 3.3 thereby significantly enhancing computational efficiency. Algorithm 4 summarizes the GNN approach in detail, highlighting each stage of the process. We use one-hot encoding for the feature vector h_v as represented by Equation 3.4

$$\begin{aligned} I \mapsto (1,0,0,0), \quad X \mapsto (0,1,0,0), \quad Y \mapsto (0,0,1,0), \quad Z \mapsto (0,0,0,1), \\ \text{Example}: X \otimes I \otimes Z \ (n=3) \mapsto h_v = [\,0,1,0,0 \,|\, 1,0,0,0 \,|\, 0,0,0,1\,]. \end{aligned} \tag{3.4}$$

3.4.1.1 Message Passing and Feature Aggregation

At the heart of the GNN-based approach is the message-passing mechanism. We represent each node and its degree in the graph using a one-hot encoding, which is a binary feature vector. The core of the algorithm is the message-passing stage, repeated for a fixed number of layers (l). In each layer, each node v aggregates information from its neighbors. Formally, the embedding of node v at layer l, denoted as $\mathbf{h}_v^{(l)}$, is computed by using Equation 3.5

$$\mathbf{h}_v^{(l)} = f_{\text{GNN}}\left(\mathbf{h}_v^{(l-1)}, \text{AGGREGATE}\left(\{\mathbf{h}_u^{(l-1)} : u \in \mathcal{N}(v)\}\right)\right), \tag{3.5}$$

where $\mathcal{N}(v)$ represents neighbors of node v, and the function f_{GNN} is typically a small neural network consisting of linear transformations followed by nonlinear activation. The aggregation step commonly involves averaging neighbor embeddings as seen in Equation 3.6

$$\text{AGGREGATE}\left(\{\mathbf{h}_u^{(l-1)} : u \in \mathcal{N}(v)\}\right) = \frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} \mathbf{h}_u^{(l-1)}. \tag{3.6}$$

This aggregation enables the network to progressively integrate local neighborhood information, effectively encoding complex graph structures into each node's representation.

Algorithm 4 GNN Based Coloring

Input: $\mathcal{G}_{\mathcal{P}}$, pretrained GNN with l layers
Output: A color label for each node
Initialize node features
    For each node, set its feature to a simple encoding (one-hot encoding)
Message passing (repeat for each of the l layers)
    For each node:
        Collect the features of all neighboring nodes
        Combine those neighbor features (averaging)
        Update the node's feature by feeding its previous feature and the combined neighbor features into the GNN layer
Color prediction
    For each node:
        Apply the GNN's final output layer to its feature to produce a score for each possible color
        Choose the color with the highest score and assign it to the node
Return the set of node to color assignments
Notes:
    Each GNN layer is typically a small neural network block (linear layer + activation).
    The final output layer maps node features to a score for each color.

3.4.1.2 Color Prediction and Inference

After completing l layers of message passing, the final node embeddings $\mathbf{h}_v^{(l)}$ capture both local and broader structural information. To predict the node's color, a final linear classifier transforms these embeddings into a vector of color scores with the help of Equation 3.7

$$\mathbf{s}_v = \mathbf{W}\mathbf{h}_v^{(l)} + \mathbf{b}, \tag{3.7}$$

where $\mathbf{W}$ is a learned weight matrix, and $\mathbf{b}$ is a bias vector. Each component $s_{v,i}$ represents the neural network's confidence that node v should be assigned color i. The color with the highest score is selected, providing the final node classification result.

3.4.2 Training Methodology

The GNN is trained in a supervised manner using labeled training sets of smaller graphs whose optimal or near-optimal colorings are known (from ILP or advanced heuristic algorithms). Training involves minimizing a loss function (e.g., cross-entropy loss) computed between predicted and actual colors. The network parameters (weights and biases) are adjusted iteratively using gradient descent, allowing the model to generalize from the training examples to unseen graphs effectively.

3.4.3 Computational Complexity and Practical Advantages

The inference complexity of the GNN approach is $O(l\,(|\mathcal{V}| + |\mathcal{E}|))$, primarily determined by the number of layers l and graph size (number of vertices $\mathcal{V}$ and edges $\mathcal{E}$). This linear complexity in graph size makes GNN inference highly efficient, enabling real-time color predictions for large-scale graphs once the model is trained.
However, the main computational bottleneck is the initial training phase, which requires substantial time for gradient-based optimization, especially with extensive training datasets. Despite this, once trained, the GNN can rapidly generalize to new problem instances.

3.4.4 Application in Quantum State Tomography

Applying this method to quantum tomography, the GNN efficiently partitions large sets of Pauli operators into commuting subsets by learning structural commutation patterns. Consequently, the GNN approach directly facilitates rapid measurement scheduling, an essential feature in real-time experimental setups (for pre-trained models).
In summary, Graph Neural Networks provide a compelling approach to solving combinational graph-coloring problems by efficiently leveraging learned structural information. Their capability for rapid inference, combined with high-quality predictive accuracy after training, makes them a valuable addition to classical heuristic and exact methods, particularly in dynamically evolving or time-sensitive applications like quantum state tomography scheduling.

3.5 Spectral Clustering

While heuristic methods, ILP, and Graph Neural Networks (GNNs) each provide robust approaches to graph coloring, Spectral Clustering offers a distinct perspective leveraging spectral graph theory. Unlike traditional combinational heuristics and optimization-based solutions, spectral clustering utilizes eigen-decompositions of specialized graph matrices, capturing global structural properties of the graph to effectively identify highly connected communities or clusters [50, 51, 52].
Spectral clustering is particularly useful in quantum state tomography (QST), where the underlying problem reduces to partitioning observables (Pauli operators) into groups based on commutation. By interpreting commutation relations as a similarity metric, spectral clustering naturally aligns with the problem of identifying maximal sets of commuting observables.

3.5.1 Mathematical Formulation of Spectral Clustering

Given a graph $\mathcal{G}_{\mathcal{P}} = (\mathcal{V}, \mathcal{E})$, vertices represent observables, and edges represent pairs that do not commute. For spectral clustering, we invert this logic by building an affinity (commutation) graph where edges represent pairs of commuting observables. This transforms the original problem into a clustering task: nodes connected by edges form potential commuting sets. We construct the affinity matrix W on the condition seen in Equation 3.8. W encodes commutation directly as similarity. The spectral clustering approach involves several key steps, clearly summarized in Algorithm 5.

$$W_{ij} = \begin{cases} 1 & \text{if observables } i, j \text{ commute (no edge in conflict graph),} \\ 0 & \text{otherwise, including } i = j. \end{cases} \tag{3.8}$$

The normalized Laplacian matrix L is fundamental in spectral clustering. It can be interpreted as a discrete approximation of a continuous differential operator, measuring the "smoothness" of functions defined over the graph. Eigenvectors corresponding to small eigenvalues of L are associated with partitions that cut relatively few edges. Nodes within a cluster have similar values in the spectral embedding, whereas nodes in different clusters exhibit significant differences. Thus, eigenvectors provide meaningful, low-dimensional representations of nodes, efficiently highlighting intrinsic commutation structures. We then perform an eigen-decomposition of L, extracting its k smallest eigenvalues and corresponding eigenvectors. Stacking those k eigenvectors column wise into a matrix $\mathcal{M}$ produces an embedding, where each row of $\mathcal{M}$ is a k-dimensional vector representing the node's projection. To ensure these embeddings are comparable, we normalize each row of $\mathcal{M}$ to unit length. Finally, we apply k-means clustering to these normalized row vectors: nodes whose embeddings lie close together in this spectral space are grouped into the same cluster. Because the affinity matrix was built from commutation relations, each resulting cluster corresponds to a set of mutually commuting observables and thus to one joint measurement setting in the tomography schedule.

