Decoherence, Predictability, and Robustness: Investigating the Preferred Division of a Quantum System into Subsystems

by
Aiden Karpf

Advisors:

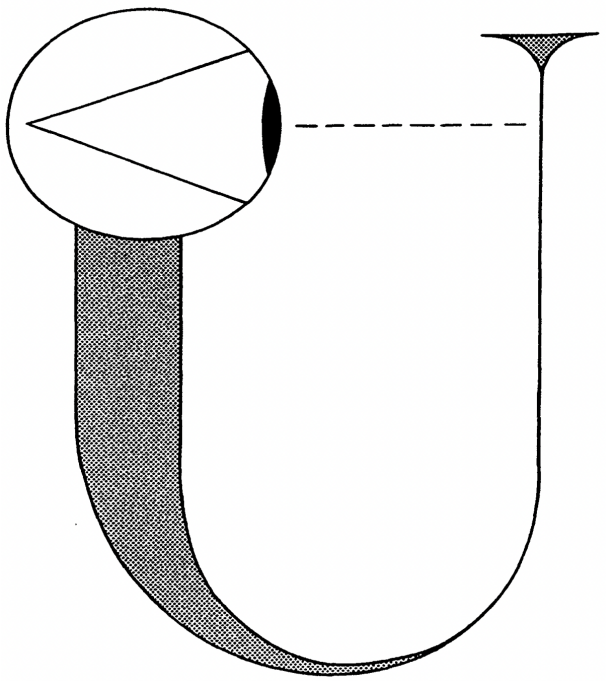
Professor Kevin Setter, Primary Advisor
Professor Dwight Whitaker, Local Advisor

A thesis submitted in partial fulfillment
of the requirements for the
Degree of Bachelor of Arts
in Physics and Astronomy

POMONA COLLEGE
Claremont, CA
May 13, 2025

"it from bit."

— John Wheeler



"everything is what it is only with respect to something else."

— Carlo Rovelli

Abstract

The emergence of classicality from its underlying quantum mechanical laws has remained a fundamental question in physics for over a century. While decoherence theory has provided a general framework to understand this quantum-to-classical transition, it critically assumes a predetermined division of a Hilbert space into subsystems (a tensor product structure) [1]. It remains an open question whether this division into subsystems can be identified given only the basic data of textbook quantum mechanics—a Hamiltonian and a Hilbert space. One promising strategy to dynamically select these subsystems is by requiring that valid tensor product structures (TPS) are able to exhibit quasi-classical dynamics [2, 3, 4]. In this thesis, we investigate two quasi-classical metrics from the literature: robustness (resistance of a system state to entangling with the environment), and predictability (localization of classical-like trajectories over time) [5]. We develop and implement a machine learning algorithm to search for the tensor product structures of a composite-system of qubits that optimize these metrics. Our numerical results strongly imply that predictability implies robustness, and furthermore that predictability is the better metric to use to inform the factorization of a Hilbert Space. Additionally, we find that quasi-classical TPSs have a significantly greater variance in their scores over different initial states than a typical TPS; this implies that a TPS with particularly stable "pointer states" must also have particularly unstable superposition states. These results provide concrete numerical evidence to inform future researchers in their search for quasi-classical tensor product structures.

gratitude

grateful for the context for which ¡ could not exist without !

this includes:

* all of the plants and animals that have lived, suffered, and died, to sustain my own life as food.
* the incredibly rare opportunity and access i have had to pursue an education throughout my life.
* the financial support that i have received in pursuit of this education.
* the deeply mysterious nature of the world we live in. in particular, the indeterminacy of quantum mechanics for cultivating my spirituality, wonder, and feeling of intimacy with the universe.
* gratitude itself, which heals all.

a final note:

in reality, there are no closed subsystems. we are all deeply and inseparably connected.

for that, i am forever grateful.

Acknowledgments

This work was made possible by the collective effort of many, many people. I would like to specifically thank:

Prof. Setter, for taking me on as a thesis student, dedicating countless hours to discussing quantum theory with me, providing incredible guidance, and—most of all—for the many moments that we've shared together expressing our joint excitement and passion for studying physics.

Mr. Powers, my high school physics teacher, for originally introducing me to the field of physics, creating an exploratory classroom environment, and encouraging my curiosity about the physical world.

My friends Gabe, Soni, and Alex, for listening to my various philosophical monologues over the years, for always being there when I needed them, and for our shared experiences "unzipping reality."

Dwight, for helping foster the incredible sense of community I immediately felt in the Physics department upon arriving at Pomona my freshman year. This played a large role in me actually becoming a physics major.

The Pomona Physics community as a whole, including all of the students, professors, and staff. I could not have done it without you.

All of the incredible friends I have met at Pomona, for everything. I know deep down that we will remain very present in each others' lives as we continue to evolve in time.

My brother, for his support, our mutual respect, and the vastly different trajectories of our lives.

And finally, my parents. For their refusal to shape me into any particularly person. For allowing me to become myself.

Contents

Chapter 1

Introduction

In our day-to-day lives, the universe seems to act in an intuitive, deterministic way, the objects we interact with possessing definite properties. Despite this, the atoms that make up our world obey the strange laws of quantum mechanics—a theory describing a probabilistic, indeterminate universe with objects that develop non-classical, "entangled" correlations. A natural question arises: how does our classical experience of the world emerge from the underlying quantum mechanical laws? What is the relationship of the quantum to the classical domains? Perhaps the most successful account of this relationship is called the "decoherence program" of quantum mechanics. Decoherence is the process by which a quantum mechanical system entangles with its environment, collapsing the system's state into a single, definite eigenstate of a preferred observable (e.g., position) [6]. These definite eigenstates align with our everyday experience of a stable, classical world. This chapter provides an overview of decoherence and introduces several relevant terms in a broad, conceptual way. These terms will be made mathematically precise in Chapter 2.

1.1 Potato in a Superposition: An Example

To begin, let's start with an example. Recall an electron in the famous double-slit experiment—before a measurement is made, the electron behaves as if it is going through both slits simultaneously, effectively in a superposition of two places at once. Now consider, on the other hand, a potato sitting on the table in front of you. Unlike the electron, we expect the potato to occupy a single, definite position. Why? If the potato is simply made up of a bunch of quantum-mechanically behaving particles, why doesn't it also enter superpositions of two places at once?

To answer this question, let's see what happens when we treat the potato as a quantum mechanical object in a superposition state of position 0 and position 1: $|\psi\rangle = c_0|0\rangle + c_1|1\rangle$ (Figure 1.1).[1] We call this a *coherent* superposition; we can think of *coherence* as referring to

[1]Though it may seem unnatural to describe a macroscopic object like a potato with a quantum state, we should be able to, in principle, apply the laws of quantum mechanics to any system. This pedagogical example will help illustrate how classicality can emerge from the underlying quantum mechanical laws.

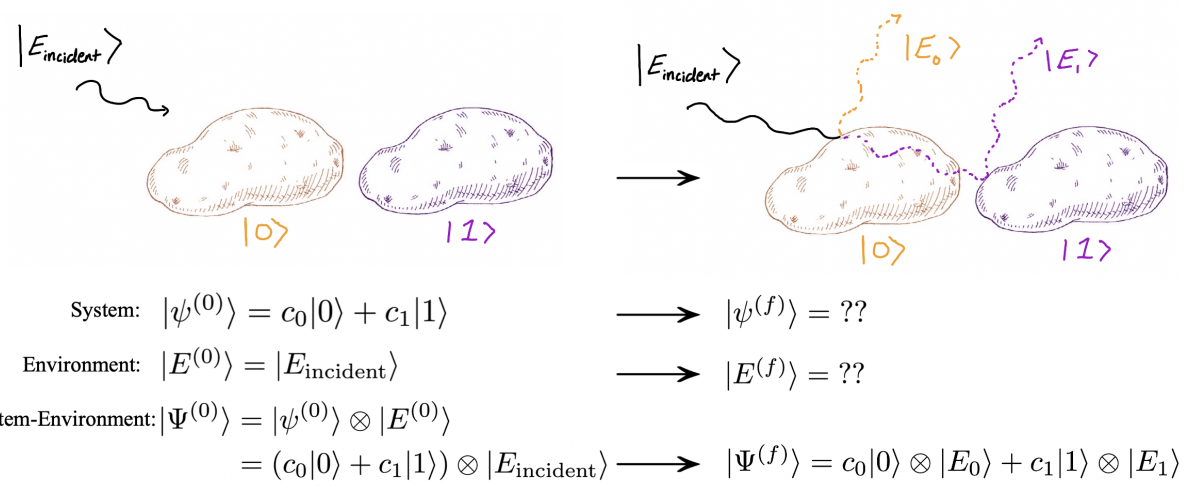


Figure 1.1: A potato in a superposition state.[2] An incident photon entangles with the potato, delocalizing coherence to the level of the system-environment. With many environmental photons (as is the case for a macroscopic object), entanglement is rapidly generated, leading to decoherence.

a well-defined relative-phase between the superposed states. Coherence allows for all of the non-classical effects of quantum mechanics (e.g., superposition, interference, entanglement).

Of course, the potato is just sitting on table, and thus it's in constant interaction with its environment (which we will also need to model quantum mechanically).[3] Consider one such environmental photon incident on the potato with initial trajectory $|E_{\text{incident}}\rangle$. It is clear that the final trajectory of the photon is dependent on the position of the potato. But since the potato itself doesn't have a definite position, the photon cannot have a definite trajectory either. Thus, we see that the initially classical state of the photon has evolved into a superposition state of two different trajectories, $|E_0\rangle$ and $|E_1\rangle$. Notice that we can no longer talk about the photon or the potato individually in our state ket formalism (as shown by the "??" in Figure 1.1)—they are fundamentally dependent on each other, together realizing a single quantum state,[4] $|\Psi^{(f)}\rangle$. The process of two subsystems becoming intrinsically dependent on each other in this way (the potato's superposition amplified to the level of the

[2]For the unfamiliar reader: the $\otimes$ symbol in this figure denotes a tensor product, which combines individual subsystems into a larger composite-system. This will be described in detail in Chapter 2.

[3]When quantum theory was first developed in the early twentieth century, quantum phenomena had only been observed on extremely microscopic scales. This motivated interpretations of quantum mechanics like the Copenhagen interpretation, which postulates a fundamental dualism between a quantum system and a classical measuring apparatus or environment. Today, however, we have observed increasingly macroscopic quantum systems [7], which supports the extension of quantum theory to the macroscopic world. Modeling the environment as a unitarily-evolving quantum mechanical entity is a central aspect of the decoherence program, and enables the framework's explanation of the emergence of classicality.

[4]Now and throughout this thesis, I will use superscripts to denote time evolution labels (e.g., initial, final), reserving subscripts to later denote subsystem indices (e.g., qubit number).

photon) is called *entanglement*.

Through entanglement, coherence has been spread from the potato to the potato-photon; the photon has "carried away" coherence from the potato alone, and making a predictable measurement of the potato (with well-defined relative phases) requires measuring both the photon and potato simultaneously. In other words, only our system-environment, $|\Psi^{(f)}\rangle$, has a well-defined quantum state.

Now consider what happens as more and more photons interact with the potato, rapidly generating entanglement and further delocalizing the coherence of our potato (Figure 1.2). With each additional entangled photon, more and more of the potato's phase information is lost to the environment, spreading over many environmental degrees of freedom and thus becoming effectively irretrievable. From the perspective of an observer embedded in the environment, the potato has lost its coherence altogether (since retaining coherence would require a joint measurement of the entire system-environment). This process of a system entangling with its environment is called *decoherence* and results in the system collapsing into a single, definite state called a *pointer state*.[5] In a way, the environment is effectively "measuring" the state of the system, and we call the final state a pointer state since it's what would be indicated by a "pointer" on a measuring device. In our potato example, decoherence would collapse the superposition into one of its two definite positions (pointer states), $|0\rangle$ or $|1\rangle$. The reader may be wondering: why exactly does the rapid generation

[5] A particularly notable example of pointer states are the coherent states of the quantum harmonic oscillator interacting with a large environment [8].

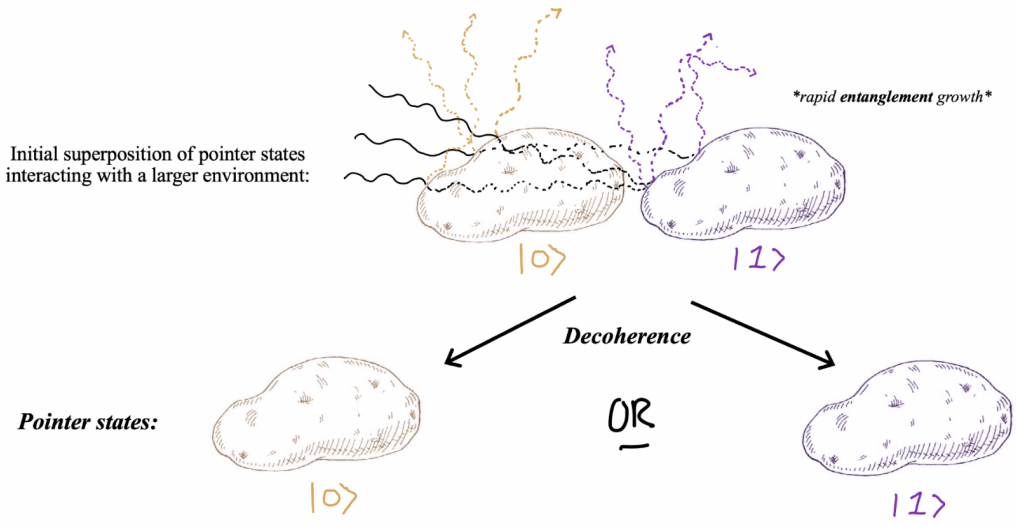


Figure 1.2: A potato initialized in the same superposition of pointer states from earlier, $|\psi^{(0)}\rangle = c_0|0\rangle + c_1|1\rangle$, but now interacting with a larger environment (more incident photons). This interaction results in the rapid generation of entanglement between the potato and its environment, causing the potato to decohere into one of its pointer states: $|0\rangle$ or $|1\rangle$.

of entanglement cause decoherence? This question is an essential part of the "measurement problem," and the answer inevitably depends on your favored interpretation of quantum mechanics.[6]

1.2 Potato in a Pointer State

What if our potato began in one of these pointer states, say position $|0\rangle$ (Figure 1.3)? Then, when photons scatter off the potato, they would all take final trajectory $|E_0\rangle$. Because the photons leave in a definite state, no entanglement is generated between the potato and its environment—thus, the potato's position is unaffected by the environment and remains constant over time. We say that the potato in this pointer state is robust to entanglement

[6]One popular interpretation is called the "many-worlds" interpretation, which provides a cartoon explanation of this potato decoherence process as follows. Consider yourself to be a part of the environment, like the photon from Figure 1.1. From your perspective as a photon, you will find yourself to have a definite trajectory (e.g., either $|E_0\rangle$ or $|E_1\rangle$). However, the total system-environment wavefunction after entanglement, $|\Psi^{(f)}\rangle = c_0\,|0\rangle \otimes |E_0\rangle + c_1\,|1\rangle \otimes |E_1\rangle$, contains amplitudes for the photon to take both trajectory $|E_0\rangle$ *and* trajectory $|E_1\rangle$. How do we reconcile this contradiction? According to the many-worlds interpretation, as soon as you—the photon—entangle with the potato, the universe "branches" into two parallel universes: one in which you occupy the state $|E_0\rangle$ and the potato has decohered to position $|0\rangle$, and one in which you occupy the state $|E_1\rangle$ and the potato has decohered to position $|1\rangle$. Though this explanation is not fully satisfying, many physicists believe it's the best interpretation we currently have. That being said, there are many other popular interpretations such as "relational quantum mechanics," "Quantum Bayesianism," and "spontaneous collapse" theories [9].

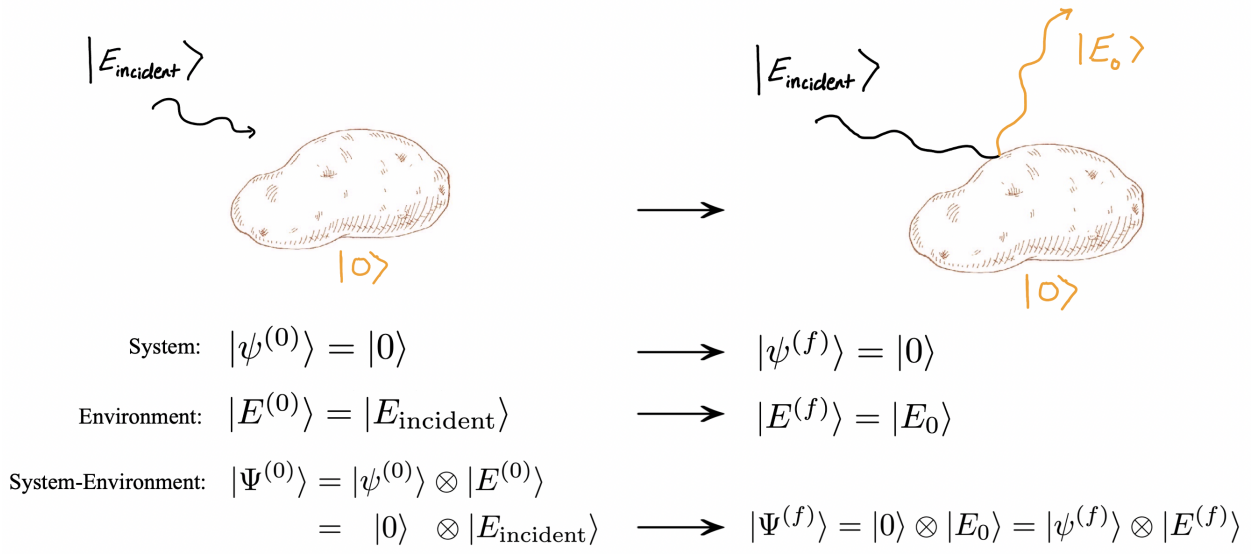


Figure 1.3: A potato in a pointer state. An incident photon scatters off the potato into a definite final state, $|E_0\rangle$. No entanglement is generated, and the pointer state persists over time.

generation. Furthermore, the pointer state is predictable (e.g., if we observe the potato, wait ten seconds, and observe it again, we expect to see it in the same position). These robust and predictable states (terms that will soon be made mathematically precise) are what we observe macroscopically!

So why then can microscopic systems like electrons act as if they're in multiple position states simultaneously? It is not, as is often misunderstood, because microscopic systems interact less with the environment—it is instead because the effect of environmental interaction is less impactful than the object's own self-interaction. For example, consider a case in which the energy available from environmental fluctuations interacting with an electron is lower than the energy level spacing of the electron itself (e.g., only small thermal fluctuations are present). Then, environmental interaction will cause the electron to decohere into an energy eigenstate (i.e., eigenstate of the self-Hamiltonian[7]) rather than a position eigenstate [10]. This is why it is so useful to deal with energy eigenstates in quantum mechanics—similar to position in classical physics (e.g., position of our potato), energies on the micro-scale remain stable to interactions with the environment![8]

Often in physics—and science as a whole—it is closed, isolated systems that are stable and easy to describe. For example, treating a ball and gravity as a closed system (while assuming environmental interaction like air resistance is negligible) greatly stabilizes the dynamics of the ball, making it simpler to model. However, in quantum mechanics, it is the radical openness of systems (their constant interaction with the environment) that makes them stable [10]. Though our potato may seem like a closed, isolated system, it is in fact constantly interacting with the environment—and this causes it to rapidly decohere to one of its pointer states that changes minimally over time. Thus, it is the openness—not closedness—of quantum systems that give rise to the stable, predictable dynamics we observe!

1.3 Potato-photon with New Subsystems

Before leaving our potato example behind, let's re-examine our superposition state from Section 1.1. There is a key observation about the nature of entanglement that is worth emphasizing.

Recall that in our potato superposition example (Figure 1.1), only the system-environment, $|\Psi^{(f)}\rangle$, had a well-defined quantum state after time evolution. Notice that this state is not separable into "potato" and "photon" (i.e., $|\Psi^{(f)}\rangle \neq |\psi\rangle \otimes |E\rangle$). This final state being inseparable is precisely what we mean when we say our potato, $|\psi\rangle$, and photon, $|E\rangle$, are

[7]The self-Hamiltonian describes the system's internal dynamics when isolated from the environment, and its eigenstates are energy eigenstates. The interaction Hamiltonian describes coupling to the environment, often via observables like position. When environmental fluctuations are too weak to induce transitions, decoherence favors energy eigenstates over position eigenstates.

[8]Consider an electron in one of an atom's s-orbitals. Because environmental interactions are too weak to distinguish between nearby positions, decoherence favors energy eigenstates over position eigenstates. Thus, the electron in an s-orbital has a definite energy, but occupies a superposition of position states—namely those within a spherical shell surrounding the nucleus.

entangled. Each subsystem has lost its individuality.

However, notice that this definition of entanglement is dependent on what exactly our subsystems are—it is a relative term that defines the relationship between two specifically chosen parts of an overall quantum state. Thus far, we have chosen the subsystems of our system-environment state, $|\Psi^{(f)}\rangle$, to be the potato, $|\psi\rangle$, and the photon, $|E\rangle$. This may seem like the only possible choice of subsystems (given its intuitive nature). However, in an abstract sense, we are dealing only with an overall quantum state (of the system-environment), and thus can split this state into subsystems however we like. In fact, there exists some other choice of subsystems (e.g., a system′ with basis $\{|0'\rangle, |1'\rangle\}$, and an environment′ with basis $\{|E_0'\rangle, |E_1'\rangle\}$) in which our overall quantum state is unentangled.[9] In Figure 1.4, this scenario is displayed, where our system′, $|\psi'\rangle$, occupies state $|0'\rangle$, and our environment′, $|E'\rangle$, occupies state $|E_0'\rangle$. We can see that the system′ and environment′ are unentangled since our total state is now separable into its subsystems: $|\Psi^{(f)}\rangle = |0'\rangle \otimes |E_0'\rangle = |\psi'\rangle \otimes |E'\rangle$. This is despite the fact that the same exact overall quantum state is entangled when we define our subsystems to be "potato" and "photon" ($|\Psi^{(f)}\rangle \neq |\psi\rangle \otimes |E\rangle$). This example highlights the fact that entanglement is a relative concept that depends on the choice of subsystems.

[9]For every pure state (of our composite-system), there exists some division into subsystems in which those subsystems are unentangled [4]. That being said, this division into subsystems is not always easy to visualize (e.g., our system′ and environment′) since our brains are so used to perceiving objects in their pointer states.

[10]It is important to emphasize that though our overall state $|\Psi^{(f)}\rangle$ is unentangled relative to our system′ and environment′, this does not mean that $\{|0'\rangle, |1'\rangle\}$ and $\{|E_0'\rangle, |E_1'\rangle\}$ are pointer states. In fact, as soon as this unentangled state $|\Psi^{(f)}\rangle = |0'\rangle \otimes |E_0'\rangle$ evolves a bit more in time (while still interacting with the environment), it will likely become quickly entangled. The point of this example is to illustrate that

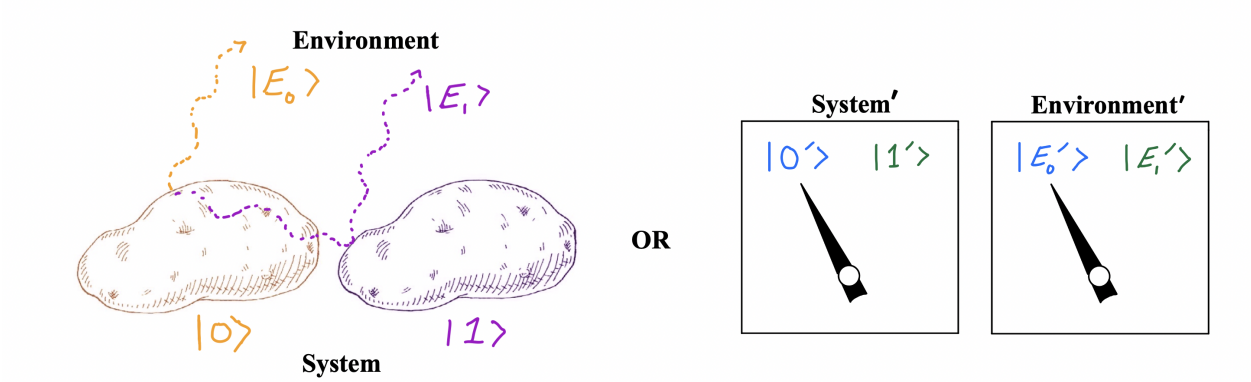


Figure 1.4: The same entangled quantum state of our system-environment from Figure 1.1, $|\Psi^{(f)}\rangle$, represented with a new division into subsystems. Here, our system′ occupies the $|0'\rangle$ state and our environment′ occupies the $|E_0'\rangle$ state. The total state $|\Psi^{(f)}\rangle$ is unentangled with this choice of subsystems.[10]

The idea of dividing an overall composite quantum system (e.g., the system-environment) into subsystems will be further described in the next section and is formalized in Chapter 2 as "tensor product structures." These tensor product structures form the focus of this thesis, and characterizing how different divisions of a composite quantum system act under time evolution is our primary objective.

1.4 The Problem of Preferred Subsystems and Tensor Product Structures

In the literature, decoherence has been claimed to effectively solve one part of the measurement problem called *the problem of preferred basis*: what makes certain observables of a quantum system remain stable and evolve predictably as the system interacts with its environment [10]? In other words, what properties allow particular quantum states to persist over time? From our potato example, we see that decoherence provides an answer to this question: the preferred states of the potato are those that generate the least entanglement with the environment under time evolution (pointer states).[11] This is a powerful statement about the nature of quantum measurement.

However, have we really solved the problem of preferred basis? Upon closer inspection, even Wojciech H. Zurek, the founder of the decoherence program, notices that an asterisk remains [11]: "one issue which has been often taken for granted is looming big, as a foundation of the whole decoherence program. It is the question of what are the 'systems' which play such a crucial role in all the discussions of the emergent classicality." Remember, entanglement itself is a relative concept—it depends on the division into subsystems within the overall quantum state. Sometimes, the division is not as clear as potato and photon. Ideally, we would be able to determine the preferred division into subsystems directly from the fundamental ingredients of quantum theory—a Hilbert space and Hamiltonian. Thus, decoherence has not made *the problem of preferred basis* disappear—it has simply pushed it back one additional layer to *the problem of preferred subsystems*.

Mathematically, dividing a quantum system into subsystems corresponds to factoring a Hilbert space (a vector space representing the system's possible states) into a particular tensor product structure (subsystem structure). What makes this question difficult is that there are many different ways to divide a given system into subsystems (there are many different tensor product structures for a given Hilbert space). Thus, in the decoherence framework—where we characterize the entanglement between subsystems—a tensor product structure (TPS) must be *assumed* for the Hilbert space.

---

entanglement is a relative concept, and furthermore that the dynamics of a chosen system (or system′) depend on the particular division of the overall system-environment into subsystems. With the introduction of Bloch balls in Chapter 2, this notion will likely become more clear.

[11] Position is not always the "preferred" observable of a quantum system. As discussed in Section 1.2, the preferred observable for an electron is energy, and the preferred states of the electron are its energy eigenstates.

That being said, we can impose an intuitive constraint on these TPSs in order to relax this assumption: we can require that the selected TPS is able to exhibit "quasi-classical" dynamics. If a TPS does not have the ability to possess classical-like pointer states, then it is not physically relevant or useful in describing processes involving decoherence; thus, this serves as a natural selection criterion. A variety of specific metrics have been developed to determine which TPSs best yield quasi-classical dynamics [3, 5, 12]. There are two main metrics of relevance to this thesis:

1. Robustness: Slow entanglement generation of a pointer state with the environment.
2. Predictability: Minimum spread in the uncertainty of a pointer observable over time.

According to Carroll and Singh [5], both metrics are required for a TPS to exhibit quasi-classical behavior. However, the relationship between these metrics is not well understood—thus, it is additionally worth investigating how the TPSs selected by each of these metrics compare.

Additionally, using either of these metrics, there remains the question of how different initial states within a TPS behave over time. For example, if a TPS has a particularly well-defined pointer state (a "quasi-classical TPS"), must it also have particularly unstable superposition states?[12]

In this chapter, we have given a broad overview of decoherence and motivated the key tasks for this thesis. We seek to investigate *the problem of preferred subsystems* left unanswered by the decoherence program and a series of adjacent questions. Specifically: Is there a particular division of a quantum system into subsystems that best yields quasi-classical dynamics? How does the answer to this question vary given different metrics of quasi-classicality? Furthermore, how do these metrics compare to each other, and in particular, are both robustness and predictability necessary to inform the factorization of a Hilbert space? Finally: do *quasi-classical* tensor product structures always exhibit a high variance in their entropy growth over different initial states (i.e., non-pointer states in that TPS will decohere quickly)?

To address these questions, I first formally define the concepts discussed thus far—including the metrics of predictability and robustness—in a mathematically precise way (Chapter 2). Next, I describe equivalent versions of these metrics as computationally-efficient cost functions, and personally derive novel analytic formulas for the mean and variance of each cost function (Chapter 3). Then, through the design and implementation of a machine learning algorithm, I search the space of possible tensor product structures for a system of qubits to find the TPSs that result in the most predictable and robust behavior (Chapter 4). Finally, I discuss the results of this investigation—comparing the TPSs selected by each metric and their implications in addressing *the problem of preferred subsystems* (Chapter 5). In the words of Zurek [11]: "a compelling explanation of what are the systems—how to define them given, say, the overall Hamiltonian in some suitably large Hilbert space—would be undoubtedly most useful."

[12] We investigate this question by using the variance of each metric, which will be described in the ensuing chapters.

Chapter 2

Theory

In this chapter, we make the concepts of Chapter 1 mathematically precise to lay a foundation for the algorithm that is the subject of this thesis. In doing so, we turn to the density matrix formalism of quantum mechanics, which allows us to capture both quantum and classical uncertainty. We first give a brief overview of density matrices and a particularly relevant operation in Section 2.1. Then, in Section 2.2, we introduce the Bloch ball—a useful tool for visualizing the states of qubits. Finally, throughout the rest of the chapter, we work through a more abstract example of three qubits (in parallel to our potato-environment from earlier) to build conceptual intuition for the mathematics introduced.

2.1 Density Matrices and the Partial Trace

Before beginning our discussion of density matrices, let's briefly clarify how we assemble the Hilbert space for our total system-environment (the one in which we are imposing a tensor product structure). This total composite Hilbert space $\mathcal{H}$ is formed by taking the *tensor product* of the system and environment Hilbert spaces:

$$\mathcal{H} = \mathcal{H}_{\text{sys}} \otimes \mathcal{H}_{\text{env}}. \tag{2.1}$$

The tensor product operation provides a general framework for representing joint states of a system and its environment. It is a standard mathematical method in quantum theory for combining subsystems into a larger whole.[1]

Now that we have solidified our understanding of tensor products, let's return to the potato-photon example of Chapter 1. Recall that it is impossible to assign a state vector to

[1]For the unfamiliar reader, we briefly show an example of the tensor product operation between two qubits: If $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ and $B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$, then their tensor product is a 4×4 matrix given by:

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B \\ a_{21}B & a_{22}B \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & a_{12}b_{11} & a_{12}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & a_{12}b_{21} & a_{12}b_{22} \\ a_{21}b_{11} & a_{21}b_{12} & a_{22}b_{11} & a_{22}b_{12} \\ a_{21}b_{21} & a_{21}b_{22} & a_{22}b_{21} & a_{22}b_{22} \end{pmatrix}.$$

a system of interest (e.g., the potato) when it is entangled with the environment (Figure 1.1). That being said, it would be incredibly useful to describe the state of our system alone since the environment typically consists of many degrees of freedom that are often inaccessible. Thus, it makes sense to incorporate classical probabilities into our formalism (mixed states), and average over the possible states of the environment when describing the final state of the system (the partial trace operation). We briefly formalize these concepts below.

Definition 1 (Density Matrix)**.** Consider a set of possible pure quantum states $|\psi_k\rangle \in \mathcal{H}$, each with classical probability p_k. We define the corresponding density matrix as

$$\rho \equiv \sum_k p_k |\psi_k\rangle\langle\psi_k|, \tag{2.2}$$

where $\sum_k p_k = 1$.

If there is no classical uncertainty (we can make a measurement of all parts of the composite-system[2]), we are in a pure quantum state ($p_i = 1$, $p_{k\neq i} = 0$ for some i) and our density matrix simplifies to

$$\rho = |\psi_i\rangle\langle\psi_i|. \tag{2.3}$$

On the other hand, when our composite-system does have classical uncertainty ($p_k \neq 1\ \forall k$), we occupy a *mixed state*.

When do mixed states become useful? Notably, when we only have access to measurements of a system that is entangled with its environment. Again, in this case, there is no way to represent the system alone with a state vector $|\psi_{\text{sys}}\rangle$ (see Figure 1.1). Since the environment is inaccessible, we have no way of knowing the exact state of the environment. Thus, we can instead average over the possible states of the environment to construct a mixed-state density matrix for the system called a *reduced density matrix*. This is done with the partial trace operation, and provides information about the system even though it cannot be represented as a pure quantum state.

Now for the mathematical definition of the partial trace. Consider an overall density matrix for our system-environment ρ and a basis set for the environment $\{\phi_k\}$. We can partial trace the environment out of the system-environment[3] with

$$\rho_{\text{sys}} = \text{Tr}_{\text{env}}(\rho) = \sum_k (\mathbb{I}_{\text{sys}} \otimes \langle\phi_k|)\rho(\mathbb{I}_{\text{sys}} \otimes |\phi_k\rangle), \tag{2.4}$$

where $\mathbb{I}_{\text{sys}}$ is the identity matrix whose dimension corresponds to our system.

Let's look at a simple example of the partial trace operation on a two-qubit system-environment, where one qubit is the system and the other is the environment. Consider an overall system-environment state:

[2]To be explicit: throughout this thesis, we will refer to our total Hilbert space as the "composite-system" and "system-environment" interchangeably. This is similar to our "potato-photon" from earlier.

[3]This is a common way to say that we trace over the environmental degrees of freedom while leaving the system degrees of freedom intact, yielding a reduced density matrix that describes the system alone.

$$\rho_{\text{Bell}} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \end{pmatrix} = \left(\begin{array}{c|c} P & Q \\ \hline R & S \end{array}\right)$$

This is the density matrix corresponding to the pure Bell state[4] $|\psi_{\text{Bell}}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Here, we have divided our density matrix into four 2×2 sub-blocks for pedagogical purposes.

Let's "trace out" the second qubit (the environment), yielding the reduced density matrix for the first qubit (the system):

$$\rho_{\text{sys}} = \text{Tr}_{\text{env}}(\rho_{\text{Bell}}) = \sum_{k=0}^{1} \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \langle k| \right] \rho_{\text{Bell}} \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes |k\rangle \right] = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$$

Equivalently, we can think of this as taking the trace over each sub-block individually [13]:

$$\rho_{\text{sys}} = \text{Tr}_{\text{env}}(\rho_{\text{Bell}}) = \begin{pmatrix} \text{Tr}(P) & \text{Tr}(Q) \\ \text{Tr}(R) & \text{Tr}(S) \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \tag{2.5}$$

This is the maximally mixed state, indicating that tracing over the environment has turned the pure entangled state (ψ_{Bell}) into a mixed state for the system alone. The off-diagonal terms[5] of this reduced density matrix are zero, indicating a complete loss of coherence. This happens because the system is entangled with an environment we can't access, and thus—from the system's perspective—our reduced density matrix is simply a probabilistic mixture of classical states. In this case, the system has an equal probability of being in the $|0\rangle$ and $|1\rangle$ states—given, of course, we lack knowledge about our entangled second qubit (the environment).[6]

The reduced density matrix allows us to model entangled systems by themselves, which will prove tremendously useful when further characterizing entanglement and quasi-classicality.

2.2 The Bloch Ball

In the previous section, we described the mathematics of density matrices and partial traces. However, conceptual intuition for these states and operations may remain a bit opaque. To help represent these density matrices, we turn to the Bloch ball—a way to visualize both mixed and pure qubit states.

[4]The four maximally entangled two-qubit states are called Bell states. They are $|\Phi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle)$ and $|\Psi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle)$.

[5]The off-diagonal terms of a density matrix are commonly referred to as *coherences*, and they encode quantum superpositions by capturing phase relationships between basis states.

[6]The key here is that the partial trace reflects a lack of access to the environmental state. If we knew the second qubit (the environment) was in the $|0\rangle$ state, for example, then we would immediately know that the first qubit (the system) was in the $|0\rangle$ state as well. However, without access to the environment, our reduced system state appears as a classical probabilistic mixture.

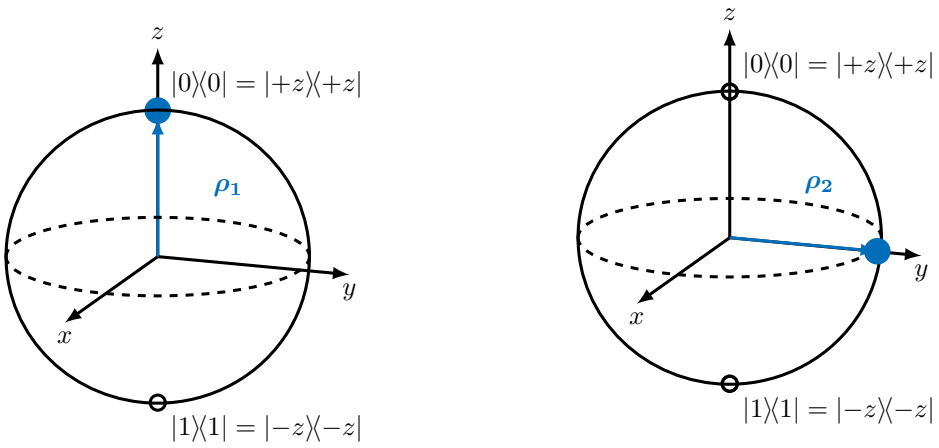


Figure 2.1: Two states on the Bloch ball: $\rho_1 = |0\rangle\langle 0|$ and $\rho_2 = |+y\rangle\langle +y|$. As both states are pure, they lie along the surface of the Bloch ball.[7]

To represent a qubit state on the Bloch ball,[8] we associate it with a vector $\vec{r} \in \mathbb{R}^3$ called the *Bloch vector*.[9] The state of the qubit with Bloch vector $\vec{r}$ is given by:

$$\rho = \frac{1}{2}(\mathbb{I} + \vec{r} \cdot \vec{\sigma}),$$

where $\vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ is the vector of Pauli matrices and $|\vec{r}| \leq 1$. Pure states correspond to Bloch vectors of unit length ($|\vec{r}| = 1$) and lie on the surface of the ball, while mixed states have $|\vec{r}| < 1$ and lie strictly within the interior. This representation provides a clear geometric picture for the state of a qubit.

A couple of different qubit states are shown in Figure 2.1:

$$\rho_1 = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad \rho_2 = |+y\rangle\langle +y| = \frac{1}{2}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix}.$$

The Bloch ball visualization makes it immediately clear that both of these states are pure since they lie on the ball's surface.

Since our Bloch ball has an interior, it can represent mixed states as well. In Figure 2.2, we see two mixed states represented: some random partially mixed state ρ_3, and our

[7] To create the Bloch balls in this figure and throughout this thesis, I modified existing code from Stack Exchange [14].

[8] For those familiar with the Bloch sphere, this is simply an extended version where the interior of the sphere represents mixed states of the qubit. The surface of the Bloch ball is the Bloch sphere.

[9] To recover the state of a qubit from its density matrix, one can use

$$\vec{r} = (x, y, z) = \big(\mathrm{Tr}[\rho\sigma_x], \mathrm{Tr}[\rho\sigma_y], \mathrm{Tr}[\rho\sigma_z]\big).$$

Here, the components are just the expectation values of the Pauli matrices in the state ρ, which tells us how significantly the qubit state is aligned with each of the x, y, and z spin directions.

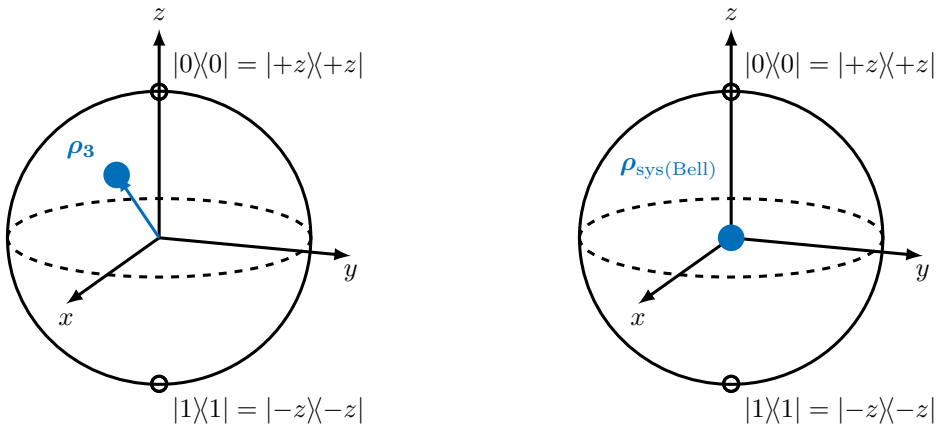


Figure 2.2: Two states on the Bloch ball: some arbitrary mixed state ρ_3 and our $\rho_{\text{sys(Bell)}}$ from earlier. As both states are mixed, they lie within the interior of the Bloch ball.

$\rho_{\text{sys(Bell)}} = \text{Tr}_{\text{env}}(\rho_{\text{Bell}})$ from the previous section. Since $\rho_{\text{sys(Bell)}}$ is maximally mixed, its Bloch vector is zero and the state is located at the origin of the Bloch ball. Thus, our Bloch vector allows us to immediately identify states as maximally mixed as well.

There is one particularly important observation to make about these Bloch balls before moving on to our three qubit example: how far a state is away from the surface of the Bloch ball indicates how mixed the state is![10]

2.3 Entanglement

Now that we are equipped with the new tools of density matrices and Bloch balls, we can continue our investigation of decoherence. Thus, let's return to the concepts introduced in Chapter 1 and begin to formalize them mathematically.