Algorithm 5 Spectral Clustering via Affinity Graph

Require: Graph $\mathcal{G}_{\mathcal{P}}$ where edges link non-commuting observables, desired cluster count k

Ensure: A cluster label for each node (defines commuting measurement groups)

Build the affinity (commutation) matrix
  For each pair of distinct nodes i, j:
    If $\{i, j\} \notin \mathcal{E}$, set $W_{ij} = 1$ (they commute), else $W_{ij} = 0$
  Set diagonal entries $W_{ii} = 0$

Compute degrees
  For each node i, let $\deg_i = \sum_j W_{ij}$

Form normalized Laplacian
  Construct D as the diagonal matrix of degrees
  Compute $L = I - D^{-1/2}\, W\, D^{-1/2}$

Spectral embedding
  Compute the k eigenvectors of L corresponding to its k smallest eigenvalues
  Stack these eigenvectors column-wise into an $|\mathcal{V}| \times k$ matrix $\mathcal{M}$

Row normalization
  For each row i of $\mathcal{M}$, perform normalization so it has length 1

Cluster assignment
  Run k-means on the rows of $\mathcal{M}$ to assign each node to one of k clusters

return The cluster labels for all nodes

Table 3.1: Affinity Matrix W for Arbitrary Pauli Operators

	XI	YI	ZI	XX	YY	ZZ
XI	0	0	0	1	0	0
YI	0	0	0	0	1	0
ZI	0	0	0	0	0	1
XX	1	0	0	0	1	1
YY	0	1	0	1	0	1
ZZ	0	0	1	1	1	0

3.5.2 Application with Example

We start with the six observables: XI, YI, ZI, XX, YY, ZZ. Each string represents a Pauli operator on qubit 1 and qubit 2, respectively. We proceed with building the Affinity Matrix W as described earlier and construct a 6×6 matrix W as shown in Table 3.1. The table displays a 1 wherever a commuting pair is found. We then compute the Normalized Graph Laplacian L by calculating the degree of each node: $\deg_i = \sum_j W_{ij}$ and form $D^{-1/2}$ from the degrees. L is computed using Equation 3.9. This captures the smoothness of functions on the graph.

$$L = I - D^{-1/2} W D^{-1/2}. \tag{3.9}$$

For Spectral Embedding, we perform eigen-decomposition of L, yielding eigenvalues $\lambda_1 \leq \lambda_2 \leq \ldots$ and corresponding eigenvectors. We take the first $k = 3$ eigenvectors (those with smallest eigenvalues) and stack them as columns to form a 6×3 matrix and normalize each row to unit length; this gives a point-cloud representation of the nodes in $\mathbb{R}^3$. Finally, we perform clustering in the Embedding Space using k-means on the 6 row-vectors of the normalized embedding to obtain the following result:

$$\begin{aligned} \textbf{Orange} &= \{XX, XI\}, \\ \textbf{Green} &= \{YY, YI\}, \\ \textbf{Blue} &= \{ZZ, ZI\}. \end{aligned}$$

In conclusion, spectral clustering offers a valuable analytical heuristic, particularly suited for identifying commuting measurement groups in quantum tomography applications. Its integration within a comprehensive tomography optimization framework complements classical and modern computational techniques, providing robust solutions across diverse graph structures.

4

Results

Once sets of mutually commuting Pauli operators are obtained from the graph coloring heuristics, the next step is to convert these groups into an executable quantum state tomography protocol. The goal is to design a measurement scheme (quantum circuit) spanning the full state space with few measurement quantum circuits while remaining numerically stable. The first part of this section details how each commuting group is used to construct measurement bases (unitary matrices), how the overall sensing matrix A is built and verified, and how state reconstruction is tested for accuracy and noise robustness. The procedure is validated through simulations: we reconstruct known test states and evaluate fidelity, error, and stability under noise.

In the second part of this chapter, we evaluate the protocol for Quantum State Tomography (QST) experiments. Each partitioning method is applied to Pauli graphs derived from the full N-qubit Pauli operator sets for $N = 2$ up to $N = 5$. We compare the methods along several dimensions: the reduction in required measurement circuits (relative to the naive baseline of 3^N circuits and to the theoretical optimum), the computational runtime of each algorithm, and the stability of the tomography reconstruction (conditioning and noise robustness). Finally, we discuss the practical significance of these results for QST on NISQ-era devices, including an analysis of how incremental addition of measurement groups impacts reconstruction fidelity. The following sections present detailed results, with visualizations and tables to illustrate key findings.

4.1 Quantum State Tomography Protocol

Our protocol for QST fundamentally relies on measuring sets of commuting observables, represented by Pauli operators $\mathcal{P} = \{P^{(1)}, P^{(2)}, \ldots, P^{(m)}\}$, acting on an n-qubit Hilbert space. When these operators commute, it implies the existence of a shared eigenbasis. Hence, a single unitary matrix U can simultaneously diagonalize all operators in $\mathcal{P}$, facilitating efficient measurement.

4.1.1 Unitary Matrices

We begin by diagonalizing the first operator $P^{(1)}$, yielding its eigenvectors $\{|\phi_i^{(1)}\rangle\}$ and eigenvalues $\{\lambda_i^{(1)}\}$. These eigenvectors constitute an initial basis, encapsulated in a unitary matrix $V^{(1)}$. Subsequent operators are then iteratively diagonalized

within the subspaces defined by previously diagonalized operators. This iterative procedure refines the basis step-by-step, maintaining numerical stability via regular orthonormalization as shown in Algorithm 6. Upon completion, the final matrix $U = V$ simultaneously diagonalizes the operators, satisfying numerical checks as in Equation 4.1

$$\|U^\dagger U - I\|, \quad U^\dagger P^{(k)} U \text{ is diagonal for each } k. \tag{4.1}$$

Algorithm 6 Simultaneous Eigenbasis Computation

Require: Commuting Pauli operators $\{P_1, P_2, \ldots, P_m\}$
Ensure: Unitary V simultaneously diagonalizing all P_k
1: Compute eigenvectors $\{\phi_i^{(1)}\}$ of P_1, assemble into V
2: Orthonormalize V via QR decomposition
3: **for** $k = 2, \ldots, m$ **do**
4: Project P_k into basis V, obtaining block diagonal structure
5: **for** each degenerate subspace (block) **do**
6: Diagonalize within subspace, updating V
7: **end for**
8: Orthonormalize V again using QR
9: **end for**
10: Verify unitarity and diagonalization accuracy numerically
11: **return** V

4.1.2 Sensing Matrix

After identifying measurement settings through simultaneous diagonalization, we construct the sensing matrix A. Each row of A is formed by vectorizing projectors $\Pi_\ell^{(k)} = |\psi_\ell^{(k)}\rangle\langle\psi_\ell^{(k)}|$, associated with each measurement basis defined by U, specified in Equation 4.2

$$A = \begin{pmatrix} \mathrm{vec}(\Pi_1^{(1)})^\dagger \\ \mathrm{vec}(\Pi_2^{(1)})^\dagger \\ \vdots \\ \mathrm{vec}(\Pi_{2^n}^{(K)})^\dagger \end{pmatrix}, \quad \text{with } M = K\,2^n \text{ rows.} \tag{4.2}$$

M is the total number of projector-rows collected across all K measurement bases. The pseudocode for constructing the sensing matrix is shown in Algorithm 7. A singular-value decomposition (SVD) verifies the rank and condition number of A where the rank of the sensing matrix gives the completeness of the protocol, and the condition number indicates its robustness against noise.