Consider a composite-system of three qubits[11] forming an eight-dimensional[12] Hilbert Space ($\mathcal{H} \cong \mathbb{C}^8$). Let's say that we as observers only have access to one qubit of information, and thus decompose our Hilbert space into one qubit for our system $\mathcal{H}_{\text{sys}} = \mathcal{H}_1 \cong \mathbb{C}^2$ and two qubits for our environment $\mathcal{H}_2 \cong \mathbb{C}^2$ and $\mathcal{H}_3 \cong \mathbb{C}^2$. Notice again that this division of system and environment assumes a tensor product structure on our Hilbert space ($\mathcal{H} \rightarrow \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$), which we will explicitly describe later in the chapter.[13]

[10] If this is not immediately evident, we can be good physicists and examine the limiting cases. When a state is pure (no mixing), it lies on the surface of the Bloch ball. When a state is maximally mixed, it lies at the center of the Bloch ball. Thus, a state somewhere in between (elsewhere on the interior), is partially mixed.

[11] This example is particularly relevant as three interacting qubits form the system-environment for one of our algorithm's simulations. For a physical picture, consider three photons entangled in their polarizations, but where two of the photons have been scattered into an inaccessible environment.

[12] Each qubit is isomorphic to $\mathbb{C}^2$ because a qubit is a two level system, and has a complex amplitude associated with each level. Thus, three qubits constitute $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2 \cong \mathbb{C}^8$.

[13] Here we have divided the environment itself into two subsystems $\mathcal{H}_2$ and $\mathcal{H}_3$. We say that both $\mathcal{H}_2$

Analogous to our potato in superposition from Section 1.1, consider a system in the initial state $|\psi^{(0)}\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$, with the corresponding density matrix $\rho_{\text{sys}}^{(0)} = |\psi^{(0)}\rangle\langle\psi^{(0)}| = \frac{1}{2}(|0\rangle\langle 0| + i\,|0\rangle\langle 1| - i\,|1\rangle\langle 0| + |1\rangle\langle 1|)$. The initial state of our environment is termed the *environmental ready state*. There are many choices for the environmental ready state (which will largely affect the dynamics), but a typical and convenient choice is the canonical ensemble in the infinite temperature limit [15], which we will use in our algorithm. This is simply the maximally mixed environmental state—thus, with this ready state, we define each individual environmental qubit as

$$\rho_{\text{env (1q)}} = \lim_{t\to\infty}\left(\frac{e^{-\frac{H}{kt}}}{\text{Tr}(e^{-\frac{H}{kt}})}\right) = \frac{1}{2}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}. \tag{2.6}$$

This is the maximally mixed state of each environmental qubit (similar to $\rho_{\text{sys(Bell)}}$), and thus the environment as a whole. It is useful to pictorially represent this system-environment with the Bloch balls introduced in the previous section, where the first Bloch ball is our system and the second two compose our environment (Figure 2.3).

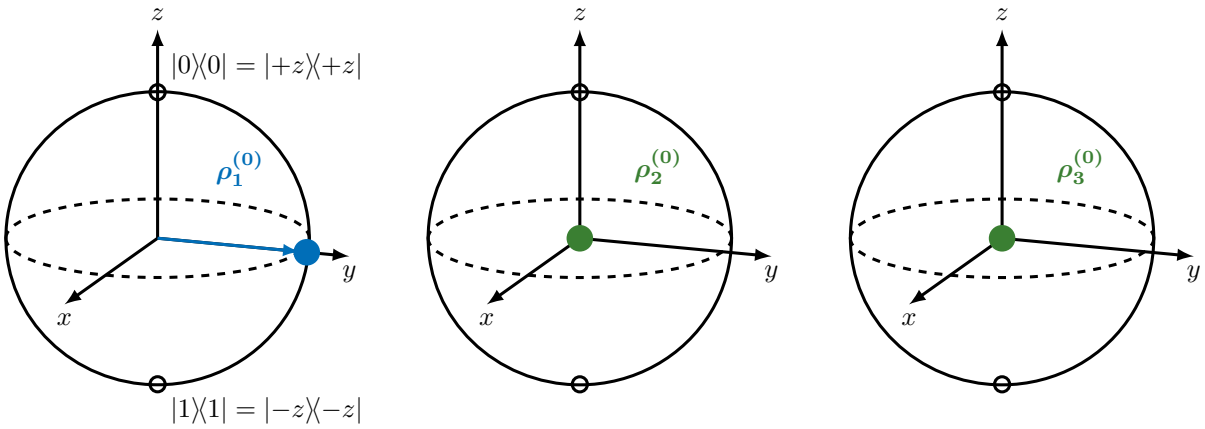


Figure 2.3: Our system-environment depicted by three Bloch balls. Our system occupies the initial state $|\psi^{(0)}\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$, which lies on the y-axis of the Bloch ball. The environmental ready state is maximally mixed, and thus the latter two balls display their states at the origin.

A natural next step to analyze this system is to determine how entangled it is with the environment. We can do this by introducing the linear entanglement entropy [5]:

Definition 2 (Linear Entanglement Entropy for our System-Environment)**.** Consider $\mathcal{H} = \mathcal{H}_{\text{sys}} \otimes \mathcal{H}_{\text{env}}$, a tensor product structure on $\mathcal{H}$. Let ρ be a density matrix of $\mathcal{H}$ for our total

and $\mathcal{H}_3$ compose the environment since we are only concerned with the dynamics of states within $\mathcal{H}_1$, our system's Hilbert space.

system-environment. We define the linear entanglement entropy as

$$S_{\text{ent}}(\rho) \equiv 1 - \text{Tr}(\rho_{\text{sys}}^2), \tag{2.7}$$

where ρ_{sys} is the reduced density matrix of our system.

Because $0 \leq \text{Tr}(\rho^2) \leq 1$ for any density matrix, it follows that $0 \leq S_{\text{ent}}(\rho) \leq 1$ as well. If $S_{\text{ent}}(\rho) = 0$, then the reduced system state is pure and the system and environment remain unentangled ($\rho = \rho_{\text{sys}} \otimes \rho_{\text{env}}$). In contrast, if $S_{\text{ent}}(\rho) > 0$, then the system has become mixed and lost coherence due to its interaction with the environment.

One can see that our initial state is unentangled since we can write the composite-system density matrix as $\rho = \rho_1 \otimes \rho_2 \otimes \rho_3$. We can also confirm this by explicitly computing the entanglement entropy. The initial composite-system density matrix is given by:

$$\rho^{(0)} = \rho_1^{(0)} \otimes \rho_2^{(0)} \otimes \rho_3^{(0)} = \frac{1}{2}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix} \otimes \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \frac{1}{8}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix} \otimes \mathbb{I}_4$$

where $\mathbb{I}_4$ is the 4×4 identity matrix. To determine $\rho_1^{(0)}$ of our system, we take the partial trace over the environment (qubits 2 and 3):

$$\rho_1^{(0)} = \text{Tr}_{\text{env}}(\rho^{(0)}) = \text{Tr}_{2,3}(\rho^{(0)}) = \text{Tr}_{2,3}\left[\frac{1}{8}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix} \otimes \mathbb{I}_4\right] = \frac{1}{2}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix}$$

recovering our original $\rho_1^{(0)}$ as expected. Plugging this into the formula for S_{ent}, we find

$$S_{\text{ent}}(\rho^{(0)}) = 1 - \text{Tr}\big[(\rho_1)^2\big] = 1 - \text{Tr}\big[(\rho_1^{(0)})^2\big] = 1 - 1 = 0, \tag{2.8}$$

confirming that our initial state is unentangled.

2.4 Decoherence and Pointer States

However, what happens when our initial superposition state evolves in time? Let's consider an example where we time-evolve an initial system superposition state $|\psi_{\text{sys}}^{(0)}\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ according to some Hamiltonian. This is analogous to our potato superposition example from Section 1.1. For this example, let's use a central spin Hamiltonian to evolve our three-qubit system. In our case, this Hamiltonian describes a single central spin (our system qubit) coupled to two surrounding environmental qubits (the "spin bath"). We can construct a relatively simple central spin Hamiltonian [16] with

$$H_{cs} = \frac{1}{2}\sigma^z \otimes \sum_{i=1}^{n-1} g \cdot \sigma_i^z, \tag{2.9}$$

where σ_i^z is the Pauli-z matrix for the ith bath spin, g is the coupling strength between the central spin and the spin bath, and n is the total number of qubits. Taking $g = 1$ for simplicity, our Hamiltonian[14] becomes

$$H_{cs} = \frac{1}{2}\sigma^z \otimes \sum_{i=1}^{n-1} \sigma_i^z. \tag{2.10}$$

Now let's act this Hamiltonian on our initial state $\rho^{(0)} = \left|\psi_{\text{sys}}^{(0)}\right\rangle\!\left\langle\psi_{\text{sys}}^{(0)}\right|$. To do this, we apply the time evolution operator $U(t) = e^{-iH_{cs}\tau/\hbar}$, where we use the characteristic time[15] $\tau = t_{\text{char}} = \frac{1}{\|H\|}$ as the length of time evolution. To time-evolve our initial state, we use

$$\begin{aligned}\rho^{(f)} &= U(t) \cdot \rho^{(0)} \cdot U^\dagger(t) \\ &= \mathrm{e}^{-iH\tau} \cdot \rho^{(0)} \cdot \mathrm{e}^{iH\tau},\end{aligned}$$

where we have adopted natural units and set $\hbar = 1$. Tracing over the environment to obtain a final reduced density matrix for our system,

$$\rho_{\text{sys}}^{(f)} = \text{Tr}_{\text{env}}(\rho^{(f)}) = \text{Tr}_{\text{env}}\left[\mathrm{e}^{-i\tau H} \cdot \rho^{(0)} \cdot \mathrm{e}^{i\tau H}\right] \approx \begin{pmatrix} 0.5 & 0.3851 \\ 0.3851 & 0.5 \end{pmatrix}. \tag{2.11}$$

One can check that $\rho^2 < 1$ for this density matrix, which indicates that our final system state is mixed. Lastly, computing the final entanglement entropy, we obtain:

$$S_{\text{ent}}(\rho^{(f)}) = 1 - \text{Tr}\left[(\rho_{\text{sys}}^{(f)})^2\right] \approx 1 - .797 = .203 \tag{2.12}$$

Thus, entanglement has been generated between the system qubit and the environmental qubits (just like when our potato began in a superposition state)! We can visually represent the new state of the composite-system with Bloch balls, where it can be seen that the state of our system has changed over time (Figure 2.4). Since our initial state was not robust to entanglement with the environment, we know that it is not a pointer state and time evolution will cause it to decohere.

Now what if we were to start the system qubit in one of its pointer states, say $|\psi^{(0)}\rangle = |0\rangle$ with $\rho_1^{(0)} = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$? From Section 1.2, we know that we should expect a pointer state like this to be robust to entanglement generation, and thus the system-environment should remain unentangled over time. Calculating $\rho_1^{(f)}$ and $S_{\text{ent}}^{(f)}$ (similar to the previous case), we obtain

$$\rho_1^{(0)} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \rho_1^{(f)}, \qquad S_{\text{ent}}^{(0)} = 0 = S_{\text{ent}}^{(f)}. \tag{2.13}$$

Thus, the system's pointer state persists over time, and the system-environment maintains zero entanglement entropy. As displayed in Figure 2.5, the system state does not drift away from the surface of the Bloch ball.

[14]The central spin Hamiltonian of Equation 2.10 is particularly relevant as this is the primary Hamiltonian used for our results in Chapter 5.

[15]The characteristic time is a good estimate for the timescale over which decoherence significantly affects the system's behavior, making it a natural choice to use in our simulations.

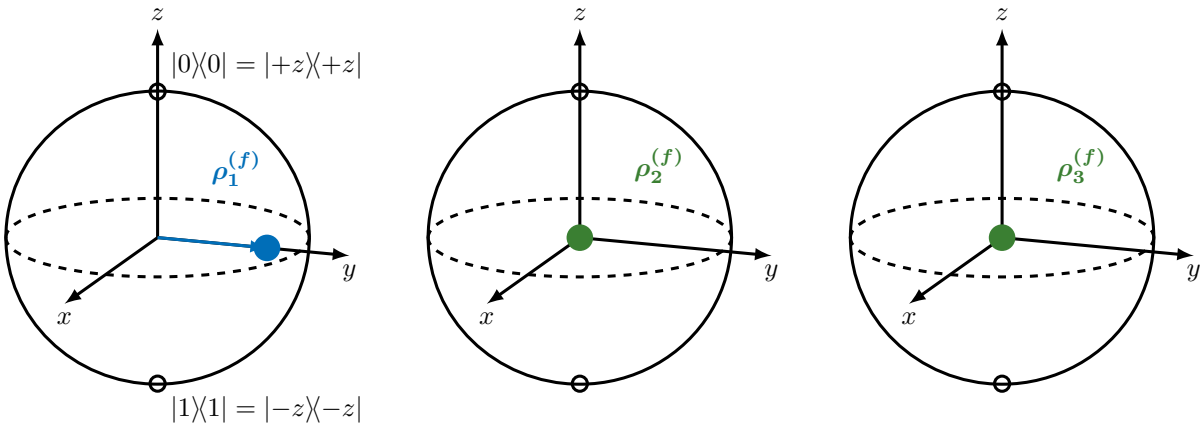


Figure 2.4: The final system-environment depicted by three Bloch balls. The system has drifted slightly towards the origin, indicating that it has entered a mixed state. Thus, the system and environment are now entangled.

2.5 Tensor Product Structures

The central aim of this thesis is to explore the space of tensor product structures (TPSs), and we are now ready to formalize TPSs mathematically. This will allow us to move forward with the implementation of our algorithm.

Recall from our three-qubit example (Sections 2.2 and 2.3) that we worked within a specific TPS on the composite system: $\mathcal{H} \to \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$. This choice helped us to define entanglement, which plays an important role in our metrics for quasi-classicality (which will be described in Section 2.6). However, remember that we would like to determine the TPSs that best yield quasi-classical dynamics given only the structure of a Hilbert Space (for our three-qubit example, this is $\mathcal{H} \cong \mathbb{C}^8$) and a Hamiltonian. Thus, a precise definition of tensor product structures is necessary before we can begin our search for quasi-classical dynamics over a large space of TPSs.

To construct a distinct TPS on a Hilbert space ($\mathcal{H} \cong \mathbb{C}^8$), we use special unitary transformations[16] following Cotler [3]:

$$\hat{U} : \mathcal{H} \to \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes ... \tag{2.14}$$

where $\hat{U} \in SU(n)$. These transformations must be unitary to preserve inner products and norms, as required by quantum mechanics. We focus on *special* unitary transformations in

[16]For the unfamiliar reader: A matrix M is"special" if $\det[M] = 1$ and "unitary" if $M^\dagger M = \mathbb{I}$. Additionally, the group of all $n \times n$ special unitary matrices is denoted $SU(n)$. This group is continuous and thus there are infinite matrices $\hat{U} \in SU(n)$ for any given value of n. The continuous nature of this group makes it computationally difficult to search the entire space of possible tensor product structures (as we do in our algorithm).

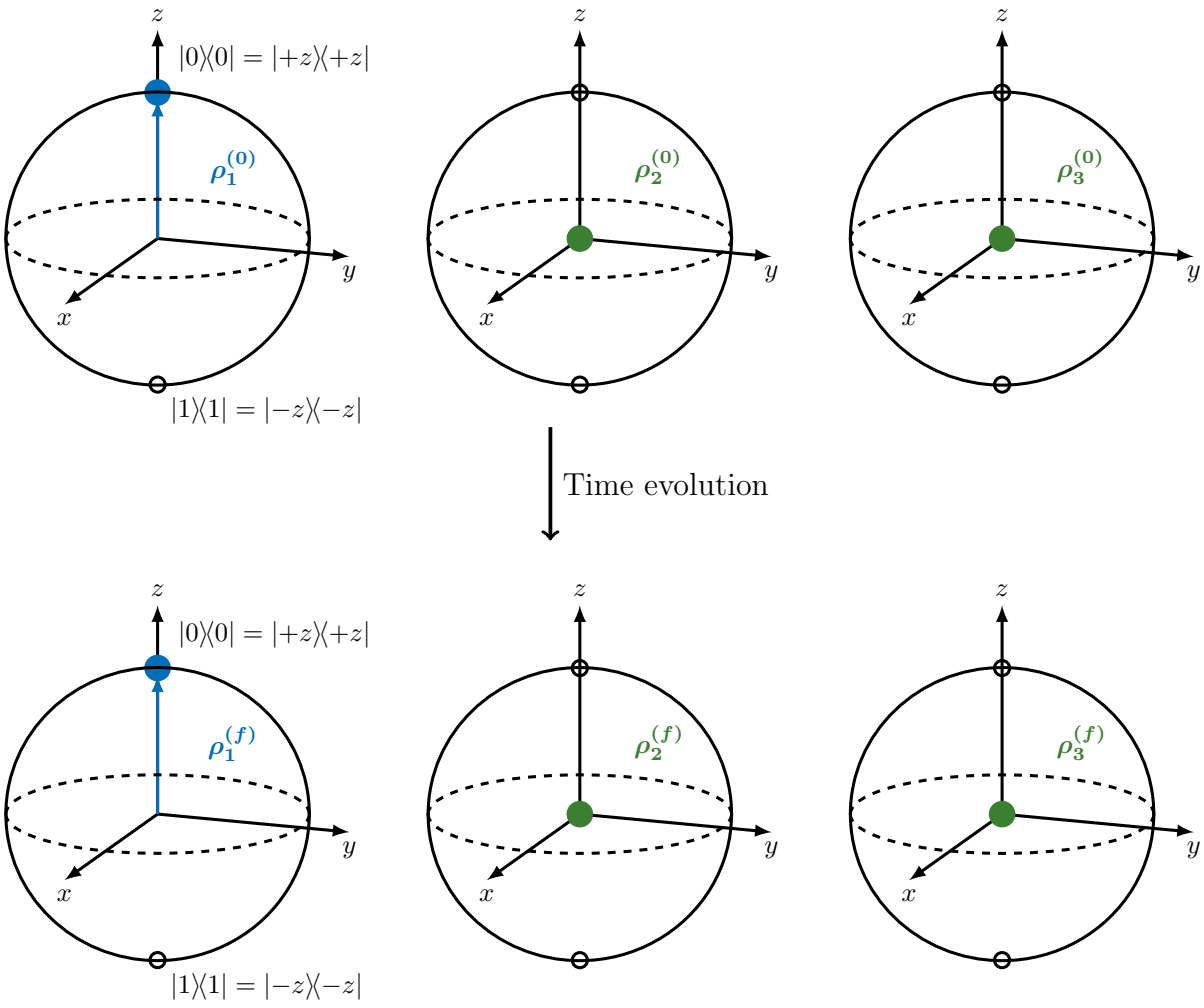


Figure 2.5: The evolution of a pointer state over time, as represented with three Bloch balls. It is clear that the system state doesn't change and no entanglement is generated between the system and the environment. The pointer state remains predictable over time and robust to entanglement generation with the environment.

particular because non-special unitaries differ by only an overall phase factor, which is not physically observable in quantum mechanics.

To help clarify the effect of these transformations, let's return to our three-qubit example. Consider the same initial state[17] of our composite-system from Section 2.3:

$$\rho^{(0)} = |\Psi\rangle\langle\Psi| = \frac{1}{8}\begin{pmatrix}1 & -i\\ i & 1\end{pmatrix} \otimes \mathbb{I}_4. \tag{2.15}$$

As will soon be described, depending on how exactly we divide up our composite-system into subsystems (represented by each individual Bloch ball), we will obtain different initial states for each qubit. Some divisions of an overall composite-system state may result in a high level of initial entanglement between the qubits. On the other hand, there may be other divisions of the same overall composite-system state that result in zero initial entanglement between the qubits (recall our system′ and environment′ from Section 1.3). This should become more clear in the following example.

First, let's consider the same division into subsystems from Section 2.3 (now represented by the special[18] unitary transformation $\hat{U} : \mathcal{H} \to \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$) that splits our Hilbert Space into a "system" (qubit 1) and "environment" (qubits 2 and 3). This $\hat{U}|\Psi\rangle$ results in the same initial superposition state from Section 2.4, namely:

$$\rho_1 = \frac{1}{8}\begin{pmatrix}1 & -i\\ i & 1\end{pmatrix}, \qquad \rho_2 = \rho_3 = \frac{1}{2}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}.$$

Now consider a different division into subsystems represented by some randomly constructed unitary transformation $\hat{U}' \in SU(n)$. This transformation defines a new tensor product structure, implementing a division of our Hilbert space into qubits 1′, 2′, and 3′ ($\hat{U}' : \mathcal{H} \to \mathcal{H}_{1'} \otimes \mathcal{H}_{2'} \otimes \mathcal{H}_{3'}$). We can think of this as expressing the same overall state ρ—originally written with respect to the product basis[19] of qubits 1, 2, and 3—as a new matrix ρ' in the product basis of qubits 1′, 2′, and 3′.

It is important to emphasize that this is the exact same overall state of our composite-system—we are simply altering the division of subsystems, and thus the representation of this state. After tracing over our new environment (qubits 2′ and 3′) and computing the

[17] Note that even though ρ here is represented with a tensor product, no TPS has been assumed because we are considering the entire 8×8 matrix (our total composite-system). In this case, the tensor product symbol is simply a notational shorthand.