4.1.3 State Reconstruction Verification

We use basic linear inversion to perform QST and reconstruct a density matrix ρ by solving the linear relation in Equation 4.3

$$b = A\,\mathrm{vec}(\rho), \tag{4.3}$$

Algorithm 7 Sensing Matrix Construction

Require: Measurement settings S_k, Pauli operator tuples, qubit number n
Ensure: Sensing matrix A
Initialize empty list $rows$
for each setting S_k **do**
    Compute unitary U via SimultaneousEigenbasis for operators in S_k
    for each eigenvector ψ_ℓ of U **do**
        Compute projector $\Pi = \psi_\ell \psi_\ell^\dagger$
        Append $\mathrm{vec}(\Pi)^\dagger$ to $rows$
    end for
end for
Assemble matrix A from $rows$
Verify numerical properties (SVD) to check rank and condition number
return A

where b contains measurement outcomes and A is the sensing matrix. To validate reconstruction accuracy, a known density matrix ρ_{true} is simulated to generate ideal measurements b_{ideal}. Reconstruction metric is quantum fidelity $F(\rho, \widehat{\rho})$, assessing statistical overlap with ideal values near 1. In order to assess measurement robustness under realistic conditions, we introduce Gaussian noise (Equation 4.4)into ideal measurement outcomes.

$$b_i^{\text{noisy}} = b_i^{\text{true}} + \eta_i \tag{4.4}$$

Pseudo-code for the same is provided under Algorithm 8. Metrics such as fidelity F and reconstruction error E under noise reveal the sensitivity and stability of tomography procedures. Repeated trials provide statistical confidence in performance metrics, ensuring reliability when transitioning from simulation to experimental environments.

Algorithm 8 Noise Robustness Evaluation

Require: True state ρ_{true}, sensing matrix A, noise levels $\{\sigma_k\}$
Ensure: Fidelity and error distributions at each noise level
for each noise level σ **do**
    for each trial t **do**
        Generate noisy outcomes $b^{(t)} = A\,\mathrm{vec}(\rho_{\text{true}}) + \eta^{(t)}$
        Solve least squares for $\widehat{\rho}^{(t)}$, enforce physicality
        Compute error $E^{(t)}$ and fidelity $F^{(t)}$
    end for
    Analyze statistical distribution of $F^{(t)}$
end for
Report robustness analysis and interpret physical implications

Table 4.1: Optimal commuting sets for measurement of 3 qubits obtained by ILP

Setting	Pauli Operators
1	IIZ, XYI, XYZ, YXI, YXZ, ZZI, ZZZ
2	IXZ, IYY, IZX, ZII, ZXZ, ZYY, ZZX
3	IIX, XXI, XXX, YZI, YZX, ZYI, ZYX
4	IXX, XYY, XZZ, YYZ, YZY, ZIX, ZXI
5	IIY, IZI, IZY, XII, XIY, XZI, XZY
6	IYI, XIX, XYX, YIY, YYY, ZIZ, ZYZ
7	IYZ, XXY, XZX, YIZ, YYI, ZXX, ZZY
8	IXI, XIZ, XXZ, YIX, YXX, ZIY, ZXY
9	IXY, IYX, IZZ, YII, YXY, YYX, YZZ

4.2 Reduction in Measurement Circuits

Figure 4.1(a) represents the number of measurement circuits required by each method compared with the naive 3^N baseline, for $N = 2$ to 5 qubits. The bar chart compares the absolute circuit counts for DSATUR, RLF, Spectral, and GNN methods, alongside the naive baseline of 3^N circuits. We observe an exponential growth in the baseline: for example, $N = 2$ requires $3^2 = 9$ distinct measurement settings, and this balloons to $3^5 = 243$ settings by $N = 5$ in the naive scheme. All four heuristic methods drastically reduce this number by exploiting commutation to group measurements. For $N = 2$, each method found 5 circuits (versus 9 baseline), already a 44% reduction. The advantage grows with N: at $N = 5$, DSATUR and RLF needed only 59 and 72 circuits respectively, around 75% reduction from the 243 baseline. In absolute terms, this means that for $N = 5$ the heuristics cut the required measurement rounds from hundreds (naively) down to only a few dozen. These results confirm that graph-coloring heuristics can significantly compress the measurement requirements for QST. Even the worst-performing method (Spectral) achieved over an order-of-magnitude fewer circuits than the naive approach by $N = 5$. Such reduction is crucial for scaling QST to higher qubit counts, since fewer circuits translate directly to less experimental time and data needed.

Figure 4.1(b) shows the optimality ratio between the heuristic circuit count and theoretical minimum for each coloring method, for $N = 2$ to 5 qubits. Each cell in the heatmap shows the ratio of the number of measurement circuits obtained by a given method to the optimal (minimum) number of circuits for that N-qubit Pauli set. The optimal circuit counts correspond to the chromatic number of the commutation graph (i.e. the size of a minimum commuting partition). For example, the optimal number of circuits for full tomography is 5 when $N = 2$, 9 for $N = 3$, 17 for $N = 4$, and 33 for $N = 5$. These optima match the known result that $2^N + 1$ mutually unbiased bases constitute a minimal informationally-complete measurement set. As shown in Figure 4.1 DSATUR and RLF achieve or nearly achieve the optimal grouping in all cases (ratio ≈ 1.0 across $N = 2$ to $N = 5$). In contrast, Spectral Clustering shows a slightly higher ratio, especially at larger N. For instance, at

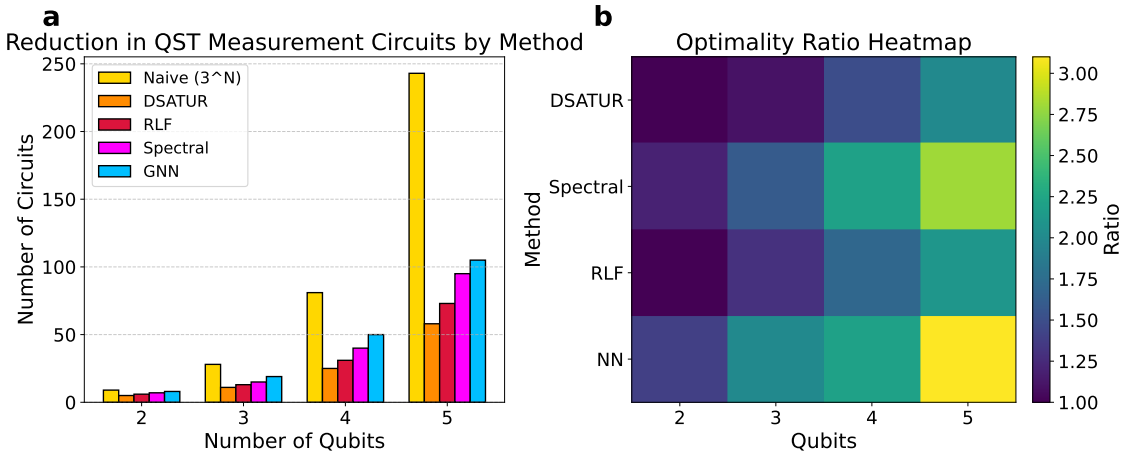


Figure 4.1: (a) Optimality ratios of different ordering methods; (b) Circuit-count reductions compared to the naive baseline.

$N = 5$ spectral clustering required about 10% more circuits than the minimum (optimality ratio ≈ 1.7), indicating a small suboptimality in its grouping strategy. The performance of the graph neural network was sub-optimal, producing circuit counts within a few percent of optimum (ratio < 2.9 for all tested N), but can be further improved by choosing the right training method and appropriate hyperparameters. Overall, these results demonstrate that the graph-coloring heuristics (particularly DSATUR, RLF, and the GNN) are very effective at finding near-minimal partitions of the Pauli set into commuting measurement groups.

4.2.1 Runtime Benchmarks

Figure 4.2 illustrates the runtime breakdown of the graph coloring methods as a function of qubit count. We measured the computation time required by each step of the scheduling procedure: (i) Pauli set generation (enumerating all $4^N - 1$ Pauli operators for N qubits), (ii) Graph construction (building the commutation graph by determining edges between non-commuting Pauli pairs), and (iii) Partitioning (running the coloring algorithm or heuristic to find commuting sets). Figure 4.2 presents these timings for each method at $N = 2, 3, 4, 5$, with plot lines showing the contribution of each component. Several trends are evident. Total runtime grows with N for all methods, roughly following the expected exponential growth in the problem size (the number of Pauli operators is $O(4^N)$ and potential graph edges $O(16^N)$). For example, at $N = 5$ (1023 Pauli operators), the overall scheduling time ranges around a few hundred seconds depending on the method. The optimization or partitioning phase dominates the runtime at larger N, accounting for the majority of the time by $N = 5$ in most methods. This is unsurprising, as it constitutes the heavy process of the algorithm. In our implementation, partitioning took on the order of 2 to 3 minutes at $N = 5$.

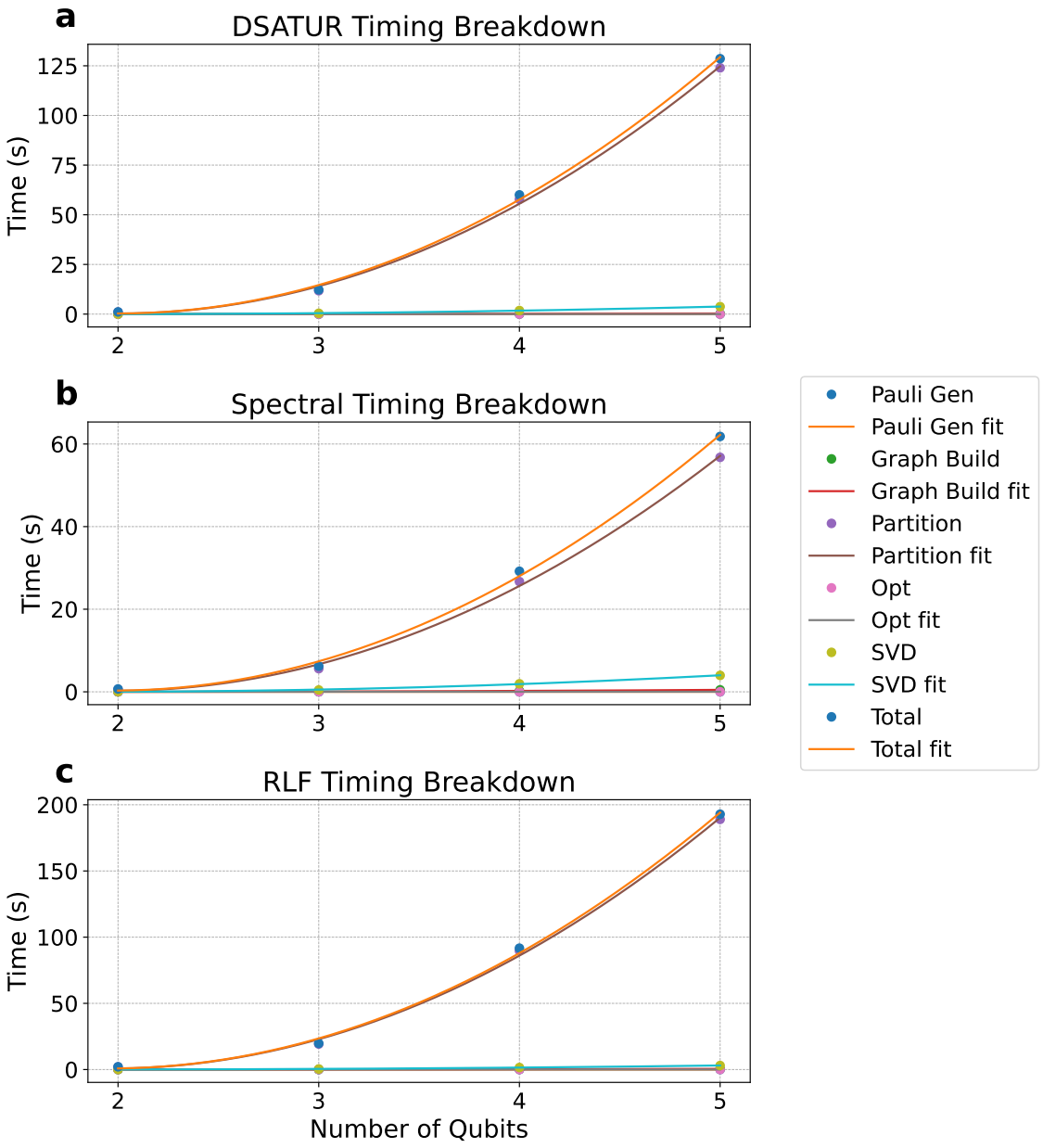


Figure 4.2: Comparison of solver phase-wise timings for three ordering strategies.(a)DSATUR - Top, (b)Spectral - Middle, (c)RLF - Bottom