[18] From this point forward, we will drop the word "special" and just refer to the transformations $U \in SU(n)$ as "unitary transformations."

[19] A *product basis* refers to a basis for the total Hilbert space constructed by taking tensor products of the basis states for each subsystem. For example, if each qubit has basis states $\{|0\rangle, |1\rangle\}$, then the product basis for a three-qubit system is $|i\rangle \otimes |j\rangle \otimes |k\rangle$, where $i, j, k \in \{0, 1\}$. Each distinct tensor product structure defines its own notion of subsystems, and thus induces a different product basis for the overall state.

resulting entanglement entropy, we obtain:

$$\rho_{\text{sys}'} = \rho_{1'} \approx \begin{pmatrix} .501 & .047 - .061i \\ .047 + .061i & .499 \end{pmatrix}$$

$$S_{\text{ent}}(\rho') = 1 - \text{Tr}\left[(\rho_{\text{sys}'})^2\right] \approx 1 - 0.512 = 0.488$$

We can see that with this new split of our Hilbert space (into qubits $1'$, $2'$, and $3'$), not only is the state of each qubit different (see Figure 2.6), but the system and environment are now entangled! Remember, this is despite the overall state of our composite-system, ρ, being the same in both cases. Thus, it is clear that the way in which we divide a Hilbert space into subsystems has a large impact on the dynamics observed and the ability of that particular division to yield quasi-classical properties (e.g., robustness, predictability).

Consider, however, another unitary transformation $\hat{U}'' = (e^{-i\frac{\pi}{4}\sigma_z} \otimes e^{-i\frac{\pi}{8}\sigma_z} \otimes e^{-i\frac{\pi}{2}\sigma_z}) \cdot \hat{U}$ that is similar to $\hat{U}$ but induces a phase shift of $\frac{\pi}{4}$ radians for qubit 1, $\frac{\pi}{2}$ radians for for qubit 2, and π radians for qubit 3, all about the z-axis.[20] This simply corresponds to rotating our Bloch balls by 90°, 45°, and 180° around the z-axis appropriately (Figure 2.7).[21]

Note that $\hat{U}$ and $\hat{U}''$ both result in the same exact division between our qubits and are related to each other by a rotation of each sub-factor. Since we can always redefine the coordinate system of each Bloch ball (e.g., rotate the coordinate system of our first qubit by $-\frac{\pi}{2}$ radians), this rotation of sub-factors should be irrelevant. By calculating the entanglement entropy

$$S_{\text{ent}}(\rho'') = 1 - \text{Tr}(\rho^2_{\text{sys}''}) = 1 - \text{Tr}\left[\begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}^2\right] = 1 - 1 = 0 \tag{2.16}$$

we confirm that $\hat{U}''$ results in an unentangled system and environment, just as $\hat{U}$ does. Thus, these two unitary operators impose the same TPS on our Hilbert Space! In fact, as long as each tensor factor is rotating each individual Bloch ball separately (regardless if they are equivalent rotations or not), the division between our subsystems—and thus, the relevant dynamics—will remain unchanged. This process of rotating each tensor factor individually is called a *local* rotation, and when two unitary operators differ only by local rotations on their respective subsystems, we say they are related by a *local unitary transformation*.

However, there is another type of transformation that also results in the same division of subsystems. Consider a transformation $\hat{U}''' = \Pi_{12}\hat{U}$ where Π_{12} permutes the first two sub-factors (qubit 2 becomes the system, qubit 1 becomes part of the environment). As shown in Figure 2.8, we have the same split of qubits (this is merely a relabeling of sub-factors),[22]

[20] The general formula for a rotation of θ around an axis $\vec{n}$ on the Bloch ball is $R_{\vec{n}}(\theta) = \exp[-i\frac{\theta}{2}\,\vec{n}\cdot\vec{\sigma}]$.

[21] Notice that since environmental qubits 2 and 3 occupy the maximally mixed state in Figure 2.7, the qubit states themselves are invariant under local rotations. This only happens to be true for the maximally mixed state (Bloch vector at the origin), which makes our choice of the infinite-temperature limit for our environmental ready-state particularly convenient.

[22] Though relabeling changes how entangled this particular state of our system is with the environment, we could easily find a system state that is unentangled since we have the same division of qubits as before.

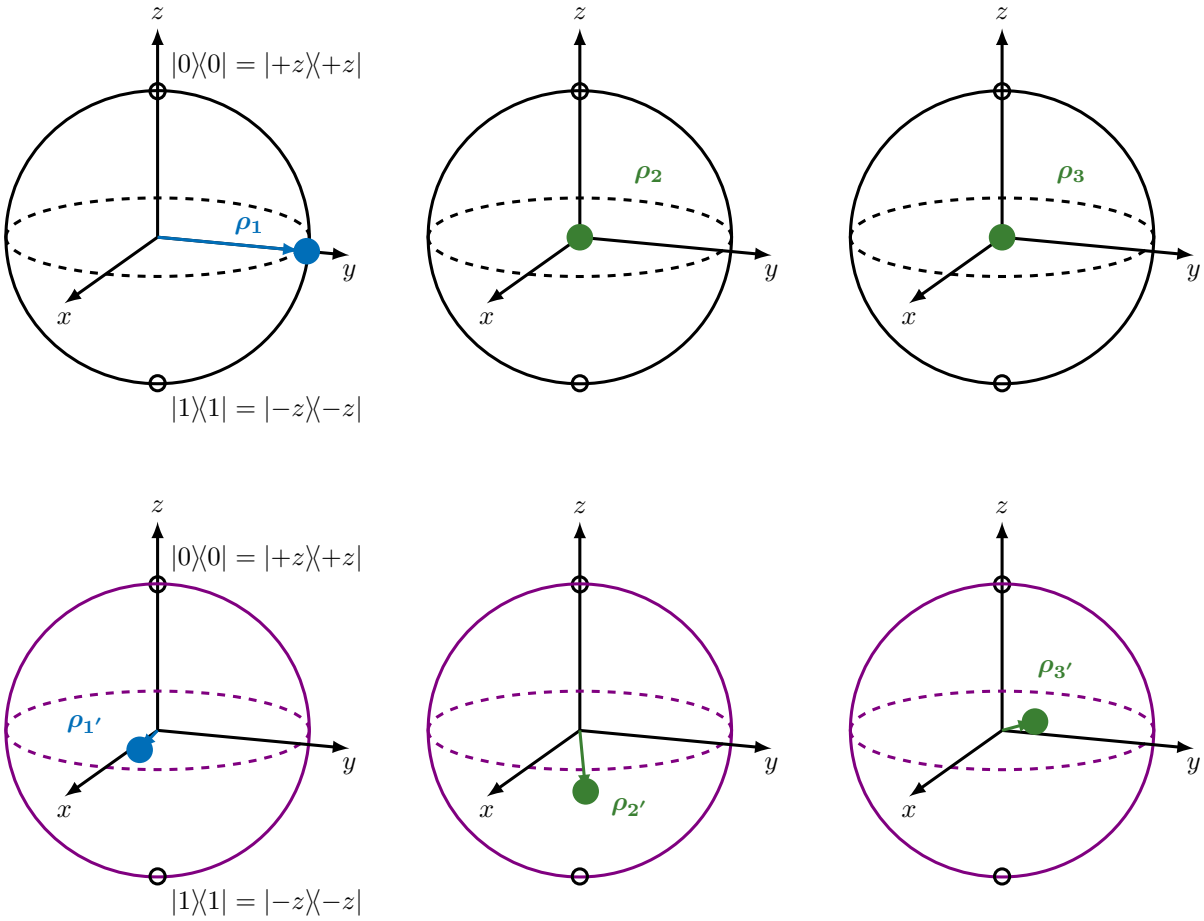


Figure 2.6: (Top, black spheres): The division of the overall state ρ into qubits 1, 2, and 3, with $\hat{U} : \mathcal{H} \to \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$. Here, our system and environment remain unentangled. (Bottom, purple spheres): The division of ρ into qubits $1'$, $2'$, and $3'$, with $\hat{U}' : \mathcal{H} \to \mathcal{H}_{1'} \otimes \mathcal{H}_{2'} \otimes \mathcal{H}_{3'}$. With this new division, our states change, and notably, our system state becomes mixed resulting in non-zero entanglement entropy. The system and environment of the same composite-system state ρ are now entangled.

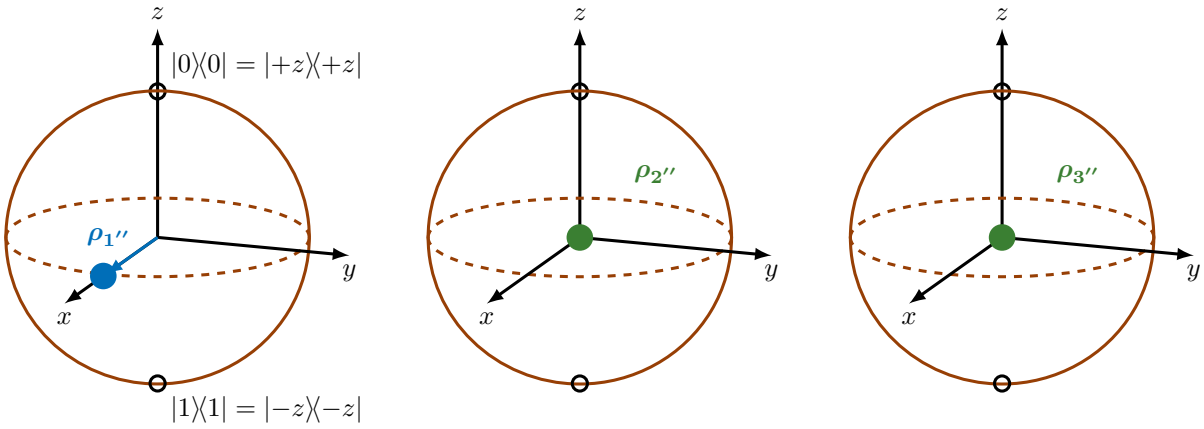


Figure 2.7: A tensor product structure $\hat{U}'' : \mathcal{H} \to \mathcal{H}_1'' \otimes \mathcal{H}_2'' \otimes \mathcal{H}_3''$ represented with Bloch balls. Since $\hat{U}$ and $\hat{U}''$ are related by a rotation of each sub-factor (local rotation), we can see that these transformations result in the same system-environment split ($\mathcal{H}_1'' \otimes \mathcal{H}_2'' \otimes \mathcal{H}_3'' \cong \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$).

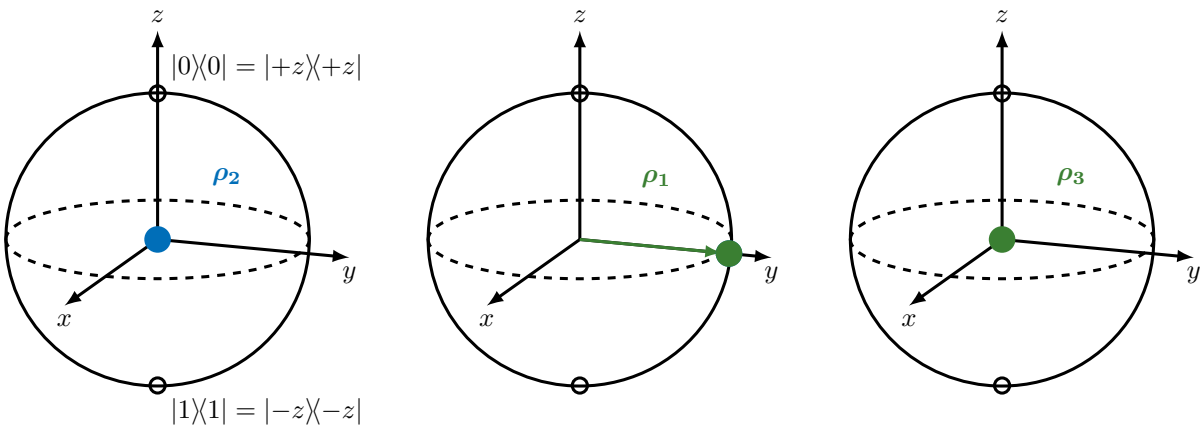


Figure 2.8: A tensor product structure $\hat{U}''' = \Pi_{12}\hat{U}$ that is similar to $\hat{U}$ but induces a permutation on qubits 1 and 2. This is simply a relabeling of tensor factors—making qubit 2 the system and qubit 1 part of the environment—and results in the same split of qubits as was induced by $\hat{U}$. In other words: the order of subsystems may change, but the underlying division into subsystems remains the same

and thus we would like to define TPSs in such a way that unitary transformations which are related by permutations are also considered equivalent.

So, in defining tensor product structures, we consider an equivalence class of special unitary operators related by both local unitaries and permutations:

Definition 3 (Tensor Product Structure)**.** Let T denote a tensor product structure. We say that T is an equivalence class of special unitary matrices $\hat{U} \in SU(n)$ such that $\hat{U}_i$ and $\hat{U}_j$ are considered equivalent if and only if they are related by a local unitary transformation, L, or a permutation of sub-factors, Π.[23]

Thus, to find two *distinct* TPSs T_i and T_j, we can induce two unitary transformations on our Hilbert space $\hat{U}_i, \hat{U}_j \in SU(n)$ such that $\hat{U}_i \neq \Pi(L(\hat{U}_j))$ for all local unitary transformations L and permutations of tensor factors Π. Notice that thus far, we have only discussed unitary transformations as acting on a Hilbert space to impose a TPS, all while keeping our Hamiltonian the same. However, there is an equivalent way to formulate this (think active vs passive transformations): we can vary the *Hamiltonian* by local unitaries and permutations while keeping the Hilbert space the same [3]. The key point is that whether we vary the Hilbert space factorization while keeping the Hamiltonian fixed, or vary the Hamiltonian while keeping the Hilbert space fixed, the effect is essentially the same. Since this latter formulation reduces computational time [17], to find distinct TPSs we will equivalently redefine T:

Definition 4 (Operational Tensor Product Structure)**.** Let T denote a tensor product structure and H a Hamiltonian. We define

$$\{T \equiv \Pi(L(H)) \mid L \in U_{LU}, \Pi \in U_P\} \tag{2.17}$$

where U_{LU} is the space of local unitary transformations and U_P the space of permutations.

Thus, mathematically a tensor product structure is simply the set of all Hamiltonians that are related to each other by local unitary transformations (local rotations of their Bloch balls) or permutations of their tensor factors (relabelings of their Bloch balls). Definition 4 is the operational definition of TPSs that we utilize in our algorithm.

With this definition in hand, we can identify when two TPSs are distinct by checking whether their corresponding Hamiltonians are related by local unitary transformations or by permutations of tensor factors. However, recall that our central goal is not merely to identify TPSs, but to determine which TPSs best yield quasi-classical dynamics. Thus, before discussing our cost functions and computational search, we must formally define the metrics of quasi-classicality that we will be using.

[23]For the group theory inclined: a tensor product structure is the *orbit* of a Hilbert space factorization under local unitaries and permutations.

2.6 Metrics

The first of the two metrics that we will be investigating is robustness: a measure of how entanglement spreads over time. To determine how well a given TPS can yield robust system states, we initialize our system in an unentangled state (for whatever TPS we are working in) and determine how quickly the system entangles with the environment. Following Carroll and Singh [5], we call this initial state the *candidate pointer observable*.[24]

In doing this, as in Section 2.4, we allow our initial state to evolve for the characteristic time $\tau = t_{\text{char}}$ and compute the resulting entanglement entropy S_{ent}. If the candidate pointer observable is truly a well-defined pointer state for the TPS, then little entanglement will be generated (think of our potato in a pointer state) and we will have a low value of S_{ent}. Thus, we consider low entanglement entropy to be a "good" score, and would like the algorithm to minimize S_{ent} in its search for quasi-classical tensor product structures.

Note, however, that we want a single score of classicality for all local unitary transformations and permutations of our Hamiltonian H. Since S_{ent} will, in general, vary over local unitaries and permutations, we must come up with a well-defined way to consider only a single calculation of S_{ent} for the entire TPS. We do this as follows. First considering permutations, we use the highest score of a given permutation of our Hamiltonian—this is because if we find a TPS seeming to exhibit quasi-classical dynamics (a "qcTPS"), we want it to be quasi-classical regardless of the way that we happen to label our qubits. Additionally, we select the lowest score over local unitary transformations of our Hamiltonian—this is because we simply want to determine if there *exists* a way of defining the axes on our Bloch balls such that the system state acts quasi-classically. With these considerations in mind, we make the following definitions:

Definition 5 (Robustness Score)**.** Consider the entanglement entropy $S_{\text{ent,k}}$ generated over the characteristic time $\tau = t_{\text{char}}$ with the k^{th} qubit of our composite-system labeled as the system. We define the robustness score, S_{rob}, as

$$S_{\text{rob}} \equiv \min_{L \in U_{LU}} (\max_{k}(S_{\text{ent,k}})) \tag{2.18}$$

With the above definition, we have a single notion of robustness for any given TPS that we use in our computational search. In Chapter 3 and in our algorithm, we refer to this robustness score S_{rob} of our evolved system as "Cost Function 1."

Consider now our second metric: predictability. This is the measure of how well a given pointer state for a TPS stays constant over time. First, we need a notion of pointer entropy for predictability, analogous to entanglement entropy for robustness. Following Carroll and Singh [5], we define:

[24] In our algorithm, our candidate pointer observable is always $\rho^{(0)} = |0\rangle\langle 0|$, where our system qubit has the basis $\{|0\rangle, |1\rangle\}$. We can assume this initial state without loss of generality since any other pure state on the Bloch sphere can be reached via a local rotation of our system qubit (which is defined to be within the same TPS).

Definition 6 (Pointer Entropy of a System Qubit)**.** Let ρ_{sys} be the reduced density matrix of our system written in the basis $\{|0\rangle, |1\rangle\}$. We define the pointer entropy of this state as

$$S_{\text{pointer}}(\rho_{\text{sys}}) \equiv 1 - \rho_{00}^2 - \rho_{11}^2, \tag{2.19}$$

where $\rho_{ii} = \langle i| \rho_{\text{sys}} |i\rangle$.

Here, ρ_{00} and ρ_{11} are simply the diagonal entries of ρ_{sys}. The pointer entropy indicates how far the system is from being in a definite eigenstate of its pointer observable (e.g., position).

Furthermore, analogous to S_{rob}, we want one score of predictability over all local unitary transformations and permutations of our Hamiltonian. Thus, we similarly define:

Definition 7 (Predictability Score)**.** Consider the pointer entropy $S_{\text{pointer,k}}$ generated over the characteristic time $\tau = t_{char}$ with the k^{th} qubit of our composite-system labeled as the system. We define the predictability score, S_{pred}, in a similar fashion:

$$S_{\text{pred}} \equiv \min_{L \in U_{LU}} (\max_{k}(S_{\text{pointer,k}})) \tag{2.20}$$

This formulation of the predictability score S_{pred} for our time-evolved system state is what we refer to as "Cost Function 3" in Chapter 3 and our algorithm.

To give the reader intuition for what constitutes predictable or robust behavior of a TPS, we construct the Bloch ball visualization of Figure 2.9. We can think of robustness as how resistant our system state is to entangling with the environment, and thus how pure our reduced system state remains over time. On the Bloch ball, this can be visually understood as how much our system state drifts towards the center. Predictability, on the other hand, is a measure of how much the system deviates from its original pointer eigenstate over time. On the Bloch ball, this can be visually understood as how much the state of our system drifts away from its initial state in general.

During this chapter, we have given an overview of the theory which serves as the backbone of this thesis, and have additionally obtained concrete notions of both predictability and robustness as Cost Function 1 (S_{rob}) and Cost Function 3 (S_{pred}). This being said, given the high computational cost of our algorithm, it is worthwhile to reformulate these cost functions in a more efficient form ("Cost Function 2" and "Cost Function 4"). Our new cost functions will also allow us to derive analytic formula for the mean and variance of each metric. These analytic results will greatly support our implementation of a machine learning algorithm to search for the TPSs that best minimize our predictability and robustness scores.