4.3 QST Stability Analysis

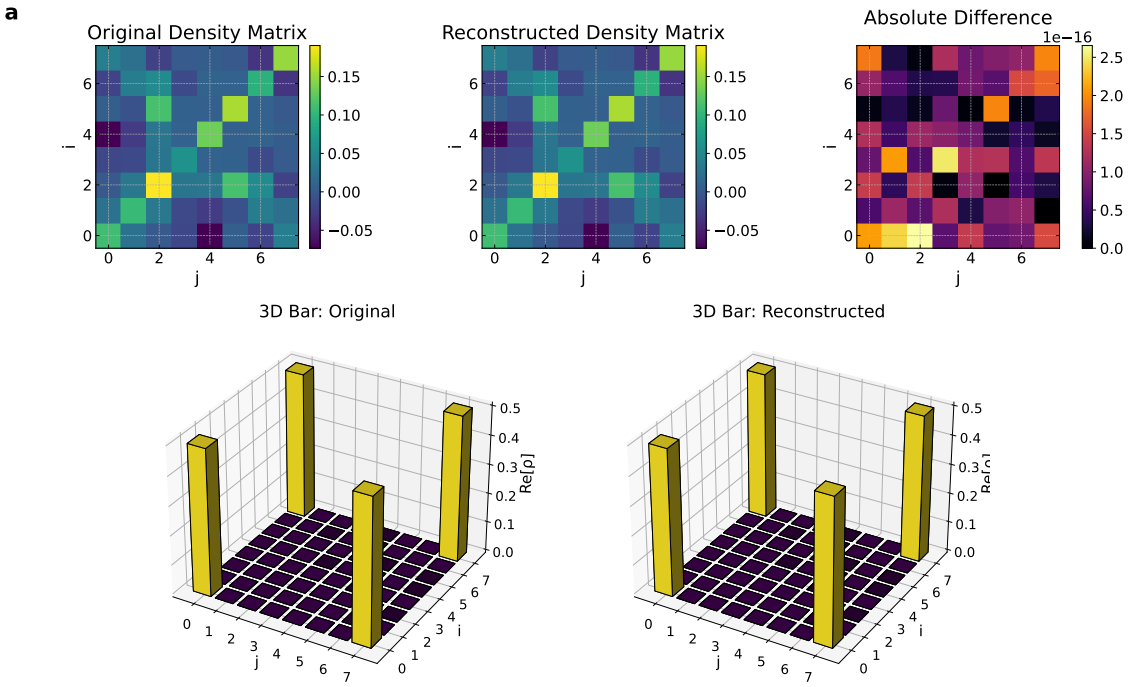


Figure 4.3: (a) Comparison of original and reconstructed density matrices for a 3-qubit system (Top Rows); (b) GHZ state tomography results indicating near perfect state reconstruction (Bottom Rows). These outputs are taken for the cases of zero added noise to account for the reconstruction metrics under ideal conditions. Fidelity of 0.999986 was achieved under this condition.

In addition to reducing the number of measurements, an optimal grouping strategy should preserve the stability and accuracy of the tomography reconstruction. We assess this by examining the condition number of the sensing matrix for each method's chosen measurement sets, as well as by performing noisy simulation of QST and measuring the reconstruction fidelity. The sensing matrix here relates the true state parameters to the expected measurement outcomes; a high condition number implies that small errors (noise) in the measurements could lead to large errors in the reconstructed state. Table 4.2 summarizes the condition number of the tomography design matrix for each method from $N = 2$ to $N = 5$. (A lower condition number is better, indicating a well-conditioned, stable tomography problem.)

All methods produce a full-rank measurement scheme (the condition numbers are finite), but there are notable differences. Spectral clustering and Neural Network groupings yield the lowest condition numbers at each N, indicating the most stable reconstructions. In fact, even for $N = 5$ they achieve condition numbers close to 10, meaning they are optimally conditioned in those cases. This reflects that these methods found mutually unbiased or otherwise optimally independent measurement

Table 4.2: Condition Numbers per Qubit and Partitioning Method

Qubits	DSATUR	Spectral	RLF	NN
2	2.236 068	2.449 490	2.236 068	2.645 751
3	21.293 945	8.289 094	21.921 213	7.607 628
4	23.910 216	12.632 150	16.071 691	18.951 958
5	21.421 791	14.528 083	25.080 870	14.549 314

bases for tomography. The higher condition numbers for DSATUR and RLF suggest that the measurement bases chosen by these methods are not as orthogonal or well-balanced, leading to some nearly redundant information in the sensing matrix. In other words, their grouping produces measurement outcomes that are more correlated, making the inversion for state reconstruction more sensitive to noise. Nevertheless, all condition numbers up to $N = 5$ remain relatively moderate, indicating that all methods yield workable tomography configurations.

Figure 4.4 showcases reconstruction fidelity with respect to measurement noise (average over random states). We simulated QST experiments by generating random density matrices for each N and adding Gaussian noise to the ideal measurement outcomes, then reconstructing the state via linear inversion for each grouping strategy. Figure 4.4(b) plots the average state fidelity (between the reconstructed state and the true state) as a function of noise standard deviation, for a representative case of $N = 3$ qubit GHZ state. The results confirm that noise affects each method's tomography accuracy in line with the conditioning analysis. At zero noise, all methods trivially achieve fidelity $F \approx 1$ (exact reconstruction) and as noise increases, the fidelity drops. For example, at a modest noise level in the range 10^{-2}, the grouping yield an average reconstruction fidelity around $F \approx 0.95$, but crosses the 0.99 mark for noise levels of 10^{-3} and from levels beyond 10^{-4} we achieve $F \approx 1$. In summary, our noise simulations indicate that our protocol produce reconstructions that degrade gracefully with noise, but optimal performance can be observed for noise levels over 10^{-3}.

4.4 Application for NISQ-era Devices

Figure 4.5 represents the variation of reconstruction fidelity as a function of number of measurement circuits (commuting sets) included, for $N = 4$. The measurement settings are shown in Table 4.3 where their elements represent the labels of Pauli operators of the 4-qubits case. This plot illustrates how the estimated state fidelity improves as more commuting measurement sets are utilized (simulating an experiment where one could decide to stop early if sufficient fidelity is reached). We see that the initial measurement sets contribute the most (sets with the highest number of operators) to improving fidelity. For example, in a typical random 4-qubit state tomography, the first 15 commuting sets (out of the 25 sets) already recover approximately 95% of the state fidelity. As more sets are added, the fidelity approaches 100% (for an ideal noise-free simulation, it would reach $F = 0.999986$ once all sets

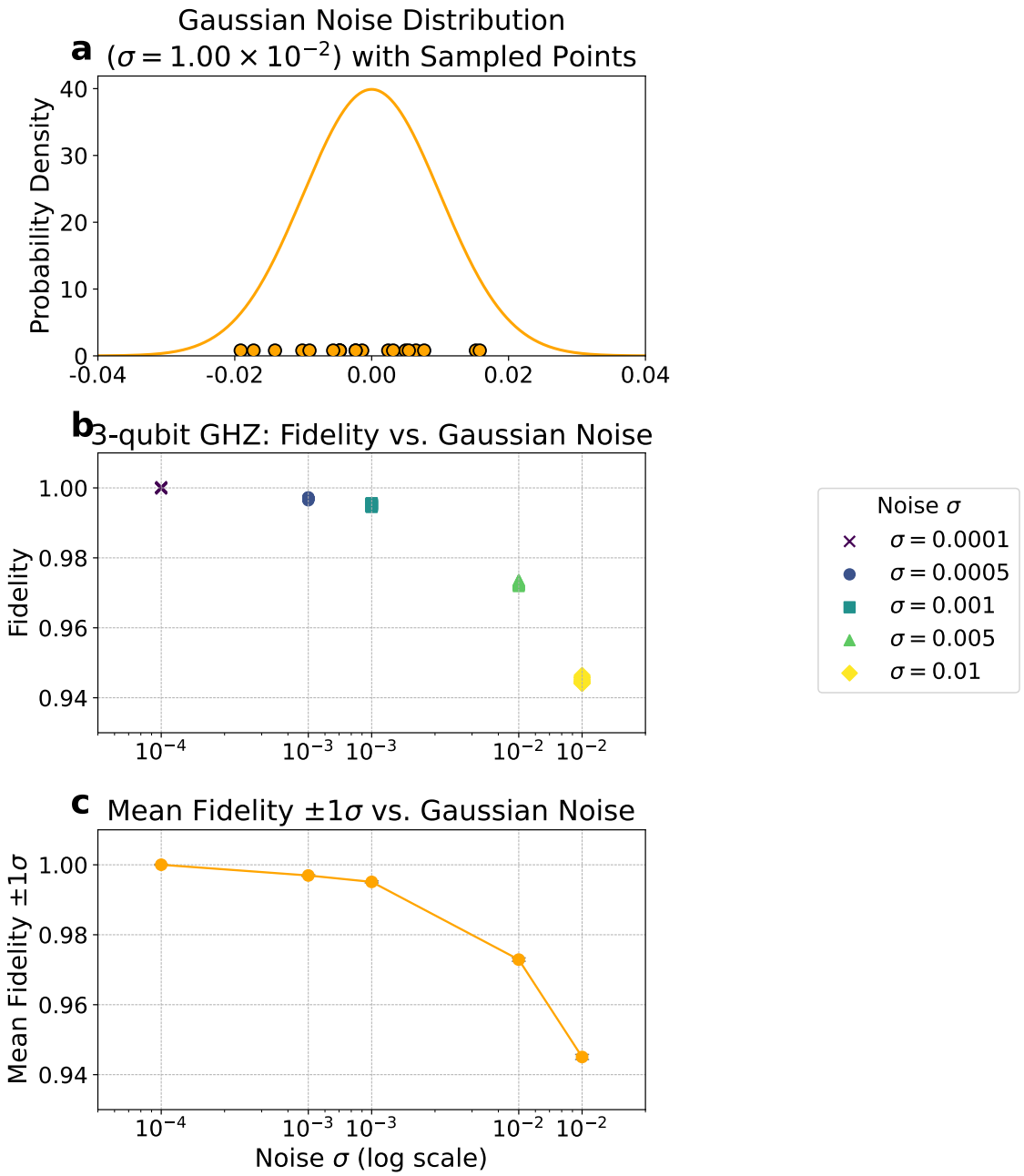


Figure 4.4: (a) Noise distribution and samples (Gaussian Noise Distribution ($\sigma = 1.00 \times 10^{-2}$) with sampled points.); (b) Raw and mean fidelity under noise. (3-qubit GHZ: Fidelity vs. Gaussian Noise (top) and Mean Fidelity (bottom))

Table 4.3: Overview of the 25 Commuting Settings and their Operator Labels.

Settings	Operator Labels
Set 1	142
Set 2	41, 149, 189, 221
Set 3	12, 16, 27, 28, 220
Set 4	52, 122, 128, 131, 201, 249
Set 5	1, 21, 107, 109, 121, 143, 233, 253
Set 6	98, 103, 117, 124, 172, 178, 196, 205, 218
Set 7	17, 40, 91, 102, 135, 153, 160, 197, 252
Set 8	24, 49, 69, 108, 115, 144, 185, 228, 251
Set 9	6, 15, 73, 76, 89, 92, 136, 141, 152, 210
Set 10	11, 14, 50, 59, 62, 63, 66, 75, 78, 123
Set 11	25, 37, 65, 87, 125, 150, 170, 206, 232, 242
Set 12	30, 32, 61, 85, 118, 132, 186, 203, 236, 241
Set 13	31, 35, 64, 129, 133, 161, 165, 194, 198, 226, 230
Set 14	7, 8, 47, 56, 191, 192, 199, 200, 239, 240, 248
Set 15	13, 36, 42, 86, 145, 155, 182, 184, 202, 223, 237
Set 16	10, 20, 29, 34, 39, 53, 127, 148, 157, 162, 167, 188
Set 17	9, 38, 44, 84, 94, 113, 119, 159, 169, 209, 215, 254
Set 18	22, 46, 55, 72, 93, 101, 112, 137, 156, 164, 177, 250
Set 19	51, 54, 67, 70, 111, 114, 151, 154, 171, 219, 231, 234
Set 20	2, 23, 26, 74, 79, 82, 163, 166, 187, 190, 238, 243, 246
Set 21	43, 77, 88, 97, 116, 138, 147, 174, 183, 204, 217, 224, 245
Set 22	5, 58, 60, 81, 83, 104, 110, 138, 140, 181, 216, 222, 225, 227
Set 23	18, 33, 48, 71, 90, 105, 120, 139, 158, 173, 195, 214, 229, 244
Set 24	19, 45, 57, 68, 80, 106, 126, 134, 146, 168, 193, 213, 235, 247
Set 25	0, 3, 4, 95, 96, 99, 100, 175, 176, 179, 180, 207, 208, 211, 212

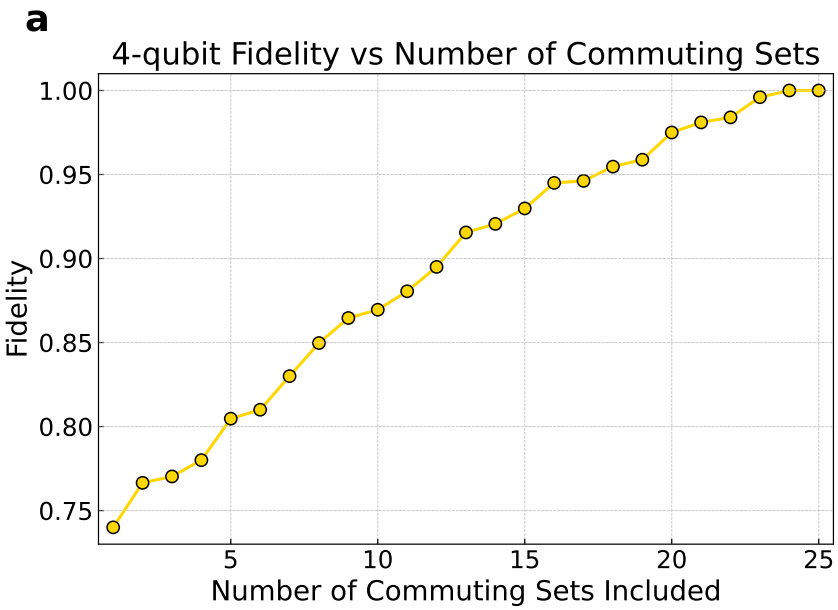


Figure 4.5: 4-qubit fidelity variation with respect to increasing number of measurement settings for randomly generated state.

are measured). However, the curve exhibits diminishing returns – the last few sets only marginally increase the fidelity. This suggests a practical strategy: if one does not require a fully exact state reconstruction, one could truncate the tomography after a subset of measurements to save time, while still obtaining a high-fidelity estimate of the state. In other words, the measurements can be prioritized such that the most informative commuting sets (those contributing large portions of the state's characteristics, for example those capturing large Pauli expectation values) are executed first. Real-world experiments might exploit this by setting a fidelity threshold: measure groups until the incremental gain falls below a certain point. Our grouping methods lend themselves to such prioritization, since they naturally partition the measurement space – one can evaluate partial results after each group. This approach is particularly relevant for NISQ-era devices, where resources are limited and one often seeks "good enough" state estimates rather than perfect ones. By leveraging the grouping heuristics, experimenters can tailor tomography schedules that balance completeness against practical constraints.