This machine learning algorithm will enable us to investigate the *problem of preferred subsystems* left unanswered by the decoherence program, and to analyze the characteristics of TPSs that best yield quasi-classical behavior. Importantly, we will explore how the predictability and robustness metrics compare to each other, and if both are truly necessary to

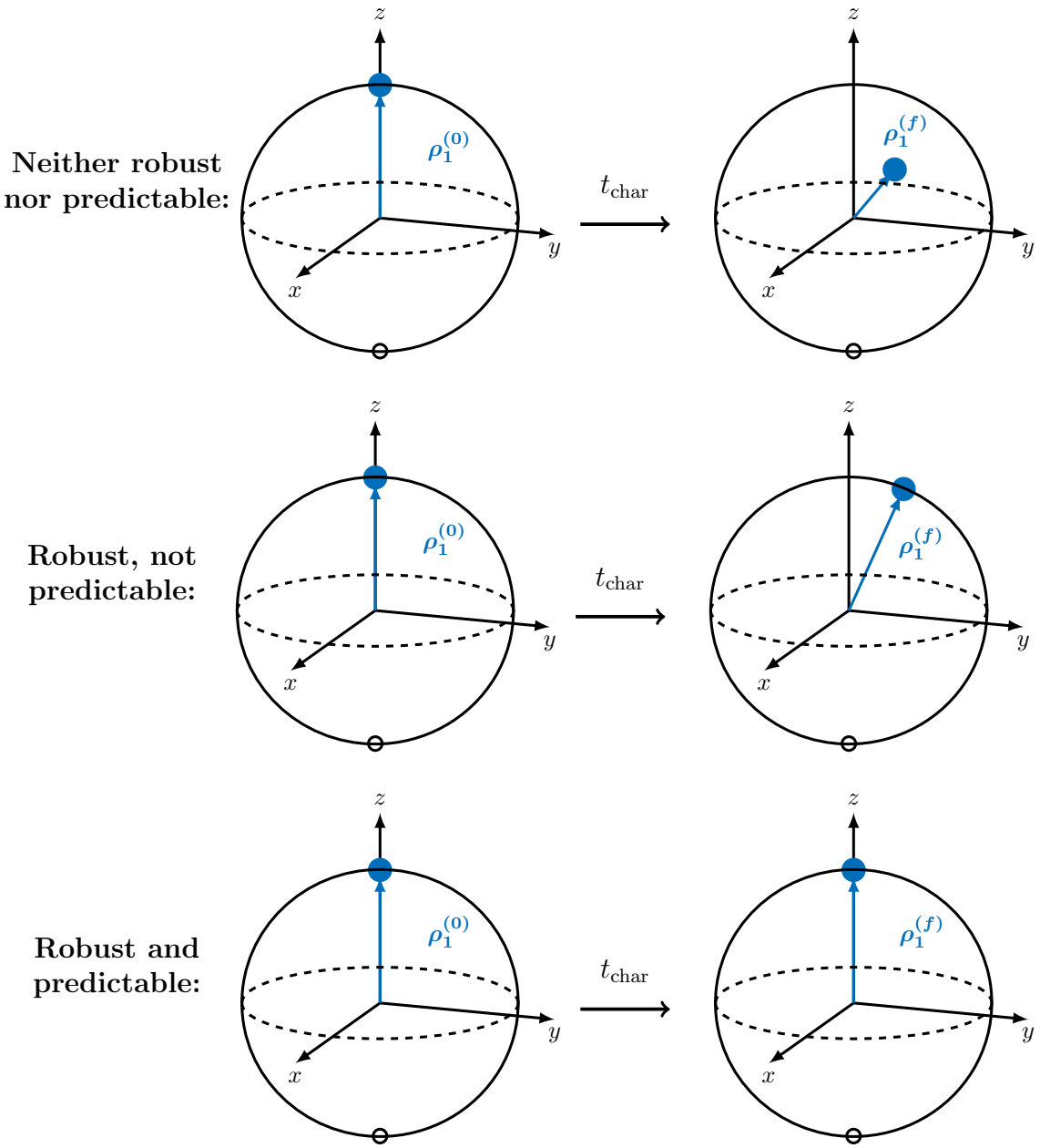


Figure 2.9: The evolution of state over for the characteristic time. (Top): The state drifts from its initial state and becomes mixed over time, lacking both robustness and predictability. (Middle): The state drifts from its initial state but remains pure, not entangling with the environment. Thus, we have robustness but not predictability. (Bottom): The state remains constant over time, and thus is both predictable and robust.

inform the factorization of a Hilbert space.[25] Furthermore, using the variance of our metrics, we will investigate how the initial states within a TPS vary in their stability. The derivations for our cost functions and their corresponding statistical formula are described in Chapter 3.

[25]Notice that in Figure 2.9, there is no time progression that is robust but not predictable. This visualization already seems to suggest that predictability implies robustness, which is a key possibility for the relationship between these metrics that we will investigate with the algorithm. What really matters, though, is not if this implication holds for an arbitrary state, but if this implication holds for the respective minima of each metric (e.g., via Cost Function 1 and Cost Function 3).

Chapter 3

Cost Functions and their Statistics

In Chapter 2, we defined the metrics of predictability and robustness that serve as the central focal points of this thesis. That being said, leaving our metrics in their Chapter 2 forms for our algorithm (Cost Functions 1 and 3) would be somewhat problematic. Specifically, the fact that we must find the minimum score over local unitary transformations (LUs) means that we would have to sample over many LUs in order to obtain a minimal score. This comes with two key issues:

1. This sampling procedure is computationally expensive.
2. Since the set of local unitary transformations on a Hilbert space form a continuous group, we would have to sample infinitely many LUs to obtain an exact minimal score.

Because of these key issues, we are motivated to reformulate our cost functions (2.18) and (2.20) in a way that avoids sampling over local unitary transformations (LUs) and instead yields an exact minimal score for a given TPS. Fortunately, both of the robustness and pointer entropies underlying our cost functions are quadratic in form (see Equations 2.7 and 2.19). This allows us to take advantage of a result from linear algebra [18]: any quadratic form over a unit vector (e.g., the Bloch vector for an initial pure state) can be maximized exactly by computing the largest eigenvalue of the associated symmetric matrix.[1] Recall from Chapter 2 that a local unitary transformation corresponds to a local rotation of the Bloch balls representing each tensor factor. This means that to find the minimum score over all possible LUs, we must find the minimum score over all possible rotations of our Bloch balls. That being said, in this case we are only concerned with the dynamics of our system, which reduces the problem to finding the minimum score over all possible rotations (i.e., initial states[2]) of our system Bloch ball!

Using this linear algebra fact to formulate equivalent versions of Cost Function 1 (CF1) and Cost Function 3 (CF3), we can avoid the computational expense and imprecision of having to sample over LUs (all of the possible rotations of the system Bloch ball). Additionally,

[1] The reason we want to *maximize* this symmetric matrix in order to *minimize* the cost function is simply associated with a negative sign in our score as will be described shortly.

[2] When testing the entropy growth of an initial state in our algorithm, we always use a pure state (i.e., candidate pointer observable). This allows us to directly observe how a given pure state for a TPS evolves over time, but also enables us to use this fact from linear algebra that only applies to unit vectors.

the quadratic form allows us to compute exact formula for the mean and variance of scores for a given TPS over all possible LUs (all possible initial states $\vec{n}$). This provides us with a fuller understanding of how the dynamics vary across different initial states of the system within a given TPS. The mean and variance expressions will prove critical in our analysis of the score behavior as we scale up the number of qubits in our composite-system.

The derivation of the symmetric matrix whose maximum eigenvalue yields a minimal CF1 score ("Cost Function 2") was done by my advisor, Prof. Setter [19]. The analogous derivation for predictability ("Cost Function 4") was done together by Prof. Setter and I. The corresponding mean and variance for each of these cost functions was derived by myself. The next two subsections will describe these derivations in detail.

To the best of my knowledge, Cost Function 2 (CF2), Cost Function 4 (CF4), and their associated statistics do not exist elsewhere in the literature. The derivation of CF4 (Section 3.2.1) and the derivation of the statistics for both CF2 and CF4 (Sections 3.1.2 and 3.2.2) constitute novel results of this thesis.

3.1 Cost Function 2: Robustness Score

To clarify: Cost Function 1 refers to the procedure of sampling the entanglement entropy S_{ent} over LUs and taking the maximum over permutations, minimum over LUs. Though this is not the ideal formula for our robustness score as described above, it is a more conceptually straightforward way to think about the robustness score of a TPS and serves as a good comparison check for our reformulated version. The reformulated version—in which S_{ent} is written in terms of a real, symmetric matrix that can be diagonalized (which allows us to minimize the function exactly)—is termed Cost Function 2 (CF2). This cost function is described in detail in Marissa Singh's thesis [6]. I will briefly outline Prof. Setter's derivation of CF2 below.[3]

3.1.1 Derivation

Consider again our composite-system of n qubits, where we have sectioned off one qubit as the "system," leaving $n-1$ qubits for the "environment." The environment forms a $d = 2^{n-1}$ dimensional Hilbert space.

Beginning with the initial state of our system as in Chapter 2 where the environment is in the infinite temperature limit (maximally mixed state), we have

$$\rho^{(0)} = \rho_{\text{sys}}^{(0)} \otimes \rho_{\text{env}}^{(0)} = \rho_{\text{sys}}^{(0)} \otimes \frac{\mathbb{I}_d}{d}. \tag{3.1}$$

Recall that our initial candidate pointer observable $\rho_{\text{sys}}^{(0)}$ is a pure state corresponding to a

[3]It is important to note that both the derivations of CF2 and CF4 assume an initially pure system state and a maximally-mixed environment.

unit Bloch vector $\vec{n}$ on the Bloch sphere:

$$\rho_{\text{sys}}^{(0)} = \frac{1}{2}\left(\mathbb{I}_2 + \vec{n}\cdot\vec{\sigma}\right) = \frac{1}{2}\left(\mathbb{I}_2 + \sum_{\alpha=1}^{3} n_\alpha \sigma_\alpha\right)$$

where $\mathbb{I}_2$ is the 2×2 Identity and $\vec{n} = (n_1, n_2, n_3) = (n_x, n_y, n_z) = (\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$. Here, θ and ϕ are the polar and azimuthal angles of spherical coordinates for our system Bloch ball. Thus, our total initial state becomes:

$$\begin{aligned}\rho^{(0)} &= \frac{1}{2}\left(\mathbb{I}_2 + \sum_{\alpha=1}^{3} n_\alpha\sigma_\alpha\right)\otimes\frac{\mathbb{I}_d}{d} \\ &= \frac{1}{2d}\left(\mathbb{I}_2\otimes\mathbb{I}_d + \sum_{\alpha=1}^{3} n_\alpha\sigma_\alpha\otimes\mathbb{I}_d\right)\end{aligned}$$

Now evolving this state for the characteristic time τ and tracing over the environment,

$$\begin{aligned}\rho_{sys}^{(f)} &= \text{Tr}_{\text{env}}[\text{e}^{-i\tau H}\cdot\rho^{(0)}\cdot\text{e}^{i\tau H}] \\ &= \text{Tr}_{\text{env}}\left[\text{e}^{-i\tau H}\cdot\left(\frac{1}{2d}\big(\mathbb{I}_2\otimes\mathbb{I}_d + \sum_{\alpha=1}^{3} n_\alpha\sigma_\alpha\otimes\mathbb{I}_d\big)\right)\cdot\text{e}^{i\tau H}\right] \\ &= \frac{1}{2d}\text{Tr}_{\text{env}}\left[\mathbb{I}_2\otimes\mathbb{I}_d + \sum_{\alpha=1}^{3} n_\alpha\, e^{i\tau H}\left(\sigma_\alpha\otimes\mathbb{I}_d\right)e^{-i\tau H}\right],\end{aligned} \tag{3.2}$$

we obtain the final, time-evolved state of our system. This state $\rho_{sys}^{(f)}$ is the state for which we would like to determine the robustness score.

For notational convenience, we define the following three $2d\times2d$ matrices:

$$Q_\alpha \equiv \frac{1}{d}e^{i\tau H}\left(\sigma_\alpha\otimes\mathbb{I}_d\right)e^{-i\tau H}, \quad \text{for } \alpha = 1,2,3. \tag{3.3}$$

Additionally, tracing Q_α out of the environment, we define the 2×2 matrices:

$$q_\alpha \equiv \text{Tr}_{\text{env}}(Q_\alpha). \tag{3.4}$$

Writing (3.2) in terms of q_α, we obtain:

$$\begin{aligned}\rho_{sys}^{(f)} &= \frac{1}{2d}\left[\text{Tr}_{\text{env}}\big[\mathbb{I}_2\otimes\mathbb{I}_d\big] + d\sum_{\alpha=1}^{3} n_\alpha q_\alpha\right] \\ &= \frac{1}{2}\mathbb{I}_2 + \frac{1}{2}\sum_{\alpha=1}^{3} n_\alpha q_\alpha\end{aligned} \tag{3.5}$$

We now need to compute the entanglement entropy of our evolved system. Plugging (3.5) into (2.7):

$$\begin{aligned} S_{\text{ent}}(\vec{n}) &= 1 - \text{Tr}\left[(\rho_{\text{sys}}^{(f)})^2\right] \\ &= 1 - \text{Tr}\left[(\frac{1}{2}\mathbb{I}_2 + \frac{1}{2}\sum_{\alpha=1}^{3} n_\alpha q_\alpha)^2\right] \\ &= 1 - \text{Tr}\left[\frac{1}{4}\mathbb{I}_2 + \frac{1}{2}\sum_{\alpha=1}^{3} n_\alpha q_\alpha + \frac{1}{4}\sum_{\alpha,\beta=1}^{3} n_\alpha n_\beta q_\alpha q_\beta\right] \\ &= \frac{1}{2} - \frac{1}{2}\sum_{\alpha=1}^{3}\text{Tr}[q_\alpha] - \frac{1}{4}\sum_{\alpha,\beta=1}^{3} n_\alpha n_\beta \text{Tr}[q_\alpha q_\beta] \end{aligned} \tag{3.6}$$

where in the last step we have used the linearity of the trace operation.
This may look rather awful, but fortunately—using a trace identity from Tegmark [2]—we see that the linear term $\frac{1}{2}\sum_{\alpha=1}^{3}\text{Tr}[q_\alpha]$ vanishes as

$$\begin{aligned} \text{Tr}[q_\alpha] &= \text{Tr}\left[\text{Tr}_{\text{env}}(Q_\alpha)\right] \\ &= \text{Tr}\left[Q_\alpha\right] \\ &= \frac{1}{d}\,\text{Tr}\left[e^{i\tau H}(\sigma_\alpha \otimes \mathbb{I}_d)e^{-i\tau H}\right] \\ &= \frac{1}{d}\,\text{Tr}(\sigma_\alpha \otimes \mathbb{I}_d) \quad \text{(cyclicity of the trace)} \\ &= \frac{1}{d}\,\text{Tr}(\sigma_\alpha)\,\text{Tr}(\mathbb{I}_d) \\ &= \text{Tr}(\sigma_\alpha) \\ \text{Tr}[q_\alpha] &= 0 \end{aligned} \tag{3.7}$$

Thus, our expression for the entanglement entropy (3.6) in terms of q_α, q_β reduces to:

$$S_{\text{ent}}(\vec{n}) = \frac{1}{2} - \frac{1}{4}\sum_{\alpha,\beta=1}^{3} n_\alpha n_\beta \text{Tr}[q_\alpha q_\beta]. \tag{3.8}$$

Finally, since the second term in our expression above is quadratic in $\vec{n}$, we can assemble our symmetric[4] matrix to diagonalize:

$$M_{\alpha\beta} \equiv \text{Tr}[q_\alpha q_\beta] \tag{3.9}$$

and thus,

$$S_{\text{ent}}(\vec{n}) = \frac{1}{2} - \frac{1}{4}\vec{n}^{\text{T}} M \vec{n}. \tag{3.10}$$

[4]How do we know that this matrix M is symmetric? It follows since the trace is invariant under cyclic permutations: $M_{\alpha\beta} = \text{Tr}(q_\alpha q_\beta) = \text{Tr}(q_\beta q_\alpha) = M_{\beta\alpha}$.

We now want to determine the minimum of S_{ent} over LUs. Again, recall that performing a local unitary transformation is analogous to rotating our Bloch balls, which simply amounts to a rotation of our system Bloch vector $\vec{n}$. Thus, finding the minimum score over LUs is equivalent to finding the minimum score over $\vec{n}$. With our new form in Equation 3.10, this amounts to finding the maximum eigenvalue $\lambda_{\max}$ of the real, symmetric matrix $M_{\alpha\beta}$, and plugging in for S_{ent} appropriately. Thus, our algorithm must simply compute

$$S_{\text{rob}} = \min_{\vec{n}}(S_{\text{ent}}(\vec{n})) = \frac{1}{2} - \frac{1}{4}\lambda_{\max} \tag{3.11}$$

to find the *exact* robustness score of a given TPS, which completes our derivation of Cost Function 2.

3.1.2 Mean and Variance

Now equipped with CF2, we can determine analytic expressions for the mean and variance of the robustness score, again over all possible initial states of our system qubit (our unit Bloch vector $\vec{n}$). We can do this because the symmetric matrix M fully captures how a given TPS induces entanglement under time evolution.[5] This will serve especially useful in investigating generic TPSs at larger qubit numbers.

We begin by calculating the mean of CF2. Starting from (3.10):

$$\begin{aligned}\langle S_{\text{ent}}\rangle &= \langle\frac{1}{2}\rangle - \langle\frac{1}{4}\vec{n}^{\mathrm{T}}M\vec{n}\rangle \\ &= \frac{1}{2} - \frac{1}{4}\sum_{i,j=1}^{3} M_{ij}\langle n_i n_j\rangle.\end{aligned} \tag{3.12}$$

We will first focus on simplifying the sum. To compute the expectation value $\langle n_i n_j\rangle$, we integrate over the uniform distribution on the Bloch (unit) sphere. This is a standard spherical integral that evaluates to [20]:

$$\langle n_i n_j\rangle = \frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi} n_i n_j \sin\theta\, d\theta\, d\phi = \frac{1}{3}\delta_{ij}.$$

Thus,

$$\sum_{i,j=1}^{3} M_{ij}\langle n_i n_j\rangle = \sum_{i=1}^{3} M_{ij}\big(\frac{1}{3}\delta_{ij}\big) = \sum_{i=1}^{3} M_{ii}\big(\frac{1}{3}\big) = \frac{1}{3}\mathrm{Tr}(M).$$

[5]Notice that in Equation 3.10, the only remaining variable is the Bloch vector $\vec{n}$ which parameterizes different initial states rather than different dynamics. Thus, M contains all of the dynamical information for the TPS. As will soon be described, averaging over $\vec{n}$ amounts to integrating a quadratic form over the unit sphere, which results in a spherical integral that can be carried out analytically.

Plugging this expression back into (3.12), we find that

$$\begin{aligned}\langle S_{\text{ent}}\rangle &= \frac{1}{2} - \frac{1}{4}\left[\frac{1}{3}\text{Tr}(M)\right] \\ &= \frac{1}{2} - \frac{1}{12}\text{Tr}(M)\end{aligned} \tag{3.13}$$

This gives us a final, analytic expression for the mean robustness score of a TPS in terms of only $\text{Tr}(M)$.

Now, to compute the variance, we use:

$$(\Delta S_{\text{ent}})^2 = \langle S_{\text{ent}}^2\rangle - \langle S_{\text{ent}}\rangle^2$$

We have already calculated $\langle S\rangle$, and thus

$$\begin{aligned}\langle S_{\text{ent}}\rangle^2 &= \big(\frac{1}{2} - \frac{1}{12}\text{Tr}(M)\big)^2 \\ &= \frac{1}{4} - \frac{1}{12}\text{Tr}(M) + \frac{1}{144}[\text{Tr}(M)]^2\end{aligned} \tag{3.14}$$

Now we must calculate the second moment, $\langle S^2\rangle$. The full calculation is omitted for brevity, but—in terms of traces—this yields

$$\langle S_{\text{ent}}^2\rangle = \frac{1}{4} - \frac{1}{12}\,\text{Tr}(M) + \frac{1}{240}\left[(\text{Tr}(M))^2 + 2\,\text{Tr}(M^2)\right] \tag{3.15}$$

Putting (3.14) and (3.15) together, we obtain

$$\begin{aligned}(\Delta S_{\text{ent}})^2 &= \langle S_{\text{ent}}^2\rangle - \langle S_{\text{ent}}\rangle^2 \\ &= \left[\frac{1}{4} - \frac{1}{12}\,\text{Tr}(M) + \frac{1}{240}\left[(\text{Tr}(M))^2 + 2\,\text{Tr}(M^2)\right]\right] \\ &\quad - \left[\frac{1}{4} - \frac{1}{12}\text{Tr}(M) + \frac{1}{144}[\text{Tr}(M)]^2\right] \\ &= \frac{1}{240}\,(\text{Tr}(M))^2 + \frac{1}{120}\,\text{Tr}(M^2) - \frac{1}{144}\,(\text{Tr}(M))^2 \\ (\Delta S_{\text{ent}})^2 &= \frac{1}{120}\,\text{Tr}(M^2) - \frac{1}{360}\,(\text{Tr}(M))^2\end{aligned} \tag{3.16}$$

This is our final expression for the variance of CF2 across the Bloch vector $\vec{n}$ of our initial system state (or equivalently, local unitary transformations of our Hamiltonian).

3.2 Cost Function 4: Predictability Score

Recall that Cost Function 3 refers to the procedure of sampling over LUs and permutations to obtain the minimum value of S_{pointer}. Again, just as for robustness, we wish to derive

an equivalent expression of CF3 in terms of a symmetric matrix that can be diagonalized (CF4).

It may seem that we can immediately start work with our equation for the pointer entropy from Chapter 2 (Equation 2.19), but there is one critical problem: this equation implicitly assumes that our candidate pointer observable is either the $|0\rangle$ or $|1\rangle$ state. In other words, it computes predictability relative to the $\{|0\rangle, |1\rangle\}$ basis (by assuming it is the system's eigenbasis), making the expression basis-dependent. For CF3, this is not a problem since we always use $|0\rangle$ as our initial system state (our candidate pointer observable). For CF4, however, we want an expression that yields a minimum over *all* initial system states $\vec{n}$, and if our underlying expression for the pointer entropy is basis-dependent, our derived CF4 will not be invariant under local unitary transformations as we require.[6]

Thus, we need a more general expression for the predictability entropy that is not tied to any particular basis. Prof. Setter and I propose the following revised expression:

$$S'_{\text{pointer}}(\rho_{sys}) \equiv 1 - \left[\text{Tr}(\rho_+ \cdot \rho_{\text{sys}}^{(f)})\right]^2 - \left[\text{Tr}(\rho_- \cdot \rho_{\text{sys}}^{(f)})\right]^2 \tag{3.17}$$

where ρ_+ represents the state of our system at time zero, and ρ_- is the corresponding antipodal point on the Bloch sphere. In the specific case where our initial state is $|0\rangle$ (as in CF3), we have:

$$\begin{aligned}
S'_{\text{pointer}} &= 1 - (\text{Tr}[\rho_+ \cdot \rho_{\text{sys}}^{(f)}])^2 - (\text{Tr}[\rho_- \cdot \rho_{\text{sys}}^{(f)}])^2 \\
&= 1 - (\text{Tr}\big[\,|0\rangle\langle 0| \cdot \rho_{\text{sys}}^{(f)}\big])^2 - (\text{Tr}\big[\,|1\rangle\langle 1| \cdot \rho_{\text{sys}}^{(f)}\big])^2 \\
&= 1 - (\text{Tr}\big[\,\langle 0|\,\rho_{\text{sys}}^{(f)}\,|0\rangle\,\big])^2 - (\text{Tr}\big[\,\langle 1|\,\rho_{\text{sys}}^{(f)}\,|1\rangle\,\big])^2 \\
&= 1 - (\langle 0|\,\rho_{\text{sys}}^{(f)}\,|0\rangle)^2 - (\langle 1|\,\rho_{\text{sys}}^{(f)}\,|1\rangle)^2 \\
&= 1 - \rho_{00}^2 - \rho_{11}^2 \\
S'_{\text{pointer}} &= S_{\text{pointer}}
\end{aligned}$$

Thus, our revised pointer entropy formula reduces to the original expression as required.

3.2.1 Derivation

Now, to obtain the symmetric matrix to diagonalize, we follow a similar procedure as for CF2. First, we will simplify the terms in Equation 3.17 that are inside the square. Looking at the first of these terms:

$$\text{Tr}\left(\rho_+ \cdot \rho_{\text{sys}}\right) = \text{Tr}\left[\frac{1}{2}(\mathbb{I} + \vec{n} \cdot \vec{\sigma}) \cdot \frac{1}{2}(\mathbb{I} + \vec{n} \cdot \vec{q})\right]$$

[6]This problem was only discovered after realizing our previous CF4 expression—which used Equation 2.19—was not invariant over local unitary transformations. We discovered this using the `check_over_LU` testing function, which will be described with other method checks in Chapter 4 (Section 4.2).

where $\mathbb{I} = \mathbb{I}_2$ is the 2×2 Identity. Writing our dot products in index notation, this becomes

$$\begin{aligned}
&= \frac{1}{4} \operatorname{Tr}\left[(\mathbb{I} + n_\alpha \sigma_\alpha)(\mathbb{I} + n_\beta q_\beta)\right] \\
&= \frac{1}{4} \operatorname{Tr}\left[\mathbb{I} + n_\alpha \sigma_\alpha + n_\beta q_\beta + n_\alpha n_\beta \sigma_\alpha q_\beta\right] \\
&= \frac{1}{4} \left[\operatorname{Tr}(\mathbb{I}) + n_\alpha \operatorname{Tr}(\sigma_\alpha) + n_\beta \operatorname{Tr}(q_\beta) + n_\alpha n_\beta \operatorname{Tr}(\sigma_\alpha q_\beta)\right].
\end{aligned}$$

Using the tracelessness of the Pauli matrices and our trace identity (3.7), we obtain

$$\begin{aligned}
&= \frac{1}{4} \left[2 + n_\alpha n_\beta \operatorname{Tr}(\sigma_\alpha q_\beta)\right] \\
&= \frac{1}{2} + \frac{1}{4} n_\alpha n_\beta \operatorname{Tr}(\sigma_\alpha q_\beta) \\
\operatorname{Tr}\left(\rho_+ \cdot \rho_{\text{sys}}\right) &= \frac{1}{2} + \frac{1}{4} n_\alpha n_\beta P_{\alpha\beta}, \qquad (3.18)
\end{aligned}$$

where we have defined $P_{\alpha\beta} \equiv \operatorname{Tr}(\sigma_\alpha q_\beta)$ for conciseness.

The simplification of $\operatorname{Tr}(\rho_- \cdot \rho_{sys})$ is similar, and thus our total predictability score comes to:

$$\begin{aligned}
S'_{\text{pointer}} &= 1 - (\operatorname{Tr}[\rho_+ \cdot \rho_{sys}])^2 - (\operatorname{Tr}[\rho_- \cdot \rho_{sys}])^2 \\
&= 1 - \big(\frac{1}{2} + \frac{1}{4} n_\alpha n_\beta P_{\alpha\beta}\big)^2 - \big(\frac{1}{2} - \frac{1}{4} n_\alpha n_\beta P_{\alpha\beta}\big)^2 \\
&= \frac{1}{2} - \frac{1}{8} \big(n_\alpha \, n_\beta \, P_{\alpha\beta}\big)^2 \qquad (3.19)
\end{aligned}$$

Though this expression is now quartic in our Bloch vector $\vec{n}$ instead of quadratic, it takes the form $(\vec{n}^{\mathrm{T}} P \vec{n})^2$: the square of a quadratic form in $\vec{n}$. Because of this structure, we can still use the linear algebra property discussed earlier. If the matrix $P_{\alpha\beta}$ is symmetric, this structure allows us to apply our standard minimization strategy—the maximum of the quartic expression $(n_\alpha n_\beta P_{\alpha\beta})^2$ over all unit vectors $\vec{n}$ is given by the maximum eigenvalue of P, just as in the quadratic case [21].

However, there is a problem: $P_{\alpha\beta} = \operatorname{Tr}(\sigma_\alpha q_\beta)$ is not symmetric in general. While the cyclic property of the trace allows us to write $\operatorname{Tr}(\sigma_\alpha q_\beta) = \operatorname{Tr}(q_\beta \sigma_\alpha)$, it does not imply that $\operatorname{Tr}(\sigma_\alpha q_\beta) = \operatorname{Tr}(\sigma_\beta q_\alpha)$ as we require. The issue is that σ_α and q_β are different matrices that do not commute in general, so switching their places changes the value of the trace.

To get around this difficulty, we define a symmetric version of $P_{\alpha\beta}$:

$$\widetilde{P}_{\alpha\beta} \equiv \frac{1}{2} \left(P_{\alpha\beta} + P_{\beta\alpha}\right) \qquad (3.20)$$

This matrix is symmetric by construction, since swapping $\alpha \leftrightarrow \beta$ leaves it unchanged:

$$\widetilde{P}_{\beta\alpha} = \frac{1}{2} \left(P_{\beta\alpha} + P_{\alpha\beta}\right) = \widetilde{P}_{\alpha\beta}$$

By working with $\widetilde{P}$ instead of P, we retain the directional information encoded within the original matrix while ensuring that our matrix is diagonalizable. Thus, replacing $P_{\alpha\beta}$ with $\widetilde{P}_{\alpha\beta}$, our final predictability score becomes

$$S'_{\text{pointer}} = \frac{1}{2} - \frac{1}{8}\left(n_\alpha\, n_\beta\, \widetilde{P}_{\alpha\beta}\right)^2 \tag{3.21}$$

and the minimum value over LUs (our TPS score) is

$$S_{\text{pred}} = \min_{\vec{n}}(S'_{\text{pointer}}) = \frac{1}{2} - \frac{1}{8}\lambda^2_{\max}, \tag{3.22}$$

where $\lambda_{\max}$ is the maximum eigenvalue of $\widetilde{P}$.

3.2.2 Mean and Variance

To derive the mean and variance of CF4, we follow a very similar procedure to CF2. This time, however, we are dealing with a form that is quartic over $\vec{n}$ instead of quadratic, which makes our life a bit more complicated. Thus, for the CF4 mean and variance derivations, we turn to Mathematica to compute our integrals and simplify our formulas.

Beginning with the mean, we have:

$$\begin{aligned}
\langle S'_{\text{pointer}}\rangle &= \left\langle \frac{1}{2} - \frac{1}{8}\left(\sum_{i,j=1}^{3} n_i n_j \widetilde{P}_{ij}\right)^2 \right\rangle \\
&= \frac{1}{2} - \frac{1}{8}\sum_{i,j,k,l=1}^{3} \widetilde{P}_{ij}\widetilde{P}_{kl}\,\langle n_i n_j n_k n_l\rangle \\
&= \frac{1}{2} - \frac{1}{8}\sum_{i,j,k,l=1}^{3} \widetilde{P}_{ij}\widetilde{P}_{kl}\left[\frac{1}{4\pi}\int_0^{\pi}\int_0^{2\pi} n_i n_j n_k n_l \sin\theta\, d\phi\, d\theta\right] \\
&= \frac{1}{2} - \frac{1}{8}\left[\frac{1}{4\pi}\int_0^{\pi}\int_0^{2\pi}\left(\sum_{i,j,k,l=1}^{3} \widetilde{P}_{ij}\widetilde{P}_{kl}\, n_i n_j n_k n_l\right)\sin\theta\, d\phi\, d\theta\right] \\
&= \frac{1}{2} - \frac{1}{8}\left[\frac{1}{4\pi}\int_0^{\pi}\int_0^{2\pi}\left(\sum_{i,j=1}^{3} n_i n_j \widetilde{P}_{ij}\right)^2 \sin\theta\, d\phi\, d\theta\right] \\
&= \frac{1}{2} - \frac{1}{8}\left[\frac{1}{4\pi}\int_0^{\pi}\int_0^{2\pi} \widetilde{P}^2(\theta,\phi)\sin\theta\, d\phi\, d\theta\right]
\end{aligned} \tag{3.23}$$

This term here in brackets is simply the expectation value of $\widetilde{P}^2$ over the Bloch sphere, $\langle\widetilde{P}^2\rangle$. To compute this, I plugged the integral into Mathematica, and then simplified using the Symmetric Reduction function to obtain an expression in terms of only power traces

of $\widetilde{P}$ and $\det(\widetilde{P})$ (see Appendix A for the full Mathematica notebook). The result of this calculation is:

$$\begin{aligned}\langle \widetilde{P}^2 \rangle &= \left[\frac{1}{4\pi} \int_0^{\pi} \int_0^{2\pi} \widetilde{P}^2(\theta, \phi) \sin\theta \, d\phi \, d\theta \right] \\ &= \frac{1}{15} \big(\mathrm{Tr}(\widetilde{P}) + 2\mathrm{Tr}(\widetilde{P}^2) \big) \end{aligned} \tag{3.24}$$

Thus, plugging back into (3.23):

$$\langle S'_{\text{pointer}} \rangle = \frac{1}{2} - \frac{1}{120} \big(\mathrm{Tr}(\widetilde{P}) + 2\mathrm{Tr}(\widetilde{P}^2) \big), \tag{3.25}$$

and our derivation for the mean of CF4 is complete.

Now, we can move on to the variance calculation for CF4, $(\Delta S'_{\text{pointer}})^2$. To approach this, we will first simplify the expression by hand, and then compute the relevant integrals. Starting from the definition of variance,

$$(\Delta S'_{\text{pointer}})^2 = \langle S^2_{\text{pointer}} \rangle - \langle S_{\text{pointer}} \rangle^2. \tag{3.26}$$

We compute $\langle S_{\text{pointer}} \rangle^2$ in terms of powers of $\widetilde{P}$:

$$\begin{aligned}\langle S_{\text{pointer}} \rangle^2 &= \left\langle \frac{1}{2} - \frac{1}{8} \widetilde{P}^2 \right\rangle^2 \\ &= \big(\frac{1}{2} - \frac{1}{8} \langle \widetilde{P}^2 \rangle \big)^2 \\ &= \frac{1}{4} - \frac{1}{8} \langle \widetilde{P}^2 \rangle + \frac{1}{64} \langle \widetilde{P}^2 \rangle^2 \end{aligned} \tag{3.27}$$

Furthermore,

$$\begin{aligned}\langle S^2_{\text{pointer}} \rangle &= \left\langle \left(\frac{1}{2} - \frac{1}{8} \widetilde{P}^2 \right)^2 \right\rangle \\ &= \left\langle \frac{1}{4} - \frac{1}{8} \widetilde{P}^2 + \frac{1}{64} \widetilde{P}^4 \right\rangle \\ &= \frac{1}{4} - \frac{1}{8} \langle \widetilde{P}^2 \rangle + \frac{1}{64} \langle \widetilde{P}^4 \rangle. \end{aligned} \tag{3.28}$$

Now, plugging (3.27) and (3.28) into (3.26), we obtain:

$$(\Delta S'_{\text{pointer}})^2 = \frac{1}{64} \big[\langle \widetilde{P}^4 \rangle - \langle \widetilde{P}^2 \rangle^2 \big] \tag{3.29}$$

We have already determined an expression for $\langle \widetilde{P}^2 \rangle$ in terms of power traces (3.24). Thus, we now need to do the same for $\langle \widetilde{P}^4 \rangle$. Following a similar procedure with Mathematica (again using the Symmetric Reduction function), we obtain the following result:

$$\langle \widetilde{P}^4 \rangle = \frac{1}{315} \left[96 \, \det(\widetilde{P}) \, \mathrm{Tr}(\widetilde{P}) - 13 \, \mathrm{Tr}(\widetilde{P})^4 + 36 \, \mathrm{Tr}(\widetilde{P})^2 \, \mathrm{Tr}(\widetilde{P}^2) + 12 \, \mathrm{Tr}(\widetilde{P})^2 \right] \tag{3.30}$$

Finally, plugging our simplified expressions (3.24) and (3.30) into our variance formula (3.29), and simplifying further, this yields:

$$(\Delta S'_{\text{pointer}})^2 = \frac{1}{12600}\left[60\det(\widetilde{P})\,\mathrm{Tr}(\widetilde{P}) - 9\,\mathrm{Tr}(\widetilde{P})^4 + 19\,\mathrm{Tr}(\widetilde{P})^2\;\mathrm{Tr}(\widetilde{P}^2) + 4\,\mathrm{Tr}(\widetilde{P}^2)^2\right] \quad (3.31)$$

Thus, we have arrived at our final analytic expression for the variance of the pointer entropy, $(\Delta S'_{\text{pointer}})^2$, within a given TPS! This expression will allow us to investigate characteristics of predictability that would otherwise be computationally inaccessible.

Though this chapter was mathematically dense, our hard work will soon pay off—the six new analytic expressions we've derived for the minimum, mean, and variance of each metric will enable us to directly investigate the leading questions of this thesis. Succinctly: how do the metrics of predictability and robustness compare for quasi-classical TPSs, and how does the stability of system states vary within a single quasi-classical TPS? In seeking answers to these questions, we perform a wide computational search over all possible tensor product structures—as is discussed in Chapter 4.

Chapter 4

Methods

4.1 Code Implementation

Now that we are equipped with analytic formulas for CF2, CF4, and their statistics, we can begin implementing them into the code.[1] Working from base code written by Marissa Singh and Brian Lee [6, 17], I focused on developing two main functions: the `generic_score` function and the `TPS_min` function. Though much of the coding during this thesis extended far beyond just these two functions, I will limit my focus to them in this section as they are the functions with the most direct connection to my results.[2]

4.1.1 Generic Score Function

The `generic_score` function generates a score and statistics for an average, generic TPS of the initial Hamiltonian. By "average" or "generic," we are referring to the mean scores and statistics over TPSs sampled from a uniform distribution.

The function operates in the following steps:

1. Initialize a Hamiltonian H_n (e.g., the 1D Ising or central spin Hamiltonian) and number of qubits n.
2. Sample a random special unitary operator $\hat{U}$ from the Haar distribution (a uniform probability distribution over $SU(n)$).
3. Apply the sampled unitary operator to the initial Hamiltonian H_n to obtain a new Hamiltonian $H' = \hat{U}^\dagger H_n \hat{U}$. As explained in Definition 4, this amounts to redefining the tensor product structure of our Hilbert space.
4. Compute the score, mean, and variance of the new Hamiltonian H' (the TPS) according to functions that utilize the analytic formulas developed in Chapter 3.

[1] I feel it is relevant to note that the majority of the work involved in this thesis was coding, and the project contains well over 2000 lines of code—much of which I personally wrote.

[2] This additional work included: expanding and restructuring the existing `cost` function, creating the `mean_variance_fn`, developing a data storage protocol to store and organize simulations in CSV files, new graphing and debugging functions, etc.

5. Append the score, mean, and variance of each cost function (CF2 and CF4) to their own respective lists (six total lists).
6. Repeat steps 2 - 5, adding scores from other, randomly-sampled TPSs to the lists.
7. Compute the mean across all six lists, obtaining distinct values for the minimum, mean, and variance of CF2 and CF4 for a "generic" TPS of the initial Hamiltonian.

This procedure may seem surprisingly simple. Because CF2 and CF4 are invariant under local unitary transformations by design, the implementation of this function can avoid much of the added complexity within each TPS. Instead, we can just compute six numbers for each TPS and we are done.

Notice also that there is no machine learning going on in this function—it is simply a matter of sampling over global[3] unitaries, and thus requires far less computational power than our minimization function. In principle, this allows us to probe larger system-environment sizes by exploring their generic TPS scores and statistics—something that would be computationally inaccessible with our minimization function.

Another powerful aspect of this function is that it gives us a reference against which we can compare our minimized-TPS scores and statistics. Without this function, it would be hard to get a sense for what really constitutes a "low" score for a TPS as opposed to a high one.

4.1.2 TPS-minimization Function

The `TPS_min` function is where we get to the actual machine learning aspect of the code. This function allows us to search the space of TPSs for ones that exhibit quasi-classical dynamics (i.e., have slow entropy growth and thus low cost function scores). For our purposes, I used the SciPy optimization function[4] with the 'BFGS' method, as well as options for the other parameters that provided the qualitatively best results. The code flow for this function is displayed below:

1. Initialize a Hamiltonian H_n, number of qubits n, cost function to minimize, number of TPSs to find, and lists for the scores and statistics of both robustness and predictability.
2. Compute the score and statistics for the Hamiltonian in its native basis H_n, adding each value to its respective list.
3. Minimize the specified cost function using SciPy's optimization routines, returning both the minimum score (either for robustness or predictability) and a set of parameters (`min_thetas`) that define the global unitary transformation used to construct the TPS.[5] This minimization is done until a specified threshold score, `threshold`, is reached.

[3] By "global" unitary, we just mean a general special unitary matrix that is not necessarily local. Thus, in general, a global unitary will map a given Hamiltonian to a new, distinct TPS.

[4] This SciPy optimization function constitutes the entirety of the "machine learning" used in this thesis. Future work on this project could be dedicated to exploring alternative minimization techniques, which could enable the exploration of higher composite-system qubit numbers.

[5] These thetas parameterize the generalized Gell-Mann matrices used to construct the global unitary operator. For more details, see the discussion in Spengler *et. al.,* [22].

4. Compute the score for the other function (CF2 or CF4) not minimized in step 3. Also compute the mean and variance for both predictability and robustness. Append all data to their respective lists.
5. Perform a Local Unitary Equivalence (LUE) check on the newly found TPS (via minimization) to ensure it is not LUE to a previously found TPS.[6] This is done via a method adapted from Martins [23]. If it remains unclear whether the TPS is LUE or not, remove its scores and statistics from their respective lists.
6. Repeat steps 3 - 5 until the specified number of quasi-classical TPSs have been found.

This constitutes the basic logical procedure of the `min_TPS` function and allows us to obtain scores and statistics for tensor product structures that exhibit quasi-classical dynamics (which we abbreviate with "qcTPS"). From here, we can compare the scores, means, and variances of qcTPSs to the generic baselines provided by the `generic_score` function.

One point to emphasize is that the `threshold` value is set to $10^{-9}\times$ (the generic score for the given Hamiltonian H_n). This is the lowest value that was accessible in reasonable search times. That being said, it is important to note that a found "minimum TPS" simply corresponds to a TPS with a score less than $\epsilon =$ `threshold`. Even if not an exact minima, finding a value below the threshold still captures the essential features of the TPS landscape, and has been done elsewhere in the literature [8].

4.2 Method Checks

A variety of method checks were implemented throughout the course of this thesis to ensure the validity and correctness of the functions, mathematics, and their outputted results.

Many of these checks involved comparing CF2 to CF1 and CF4 to CF3. These were done via an adapted version of our `generic_score` function called `generic_score_check`. Recall that from Equation 2.18, when computing the robustness score of a TPS we seek the minimum entanglement entropy over local unitary transformations. To do this with CF1, we must sample over these local unitary transformations and return the minimum score, whereas with CF2, we can use our analytic formula to compute this minimum without sampling. That being said, for a large number of samples, the minimum sampled score from CF1 should converge to CF2. Over 10,000 sampled local unitary transformations within a given TPS, the minimum CF1 score and analytic CF2 score were consistently equivalent within ± 0.001. This remained true when testing multiple distinct TPSs. In a similar manner, CF3 was found to converge to CF4 over large LU sample sizes. This check provided us with confidence that our analytically derived formulas were accurately reproducing the expected results.

In addition to checking for equivalence between the scores of each pair of functions, another important check was that CF2 $\leq$ CF1 and CF4 $\leq$ CF3 for all sampled scores

[6] We remark here that the invariance of CF2 and CF4 over local unitary transformations itself serves as LUE check. This note should be further explored and has the potential to aid future researchers in determining LUE.

within the local unitary orbit[7] (TPS). This should hold since CF2 and CF4 provide an *exact* minimum score, as opposed to a sampled one. Adapting the code to throw an error if the condition CF2 > CF1 or CF4 > CF3 was met for a given local unitary transformation within a TPS, no error was thrown over 10,000 LU samples. This added to our confidence that CF2 and CF4 were behaving as expected.

A final check on the behavior of the scores returned by CF2 and CF4 was done in a separate testing function titled `check_over_LU`. This function computed CF2 and CF4 many times within a single local unitary orbit (TPS) to ensure that regardless of the local unitary applied to the Hamiltonian, the scores for CF2 and CF4 were invariant. This should hold since CF2 and CF4 return the *minimum* score across the entire local unitary orbit, and thus regardless of where the Hamiltonian happens to be within the orbit, the returned score should be the same. Again, across 10,000 LU samples, both CF2 and CF4 passed this check.

To test our analytic formulas for the mean and variance of scores within a TPS, we repeated the above checks but instead of taking the minimum across sampled scores with CF1 and CF3, we instead took the mean and variance across the sampled scores. The exact variance of CF2 and CF4 agreed well with the sampled variance of CF1 and CF3. Similarly, the exact mean agreed well with the sampled mean. Furthermore, the mean and variance were invariant over local unitary transformations[8] (shown via the `check_over_LU` function), as expected.

These method checks allowed us to move forward into the simulation part of this thesis with confidence—the functions we are relying on behave as expected. The results of the simulations done with the `generic_score` and `TPS_min` functions are detailed in Chapter 5.

[7]By "local unitary orbit," we mean the set of all Hamiltonians that are related to a given Hamiltonian H by local unitary transformations. These are considered equivalent and belong to the same tensor product structure (see Definition 4).

[8]As discussed in a previous footnote, the basis-dependence of our original version of Cost Function 4 was caught by the `check_over_LU` function. This prompted our new expression for the pointer entropy, $S'_{pointer}$, that we used in our derivation (Equation 3.17).

Chapter 5

Results and Analysis

Before beginning this section, I would like to re-emphasize the two primary questions guiding this thesis:

1. What is the relationship between the metrics of "predictability" (2.20) and "robustness" (2.18)? Specifically: are both metrics necessary to inform a quasi-classical Tensor Product Factorization of a Hilbert Space?
2. What does the variance of these metrics look like across different initial states of a given Tensor Product Structure? Specifically: what is the relationship between the quasi-classical TPS variance and generic TPS variance?

The findings presented in this chapter offer strong evidence addressing both of these questions. The code that generated this data was built from base code [6, 17] and extended by me via the analytic calculations described in Chapter 3, the methods described in Chapter 4, and a variety of other improvements (e.g., data storage system, improvements to the machine learning model, etc.). All of the data displayed in this section uses the central spin Hamiltonian as the native Hamiltonian (Equation 2.10), though the same qualitative results held for the 1D Ising, CNOT, and Heisenberg Hamiltonians as well.[1]

5.1 Robustness and Predictability: Redundant Metrics?

Our first result is that predictability implies robustness. Sampling over many LUs, we observe that the value of CF2 is always greater than or equal to the value of CF4. This aligns well with our intuition from Figure 2.9: the distance from an initial pure state towards the center on the Bloch ball (robustness) must be less than or equal to the *total* distance (both towards the center and along the surface) away from the initial state on the Bloch ball (predictability).

The real question, however, is not if this is the case for randomly sampled TPSs, but if it is true for the quasi-classical TPSs (qcTPSs) that constitute the focus of this thesis.

[1] For graphs corresponding to these other initial Hamiltonians, see the "Results" folder on GitHub (link in the Appendix).

Here, our results similarly hold for a variety of initial Hamiltonians at qubit numbers $n = 2, 3,$ and 4. Figure 5.1 displays the minimized TPSs[2] found for the initial 4-qubit central spin Hamiltonian. From the graph, it is clear that a minimal predictability score for a TPS implies a minimal robustness score, however the converse does not hold—a minimal robustness score does not imply a minimal predictability score. This plot—along with the similar results for other initial Hamiltonians and qubit numbers—provides a definitive answer to one of our primary guiding research questions: predictability implies robustness.

This result is highly significant for future research on quasi-classical tensor product factorizations: robustness and predictability are redundant metrics. This supports the claim that—contrary to the proposal of Carroll and Singh [5]—only one of these metrics is necessary to inform the quasi-classical factorization of a Hilbert Space. A natural question arises: if these are truly redundant metrics, which one should be used in future studies of qcTPSs? At the moment, robustness seems to be a more popular choice in the literature [8, 11].

Despite predictability's lack of prevalence in the literature, given that it is the harder metric to satisfy it seems like the natural choice as the better metric to minimize. That being said, it would be nice to have numerical evidence to support this intuition. In Figure 5.2 we directly compare the minimal scores selected by minimizing each function, again for the 4-qubit central spin Hamiltonian. Minimization of both the predictability and robustness cost functions was done to the same threshold value and with the same optimization parameters.

Though these results are less conclusive than the previous, it seems that, in general, predictability-minimized TPSs result in lower scores for both predictability *and* robustness. Thus, even if you feel that robustness is a better metric for quasi-classicality, minimizing predictability will actually allow you to obtain lower robustness scores than if you just attempted to minimize robustness alone. Additionally, given the same machine learning parameters and threshold values, the predictability-minimization consistently ran approximately 40% faster than robustness-minimization. This provides further evidence that predictability is the better metric to minimize.

Why exactly does predictability-minimization appear to have such substantial advantages over robustness-minimization? Some intuition comes from imagining the fitness landscape of each metric. The predictability fitness landscape naturally has fewer, steeper troughs, and thus it relatively easy to find extremely low values for its score. The robustness landscape, on the other hand, has more frequent, shallower troughs, which makes it substantially harder to minimize than predictability. By further optimizing the machine learning parameters, this effect would likely become even more pronounced. Thus, our results support the recommendation that future work in this area (searching for qcTPSs) should use predictability as the metric to minimize rather than robustness.

[2]Keep in mind that when we say "minimized TPSs," we really mean the *Hamiltonians* that minimize our cost functions, as we have adopted the operational definition for Tensor Product Structures in terms of Hamiltonians (see Definition 4). Accordingly, each pair of bars in our graph represents a different Hamiltonian.

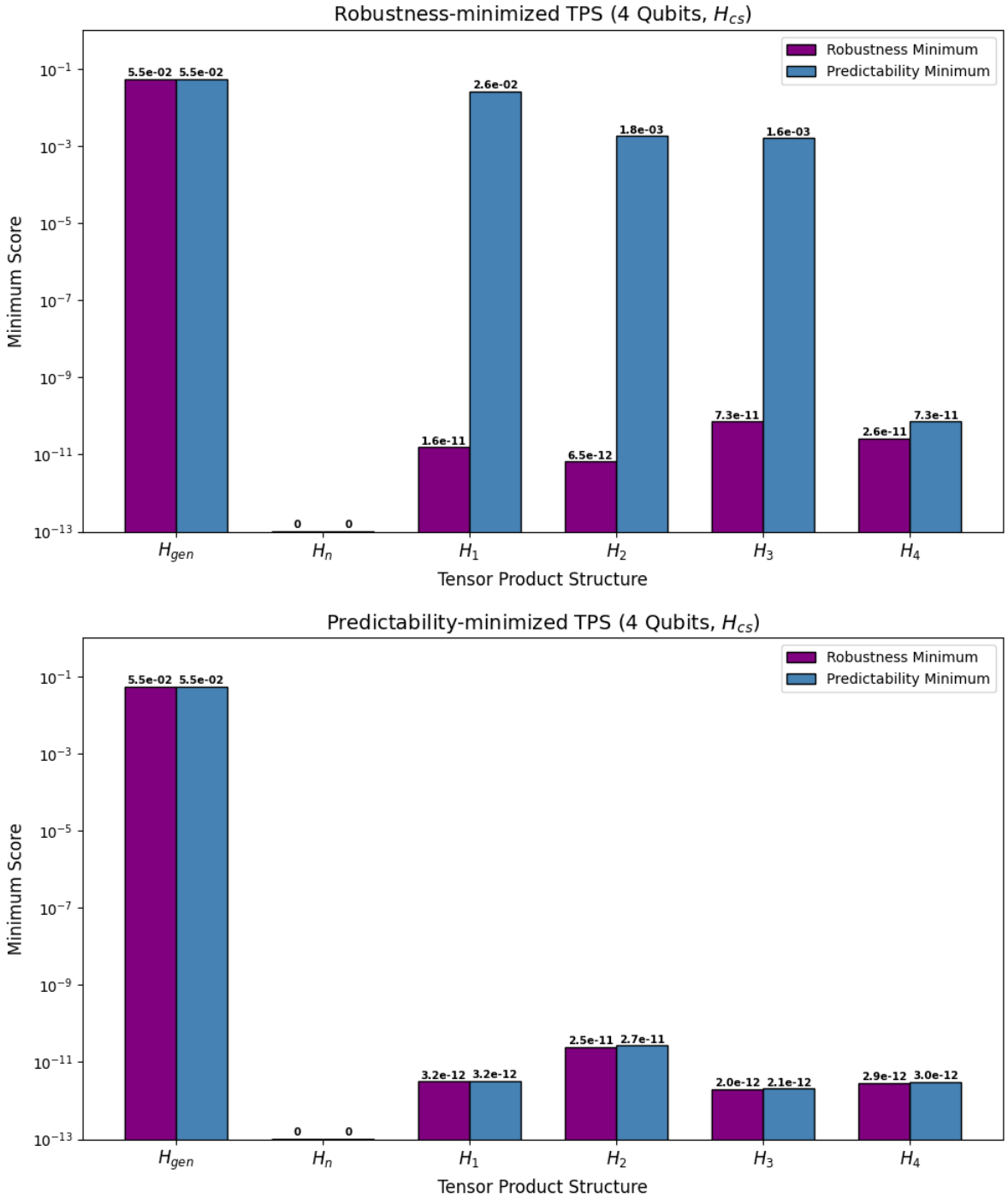


Figure 5.1: Robustness-minimized TPSs (Top) and Predictability-minimized TPSs (Bottom) for the 4-qubit central spin Hamiltonian (Equation 2.10) on a log scale. Each simulation was run with the same threshold value and optimization parameters. The H_{gen} bars display the generic scores for a typical TPS, the H_n bars represent the scores for the initial, "native" Hamiltonian before scrambling with a global unitary matrix, and the remaining bars represent the cost function minimized-TPSs. TPSs with minimal predictability scores also have minimal robustness scores (Bottom), but the converse is not true (Top).

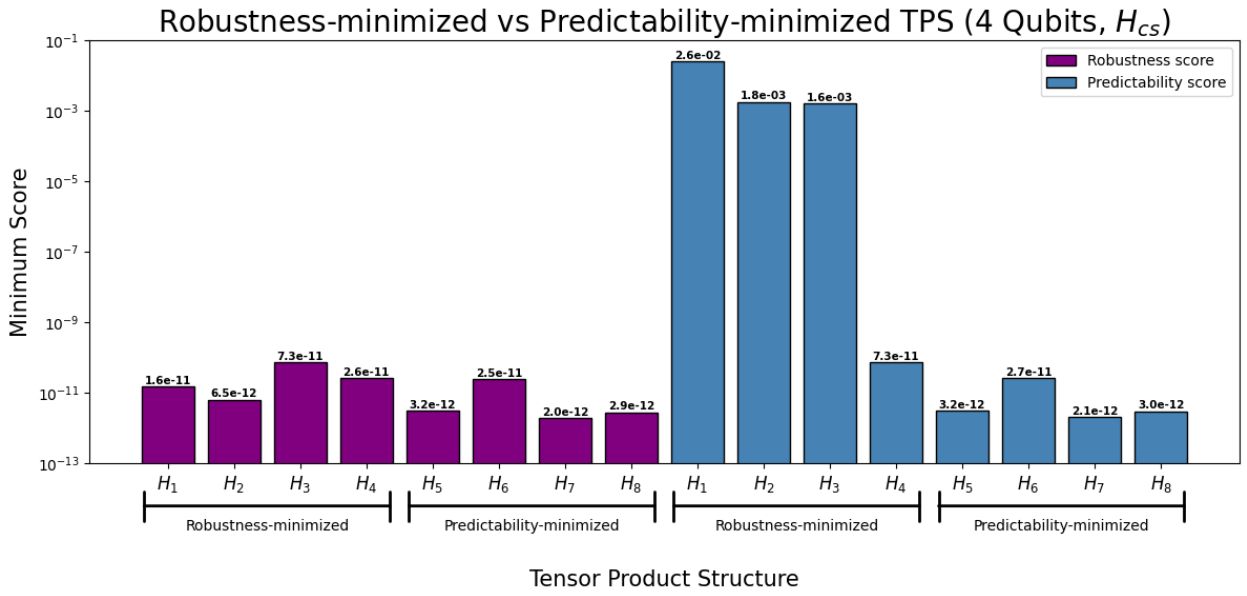


Figure 5.2: A comparison between the effectiveness of predictability-minimization and robustness-minimization for the same optimization parameters. In general, it appears that predictability-minimization tends to result in better scores for both predictability and robustness.

5.2 Generic vs qcTPS Variance

The second primary investigation at the heart of this thesis relates to the variance of scores over a TPS. This direction was largely motivated by a simple hypothesis: if there exists a well-defined pointer state within a given TPS, then there also must exist initial states that rapidly generate (pointer or entanglement) entropy and quickly decohere to a pointer state. Equivalently, in the language of our algorithm: if there exists one initial state for a given TPS with a minimal score, then there also exists other initial states in the same TPS with a really high score. Thus, within such a TPS (a qcTPS), there should exist a high variance in the scores over initial states. Since a LU transformation can rotate our system Bloch ball to any initial state, this variance over initial states simply corresponds to the variance of our cost functions (see Sections 3.1.2 and 3.2.2).

To test this hypothesis, we turned to the variance formulas for CF2 and CF4 derived in Chapter 4 (Equations 3.16 and 3.31). These analytic expressions allowed us to compute the variance of initial scores over local unitary transformations in a computationally-efficient manner. In Figure 5.3, we again use our 4-qubit central spin Hamiltonian to compare the variance of a generic TPS with the variance of TPSs selected via our minimization procedure.[3]

[3]Figure 5.3 also seems to reveal that predictability-minimized TPSs exhibit a larger variance in their scores than robustness-minimized TPSs. This is likely just an artifact of predictability-minimization yielding lower minimal scores than robustness-minimization.

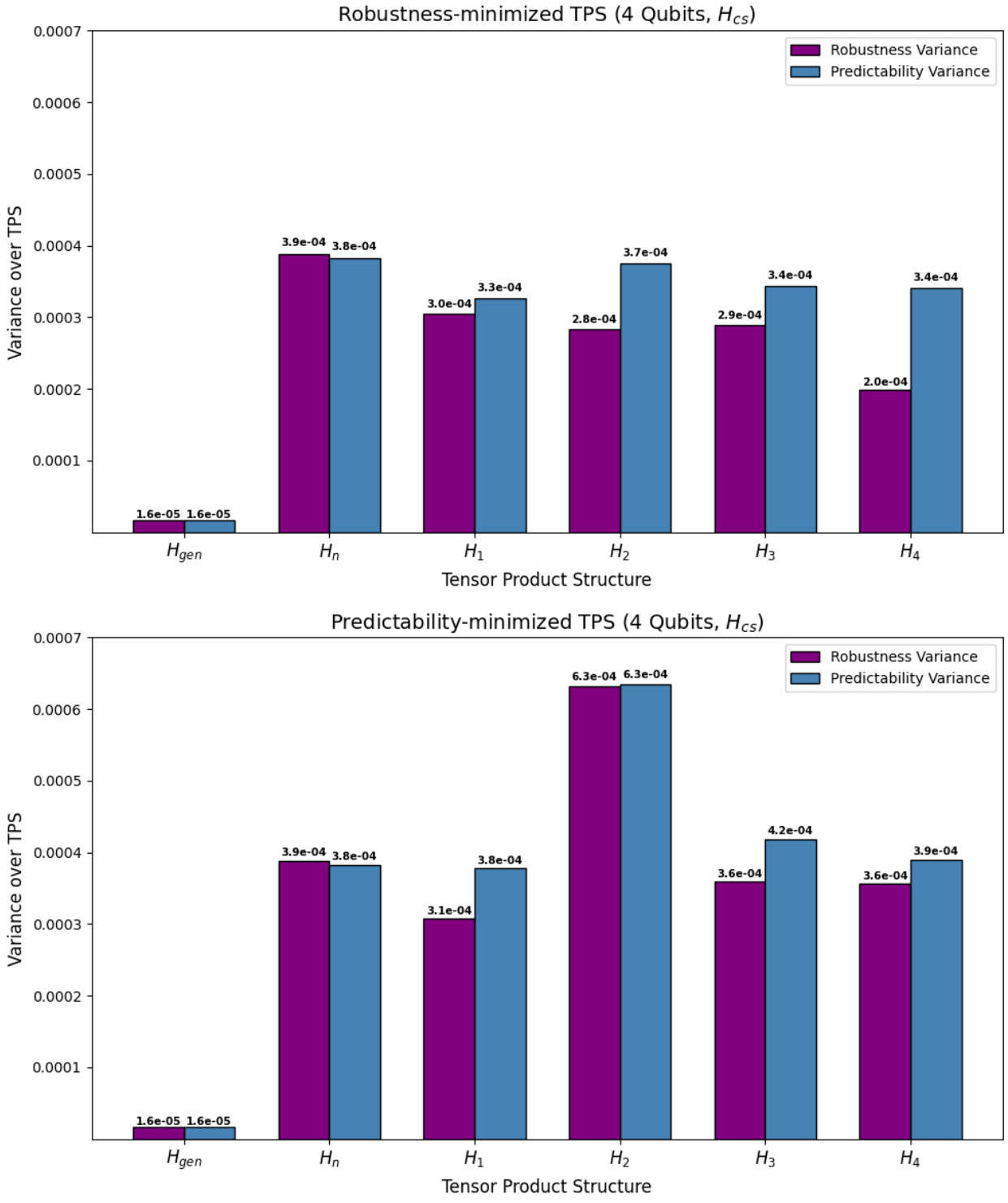


Figure 5.3: The variance of a generic TPS and selected qcTPSs for the 4-qubit central spin Hamiltonian on a linear scale. It is clear that qcTPSs exhibit a significantly higher variance than a generic TPS.

From the figure, it is apparent that the variances of scores across qcTPSs are much higher than the variance of scores across a generic TPS. This supports our hypothesis that if a TPS has a well-defined pointer state, then it also has many states that generate rapid entropy growth and thus decohere quickly. Think back to our potato example from Chapter 1: if our division of subsystems (TPS) is potato and photon—a TPS that does have well-defined pointer states (see Section 1.2)—then it also has states that rapidly generate entanglement and decohere quickly (e.g., the superposition of pointer states from Section 1.1). Thus, our results align nicely with conceptual intuition.

As a final note, the mean predictability and robustness scores across initial states for our 4-qubit central spin Hamiltonian are shown in Figure 5.4. Notice that the mean scores for both the generic TPS and minimized qcTPSs are similar. This is consistent with our variance results and, in fact, helps explain them: if all tensor product structures exhibit similar mean scores, then any TPS with particularly low scores (such as a qcTPS) must also exhibit initial states with particularly high scores. Thus, qcTPSs must have a larger variance.

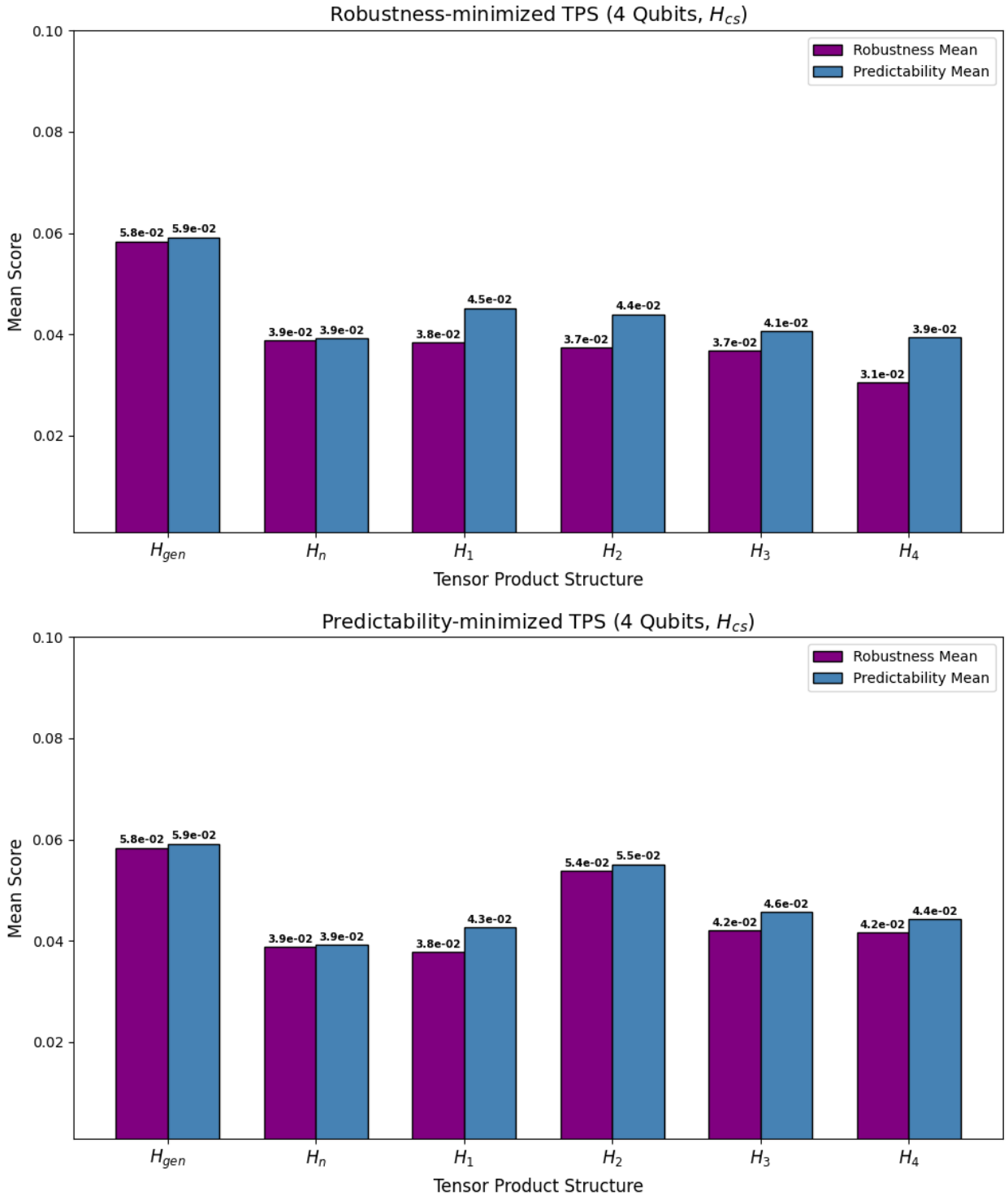


Figure 5.4: The mean of the generic TPS and selected qcTPSs for the 4-qubit central spin Hamiltonian on a linear scale. The mean score across LUs does not differ much between qcTPSs and a generic TPS. These results directly imply that qcTPSs must have a large variance in their scores over initial states.

Chapter 6

Conclusion

Throughout this thesis, we have motivated the *problem of preferred subsystems* left unanswered by the decoherence program, built intuition for the mathematical theory behind this problem, developed novel analytic formulas to aid in a computational search for TPSs with quasi-classical dynamics, performed that computational search with a machine learning algorithm, and obtained numerous original results that will inform future studies in this area. Specifically, we provide numerical evidence that supports the claims that:

1. Predictability implies robustness. In particular, a TPS with a minimal predictability score must also have a minimal robustness score. The converse is not true in general.
2. Quasi-classical TPSs exhibit a significantly higher variance of scores across their initial states than a generic TPS.

There are many areas of ongoing work with this project. One includes characterizing the generic scores and statistics across higher numbers of qubits (e.g., $n = 5, 6, 7, 8...$)—the code to do this has already been implemented, and thus it is simply a matter of the additional computational time to run the simulations. Additionally, we have recently become aware that the scores and statistics remain invariant under local unitaries not only in the infinite temperature limit, but also when the environment is initialized in a finite-temperature thermal state. This provides an entirely new parameter to tune (the "temperature" or "environmental ready state"), and would help our work connect more directly with the literature [12, 24]. Furthermore, the code is currently being set up to run on the Pomona College supercomputing cluster with hopes of running simulations that have previously been computationally inaccessible.

There is also much exciting work on this project that will be left for future students. For one, the space of initial Hamiltonians is a parameter that could be further explored, and there is particular interest in the "generic Hamiltonian" [25]—a general Hamiltonian without any special form—that may help us better generalize our results.[1] Additionally, the machine learning aspect of the code has tremendous room for improvement, which could enable more thorough investigations at higher values of n. Furthermore, given the novelty of system qubit

[1] Confusingly, this is not related to the "generic" TPS of our `generic_score_fn`. In Chapter 4 we are referring to a "generic" TPS for a given initial Hamiltonian (e.g., the central spin), whereas here we are discussing a "generic" initial Hamiltonian itself.

permutations as part of the selection criteria [6], it could be worth further characterizing how it affects the scores. Finally, following Cotler and Lee [3, 17], there is interest in describing the interplay between locality and the emergence of classicality—another area suitable for investigation with this algorithm. All of these future directions are promising and have the potential for their own significant new insights.

It would also be worth further examining the structure of the selected quasi-classical Hamiltonians, and potentially developing analytic arguments suggesting that Hamiltonians with certain structural features yield minimal TPS scores. Preliminary investigations indicate that coupling between pairs of qubits may lead to minimum TPSs.

To conclude, I will leave the reader with some final (more philosophical) questions for marination: given the large space of possible tensor product structures, why does reality seem to favor TPSs with states that act quasi-classically? How exactly does a reality of determinacy seem to emerge out of a chasm of complete indeterminacy? And what really is a quantum measurement?

Perhaps there is no question deeper than that.

Appendix A
GitHub Repository

The code used throughout this thesis, as well as results for additional Hamiltonians beyond the central-spin model, can be found in the following GitHub repository:

`https://github.com/coalescion/decoherence-selected-TPSs`

Appendix B

Mathematica Notebooks

Raw notebook files are contained within the GitHub repository linked in Appendix A. For convenience, a pdf of the notebooks are pasted below:

B.1 CF4 Mean and Variance

During Chapter 3, the calculations for the mean variance of Cost Function 4 were relegated to a Mathematica notebook. A full pdf of this notebook—including a test formula that checks its results—are displayed beginning on the next page.

Analytic Formula for Variance of CF4

We wish to derive expressions for the mean and variance of the predictability entropy, which is in terms of a quartic from on the unit sphere. The quartic form is specified by a real, symmetric, 3x3 matrix P_tilda (notated below as P for simplicity). All such matrices are orthogonally diagonalizable, meaning that there exists some choice of axes x,y,z (the eigenvectors of P) where P takes the form:

In[1]:=
```
P = (λ[1] x^2 + λ[2] y^2 + λ[3] z^2) /. {
  x → Sin[θ] Cos[ϕ],
  y → Sin[θ] Sin[ϕ],
  z → Cos[θ]
}
```

Out[1]= $\text{Cos}[\phi]^2\ \text{Sin}[\theta]^2\ \lambda[1] + \text{Sin}[\theta]^2\ \text{Sin}[\phi]^2\ \lambda[2] + \text{Cos}[\theta]^2\ \lambda[3]$

where $\lambda[1]$, $\lambda[2]$, $\lambda[3]$ are the three eigenvalues of M.

In[2]:=
```
$Assumptions = {0 < θ < π, 0 < ϕ < 2 π}
```

Out[2]= $\{0 < \theta < \pi,\ 0 < \phi < 2\ \pi\}$

First, we compute the mean of CF4. Starting with the second moment <P^2>:

In[3]:=
```
secondMoment = 1/(4 π) ∫_π^0 ∫_(2 π)^0 P^2 (Sin[θ]) dϕ dθ // Simplify
```

Out[3]= $\frac{1}{15}\left(3\ \lambda[1]^2 + 3\ \lambda[2]^2 + 2\ \lambda[2] \times \lambda[3] + 3\ \lambda[3]^2 + 2\ \lambda[1]\ (\lambda[2] + \lambda[3])\right)$

SymmetricReduction will allow us to rewrite the expression in terms of traces and det(P). The first returned element is the symmetric part and the second returned element is the remainder. We want the symmetric part for our simplification.

```
symRedP2 = SymmetricReduction[secondMoment, {λ[1], λ[2], λ[3]}]
```

The symmetric part:

In[19]:=
```
symRedP2[[1]]
```

Out[19]=

$$\frac{1}{5}\ (\lambda[1] + \lambda[2] + \lambda[3])^2 - \frac{4}{15}\ (\lambda[1] \times \lambda[2] + \lambda[1] \times \lambda[3] + \lambda[2] \times \lambda[3])$$

Specifying the traces of P, P^2, P^3, P^4, and det(P) in terms of lambda (will be useful for both mean and variance):

In[20]:=
```
rules = {λ[1] + λ[2] + λ[3] → t1,
    λ[1]* λ[2] + λ[1]* λ[3] + λ[2]* λ[3] → (t1^2 - t2)/2,
    λ[1]* λ[2] *λ[3] → d,
    λ[1]^2 + λ[2]^2 + λ[3]^2 → t2,
    λ[1]^3 + λ[2]^3 + λ[3]^3 → t3,
    λ[1]^4 + λ[2]^4 + λ[3]^4 → t4};
```

Thus, simplifying our expression and calculating the mean:

In[48]:=
```
MeanSphereFinal = FullSimplify[symRedP2[[1]] //. rules,
   Assumptions → {λ[1] ∈ Reals, λ[2] ∈ Reals, λ[3] ∈ Reals}];
MeanFinal = (1/2) - (1/8)*(MeanSphereFinal)
```

Out[49]=

$$\frac{1}{2}+\frac{1}{120}\left(-\text{t1}^2-2\,\text{t2}\right)$$

The mean written explicitly in terms of our matrix P:

```
MeanFinalFinal = MeanFinal /. {t1 → Tr[Pmat], t2 → Tr[Pmat], d → Det[Pmat]}
```

Out[50]=

$$\frac{1}{2}+\frac{1}{120}\left(-\text{Tr}[\text{Pmat}]^2-2\,\text{Tr}[\text{Pmat.Pmat}]\right)$$

Now, we compute the variance. Starting with the fourth moment <P^4>:

In[51]:=
```
fourthMoment = 1/(4 π) ∫_π^0 ∫_(2 π)^0 P^4*(Sin[θ]) dϕ dθ // Simplify
```

Out[51]=

$$\frac{1}{315}\left(35\,\lambda[1]^4+35\,\lambda[2]^4+20\,\lambda[2]^3\,\lambda[3]+18\,\lambda[2]^2\,\lambda[3]^2+20\,\lambda[2]\,\lambda[3]^3+35\,\lambda[3]^4+20\,\lambda[1]^3\,(\lambda[2]+\lambda[3])+6\,\lambda[1]^2\left(3\,\lambda[2]^2+2\,\lambda[2]\times\lambda[3]+3\,\lambda[3]^2\right)+4\,\lambda[1]\,(\lambda[2]+\lambda[3])\left(5\,\lambda[2]^2-2\,\lambda[2]\times\lambda[3]+5\,\lambda[3]^2\right)\right)$$

I calculated by hand the variance formula in terms of the <P^4> and <P^2>:

In[30]:=
```
varianceS = (1/64)*(fourthMoment - (secondMoment)^2) // FullSimplify
```

Out[30]=

$$\frac{1}{6300}\left(7\,\lambda[1]^4+7\,\lambda[2]^4+\lambda[2]^3\,\lambda[3]-4\,\lambda[2]^2\,\lambda[3]^2+\lambda[2]\,\lambda[3]^3+7\,\lambda[3]^4+\lambda[1]^3\,(\lambda[2]+\lambda[3])+\lambda[1]\,(\lambda[2]+\lambda[3])\left(\lambda[2]^2-6\,\lambda[2]\times\lambda[3]+\lambda[3]^2\right)-\lambda[1]^2\left(4\,\lambda[2]^2+5\,\lambda[2]\times\lambda[3]+4\,\lambda[3]^2\right)\right)$$

In[32]:=
```
symRedP4 = SymmetricReduction[varianceS, {λ[1], λ[2], λ[3]}];
```

Our final expression in terms of P, P^2, P^3, P^4, and det(P):

In[34]:=
```
finalVar =
 FullSimplify[symRedP4[[1]] //. rules, Assumptions → {λ[1] ∈ Reals, λ[2] ∈ Reals, λ[3] ∈ Reals}]
```

Out[34]=

$$\frac{60\,d\,t1 - 9\,t1^4 + 19\,t1^2\,t2 + 4\,t2^2}{12\,600}$$

The variance written explicitly in terms of P:

In[46]:=
```
finalfinalVar = finalVar /. {t1 → Tr[Pmat], t2 → Tr[Pmat.Pmat], d → Det[Pmat]}
```

Out[46]=

$$\frac{60\,\text{Det[Pmat]} \times \text{Tr[Pmat]} - 9\,\text{Tr[Pmat]}^4 + 19\,\text{Tr[Pmat]}^2\,\text{Tr[Pmat.Pmat]} + 4\,\text{Tr[Pmat.Pmat]}^2}{12\,600}$$

Test formula

Let's draw a large number of random points on the unit sphere in order to statistically test the equations.

In[84]:=
```
Nsamples = 10 000;
n = RandomPoint[Sphere[], Nsamples];
```

Let's choose a random symmetric matrix M by choosing a random matrix A and then multiplying $M = A^T A$:

In[94]:=
```
A = RandomVariate[NormalDistribution[0, 2], {3, 3}];
Psample = Transpose[A].A;
% // MatrixForm
```

Out[96]//MatrixForm=

$$\begin{pmatrix} 15.3984 & 3.63008 & 10.2383 \\ 3.63008 & 2.95746 & 2.13675 \\ 10.2383 & 2.13675 & 7.10017 \end{pmatrix}$$

As a function of the three coordinates n = {n[[1]], n[[2], n[[3]]}, the quadratic form is:

In[97]:=
```
Ssample[n_] := 1/2 - 1/8 (Transpose[n].Psample.n)^2;
```

In[98]:=
```
Ssamples = Ssample /@ n;
meanSsample = Mean[Ssamples];
varianceSsample = Variance[Ssamples]
```

Out[100]=

```
324.605
```

In[101]:=
```
finalfinalVar /. {Pmat → Psample}
```

Out[101]=

```
323.155
```

Bibliography

[1] W. H. Zurek. "Pointer basis of quantum apparatus: Into what mixture does the wave packet collapse?" en. In: *Physical Review D* 24.6 (Sept. 1981), pp. 1516–1525. ISSN: 0556-2821. DOI: 10.1103/PhysRevD.24.1516. URL: https://link.aps.org/doi/10.1103/PhysRevD.24.1516 (visited on 09/09/2024).

[2] Max Tegmark. "Consciousness as a state of matter". en. In: *Chaos, Solitons & Fractals* 76 (July 2015), pp. 238–270. ISSN: 09600779. DOI: 10.1016/j.chaos.2015.03.014. URL: https://linkinghub.elsevier.com/retrieve/pii/S0960077915000958 (visited on 09/22/2024).

[3] Jordan S. Cotler, Geoffrey R. Penington, and Daniel H. Ranard. "Locality from the Spectrum". en. In: *Communications in Mathematical Physics* 368.3 (June 2019), pp. 1267–1296. ISSN: 0010-3616, 1432-0916. DOI: 10.1007/s00220-019-03376-w. URL: http://link.springer.com/10.1007/s00220-019-03376-w (visited on 09/25/2024).

[4] Paolo Zanardi et al. "Operational Quantum Mereology and Minimal Scrambling". en. In: *Quantum* 8 (July 2024), p. 1406. ISSN: 2521-327X. DOI: 10.22331/q-2024-07-11-1406. URL: http://arxiv.org/abs/2212.14340 (visited on 04/10/2025).

[5] Sean M. Carroll and Ashmeet Singh. "Quantum mereology: Factorizing Hilbert space into subsystems with quasiclassical dynamics". English. In: *PHYSICAL REVIEW A* 103.2 (Feb. 2021), p. 022213. ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.103.022213. URL: https://journals.aps.org/pra/abstract/10.1103/PhysRevA.103.022213 (visited on 09/04/2024).

[6] Marissa Singh. "Decoherence and Preferred Tensor Product Structures for Systems of Qubits". en. In: (2023).

[7] K. G. Johnson et al. "Ultrafast creation of large Schrödinger cat states of an atom". en. In: *Nature Communications* 8.1 (Sept. 2017), p. 697. ISSN: 2041-1723. DOI: 10.1038/s41467-017-00682-6. URL: https://www.nature.com/articles/s41467-017-00682-6 (visited on 04/21/2025).

[8] Arsalan Adil et al. *A Search for Classical Subsystems in Quantum Worlds*. en. May 2024. URL: http://arxiv.org/abs/2403.10895 (visited on 09/09/2024).

[9] Wikipedia contributors. *Interpretations of quantum mechanics*. 2024. URL: https://en.wikipedia.org/wiki/Interpretations_of_quantum_mechanics#Influential_interpretations.

[10] Maximilian Schlosshauer. *Decoherence and the Quantum-To-Classical Transition*. en. Frontiers Collection. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. ISBN: 978-3-540-35773-5 978-3-540-35775-9. URL: http://link.springer.com/10.1007/978-3-540-35775-9 (visited on 09/22/2024).

[11] Wojciech H. Zurek. "Decoherence, Einselection, and the Existential Interpretation (the Rough Guide)". en. In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 356.1743 (Aug. 1998), pp. 1793–1821. ISSN: 1364-503X, 1471-2962. DOI: 10.1098/rsta.1998.0250. URL: http://arxiv.org/abs/quant-ph/9805065 (visited on 10/22/2024).

[12] Nicolas Loizeau and Dries Sels. "Quantum mereology and subsystems from the spectrum". en. In: *Foundations of Physics* 55.1 (Feb. 2025), p. 3. ISSN: 0015-9018, 1572-9516. DOI: 10.1007/s10701-024-00813-2. URL: http://arxiv.org/abs/2409.01391 (visited on 04/08/2025).

[13] Qubit.guide contributors. *8.7 Partial Trace Revisited*. en. https://qubit.guide/8.7-partial-trace-revisited. Apr. 2025. (Visited on 04/22/2025).

[14] tobiasBora. *How to Draw a Bloch Sphere?* July 2021. URL: https://tex.stackexchange.com/questions/345420/how-to-draw-a-bloch-sphere (visited on 12/12/2024).

[15] Fengping Jin et al. "Quantum decoherence scaling with bath size: Importance of dynamics, connectivity, and randomness". In: *Phys. Rev. A* 87.2 (Feb. 2013), p. 022117. DOI: 10.1103/PhysRevA.87.022117. URL: https://link.aps.org/doi/10.1103/PhysRevA.87.022117.

[16] Rafael I. Nepomechie and Xi-Wen Guan. "The spin-s homogeneous central spin model: exact spectrum and dynamics". en. In: *Journal of Statistical Mechanics: Theory and Experiment* 2018.10 (Oct. 2018), p. 103104. ISSN: 1742-5468. DOI: 10.1088/1742-5468/aae2d9. URL: http://arxiv.org/abs/1810.03937 (visited on 04/17/2025).

[17] Brian Lee. "Existence of Well-Defined Pointer Observable Selects Tensor Product Factorizations of Quantum Systems". en. In: (2023).

[18] Wikipedia contributors. *Rayleigh quotient*. en. https://en.wikipedia.org/wiki/Rayleigh_quotient. Apr. 2025. (Visited on 04/22/2025).

[19] Kevin Setter. *Written Notes*.

[20] Howard Haber. *Angular Integrals for Spherical Harmonics*. 2025. URL: https://scipp.ucsc.edu/~haber/ph214/Angular_Integrals.pdf (visited on 04/17/2025).

[21] Xiaoli Cen and Yong Xia. "Globally maximizing the sum of squares of quadratic forms over the unit sphere". In: *Optimization Letters* 14 (Oct. 2020), pp. 1–13. DOI: 10.1007/s11590-019-01498-7.

[22] Christoph Spengler, Marcus Huber, and Beatrix C. Hiesmayr. "A composite parameterization of unitary groups, density matrices and subspaces". en. In: *Journal of Physics A: Mathematical and Theoretical* 43.38 (Sept. 2010), p. 385306. ISSN: 1751-8113, 1751-8121. DOI: 10.1088/1751-8113/43/38/385306. URL: http://arxiv.org/abs/1004.5252 (visited on 04/09/2025).

[23] A. M. Martins. "Necessary and Sufficient Conditions for Local Unitary Equivalence of Multi-qubit States". en. In: *Physical Review A* 91.4 (Apr. 2015), p. 042308. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.91.042308. URL: http://arxiv.org/abs/1408.3840 (visited on 04/09/2025).

[24] M. A. Novotny et al. "Quantum Decoherence and Thermalization at Finite Temperature within the Canonical Thermal State Ensemble". en. In: *Physical Review A* 93.3 (Mar. 2016), p. 032110. ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.93.032110. URL: http://arxiv.org/abs/1601.04209 (visited on 04/08/2025).

[25] Mario Bessa et al. *Generic Hamiltonian Dynamics*. en. Feb. 2015. DOI: 10.48550/arXiv.1203.3849. URL: http://arxiv.org/abs/1203.3849 (visited on 04/20/2025).