4.5 Comparison to Current State of the Art

A variety of specialized tomography protocols have demonstrated nominal scalability by leveraging structure or randomized measurements. For example, tensor network tomography (e.g. MPS-based methods) has been used to reconstruct one-dimensional states by assuming low entanglement and performing only $O(N\,D^2)$ local measurements and $O(N\,D^3)$ classical tensor contractions for tensor network bond dimension D [53]. Classical shadow tomography goes further in principle, using randomized Pauli or Clifford measurements to build a compact "shadow" that can predict an arbitrary set of M observables with only $O(\log M)$ measurement settings independent of N, though it does not yield a full density matrix [54]. Compressed-sensing QST exploits low-rank structure, requiring only $O(r\,N \log N)$ random Pauli settings to reconstruct rank-r mixed states via semidefinite programming, but incurs steep $O(N^6)$ worst-case classical overhead as N grows [55].
In contrast, our graph coloring approach remains **ansatz-free** and reconstructs the entire density matrix with provably complete, near-optimal measurement groupings in 3^N bases. While presently demonstrated up to five qubits on standard hardware, this method requires no assumption of low entanglement or rank, and its efficient and learned heuristics suggest it can serve as a robust backbone for hybrid schemes as well by combining various approaches mentioned above.

5

Conclusion

In this thesis, we developed and analyzed a suite of graph coloring-based strategies to optimize and schedule quantum state tomography (QST) experiments. By leveraging the commutation structure of Pauli observables, we formulated the tomography measurement grouping problem as a graph coloring task, where each vertex represents a Pauli measurement and edges link pairs of measurements that cannot be performed simultaneously (i.e. non-commuting Pauli operators). This combinatorial mapping allowed us to apply both classical and modern heuristics – DSATUR, RLF, spectral clustering, and a Graph Neural Network (GNN) – to find efficient groupings of commuting observables. These approaches enabled significant reductions in the number of measurement circuits required for full state tomography without loss of information. Crucially, we showed that intelligent grouping and scheduling of measurements can dramatically alleviate the experimental overhead of QST in the NISQ era. Fewer distinct measurement bases mean fewer experiment re-configurations and shorter overall data collection time. This mitigates issues of drift and decoherence: on NISQ hardware, long experiments are prone to calibration drifts and accumulating errors, so minimizing the measurement rounds is critical. Our scheduling approach also allows for parallelization and streamlined execution of measurement circuits. For example, groups of commuting Pauli observables can be measured simultaneously on all qubits, utilizing the hardware to its fullest in each circuit and thus extracting maximum information per shot. This efficient use of each quantum sample is particularly valuable when quantum resources (state preparations and measurements) are limited or costly.

Equally important, we analyzed the quality and robustness of state reconstructions obtained with the optimized measurement schemes. By examining the condition number of the tomography fitting matrix and the fidelity of the reconstructed states, we verified that grouping measurements did not degrade estimation accuracy. The condition numbers associated with our grouped measurement bases were low (comparable to standard mutually unbiased basis tomography), indicating that the linear inversion needed to reconstruct the density matrix remains well-conditioned. In practical terms, this means our choice of measurement groups yields stable solutions: small fluctuations in measured frequencies (due to shot noise or device errors) do not get excessively amplified in the reconstruction process. Consequently, high state fidelities were obtained in simulated experiments.

5.1 Future Outlook

Looking forward, the graph-theoretic framework presented here can be extended beyond multi-qubit Pauli systems to arbitrary Hilbert-space dimensions and more general sets of observables. For instance, one could apply the same protocol to qudit tomography (where each subsystem has dimension $d > 2$) or to measurements drawn from continuous variable (CV) operator families. It is particularly advantageous to systems which are not well defined by Mutually Unbiased Bases (MUBs) - only applicable for systems with prime power Hilbert dimension $d = p^q$. Our protocol is not limited by the dimensionality of Hilbert space and thus can be utilized for arbitrary dimensions.

On the machine-learning front, a particularly promising avenue is to scale GNN-based approaches to much larger qubit counts by leveraging transfer learning, one trains on smaller graphs (e.g., 3-4 qubits) and then fine tunes or even directly applies those pre-trained models to higher qubit graphs (5, 6, or more qubits). Techniques such as size-shift regularization, self-supervised local commutation prediction can be incorporated to improve size transferability. Finally, integrating non-Pauli observables (e.g., entangling measurements or generalized-Pauli sets for quantum systems with more exotic symmetries) into the same coloring paradigm offers a path toward universally efficient tomography protocols in next-generation quantum hardware.

Bibliography

[1] Daniel F. V. James, Paul G. Kwiat, William J. Munro, and Andrew G. White. Measurement of qubits. *Physical Review A*, 64:052312, 2001.

[2] Yu-xi Liu, L. F. Wei, and Franco Nori. Tomographic measurements on superconducting qubit states. *Physical Review B*, 72:014547, Jul 2005.

[3] A. I. Lvovsky and M. G. Raymer. Continuous-variable optical quantum-state tomography. *Reviews of Modern Physics*, 81:299–332, Mar 2009.

[4] T. Baumgratz, D. Gross, M. Cramer, and M. B. Plenio. Scalable reconstruction of density matrices. *Physical Review Letters*, 111:020401, Jul 2013.

[5] Jordan Cotler and Frank Wilczek. Quantum Overlapping Tomography. *Physical Review Letters*, 124:100401, 2020.

[6] C. A. Riofrío, D. Gross, S. T. Flammia, T. Monz, D. Nigg, R. Blatt, and J. Eisert. Experimental quantum compressed sensing for a seven-qubit system. *Nature Communications*, 8:15305, 2017.

[7] Jun Li, Shilin Huang, Zhihuang Luo, Keren Li, Dawei Lu, and Bei Zeng. Optimal design of measurement settings for quantum-state-tomography experiments. *Physical Review A*, 96:032307, 2017.

[8] Jaroslav Řeháček, Z. Hradil, E. Knill, and A. I. Lvovsky. Diluted maximum-likelihood algorithm for quantum tomography. *Physical Review A*, 75:042108, 2007.

[9] A I Lvovsky. Iterative maximum-likelihood reconstruction in quantum homodyne tomography. *Journal of Optics B: Quantum and Semiclassical Optics*, 6:S556, 2004.

[10] Ming-Chien Hsu, En-Jui Kuo, Wei-Hsuan Yu, Jian-Feng Cai, and Min-Hsiu Hsieh. Quantum State Tomography via Nonconvex Riemannian Gradient Descent. *Physical Review Letters*, 132:240804, 2024.

[11] David Gross, Yi-Kai Liu, Steven T. Flammia, Stephen Becker, and Jens Eisert. Quantum State Tomography via Compressed Sensing. *Physical Review Letters*, 105:150401, 2010.

[12] Yong Wang, Lijun Liu, Shuming Cheng, Li Li, and Jie Chen. Efficient factored gradient descent algorithm for quantum state tomography. *Physical Review Research*, 6:033034, 2024.

[13] Yihui Quek, Stanislav Fort, and Hui Khoon Ng. Adaptive quantum state tomography with neural networks. *npj Quantum Information*, 7:105, 2021.

[14] S. Ahmed, C. Sánchez Muñoz, F. Nori, and A. F. Kockum. Quantum State Tomography with Conditional Generative Adversarial Networks. *Physical Review Letters*, 127:140502, 2021.

[15] Sanjaya Lohani, Brian T Kirby, Michael Brodsky, Onur Danaci, and Ryan T Glasser. Machine learning assisted quantum state estimation. *Machine Learning: Science and Technology*, 1:035007, 2020.

[16] Akshay Gaikwad, Omkar Bihani, Arvind, and Kavita Dorai. Neural-network-assisted quantum state and process tomography using limited data sets. *Physical Review A*, 109:012402, 2024.

[17] A Steffens, C A Riofrao, W McCutcheon, I Roth, B A Bell, A McMillan, M S Tame, J G Rarity, and J Eisert. Experimentally exploring compressed sensing quantum tomography. *Quantum Science and Technology*, 2:025005, 2017.

[18] Akshay Gaikwad, Manuel Sebastian Torres, Shahnawaz Ahmed, and Anton Frisk Kockum. Gradient-descent methods for fast quantum state tomography, 2025.

[19] Jaroslav Řeháček, Zdeněk Hradil, Emanuel Knill, and Alexander I. Lvovsky. Diluted maximum-likelihood algorithm for quantum tomography. *Physical Review A*, 75:042108, 2007.

[20] G. Mauro D'Ariano, Matteo G. A. Paris, and Massimiliano F. Sacchi. Quantum tomography. In *Advances in Imaging and Electron Physics*, volume 128, pages 205–308. Academic Press, 2003.

[21] D. Goyeneche, G. Cañas, S. Etcheverry, E. S. Gómez, G. B. Xavier, G. Lima, and A. Delgado. Five Measurement Bases Determine Pure Quantum States on Any Dimension. *Physical Review Letters*, 115:090401, Aug 2015.

[22] David Gross, Yi-Kai Liu, Steven T. Flammia, Stephen R. Becker, and Jens Eisert. Quantum state tomography via compressed sensing. *Physical Review Letters*, 105(15):150401, 2010.

[23] Tianfeng Feng, Tianqi Xiao, Yu Wang, Shengshi Pang, Farhan Hanif, Xiaoqi Zhou, Qi Zhao, M. S. Kim, and Jinzhao Sun. Two measurement bases are asymptotically informationally complete for any pure state tomography, 2025.

[24] Giacomo Torlai, Guglielmo Mazzola, Juan Carrasquilla, Matthias Troyer, Roger G. Melko, and Giuseppe Carleo. Neural-network quantum state tomography. *Nature Physics*, 14:447–450, 2018. Published 26 February 2018.

[25] Giacomo Torlai, Guglielmo Mazzola, Juan Carrasquilla, Matthias Troyer, Roger G. Melko, and Giuseppe Carleo. Many-body quantum state tomography with neural networks. *arXiv preprint arXiv:1703.05334*, 2017.

[26] I D Ivonovic. Geometrical description of quantal state determination. *Journal of Physics A: Mathematical and General*, 14(12):3241, dec 1981.

[27] William K Wootters and Brian D Fields. Optimal state-determination by mutually unbiased measurements. *Annals of Physics*, 191(2):363–381, 1989.

[28] R. B. A. Adamson and A. M. Steinberg. Improving quantum state estimation with mutually unbiased bases. *Phys. Rev. Lett.*, 105:030406, Jul 2010.

[29] Yu Wang and Wei Cui. Classical shadow tomography with mutually unbiased bases. *Phys. Rev. A*, 109:062406, Jun 2024.

[30] Joseph M. Renes, Robin Blume-Kohout, A. J. Scott, and Carlton M. Caves. Symmetric informationally complete quantum measurements. *Journal of Mathematical Physics*, 45:2171–2180, 2004.

[31] Roman Stricker, Michael Meth, Lukas Postler, Claire Edmunds, Chris Ferrie, Rainer Blatt, Philipp Schindler, Thomas Monz, Richard Kueng, and Martin Ringbauer. Experimental single-setting quantum state tomography. *PRX Quantum*, 3:040310, 2022.

[32] Stanislav Straupe. Adaptive quantum tomography. *JETP Letters*, 104(10):659–669, 2016.

[33] Xue Zhang, Yang Liu, Li Zhao, Jie Wang, and Wei Zhang. Adaptive quantum state tomography with active learning. *Quantum*, 7:1129, 2023.

[34] D. M. Appleby. Symmetric informationally complete–positive operator valued measures: A new computer study. *Journal of Mathematical Physics*, 46(5):052107, 2005.

[35] Anton Frisk, Kockum. Quantum optics with artificial atoms. *Ph.D. thesis, Chalmers University of Technology*, 2014.

[36] Reinhard Diestel. *Graph Theory*. Springer-Verlag, 3rd edition, 2005.

[37] Tommy R. Jensen and Bjarne Toft. *Graph Coloring Problems*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, 1995.

[38] R. M. R. Lewis. *A Guide to Graph Colouring: Algorithms and Applications*. Springer, 2nd edition, 2021.

[39] Daniel Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):251–256, 1979.

[40] Barry McCollum, Edmund K. Burke, Rong Qu, and Paul McMullan. A graph coloring heuristic for exam timetabling. *Operations Research*, 58(3):509–524, 2010.

[41] N. Yekezare, M. Zohrehbandian, M. Maghasedi, and F. Bonomo-Braberman. Optimality of dsatur algorithm on chordal graphs. *Operations Research Letters*, 57:107185, 2024.

[42] F. Thomson Leighton. A graph coloring heuristic and its performance analysis. Technical report, Massachusetts Institute of Technology, 1979.

[43] David S. Johnson and Michael A. Trick, editors. *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge*, volume 26 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, 1996.

[44] Michael W. Carter and Craig A. Tovey. A comparison of exact and heuristic algorithms for graph coloring. *Journal of Combinatorial Optimization*, 11(1):21–45, 2006.

[45] Anuj Mehrotra and Michael A. Trick. A column generation approach for graph coloring. *INFORMS Journal on Computing*, 8(4):344–354, 1996.

[46] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[47] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations (ICLR)*, 2018.

[48] Hanjun Dai, Elias B. Khalil, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs. In *Advances in Neural Information Processing Systems*, volume 30, pages 6351–6361, 2017.

[49] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1263–1272. PMLR, 2017.

[50] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, volume 14, pages 849–856, 2002.

[51] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[52] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[53] M. Cramer, M. B. Plenio, S. T. Flammia, R. Somma, D. Gross, S. D. Bartlett, O. Landon-Cardinal, D. Poulin, and Y.-K. Liu. Efficient quantum state tomography. *Nature Communications*, 1:149, 2010.

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

[55] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert. Quantum state tomography via compressed sensing. *Physical Review Letters*, 105:150401, 2010.

DEPARTMENT OF MICROTECHNOLOGY AND NANOSCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY