

# **DYNAMICAL DECOUPLING UNDER STRUCTURED NOISE: A SIMULATION-BASED STUDY**

A THESIS SUBMITTED FOR THE COMPLETION OF  
REQUIREMENTS FOR THE DEGREE OF

**Master of Technology**

IN

**Quantum Technology**

BY

**Dorbala Trivikramahimamshu Shekhar**

Postgraduate Programme  
Indian Institute of Science



Under the supervision of

**Prof. Baladitya Suri**

Department of Instrumentation and Applied Physics  
Indian Institute of Science

**May 2025**

# Acknowledgements

I would first like to express my sincere gratitude to my guide and project advisor, Prof. Baladitya Suri, for his constant support and guidance throughout my M.Tech. Your encouragement and clarity have been invaluable to the completion of this project. I also feel fortunate to have taken multiple courses with you-Physics and Engineering Foundation for Quantum Technology, Introduction to Quantum Measurements, and Quantum Optics-which laid the conceptual groundwork for everything quantum. I hope to continue learning under your mentorship in the future as well.

I am deeply grateful to my parents for their love and unwavering support. I would like to thank all the teachers who guided me through my school years, for their kindness and encouragement that helped me grow. I am especially thankful to my intermediate lecturers, who inspired me to pursue academics as a career.

I am extremely grateful to my friends. Vivek, thank you for always being in my corner, even when everything seemed to be going wrong. I also want to thank my batchmates Gouranga, Sannzay, and Guru for listening to my endless ramblings about things I found fascinating, especially my project, and for offering invaluable suggestions along the way.

I would also like to thank Prof. Apoorva Patel, Prof. Shayan Garani, Prof. Navin Kashyap, and Prof. Phani Motamarri for their inspiring teaching during the course of my M.Tech. Though not directly related to my project, their courses broadened my understanding of quantum theory and scientific computing, and played an important role in shaping my academic journey.

# Abstract

Dynamical decoupling (DD) is a lightweight error-mitigation technique that combats decoherence in noisy intermediate-scale quantum (NISQ) devices. This thesis presents a comprehensive, simulation-based evaluation of several widely used DD sequences under realistic noise models, using a custom Julia package `QuantumDD.jl`. The study has two goals: **(i)** to quantify how effectively different DD protocols preserve qubit coherence across structured noise environments, and **(ii)** to demonstrate how the filter-function formalism can both predict and, in principle, reconstruct features of the underlying noise spectrum.

A time-dependent pure-dephasing Hamiltonian  $H(t) = \beta(t)\sigma_z$  is adopted as the baseline, with  $\beta(t)$  generated from user-specified power spectral densities (PSDs). White,  $1/f$ , and Ornstein–Uhlenbeck noise are investigated, together with composite multiaxial variants that inject weaker transverse fluctuations.

The DD protocols analysed are: Free Induction Decay (FID), Hahn Echo, Carr-Purcell-Meiboom-Gill (CPMG), Periodic DD (PDD), Concatenated DD (CDD), and Uhrig DD (UDD), with pulse counts up to  $N = 800$ . Closed-form filter functions are given where available (FID, Hahn, CPMG); otherwise they are computed numerically from the modulation function. Simulated fidelity curves reveal:

- CPMG delivers nearly an order-of-magnitude extension of the effective coherence time  $T_{2,\text{eff}}$  under white noise at  $N = 800$ , but offers diminishing returns against strongly coloured noise.
- UDD outperforms CPMG for  $1/f$  and OU spectra, exhibiting stretch exponents  $\beta \gtrsim 2$  that indicate a crossover from exponential to Gaussian decay.
- Multiaxial simulations confirm that uniaxial sequences lose protection when transverse noise is comparable in strength, motivating future deployment of XY-type sequences.

All code, notebooks, and plotted data are available in the accompanying repository `Quantum-Dynamical-Decoupling-Toolkit` (GitHub). All simulations were performed using custom Julia code available at [my GitHub repository](#).

The thesis concludes with a critical discussion of limitations—ideal pulse shapes, classical-noise assumption, single-qubit scope—and outlines extensions to shaped or randomized DD, multi-qubit settings, and full noise-spectroscopy.

# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Mechanisms of Noise in qubits . . . . .	1
1.1.1 Dephasing . . . . .	1
1.1.2 Relaxation . . . . .	2
1.1.3 Decoherence . . . . .	2
<b>2 Theoretical Foundations of Dynamical Decoupling</b>	<b>4</b>
2.1 Setup . . . . .	4
2.2 Decoupling set . . . . .	5
2.3 Decoupling cycle . . . . .	6
2.4 Repeated Cycles and Long-Time Limit . . . . .	8
2.5 Bounding the Approximation Error . . . . .	9
2.6 System-Environment Interaction . . . . .	10
2.6.1 Sufficiency Condition for Effective Dynamical Decoupling . . . . .	10
2.7 Effect of Finite-Duration Pulses . . . . .	11
2.8 Note on Non-Uniform Pulse Spacing . . . . .	12
2.9 Summary of the Formalism . . . . .	12
<b>3 Filter Function Formalism</b>	<b>14</b>
3.1 Motivation for Frequency Domain Analysis . . . . .	14
3.2 The Model Hamiltonian . . . . .	14
3.3 Time Evolution Under Dephasing Noise . . . . .	15
3.4 Control Pulses and the Modulation Function . . . . .	16
3.5 Decoherence Functional and the Filter Function . . . . .	17
3.6 Extension of Filter Function Formalism to Multiaxial Noise . . . . .	20
3.7 Filter Function Formalism for Noise Spectroscopy . . . . .	21
<b>4 Common DD sequences and their Filter Functions</b>	<b>23</b>
4.1 Free Induction Decay (FID) . . . . .	23
4.1.1 FID Filter Function . . . . .	24
4.2 Hahn Echo . . . . .	25
4.2.1 Hahn Echo Pulse Scheduling and Modulation Plots . . . . .	26
4.2.2 Hahn Echo Filter Function . . . . .	26
4.3 Carr-Purcell-Meiboom-Gill (CPMG) . . . . .	28
4.3.1 CPMG Pulse Scheduling and Modulation Plots . . . . .	29

4.3.2	CPMG Filter Function . . . . .	30
4.4	Periodic Dynamical Decoupling (PDD) . . . . .	33
4.4.1	PDD Pulse Scheduling and Modulation Plots . . . . .	34
4.4.2	PDD Filter Function . . . . .	35
4.5	Concatenated Dynamical Decoupling (CDD) . . . . .	37
4.5.1	CDD Pulse Scheduling and Modulation Plots . . . . .	38
4.5.2	$\ell$ th level CDD Filter Function . . . . .	39
4.6	Uhrig Dynamic Decoupling (UDD) . . . . .	42
4.6.1	UDD Pulse Scheduling and Modulation function . . . . .	43
4.6.2	UDD Filter Function plots . . . . .	44
4.7	Multiaxial Dynamical Decoupling Sequences . . . . .	47
4.7.1	The XY Family of DD Sequences . . . . .	47
4.7.2	Other Notable Multiaxial DD Sequences . . . . .	48
<b>5</b>	<b>Simulation Framework and Workflow</b>	<b>49</b>
5.1	Overview of the Simulation Framework . . . . .	49
5.2	Assumptions and Their Justification . . . . .	50
5.3	Noise Generation . . . . .	50
5.3.1	Spectral Density Specification . . . . .	50
5.3.2	Time-Domain Synthesis of $\beta(t)$ . . . . .	52
5.3.3	Validation . . . . .	52
5.4	Control Pulse Specification and Scheduling . . . . .	55
5.5	Simulation Setup . . . . .	55
5.6	Filter Functions and Theoretical Coherence Estimation . . . . .	56
5.6.1	Simulation Workflow: CPMG under White Noise in Longitudinal axis and OU Noise in the Transverse axes . . . . .	57
5.6.2	Note on Noise Strength Calibration . . . . .	58
<b>6</b>	<b>Results and Discussion</b>	<b>59</b>
6.1	Simulation Results . . . . .	61
6.1.1	Hahn Echo . . . . .	61
6.1.2	CPMG Sequence . . . . .	64
6.1.3	UDD Sequence . . . . .	66
6.1.4	CDD Sequence . . . . .	67
6.1.5	CPMG under Longitudinal White Noise with Transverse Noise . . . . .	69
6.1.6	CPMG under Longitudinal 1/f Noise with Transverse Noise . . . . .	72
6.1.7	CPMG under Longitudinal OU Noise with Transverse Noise . . . . .	73
6.1.8	Final Remarks on Results and Limitations . . . . .	75
6.2	Conclusion . . . . .	76

# List of Figures

2.1	The application of control pulses $\gamma_k$ and free evolution under the system Hamiltonian $\hat{\mathcal{H}}(t)$ . . . . .	6
2.2	The interpretation of controlled evolution as a sequence of conjugated Hamiltonians $\hat{\mathcal{H}}_k = \mathcal{V}_k \hat{\mathcal{H}} \mathcal{V}_k^\dagger$ , where the effect of each pulse is to rotate the system Hamiltonian before each free evolution interval. . . . .	7
4.1	Filter function plots for FID with increasing total duration $T$ . The horizontal axis represents frequency $\omega$ and the vertical axis shows the filter function $F(\omega, T)$ . . . . .	25
4.2	Hahn Echo: Pulse schedule and corresponding modulation function. . . . .	26
4.3	Filter function plots for Hahn Echo with increasing total duration $T$ . The horizontal axis represents frequency $\omega$ and the vertical axis shows the filter function $F(\omega, T)$ . . . . .	28
4.4	Pulse schedules and modulation functions for the CPMG sequence with increasing number of pulses $N$ . . . . .	29
4.5	filter function plots for CPMG for fixed number of pulses $N = 8$ , and increasing $T$ . The horizontal axis represents frequency $\omega$ and the vertical axis shows the filter function $F(\omega, T)$ . . . . .	32
4.6	PDD Pulse Schedules and Modulation Functions for different pulse counts $N$ . . . . .	34
4.7	Actual PDD filter functions for increasing total durations $T$ , with fixed number of pulses $N = 10$ . These use the full filter function expression, not the small-angle approximation. The horizontal axis represents frequency $\omega$ and the vertical axis shows the filter function $F(\omega, T)$ . . . . .	36
4.8	CDD Pulse Schedules and Corresponding Modulation Functions for Recursion Levels 1 to 4 . . . . .	38
4.9	CDD Filter Function $F_\ell(\omega, \tau)$ for $\tau = 1, 2$ and $\ell = 1$ to 4 . . . . .	40
4.10	CDD Filter Function $F_\ell(\omega, \tau)$ for $\tau = 3, 4$ and $\ell = 1$ to 4 . . . . .	41
4.11	Pulse scheduling and modulation functions for UDD sequences at different pulse counts . . . . .	43
4.12	UDD pulse schedules and modulation functions showing strong nonlinearity as $N$ increases. The pulses become increasingly concentrated toward the boundaries. . . . .	44
4.13	UDD filter functions for $T = 2$ with varying pulse numbers. . . . .	45
4.14	UDD filter functions for $T = 4$ with varying pulse numbers. . . . .	46
5.1	A single realization $\beta(t)$ generated using the target $1/f$ spectral density.	53
5.2	Validation plots for noise generation. Each trace reflects the intended $1/f$ spectral behavior, with averaging improving spectral accuracy. . . . .	54

6.1	Hahn echo applied to different noise models. The FID plots are almost completely overlapped by the Hahn echo curves, demonstrating that Hahn echo offers negligible improvement under white, $1/f$ , or OU noise. . . . .	62
6.2	Comparison of Hahn echo performance under different noise types with added DC component. Unlike the pure stochastic cases, here a clear recovery is observed due to cancellation of quasi-static detuning. . . . .	63
6.3	Ensemble-averaged fidelity for CPMG sequence subjected to three representative noise spectra. Each panel shows FID and CPMG curves for $N = 100–800$ pulses, together with stretched-exponential fits. . . . .	65
6.4	Performance of UDD sequences against different noise types at high pulse counts. . . . .	66
6.5	Performance of CDD under different noise models: white, $1/f$ , and OU. .	68
6.6	CPMG under multiaxial noise with white noise along the longitudinal axis and varying transverse noise models: (Top) white, (Middle) $1/f$ , (Bottom) Ornstein–Uhlenbeck (OU). . . . .	71
6.7	CPMG simulations under longitudinal $1/f$ noise combined with different transverse noise models: (top) transverse white, (bottom) transverse $1/f$ , .	72
6.8	CPMG simulations under longitudinal $1/f$ noise combined with transverse OU noise. . . . .	73
6.9	CPMG performance under longitudinal OU noise combined with transverse white noise (top), transverse $1/f$ noise (middle), and transverse OU noise (bottom). . . . .	74

# List of Tables

2.1	Pulse operators and timings in the decoupling sequence . . . . .	6
5.1	Interpretation of parameters in the general spectral density function $S(\omega)$	51
6.1	Relative increase in effective coherence time $T_{2,\text{eff}}/T_2^*$ for CPMG under various noise models. . . . .	64
6.2	Comparison of $T_{2,\text{eff}}/T_2^*$ and stretch exponent $\beta$ for UDD under different noise models . . . . .	67
6.3	Comparison of $T_{2,\text{eff}}/T_2^*$ and stretch exponent $\beta$ for CDD under different noise models . . . . .	69
6.4	Comparison of $T_{2,\text{eff}}/T_2^*$ and stretch exponent $\beta$ for CPMG under longitudinal white noise with varied transverse noise . . . . .	70
6.5	Comparison of $T_{2,\text{eff}}$ for CPMG under uniaxial white noise vs white noise combined with transverse components. Only common pulse counts are compared. . . . .	70
6.6	Percentage decrease in $T_{2,\text{eff}}$ compared to uniaxial white noise case for CPMG simulations. Here, W = White, $1/f = 1/f$ noise, and OU = Ornstein–Uhlenbeck noise. . . . .	70
6.7	Comparison of $T_{2,\text{eff}}/T_2^*$ and stretch exponent $\beta$ for CPMG under longitudinal OU noise with different transverse noise components. . . . .	73
6.8	Percentage decrease in $T_{2,\text{eff}}$ compared to uniaxial OU noise case for CPMG simulations. Transverse noise models include white, $1/f$ , and OU. . . . .	75

# Chapter 1

## Introduction

### 1.1 Mechanisms of Noise in qubits

The noise affecting qubits is seen in the form of the decoherence, where after preparing the qubit in a superposition state (typically  $|+\rangle$ ), as the time passes the qubit loses its state and eventually ends up in the ground state. The phenomenon of decoherence is explained by the two mechanisms acting together, namely dephasing (sometimes called pure dephasing), and relaxation.

#### 1.1.1 Dephasing

This is the process where the qubit slowly drifts in phase due to random noise from interaction with the environment. In this process there is no energy loss in the qubit. On the Bloch sphere, dephasing appears as random drift in the x–y plane, with no movement along the z-axis (assuming the computational basis along z). These fluctuations arise from environmental noise<sup>1</sup>, often modeled as longitudinal noise

$$H_{\text{dephasing}} = \beta(t)\hat{\sigma}_z \quad (1.1)$$

Here  $\beta(t)$  is stochastic process representing environmental fluctuations. Dephasing is characterized by dephasing time  $T_\varphi$  which represents the characteristic time over which the off-diagonal elements (the coherences) of the qubit's density matrix decay, without any population transfer between energy levels.<sup>2</sup>

In the presence of only dephasing, the density matrix for a qubit initially in a superposition evolves as

---

<sup>1</sup>See Ithier, G., et al. (2005). Decoherence in a superconducting quantum bit. <https://doi.org/10.1103/PhysRevB.72.134519>

<sup>2</sup>Widely used model for pure dephasing; see standard quantum information textbooks or Ithier et al.

$$\rho(t) = \begin{bmatrix} \rho_{00} & \rho_{01}e^{-t/T_\varphi} \\ \rho_{10}e^{-t/T_\varphi} & \rho_{11} \end{bmatrix} \quad (1.2)$$

### 1.1.2 Relaxation

This is the process where the qubit releases energy into the environment and instantaneously goes back to the ground state  $|0\rangle$ . Unlike dephasing which preserves energy but disturbs phase coherence, relaxation irreversibly removes excitation from the system. Relaxation occurs over a characteristic timescale denoted by  $T_1$ .

In physical systems like superconducting qubits, this energy loss arises due to coupling with electromagnetic modes or other environmental degrees of freedom. On the Bloch sphere, relaxation pulls the state vector up along the z-axis toward  $|0\rangle$ .

This process is modeled by the amplitude damping channel. For completeness, the evolution of a qubit under relaxation can be expressed using Kraus operators<sup>3</sup>

$$K_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix} \quad K_1 = \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix} \quad (1.3)$$

where  $\gamma$  is the probability of decay from  $|1\rangle$ , and it is related to the relaxation time  $T_1$  as  $\gamma = 1 - e^{-t/T_1}$

### 1.1.3 Decoherence

This is an umbrella term for the combined process of dephasing and relaxation. Together, these processes cause a quantum system to gradually lose its quantum character and behave more classically. Decoherence limits the performance of quantum computation and communication, making its suppression a central challenge in quantum technologies. Decoherence is characterized by the decoherence time  $T_2$  and is related to the dephasing and relaxation times by the following relation<sup>4</sup>

$$\frac{1}{T_2} = \frac{1}{2T_1} + \frac{1}{T_\varphi} \quad (1.4)$$

If  $T_1 \gg T_\varphi \implies T_2 \approx T_\varphi$ . In my project I have mainly operated under this assumption. This approximation is often necessary to make simulation and analysis feasible in the literature.

A natural question that arises after understanding how a qubit loses information is: **“What can we do to prevent this from happening?”** There are several strategies

---

<sup>3</sup>Nielsen, M. A., & Chuang, I. L. *Quantum Computation and Quantum Information* (2010).

<sup>4</sup>Standard decoherence relation; derived in many quantum optics texts, e.g., Scully & Zubairy.

developed to tackle this problem, each with its own complexity and limitations<sup>5</sup>. Broadly, these fall into three categories:

1. **Error Correction Codes:** These encode logical qubits into multiple physical qubits, allowing for detection and correction of errors without directly observing the quantum state. While powerful, they require significant hardware overhead and high gate fidelities.
2. **Improved Qubit Design and Isolation:** Engineering better qubits—such as transmons with optimized geometry—and improving shielding from environmental noise can help increase coherence times.
3. **Dynamical Decoupling (DD):** Instead of preventing noise, DD sequences fight back by rapidly flipping the qubit’s state, effectively averaging out certain types of noise over time<sup>6</sup>. This technique is relatively low-cost, doesn’t require ancilla qubits, and is what I focus on in this project.

While no single approach is a silver bullet, dynamical decoupling offers a lightweight and practical solution to improve coherence, especially in systems where full error correction is not yet feasible.

### Dynamical Decoupling:

Dynamical Decoupling (DD) is a noise suppression technique that combats decoherence by deliberately applying control pulses to the qubit. These pulses are designed to systematically reverse or average out the unwanted evolution caused by environmental noise. DD techniques exploit symmetries in the noise and the control operations to suppress specific types of decoherence, particularly dephasing noise. Different DD sequences (like Hahn echo, CPMG, UDD, etc.) vary in complexity and performance depending on the noise spectrum and pulse timing<sup>7</sup>. All of this and more is discussed in the following chapters.

---

<sup>5</sup>Summarized in: Preskill, J. (2018). *Quantum Computing in the NISQ era and beyond*, arXiv:1801.00862

<sup>6</sup>For DD as noise filtering, see Biercuk et al., Nature, 458, 996 (2009) and arXiv:1012.4262

<sup>7</sup>Cywiński, L., Lutchyn, R. M., Nave, C. P., & Das Sarma, S. (2008). How to enhance dephasing time in superconducting qubits. *Phys. Rev. B*, **77**(17), 174509.

# Chapter 2

## Theoretical Foundations of Dynamical Decoupling

The theoretical framework behind DD sequences draws from the idea that by interspersing the system's evolution with unitary operations, it is possible to average out or nullify the net effect of unwanted Hamiltonian dynamics over time. This concept was first demonstrated in the context of spin echo experiments in nuclear magnetic resonance (NMR)<sup>1</sup> and later generalized to arbitrary quantum systems<sup>2</sup>.

Importantly, this approach does not require any prior knowledge of the exact form of the system-environment interaction Hamiltonian, making it highly practical in experimental contexts.<sup>3</sup> The control pulses are applied solely to the system, no feedback or intervention on the environment is assumed or required.

### 2.1 Setup

Consider a finite dimensional Hilbert space  $\mathcal{H}_s$  and a certain Hamiltonian  $\hat{\mathcal{H}} = \hat{\mathcal{H}}^\dagger \in B(\mathcal{H}_s)$ . Where  $B(\mathcal{H}_s)$  this is the set of all Bounded Linear operators on  $\mathcal{H}_s$ .

To implement a dynamical decoupling protocol, we interleave periods of free evolution with the application of unitary control pulses  $\gamma_k \in V$ , drawn from a finite control set  $V \subset U(\mathcal{H}_s)$ . Where  $U(\mathcal{H}_s)$  is the set of all unitaries on  $\mathcal{H}_s$ . A typical pulse sequence proceeds as follows:

1. At time  $t = 0$  we apply an initial pulse  $\gamma_0$
2. The system then evolves for a time  $\tau$  under the unitary  $\exp(-i\tau\hat{\mathcal{H}})$

---

<sup>1</sup>Hahn, E. L. (1950). Spin echoes. *Physical Review*, **80**(4), 580.

<sup>2</sup>Viola, L., Knill, E., & Lloyd, S. (1999). Dynamical decoupling of open quantum systems. *Phys. Rev. Lett.*, **82**(12), 2417.

<sup>3</sup>Viola, L., & Lloyd, S. (1998). Dynamical suppression of decoherence in two-state quantum systems. *Phys. Rev. A*, **58**(4), 2733.

3. Apply a different pulse  $\gamma_1$
4. Repeat steps 2 and 3, applying pulses and allowing free evolution, until time  $t = N\tau$ , where  $N$  is the total number of pulses applied and  $\gamma_N$  is the final pulse.

A commonly used control set is Pauli group  $\{X, Y, Z, I\}$ .

It is assumed that the duration of each pulse  $\tau_d$  is much smaller than the interval  $\tau$  between pulses  $\tau_d \ll \tau$ . In practice,  $\tau_d$  may scale with  $\tau^2$ , but is often idealized as zero in theoretical treatments.

## 2.2 Decoupling set

This setup leads to the central question

**Can we design a sequence of such pulses so that the system's natural evolution under  $e^{-it\hat{\mathcal{H}}}$  is effectively canceled or suppressed?**

To answer this, we introduce the concept of a **decoupling set**, a carefully chosen collection of unitary operators that, when used as control pulses, can symmetrize or average out arbitrary system-environment interactions over time.

A decoupling set is a finite subset of unitary operators  $V \subset U(\mathcal{H}_s)$ , such that for any operator  $X \in B(\mathcal{H}_s)$ , the group-averaged conjugation satisfies:

$$\frac{1}{|V|} \sum_{v \in V} v X v^\dagger = c \cdot \hat{\mathbf{1}}_{\mathcal{H}_s} \quad (2.1)$$

for some constant  $c$ . This means that the action of the entire decoupling set uniformly "scrambles" any operator into a multiple of the identity, effectively neutralizing its directional bias in Hilbert space.

We can calculate the value of the constant  $c$  from the fact that both sides of the equation must have the same trace

$$\text{Tr}(X) = \text{Tr}(c \cdot \hat{\mathcal{W}}_{\mathcal{H}_s}) = c \cdot d \quad \Rightarrow \quad c = \frac{\text{Tr}(X)}{d} \quad (2.2)$$

where  $d = \dim(\mathcal{H}_s)$ .

In practice, a typical decoupling set might be  $V = \{X, Y, Z\}$ , composed of Pauli operations. If the Hilbert space is infinite-dimensional, however, no finite decoupling set can fully average out all bounded operators, and more advanced techniques are needed.

## 2.3 Decoupling cycle

The decoupling cycle consists of a sequence of unitary operators  $(\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_N)$ , where each  $\mathcal{V}_i \in V$ , the decoupling set. The length of the cycle  $N$  is taken to be a multiple of  $|V|$ , ensuring that each element of  $V$  appears an equal number of times.

The average action of such a sequence on an operator  $X$  satisfies the identity:

$$\frac{1}{N} \sum_{k=1}^N \mathcal{V}_k X \mathcal{V}_k^\dagger = \frac{1}{|V|} \sum_{v \in V} v X v^\dagger \quad (2.3)$$

The control pulses  $\gamma_k$  in the decoupling cycle are constructed from elements of the decoupling set  $V$  as follows

Pulse	Operator	Timing
$\gamma_0$	$\mathcal{V}_1^\dagger$	0
$\gamma_1$	$\mathcal{V}_2^\dagger \mathcal{V}_1$	$\tau$
$\vdots$	$\vdots$	
$\gamma_{N-1}$	$\mathcal{V}_N^\dagger \mathcal{V}_{N-1}$	$(N-1)\tau$
$\gamma_N$	$\mathcal{V}_N$	$N\tau$

Table 2.1: Pulse operators and timings in the decoupling sequence

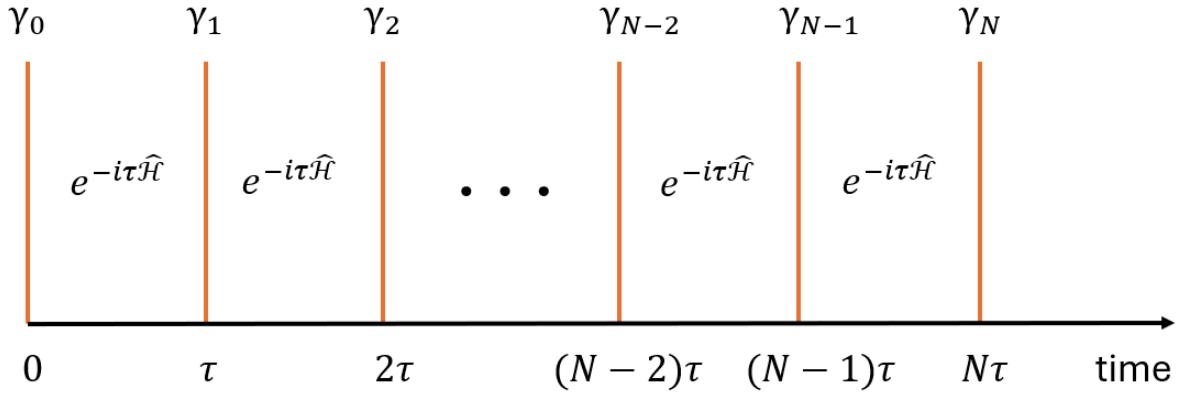


Figure 2.1: The application of control pulses  $\gamma_k$  and free evolution under the system Hamiltonian  $\hat{H}(t)$

The overall evolution of the state under this sequence of pulses is described by the operator

$$\hat{U}_{\text{cycle}}(N\tau) = \mathcal{V}_N e^{-i\tau\hat{H}} (\mathcal{V}_N^\dagger \mathcal{V}_{N-1}) e^{-i\tau\hat{H}} (\mathcal{V}_{N-1}^\dagger \mathcal{V}_{N-2}) \dots (\mathcal{V}_3^\dagger \mathcal{V}_2) e^{-i\tau\hat{H}} (\mathcal{V}_2^\dagger \mathcal{V}_1) e^{-i\tau\hat{H}} \mathcal{V}_1^\dagger \quad (2.4)$$

This rather bulky expression can be rewritten more cleanly using the identity  $U e^D U^\dagger = e^{UDU^\dagger}$ .

$$\hat{U}_{\text{cycle}}(N\tau) = e^{-i\tau\mathcal{V}_N \hat{H} \mathcal{V}_N^\dagger} e^{-i\tau\mathcal{V}_{N-1} \hat{H} \mathcal{V}_{N-1}^\dagger} \dots e^{-i\tau\mathcal{V}_2 \hat{H} \mathcal{V}_2^\dagger} e^{-i\tau\mathcal{V}_1 \hat{H} \mathcal{V}_1^\dagger} \quad (2.5)$$

The evolution interleaves these control operations and free dynamics. As shown in the figure below, this can be interpreted as a sequence of conjugated Hamiltonians Let us define  $\hat{\mathcal{H}}_k = \mathcal{V}_k \hat{\mathcal{H}} \mathcal{V}_k^\dagger$

$$\hat{U}_{\text{cycle}}(N\tau) = e^{-i\tau\hat{\mathcal{H}}_N} e^{-i\tau\hat{\mathcal{H}}_{N-1}} \dots e^{-i\tau\hat{\mathcal{H}}_2} e^{-i\tau\hat{\mathcal{H}}_1} = \prod_{k=N}^1 e^{-i\tau\hat{\mathcal{H}}_k} \quad (2.6)$$

Where the product is ordered from left to right, with earlier operations appearing to the right.

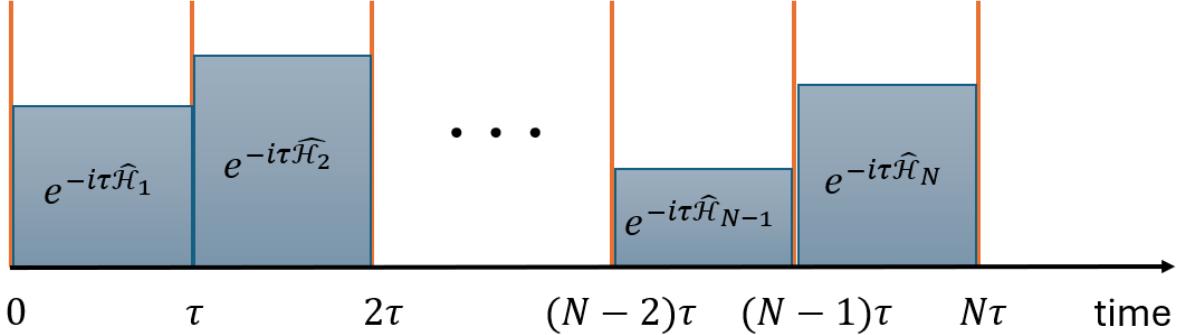


Figure 2.2: The interpretation of controlled evolution as a sequence of conjugated Hamiltonians  $\hat{\mathcal{H}}_k = \mathcal{V}_k \hat{\mathcal{H}} \mathcal{V}_k^\dagger$ , where the effect of each pulse is to rotate the system Hamiltonian before each free evolution interval.

This structure resembles the Trotter product formula, which tells us that for sufficiently small  $\tau$ , the product approximates evolution under the average Hamiltonian  $\bar{\mathcal{H}} = \frac{1}{N} \sum_{k=1}^N \mathcal{V}_k \hat{\mathcal{H}} \mathcal{V}_k^\dagger$

$$\hat{U}_{\text{cycle}}(N\tau) \approx \exp(-iN\tau\bar{\mathcal{H}}) \quad (2.7)$$

If the control set  $V$  forms a valid decoupling set and the pulse spacing  $\tau$  is small, this average  $\bar{\mathcal{H}}$  becomes proportional to the identity, effectively halting the system's evolution. In principle, knowing the exact form of  $\hat{\mathcal{H}}$  allows us to design such a sequence exactly, but crucially, dynamical decoupling works even when the Hamiltonian is unknown.

**Claim:**  $\hat{U}_{\text{cycle}}(N\tau) \approx \exp(-iN\tau\bar{\mathcal{H}})$

**Proof:** Expand each exponential using the first-order approximation of the matrix exponential for small  $\tau$ , we have  $e^{-i\tau\hat{\mathcal{H}}_k} = \hat{\mathbf{1}} - i\tau\hat{\mathcal{H}}_k + \mathcal{O}(\tau^2)$ . Taking the ordered product (as described earlier) of all  $N$  such exponentials, we get

$$\prod_{k=N}^1 (\hat{\mathbf{1}} - i\tau\hat{\mathcal{H}}_k + \mathcal{O}(\tau^2)) \quad (2.8)$$

To first order in  $\tau$ , we can write this product as

$$\hat{U}_{\text{cycle}}(N\tau) = \hat{\mathbf{1}} - i\tau \sum_{k=1}^N \hat{\mathcal{H}}_k + \mathcal{O}(N^2\tau^2) \quad (2.9)$$

where the  $\mathcal{O}(N^2\tau^2)$  term comes from collecting all second-order and higher-order cross terms in the product expansion.

Now from the definition of the average Hamiltonian  $\bar{\mathcal{H}} = \frac{1}{N} \sum_{k=1}^N \hat{\mathcal{H}}_k$  we have

$$\hat{U}_{\text{cycle}}(N\tau) = \hat{\mathbf{1}} - i\tau N \bar{\mathcal{H}} + \mathcal{O}(N^2\tau^2) \quad (2.10)$$

Finally, noting that this matches the first-order expansion of the exponential

$$\exp(-i\tau N \bar{\mathcal{H}}) = \hat{\mathbf{1}} - i\tau N \bar{\mathcal{H}} + \mathcal{O}(\tau^2 N^2) \quad (2.11)$$

we conclude that

$$\hat{U}_{\text{cycle}}(N\tau) = \exp(-i\tau N \bar{\mathcal{H}}) + \mathcal{O}(N^2\tau^2) \quad (2.12)$$

This shows that the sequence of pulses effectively simulates evolution under the average Hamiltonian, up to second-order corrections in  $\tau$ . ■

## 2.4 Repeated Cycles and Long-Time Limit

The core idea behind repeated decoupling cycles is straightforward, if the sequence is applied frequently enough, the system's decoherence is suppressed. To formalize this, imagine repeating the decoupling cycle  $n$  times. If the length of the experiment is  $T$ , then each cycle is of length  $T/n$ . From previous analysis, each cycle has  $N$  pulses with each subsequent pulse applied with a time difference of  $\tau$ . So,  $N\tau$  is the length of a given pulse but it is also  $T/n$ .

$$\therefore N\tau = \frac{T}{n} \implies \tau = \frac{T}{nN} \quad (2.13)$$

As  $n \rightarrow \infty$ , the time between pulses shrinks, and the higher-order error terms  $\mathcal{O}(N^2\tau^2) = \mathcal{O}\left(\frac{T^2}{n^2}\right)$  vanish. In this limit, the total evolution over the whole period,  $\hat{U}(T) = \hat{U}_{\text{cycle}}^n\left(\frac{T}{n}\right)$ , converges as shown

$$\hat{U}(T) = \left(\hat{U}_{\text{cycle}}\left(\frac{T}{n}\right)\right)^n = \left[\exp\left(-i\frac{T}{n} \bar{\mathcal{H}}\right) + \mathcal{O}\left(\frac{T^2}{n^2}\right)\right]^n \quad (2.14)$$

$$\implies \hat{U}(T) = \lim_{n \rightarrow \infty} \left[ \exp\left(-i\frac{T}{n} \bar{\mathcal{H}}\right) + \cancel{\mathcal{O}\left(\frac{T^2}{n^2}\right)}^0 \right]^n \quad (2.15)$$

$$\implies \hat{U}(T) = \lim_{n \rightarrow \infty} \left[ \exp\left(-i\frac{T}{n} \bar{\mathcal{H}}\right) \right]^n \quad (2.16)$$

$$\implies \hat{U}(T) = \exp(-iT\bar{\mathcal{H}}) \quad (2.17)$$

Now if the sequence is well designed and the decoupling set is valid, the resulting average Hamiltonian becomes proportional to identity,  $\bar{\mathcal{H}} \approx \frac{\text{Tr}(\hat{\mathcal{H}})}{d} \hat{\mathbf{1}}_d$ , that is, the system rotates trivially up to a global phase

$$\implies \hat{U}(T) \approx \exp\left(-iT \cdot \frac{\text{Tr}(\hat{\mathcal{H}})}{d}\right) \cdot \hat{\mathbf{1}}_d \quad (2.18)$$

## 2.5 Bounding the Approximation Error

We've shown that the repeated application of a decoupling cycle approximates evolution under an average Hamiltonian  $\bar{\mathcal{H}}$ . But naturally, the question arises,

### How good is this approximation?

To answer that, we can lean on a well-known result from operator theory, one that offers a quantitative upper bound on how far apart two unitary evolutions can drift over time.

Suppose  $U_1(t)$  and  $U_2(t)$  are two time-evolution operators generated by Hamiltonians  $H_1(t)$  and  $H_2(t)$ , respectively. Then the following bound holds<sup>4</sup>

$$\|\hat{U}_2(t) - \hat{U}_1(t)\| \leq \|S_{21}\| + \int_0^t dx \|S_{21}(x)\| \cdot (\|\hat{\mathcal{H}}_1(x)\| + \|\hat{\mathcal{H}}_2(x)\|) \quad (2.19)$$

Here the  $S_{21}(t)$  is the action integral defined as

$$S_{21}(t) = \int_0^t ds [\hat{\mathcal{H}}_2(s) - \hat{\mathcal{H}}_1(s)] \quad (2.20)$$

Now applying this bound, to our specific case with

- $\hat{U}_2(T)$  is true time evolution via the decoupling sequence

$$\hat{U}_2(T) = \hat{U}_{\text{cycle}} = \prod_{k=N}^1 \mathcal{V}_k e^{-it\hat{\mathcal{H}}} \mathcal{V}_k^\dagger$$

- $\hat{U}_1(T)$  is the ideal evolution under the average Hamiltonian  $\bar{\mathcal{H}}$ ,

$$\hat{U}_1(T) = e^{-it\bar{\mathcal{H}}}$$

Recall that the transformed Hamiltonians  $\hat{\mathcal{H}}_k = \mathcal{V}_k \hat{\mathcal{H}} \mathcal{V}_k^\dagger$  has the same norm as the original,  $\|\hat{\mathcal{H}}_k\| = \|\hat{\mathcal{H}}\|$ .

Now the action integral can be approximately calculated as follows,

$$S_{21} = \int_0^t ds (\hat{\mathcal{H}}_2(s) - \bar{\mathcal{H}}) \approx \sum_k \hat{\mathcal{H}}_k \frac{T}{nN} - T\bar{\mathcal{H}} = \mathcal{O}\left(\frac{T}{n} \|\hat{\mathcal{H}}\|\right) \quad (2.21)$$

---

<sup>4</sup>A. Buryak, P. Facchi, M. Gramegna, K. Yuasa, “One bound to rule them all: A universal quantum speed limit,” *Quantum*, **6**, 774 (2022). <https://arxiv.org/abs/2111.08961>

Now using this we have,

$$\|\hat{U}_{\text{actual}} - \hat{U}_{\text{ideal}}\| \leq \frac{2T}{n} \cdot \|\hat{\mathcal{H}}\| \cdot (1 + 2T\|\hat{\mathcal{H}}\|) = \mathcal{O}\left(\frac{1}{n}\right) \quad (2.22)$$

Therefore repeating the cycle enough times makes the approximation arbitrarily good.

## 2.6 System-Environment Interaction

Up to this point, the analysis has focused solely on the qubit system, ignoring its coupling to the environment. In reality, nothing exists in isolation, especially fragile decoherence-prone quantum systems of NISQ (Noisy Intermediate-Scale Quantum) era. Let's consider a more realistic scenario where the system is coupled to the environment. We define the full Hilbert space as  $\mathcal{H} = \mathcal{H}_s \otimes \mathcal{H}_e$  with the total Hamiltonian,  $\hat{\mathcal{H}}$ , governing the joint dynamics.

**Note:** Generally this total Hamiltonian contains system-only terms, environment-only terms and interaction terms  $\hat{\mathcal{H}} = \hat{\mathcal{H}}_s \otimes \hat{\mathbf{1}}_e + \hat{\mathbf{1}}_s \otimes \hat{\mathcal{H}}_e + \hat{\mathcal{H}}_{int}$  and is Hermitian.

We assume that the decoupling pulses act only on the system as described earlier and the environment is left untouched. So, each control operation is of the form  $\mathcal{V}_k \otimes \hat{\mathbf{1}}_e$ .

The evolution after  $n$  repeated decoupling cycles, each of duration  $t/n$ , becomes

$$\hat{U}_{\text{cycle}}^n\left(\frac{t}{n}\right) = \left( \prod_{k=N}^1 (\mathcal{V}_k \otimes \hat{\mathbf{1}}) \exp\left(\frac{-it}{nN} \hat{\mathcal{H}}\right) (\mathcal{V}_k^\dagger \otimes \hat{\mathbf{1}}) \right)^n \quad (2.23)$$

$$= \left( \prod_{k=N}^1 \exp\left(-\frac{it}{nN} \hat{\mathcal{H}}_k\right) \right) \quad (2.24)$$

where each  $\hat{\mathcal{H}}_k = (\mathcal{V}_k \otimes \hat{\mathbf{1}}_e) \hat{\mathcal{H}} (\mathcal{V}_k^\dagger \otimes \hat{\mathbf{1}}_e)$  is the conjugated Hamiltonian for that pulse step.

### 2.6.1 Sufficiency Condition for Effective Dynamical Decoupling

$$\text{s-} \lim_{n \rightarrow \infty} \hat{U}_{\text{cycle}}^n\left(\frac{t}{n}\right) = \hat{\mathbf{1}}_s \otimes \exp(-it\hat{B}) \quad (2.25)$$

$$\text{s-} \lim_{n \rightarrow \infty} \hat{U}_{\text{cycle}}^n\left(\frac{t}{n}\right) = \exp(-it \hat{\mathbf{1}}_s \otimes \hat{B}) \quad (2.26)$$

(Here, s-lim denotes the strong operator limit. Both expressions are equivalent in our context.) Here,  $\hat{B}$  is some Hermitian operator acting only on the environment. This result tells us that the system part of the evolution becomes identity.

**Sufficiency condition for existence of the limit (2.26):**

A sufficient condition for the limit above to exist is based on the self-adjoint-ness of the averaged Hamiltonian, similar to the requirement in the Trotter product formula. If we define the average Hamiltonian as

$$\bar{\mathcal{H}} = \frac{1}{|V|} \sum_{v \in V} (\mathcal{V} \otimes \hat{\mathbf{1}}_e) \hat{\mathcal{H}} (\mathcal{V}^\dagger \otimes \hat{\mathbf{1}}_e) \quad (2.27)$$

then we require it to be self-adjoint on the domain  $\bigcap_{v \in V} (\mathcal{V} \otimes \hat{\mathbf{1}}) D(\hat{\mathcal{H}})$  where,  $D(\hat{\mathcal{H}})$  is the domain of the original Hamiltonian  $\hat{\mathcal{H}}$ . This ensures that the average dynamics are well-behaved and the decoupling cycles behave as expected in the infinite limit.

## 2.7 Effect of Finite-Duration Pulses

Much of the theoretical analysis above rests on the common idealization that control pulses are instantaneous, modeled as delta functions that apply perfect unitaries in zero time. This assumption makes the math clean and the convergence to an effective average Hamiltonian straightforward. However, in practical settings, pulses have non-zero duration, and this has direct consequences for dynamical decoupling performance.

In this project, we have explicitly simulated using non-ideal, finite-duration pulses, and the results show a clear trend

**As the pulse duration increases the coherence decreases.**

This degradation occurs because

- The noise acts concurrently with the pulse.
- This causes the pulse Hamiltonian and system-environment Hamiltonian act simultaneously and cause unwanted cross-terms.
- Longer pulses reduce the time available for free evolution between pulses, deviating from the assumptions of the theoretical model.

As a result,

- The assumptions underlying the bounds we found in earlier sections no longer hold strictly.
- The strong limit of repeated cycles is expected to converge to an identity on the system, but simulations show that the fidelity does not remain near 1 even at very high pulse counts, indicating a breakdown of this convergence in the presence of finite-duration pulses.

- Even with high pulse counts, increasing pulse duration leads to a noticeable drop in fidelity.

This highlights an important real-world limitation of DD.

## 2.8 Note on Non-Uniform Pulse Spacing

The formalism and convergence results discussed so far assume that all  $\pi$  pulses are applied at **uniform intervals**, i.e., with a constant inter-pulse spacing  $\tau = \frac{T}{nN}$  across  $N$  pulses per cycle repeated  $n$  times. This assumption simplifies the analysis and is sufficient to demonstrate the core idea of dynamical decoupling through time-symmetric averaging.

However, in practice, and in more advanced theoretical constructions, it is often beneficial to relax this constraint and allow non-uniform pulse spacings. This means choosing a set of pulse times  $\{t_1, t_2, \dots, t_N\}$  within  $[0, T]$  that are not equally spaced.

Several well-known sequences, such as **Uhrig Dynamical Decoupling (UDD)**, deliberately use non-uniform timing to optimize the cancellation of specific frequency components of the noise. These designs are particularly effective against structured or high-frequency noise, and their construction is often guided by filter function analysis which is discussed in the next chapter.

In the general case, the modulation function  $y(t)$  is no longer a periodic square wave but a piecewise-constant function with irregular flip points, which results in a qualitatively different filter function  $F(\omega, T)$ . Despite the increased complexity, the filter function formalism remains valid. Thus, non-uniform pulse timing opens up a broader design space for tailoring dynamical decoupling to specific noise environments, and many optimal DD strategies rely on exploiting this freedom.

## 2.9 Summary of the Formalism

In this chapter, we developed the theoretical framework underlying dynamical decoupling (DD). Starting from the basic assumption of bounded Hamiltonian dynamics, we showed how sequences of unitary pulses can effectively average out unwanted interactions both within the system and between the system and its environment.

Using the language of conjugated Hamiltonians, decoupling sets, and Trotter expansions, we demonstrated that

- The system's evolution under a carefully designed DD cycle converges to an identity (up to a global phase), effectively halting decoherence.
- This behavior holds in the ideal limit of infinitely fast, zero-duration pulses and a bounded system-environment Hamiltonian.

- We derived bounds on the approximation error, showing  $\mathcal{O}(1/n)$  scaling in the number of repeated cycles under ideal assumptions.

We then extended the analysis to composite systems (system + environment), showing that if the decoupling pulses act only on the system, the system's evolution can still be frozen in the strong limit, provided the averaged Hamiltonian is self-adjoint.

However, a key caveat emerges from both theory and simulation

*All preceding analysis assumes idealized, delta-function pulses, an assumption that breaks down in practical implementations.*

In practical scenarios, pulses have finite duration, and the clean separation between control and noise breaks down. The bounds and convergence results are no longer guaranteed. Through numerical simulations in this project, we observe that longer pulses degrade performance and delay convergence, highlighting the importance of taking physical control limitations into account.

This motivates the next chapter, where we move from general theory to some well-known decoupling sequences: CPMG, UDD, CDD, and others. We will examine their pulse schedules, filter functions, and performance under various noise models, building a more complete picture of how theoretical DD protocols behave when implemented in simulations of noisy quantum systems.

*Note:* The theoretical structure presented here follows closely from [Quantum control and dynamical decoupling, Paolo Facchi - 25/05/23](#), with original elaborations and simulation-based interpretation

# Chapter 3

## Filter Function Formalism

### 3.1 Motivation for Frequency Domain Analysis

In Chapter 2, we analyzed dynamical decoupling (DD) from the perspective of averaging out unwanted unitary evolution using sequences of control pulses. However, a complementary and powerful approach is to interpret decoherence as the result of environmental noise with a particular spectral distribution, and to understand how different DD sequences selectively suppress this noise—hence the name **filter function formalism**.

In this chapter, we will build a model Hamiltonian, explain the effect of DD pulses, analyze how the noise is modulated by the applied control, and then define the filter function corresponding to a given DD sequence.

From this analysis, we will see that by applying a DD sequence with a known filter function, while varying parameters such as the number of pulses, the total experiment duration, and measuring the resulting coherence decay of the qubit, we can infer the nature of the noise affecting the system. This provides a powerful technique for determining the power spectral density (PSD) of an unknown noise environment, making the filter function formalism an essential diagnostic tool for characterizing quantum systems.

### 3.2 The Model Hamiltonian

To understand the effect of dynamical decoupling (DD) from a frequency-domain perspective, we begin with a simplified noise model that captures the essential physics of decoherence. We consider the qubit to be subjected to a stochastic classical noise field along the  $z$ -axis. The corresponding Hamiltonian is

$$\hat{\mathcal{H}}(t) = \beta(t) \hat{\sigma}_z \quad (3.1)$$

Here,  $\beta(t)$  is a real-valued, time-dependent stochastic process representing fluctuations in the qubit energy splitting due to environmental interactions. This model captures

pure dephasing noise, where the qubit's phase coherence decays over time without energy exchange with the environment. The choice of  $\hat{\sigma}_z$  is justified because in many qubit platforms, such as superconducting qubits and spin qubits, the dominant noise source at low frequencies is longitudinal (i.e., coupling to  $\hat{\sigma}_z$ ).

This Hamiltonian serves as a useful starting point for analyzing decoherence since it isolates the effects of dephasing from those of relaxation, which would require modeling coupling to  $\hat{\sigma}_x$  or  $\hat{\sigma}_y$ . Additional noise terms along  $x$  and  $y$  can be added later to build more comprehensive models, but the  $z$ -noise model already reveals many important features of how DD sequences suppress decoherence and how they can be tailored to the spectral characteristics of the noise.

### 3.3 Time Evolution Under Dephasing Noise

The time evolution operator under the Hamiltonian, eq-34, can be calculated from the Schrodinger's equation, if the initial state of the qubit is some pure state  $|\psi(0)\rangle$

$$i\hbar \frac{\partial |\psi(t)\rangle}{\partial t} = \hat{\mathcal{H}}(t) |\psi(t)\rangle = \beta(t) \hat{\sigma}_z |\psi(t)\rangle \quad (3.2)$$

We work in units where  $\hbar = 1$ , and since the Hamiltonian is time-dependent but always proportional to  $\hat{\sigma}_z$ , it commutes with itself at different times

$$[\hat{\mathcal{H}}(t), \hat{\mathcal{H}}(t')] = 0 \quad (3.3)$$

This allows us to drop time-ordering and write the formal solution to the Schrödinger equation as<sup>1</sup>

$$|\psi(t)\rangle = \hat{U}(t) |\psi(0)\rangle, \quad \text{where} \quad \hat{U}(t) = \exp\left(-i \int_0^t \beta(s) ds \cdot \hat{\sigma}_z\right) \quad (3.4)$$

We define the accumulated stochastic phase as

$$\phi(t) = \int_0^t \beta(\tau) d\tau \quad (3.5)$$

so that the time evolution operator becomes  $\hat{U}(t) = \exp(-i\phi(t) \hat{\sigma}_z)$ . This operator describes a rotation about the  $z$ -axis of the Bloch sphere by a random angle  $\phi(t)$ , where  $\phi(t)$  depends on the specific realization of the stochastic noise  $\beta(t)$ .

Suppose, the qubit is initially prepared in a general superposition state  $|\psi(0)\rangle = a|0\rangle + b|1\rangle$ . Under the action of the unitary  $\hat{U}(t)$ , the state evolves to  $|\psi(t)\rangle = ae^{-i\phi(t)}|0\rangle + be^{i\phi(t)}|1\rangle$ . Up to a global phase, this state can be rewritten as  $|\psi(t)\rangle \sim a|0\rangle + be^{2i\phi(t)}|1\rangle$ . The corresponding density matrix at time  $t$  is

$$\rho(t) = |\psi(t)\rangle \langle \psi(t)| = \begin{bmatrix} |a|^2 & ab^* e^{-2i\phi(t)} \\ a^* b e^{2i\phi(t)} & |b|^2 \end{bmatrix} \quad (3.6)$$

---

<sup>1</sup>A derivation of this form appears in Bylander et al., *Nature Physics*, 7, 565–570 (2011).

**Note:** If the  $\phi(t) = 0$ , then there is no evolution of state. We aim to make  $\phi(t)$  as small as possible.

Now, because  $\phi(t)$  is a random variable (different for each realization of the noise), the off-diagonal terms acquire a fluctuating phase factor. Taking the ensemble average over many realizations of  $\beta(t)$  gives the reduced state:

$$\langle \rho(t) \rangle = \begin{bmatrix} |a|^2 & ab^* \langle e^{-2i\phi(t)} \rangle \\ a^*b \langle e^{2i\phi(t)} \rangle & |b|^2 \end{bmatrix} \quad (3.7)$$

Since  $\langle e^{\pm 2i\phi(t)} \rangle$  decays with time (due to destructive interference among random phase realizations), the off-diagonal coherence terms vanish, leading to dephasing.

This analysis shows that the noise  $\beta(t)$  induces a relative phase between the components  $|0\rangle$  and  $|1\rangle$ , and averaging over noise realizations leads to **decay of coherence**. In this model, decoherence manifests entirely as **random phase accumulation**, and **suppressing this accumulated phase** is precisely the goal of dynamical decoupling.

## 3.4 Control Pulses and the Modulation Function

To counteract the phase accumulation induced by the noise  $\beta(t)$ , we apply a sequence of control pulses to the qubit. These pulses are typically chosen to be  $\pi$  rotations about say, the  $x$ -axis.

More importantly, they flip the sign of the Hamiltonian, thereby reversing the direction of phase accumulation. This can be seen mathematically as follows

$$\hat{\sigma}_x \hat{H}(t) \hat{\sigma}_x^\dagger = \hat{\sigma}_x (\beta(t) \hat{\sigma}_z) \hat{\sigma}_x = \beta(t) \hat{\sigma}_x \hat{\sigma}_z \hat{\sigma}_x = -\beta(t) \hat{\sigma}_z \quad (3.8)$$

Therefore, after the pulse, the effective Hamiltonian becomes  $-\beta(t) \hat{\sigma}_z$ . If we consider the total evolution split into two segments, before and after the pulse applied at some time  $t_1$  and the total time of the experiment is  $T$ , then before the pulse the accumulated phase is  $\phi_1 = \int_0^{t_1} \beta(\tau) d\tau$ , and after the pulse, the Hamiltonian has flipped sign, so the phase accumulated is  $\phi_2 = \int_{t_1}^T (-\beta(\tau)) d\tau = - \int_{t_1}^T \beta(\tau) d\tau$

Thus, the total phase accumulated becomes

$$\phi(t) = \phi_1 + \phi_2 = \int_0^{t_1} \beta(\tau) d\tau - \int_{t_1}^T \beta(\tau) d\tau \quad (3.9)$$

Therefore, every time we apply a  $R_x(\pi)$  pulse the direction of phase accumulation is reversed. Extending this idea to multiple  $\pi$  pulses, and carefully placing such  $R_x(\pi)$

pulses, at times  $\{t_1, t_2, \dots, t_n\}$ , the phase alternates in sign after every pulse and the accumulated phase becomes

$$\phi(t) = \int_0^{t_1} \beta(\tau) d\tau - \int_{t_1}^{t_2} \beta(\tau) d\tau + \cdots + (-1)^n \int_{t_n}^T \beta(\tau) d\tau \quad (3.10)$$

Therefore, can cause different contributions to the phase to **interfere destructively**, and thus suppress the overall decoherence. This is the basic mechanism behind dynamical decoupling.

To capture this idea precisely, we define the **modulation function**  $y(t)$ , which takes values  $\pm 1$  and encodes the sign of the effective Hamiltonian at any given time. The Hamiltonian under control becomes

$$\hat{\mathcal{H}}_{\text{eff}}(t) = y(t) \beta(t) \hat{\sigma}_z \quad (3.11)$$

so the total accumulated phase becomes

$$\phi(t) = \int_0^t y(\tau) \beta(\tau) d\tau \quad (3.12)$$

This means the control pulses are now entirely encoded in the binary-valued modulation function  $y(t)$ , which flips sign at the pulse locations.

### For example:

- For a **Free Induction Decay (FID)** experiment with no pulses,  $y_{\text{FID}}(t) = 1$  for all  $t$ .
- For a **Hahn Echo** with a single  $\pi$  pulse at  $t = T/2$ , we have

$$y_{\text{echo}}(t) = \begin{cases} +1 & \text{for } 0 \leq t < T/2 \\ -1 & \text{for } T/2 \leq t \leq T \end{cases} \quad (3.13)$$

- In a **CPMG** sequence with  $n$  evenly spaced  $\pi$  pulses at times  $t_k = (k-1/2) \cdot T/n$ , the modulation function switches sign at each pulse. The shape of  $y_{\text{CPMG}}(t)$  becomes a square wave with  $n$  sign flips.

Thus, the modulation function serves as a compact representation of the pulse sequence, and plays a central role in determining how effectively the sequence suppresses noise. In the next section, we will use the modulation function,  $y(t)$ , to define the **filter function** of a given DD sequence, which quantifies how much each frequency component of the noise contributes to decoherence.

## 3.5 Decoherence Functional and the Filter Function

As discussed earlier, the total accumulated stochastic phase due to noise, in the presence of a DD sequence encoded by modulation function  $y(t)$ , is given by

$$\phi(T) = \int_0^T y(t) \beta(t) dt \quad (3.14)$$

This random phase induces dephasing. The coherence of the qubit is quantified by the expectation value of the off-diagonal element of the density matrix, which for a state initially in  $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$  is

$$\langle \sigma_+(T) \rangle = \langle e^{-2i\phi(T)} \rangle \quad (3.15)$$

Assuming that the noise  $\beta(t)$  is a zero-mean **stationary Gaussian process**, the average can be computed exactly using

$$\langle e^{-2i\phi(T)} \rangle = e^{-2\langle \phi^2(T) \rangle} \quad (3.16)$$

This defines the **decoherence functional**

$$\chi(T) \triangleq 2\langle \phi^2(T) \rangle = 2 \left\langle \left( \int_0^T y(t) \beta(t) dt \right)^2 \right\rangle \quad (3.17)$$

Using the autocorrelation function of  $\beta(t)$ ,

$$\langle \beta(t)\beta(t') \rangle = C(t-t') \quad (3.18)$$

we expand the square

$$\chi(T) = 2 \int_0^T \int_0^T y(t)y(t') C(t-t') dt dt' \quad (3.19)$$

Now, using the **Wiener–Khinchin theorem**, the autocorrelation function is related to the **power spectral density (PSD)**  $S(\omega)$  via Fourier transform

$$C(t-t') = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) e^{-i\omega(t-t')} d\omega \quad (3.20)$$

Since  $\beta(t)$  is a real-valued stochastic process, its PSD  $S(\omega)$  is real and even, that is,  $S(-\omega) = S(\omega)$ .

Substituting into the double integral

$$\chi(T) = 2 \int_0^T \int_0^T y(t)y(t') \left[ \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) e^{-i\omega(t-t')} d\omega \right] dt dt' \quad (3.21)$$

interchanging the order of integration

$$\chi(T) = \frac{1}{\pi} \int_{-\infty}^{\infty} S(\omega) \left[ \int_0^T \int_0^T y(t)y(t') e^{-i\omega(t-t')} dt dt' \right] d\omega \quad (3.22)$$

Now the integrals inside the bracket can be simplified as shown

$$\text{Claim: } \int_0^T \int_0^T y(t)y(t') e^{-i\omega(t-t')} dt dt' = \left| \int_0^T y(t) e^{-i\omega t} dt \right|^2$$

**Proof:** let  $f(\omega) = \int_0^T y(t) e^{-i\omega t} dt$ , then the conjugate is  $f^*(\omega) = \int_0^T y(t') e^{i\omega t'} dt'$

Now their product is  $|f(\omega)|^2 = \left( \int_0^T y(t) e^{-i\omega t} dt \right) \left( \int_0^T y(t') e^{i\omega t'} dt' \right)$ . This can be easily simplified to  $\int_0^T \int_0^T y(t)y(t')e^{-i\omega(t-t')} dt dt'$

■

$$\therefore \chi(T) = \frac{1}{\pi} \int_{-\infty}^{\infty} S(\omega) \left| \int_0^T y(t) e^{i\omega t} dt \right|^2 d\omega \quad (3.23)$$

Since  $S(\omega)$  is even and the modulus term is also even, we have

$$\chi(T) = \frac{2}{\pi} \int_0^{\infty} S(\omega) \left| \int_0^T y(t) e^{i\omega t} dt \right|^2 d\omega \quad (3.24)$$

The term inside the modulus is the Fourier transform of  $y(t)$ . Based on this, we define the **filter function** as

$$F(\omega, T) \triangleq \left| \int_0^T y(t) e^{i\omega t} dt \right|^2 \quad (3.25)$$

Thus, the **final expression for decoherence** is

$$\boxed{\chi(T) = \frac{2}{\pi} \int_0^{\infty} S(\omega) F(\omega, T) d\omega} \quad (3.26)$$

**Note:**

- Some literature defines the filter function with an additional factor of  $\omega^2$  absorbed into it  $F_{alt}(\omega, T) \triangleq \omega^2 \left| \int_0^T y(t) e^{i\omega t} dt \right|^2$ , which leads to  $\chi(T) = \frac{1}{\pi} \int_0^{\infty} \frac{S(\omega)}{\omega^2} F_{alt}(\omega, T) d\omega$ .
- In this work, I use the unscaled form for consistency with my code and theoretical derivation.

This is the core result of the filter function formalism. It tells us that the **coherence decay** of the qubit depends on

- the **noise spectrum**  $S(\omega)$  of the environment, and
- the **filter function**  $F(\omega, T)$  of the pulse sequence.

If  $F(\omega, T)$  is small where  $S(\omega)$  is large, decoherence is suppressed. By analyzing  $F(\omega, T)$  for different DD sequences, we can both design sequences to filter out specific noise frequencies.

## 3.6 Extension of Filter Function Formalism to Multiaxial Noise

In realistic environments, the noise acting on a qubit is not restricted to a single axis. Instead, the system-bath coupling often takes the general form

$$\hat{H}_{\text{int}}(t) = \sum_{\mu=x,y,z} \beta_\mu(t) \hat{\sigma}_\mu, \quad (3.27)$$

where  $\beta_\mu(t)$  are classical stochastic processes representing noise along each axis.

**Uncorrelated Noise Case:** If the noise components  $\beta_x(t)$ ,  $\beta_y(t)$ , and  $\beta_z(t)$  are statistically independent and zero-mean, then the total decoherence function  $\chi(t)$  can be written as a sum:

$$\chi(t) = \sum_{\mu=x,y,z} \int_0^\infty \frac{d\omega}{2\pi} S_\mu(\omega) F_\mu(\omega t),$$

where  $S_\mu(\omega)$  is the spectral density of noise along axis  $\mu$ , and  $F_\mu(\omega t)$  is the corresponding filter function.

This decomposition is very convenient, as it allows us to analyze each noise axis and each control modulation independently. In our simulations, we use this form to compute coherence decay under multiaxial DD sequences when the noise is modeled as uncorrelated across axes.

**Correlated Noise<sup>2</sup> Case:** If the noise across different axes is correlated (e.g.,  $\langle \beta_x(t) \beta_y(0) \rangle \neq 0$ ), then cross terms appear in  $\chi(t)$ . In this case, the decoherence functional becomes... cross-spectral density  $S_{\mu\nu}(\omega)$ . :

$$\chi(t) = \sum_{\mu=x,y,z} \sum_{\nu=x,y,z} \int_0^\infty \frac{d\omega}{2\pi} S_{\mu\nu}(\omega) F_{\mu\nu}(\omega t), \quad (3.28)$$

where  $S_{\mu\nu}(\omega)$  is the cross-spectral density between axes  $\mu$  and  $\nu$ , and  $F_{\mu\nu}(\omega t)$  is a generalized filter function involving correlations between modulation functions along different axes:

$$F_{\mu\nu}(\omega t) = \left| \int_0^t e^{i\omega t'} y_\mu(t') dt' \cdot \int_0^t e^{-i\omega t''} y_\nu(t'') dt'' \right|. \quad (3.29)$$

These cross terms make analytical evaluation of coherence decay significantly more difficult. In such cases, numerical simulation is often the only tractable approach. In this project, we restrict ourselves to the uncorrelated case.

If the noise components  $\beta_x(t)$ ,  $\beta_y(t)$ , and  $\beta_z(t)$  are uncorrelated and have independent spectral densities  $S_x(\omega)$ ,  $S_y(\omega)$ , and  $S_z(\omega)$ , then the total decoherence function  $\chi(t)$  can be written as a sum over the axes:

$$\chi(t) = \sum_{\mu=x,y,z} \int_0^\infty \frac{d\omega}{2\pi} S_\mu(\omega) F_\mu(\omega t) \quad (3.30)$$

---

<sup>2</sup>See General Transfer-Function Approach to Noise Filtering in Open-Loop Quantum Control by Paz-Silva, Gerardo A. and Viola, Lorenza <https://doi.org/10.1103/PhysRevLett.113.250501>

Here,  $F_\mu(\omega t)$  is the filter function corresponding to the modulation  $y_\mu(t)$  applied along axis  $\mu$ . This allows for a straightforward incorporation of multiaxial control and multiaxial noise within the same formalism<sup>3</sup>.

We used this formalism to compute theoretical predictions for coherence under multiaxial DD, and validated them against simulations under various composite noise models such as white +  $1/f$  noise in all directions.

## 3.7 Filter Function Formalism for Noise Spectroscopy

The filter–function formalism not only predicts the decoherence functional

$$\chi(T) = \frac{1}{\pi} \int_0^\infty S(\omega) F(\omega, T) d\omega \quad (3.26)$$

for a *known* noise spectrum  $S(\omega)$ , but also permits the inverse problem: estimating an *unknown*  $S(\omega)$  from measured coherence data. This approach ,often called **noise spectroscopy**, proceeds as follows

The key idea is to use different pulse configurations to sample distinct regions of the frequency domain. This is achieved through two complementary levers:

- **Varying the number of pulses  $n$**  in a fixed sequence (e.g., CPMG), which sharpens and shifts the passband of the filter function.
- **Varying the total evolution time  $T$** , which compresses or stretches the filter function in frequency space, thereby offering finer *or* broader resolution of spectral features.

### Inversion Strategy

The noise spectroscopy workflow proceeds as follows:

1. Choose a dynamical decoupling sequence (e.g., CPMG).
2. For multiple choices of pulse count  $n$ , and multiple values of total duration  $T$ , compute the decoherence function  $\chi_{j,k}$  for each combination.
3. For each pair  $(n_j, T_k)$ , calculate the corresponding filter function  $F_{j,k}(\omega)$ .
4. Construct a set of integral equations:

$$\chi_{j,k} = \frac{1}{\pi} \int_0^\infty S(\omega) F_{j,k}(\omega) d\omega.$$

---

<sup>3</sup>See, e.g., W. Cywiński *et al.*, Phys. Rev. B **77**, 174509 (2008); W. Yang and R.-B. Liu, Phys. Rev. B **79**, 144302 (2009).

5. Solve this linear inverse problem to estimate  $S(\omega)$ .<sup>4</sup>

Since both pulse number and total evolution time are varied, the filter function becomes a two-dimensional kernel  $F_{j,k}(\omega)$ , indexed over pulse count  $j$  and time  $T_k$ . The collection  $\{F_{j,k}\}$  is often called the “transfer-function matrix”. Collectively, these filter functions form a structured basis that samples the noise spectrum with richer resolution, turning the inversion problem into a tensor contraction against this kernel.

This two-dimensional variation, along both pulse number and experiment time, greatly enriches the sampling of the frequency axis, providing a more robust and well-conditioned dataset for inversion.

## Parametric Reconstruction

As direct inversion is often ill-posed<sup>5</sup>, one typically resorts to regularized fitting using flexible parametric forms. A representative ansatz might be:

$$S(\omega) = \frac{a}{\omega^2 + \varepsilon} + \frac{b}{(\omega + \varepsilon)^\alpha} + c + d\omega + e\omega^2 + f\omega^3 + A \exp\left(-\frac{(\omega - \mu)^2}{2\sigma^2}\right),$$

where  $\varepsilon \ll 1$  avoids division by zero.

This model can simultaneously capture:

- Low-frequency divergences,
- Smooth polynomial background,
- Narrowband spectral features.

**Important Caveat.** All simulations in this project were carried out with *explicitly known* spectral densities supplied as input. Performing a full reconstruction of the very same spectrum from simulated coherence data would therefore provide no additional insight, and was *not* attempted here. The formulation above is included to emphasise that filter–function inversion is feasible in principle and is especially valuable in experimental settings where  $S(\omega)$  is genuinely unknown and must be inferred from qubit-coherence measurements.

---

<sup>4</sup>See, for example, G. A. Paz-Silva and L. Viola, *Phys. Rev. Lett.* **113**, 250501 (2014); L. M. Norris *et al.*, *Phys. Rev. Lett.* **116**, 150503 (2016).

<sup>5</sup>Regularisation (Tikhonov, ridge, Bayesian, etc.) is standard when inverting Eq. (3.26); see the discussion in W. M. Witzel *et al.*, *Phys. Rev. Lett.* **105**, 187602 (2010).

# Chapter 4

## Common DD sequences and their Filter Functions

In the previous chapter, we introduced the filter function formalism as a tool for understanding how dynamical decoupling (DD) sequences suppress decoherence by selectively attenuating noise in specific frequency bands. In this chapter, we explore some of the most widely used DD sequences, analyze their modulation functions, and compute their corresponding filter functions.

These sequences differ in pulse placement, symmetry, and design principles, which leads to different spectral selectivity and robustness. We will discuss

- Pulse schedules and modulation functions for each DD sequence,
- Filter function plots for increasing pulse numbers,
- Brief remarks on each sequence's strengths, limitations, and typical use cases.

The filter functions shown here are computed numerically using the definitions developed in Chapter 3. We will use pulse counts  $N \in \{2, 4, 8, 16, 64\}$  to highlight how spectral response sharpens with increasing control granularity.

*Note:* Unless otherwise specified, all simulations in this work begin with the qubit initialized in the superposition state  $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ . This choice ensures maximal sensitivity to dephasing noise and allows direct interpretation of coherence decay.

### 4.1 Free Induction Decay (FID)

The Free Induction Decay (FID) experiment serves as the baseline case for analyzing decoherence in the absence of any control pulses. After preparing the qubit in a superposition state, and let it evolve freely under the influence of environmental noise. Since there are no control pulses, the total Hamiltonian is just the noise Hamiltonian described

in Eq. (1.1).

$$H_{\text{dephasing}} = \beta(t)\hat{\sigma}_z$$

In the filter function formalism, the modulation function  $y(t)$  is simply constant

$$y_{\text{FID}}(t) = 1 \quad \text{for all } t \in [0, T] \quad (4.1)$$

This means that the filter function for FID has maximal overlap with low-frequency noise, making FID highly susceptible to dephasing. As a result, it provides a useful reference for quantifying the suppression achieved by more sophisticated DD sequences.

#### 4.1.1 FID Filter Function

The FID filter function can be calculated using the formula Eq. (3.25)

$$\begin{aligned} F^{\text{FID}}(\omega T) &= \left| \int_0^T y_{\text{FID}}(t) e^{i\omega t} dt \right|^2 = \left| \int_0^T 1 \cdot e^{i\omega t} dt \right|^2 \\ &= \left| \frac{e^{i\omega t}|_0^T}{i\omega} \right|^2 = \left| \frac{e^{i\omega T} - 1}{i\omega} \right|^2 \\ &= \frac{e^{i\omega T} - 1}{i\omega} \cdot \frac{e^{-i\omega T} - 1}{-i\omega} \\ &= \frac{1 - e^{-i\omega T} - e^{i\omega T} + 1}{\omega^2} = \frac{2 - 2 \cos(\omega T)}{\omega^2} \\ \implies F^{\text{FID}}(\omega T) &= \frac{\sin^2\left(\frac{\omega T}{2}\right)}{\left(\frac{\omega}{2}\right)^2} \end{aligned} \quad (4.2)$$

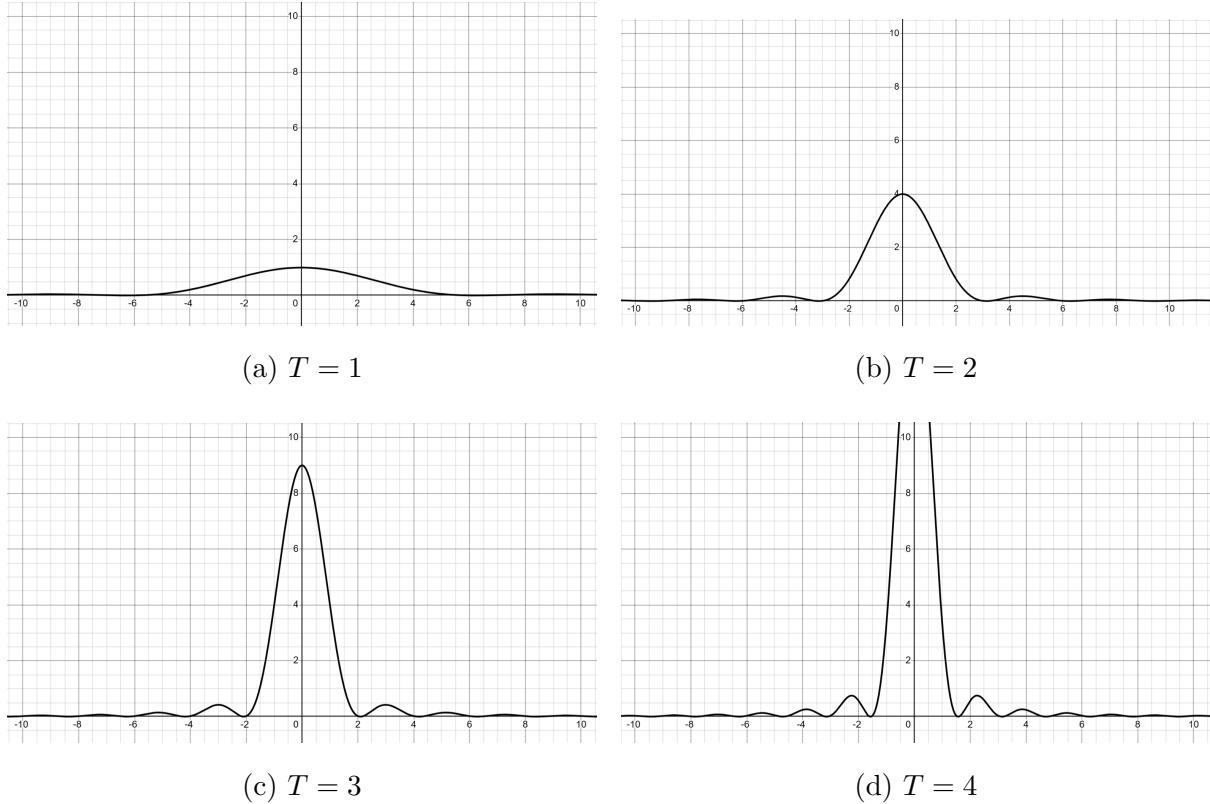


Figure 4.1: Filter function plots for FID with increasing total duration  $T$ . The horizontal axis represents frequency  $\omega$  and the vertical axis shows the filter function  $F(\omega, T)$ .

**Note:** Although only positive frequencies appear in the integral for decoherence (Eq. (3.26)), we plot the full symmetric filter function over both positive and negative frequencies to highlight the even symmetry  $F(-\omega) = F(\omega)$ , which arises due to the real-valued modulation function  $y(t)$ .

## 4.2 Hahn Echo

The Hahn Echo sequence, introduced by Erwin Hahn in 1950, was the first dynamical decoupling technique designed to refocus dephasing in spin systems. It remains foundational in quantum control and NMR.

In its simplest form, a  $R_x(\pi)$  pulse is applied halfway through the total evolution time  $T$ , effectively inverting the phase accumulation caused by low-frequency noise.

The modulation function  $y(t)$  becomes

$$y_{\text{echo}}(t) = \begin{cases} +1 & \text{for } 0 \leq t \leq T/2 \\ -1 & \text{for } T/2 < t \leq T \end{cases} \quad (4.3)$$

This modulation cancels noise that is approximately constant over the time interval  $T$ , making Hahn Echo particularly effective against low-frequency (quasi-static) noise.

#### 4.2.1 Hahn Echo Pulse Scheduling and Modulation Plots

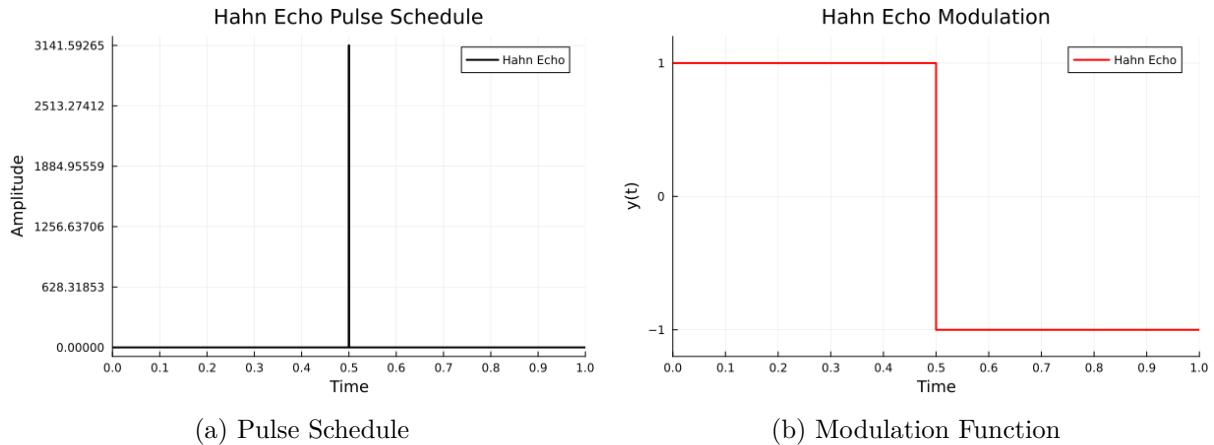


Figure 4.2: Hahn Echo: Pulse schedule and corresponding modulation function.

The amplitude of each  $\pi$  pulse is chosen such that the area under the pulse equals  $\pi$ , ensuring the correct unitary rotation. Since the pulse duration is fixed at  $10^{-3}$ , the amplitude becomes  $\frac{\pi}{10^{-3}} = 10^3\pi$ . This leads to visually tall pulses in the scheduling plots, but the integrated area (not the height) determines the physical effect. In simulations, a short pulse duration (e.g.,  $10^{-3}$  units) is used to closely approximate instantaneous  $\pi$  pulses while remaining numerically stable and integrable over time. This duration is small relative to the total evolution time  $T$ . The timing and spacing of pulses directly influence the system's ability to refocus phase errors and filter out specific frequency components of environmental noise. Sequences like Hahn, CPMG, and UDD differ precisely in how they schedule these  $\pi$  pulses.

### 4.2.2 Hahn Echo Filter Function

The Hahn Echo sequence applies a single  $\pi$  pulse at time  $t = T/2$ , flipping the sign of the effective Hamiltonian.

Using the general definition of the filter function from Eq. (3.25), we write

$$F^{\text{echo}}(\omega T) = \left| \int_0^T y_{\text{echo}}(t) e^{i\omega t} dt \right|^2$$

$$= \left| \int_0^{T/2} e^{i\omega t} dt - \int_{T/2}^T e^{i\omega t} dt \right|^2$$

Evaluating each integral

$$\int_0^{T/2} e^{i\omega t} dt = \frac{e^{i\omega t}}{i\omega} \Big|_0^{T/2} = \frac{e^{i\omega T/2} - 1}{i\omega}$$

$$\int_{T/2}^T e^{i\omega t} dt = \frac{e^{i\omega t}}{i\omega} \Big|_{T/2}^T = \frac{e^{i\omega T} - e^{i\omega T/2}}{i\omega}$$

So, their difference is  $\int_0^T y_{\text{echo}}(t) e^{i\omega t} dt$

$$\begin{aligned}
 &= \frac{e^{i\omega T/2} - 1}{i\omega} - \frac{e^{i\omega T} - e^{i\omega T/2}}{i\omega} \\
 &= \frac{2e^{i\omega T/2} - 1 - e^{i\omega T}}{i\omega} \\
 &= \frac{1}{i\omega} \left[ 2 \cos\left(\frac{\omega T}{2}\right) + 2i \sin\left(\frac{\omega T}{2}\right) - 1 - \cos(\omega T) - i \sin(\omega T) \right] \\
 &= \frac{1}{i\omega} \left[ 2 \cos\left(\frac{\omega T}{2}\right) + 2i \sin\left(\frac{\omega T}{2}\right) - 2 \cos^2\left(\frac{\omega T}{2}\right) - 2i \sin\left(\frac{\omega T}{2}\right) \cos\left(\frac{\omega T}{2}\right) \right] \\
 &= \frac{1}{i\omega} \left[ 2 \cos\left(\frac{\omega T}{2}\right) \left(1 - \cos\left(\frac{\omega T}{2}\right)\right) + 2i \sin\left(\frac{\omega T}{2}\right) \left(1 - \cos\left(\frac{\omega T}{2}\right)\right) \right] \\
 &= \frac{2}{i\omega} \left(1 - \cos\left(\frac{\omega T}{2}\right)\right) \cdot \left(\cos\left(\frac{\omega T}{2}\right) + i \sin\left(\frac{\omega T}{2}\right)\right) \\
 &= \frac{2}{i\omega} \left(1 - \cos\left(\frac{\omega T}{2}\right)\right) \cdot e^{i\omega T/2} \\
 &= \frac{4}{\omega} \cdot \sin^2\left(\frac{\omega T}{4}\right) \exp\left(i\frac{\omega T - \pi}{2}\right)
 \end{aligned}$$

$$\begin{aligned}
 \therefore F^{\text{echo}}(\omega T) &= \left| \int_0^T y_{\text{echo}}(t) e^{i\omega t} dt \right|^2 \\
 &= \left| \frac{4}{\omega} \cdot \sin^2\left(\frac{\omega T}{4}\right) \exp\left(i\frac{\omega T - \pi}{2}\right) \right|^2 \\
 &= \frac{4}{\omega^2} \cdot \sin^4\left(\frac{\omega T}{4}\right) \cdot 1
 \end{aligned}$$

Therefore, the final expression is

$$F^{\text{echo}}(\omega T) = \frac{4 \sin^4\left(\frac{\omega T}{4}\right)}{\omega^2} \quad (4.4)$$

This expression reveals that the Hahn Echo filter function suppresses low-frequency noise more effectively than FID. The  $\sin^4$  dependence leads to a faster decay in the low-frequency region, giving a stronger rejection of slow drifts in  $\beta(t)$ .

**Note:** This is a special case of the CPMG filter function with  $N = 1$  pulse.

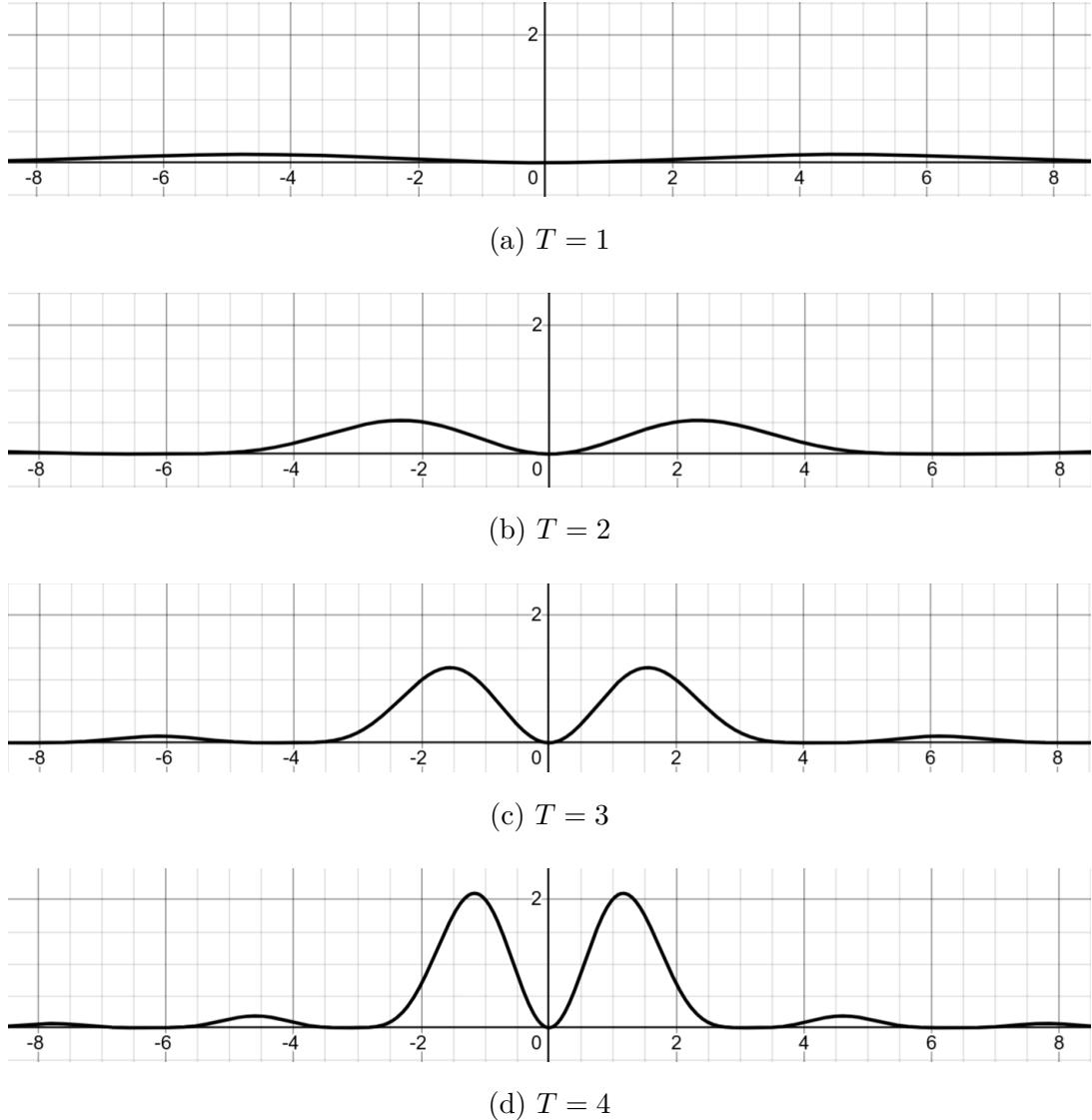


Figure 4.3: Filter function plots for Hahn Echo with increasing total duration  $T$ . The horizontal axis represents frequency  $\omega$  and the vertical axis shows the filter function  $F(\omega, T)$ .

### 4.3 Carr-Purcell-Meiboom-Gill (CPMG)

The Carr-Purcell-Meiboom-Gill (CPMG) sequence is an extension of the Hahn Echo, introduced to suppress dephasing more effectively by applying multiple  $\pi$  pulses. These pulses are equally spaced and flip the qubit state to counteract phase errors accumulated.

The modulation function for CPMG alternates between  $+1$  and  $-1$  across  $N$  pulses placed at times  $t_k = \frac{kT}{N}$  for  $k = 1, 2, \dots, N$ .

$$y_{\text{CPMG}}(t) = \begin{cases} +1 & t \in [0, t_1) \cup [t_2, t_3) \cup \dots \\ -1 & t \in [t_1, t_2) \cup [t_3, t_4) \cup \dots \end{cases} \quad (4.5)$$

### 4.3.1 CPMG Pulse Scheduling and Modulation Plots

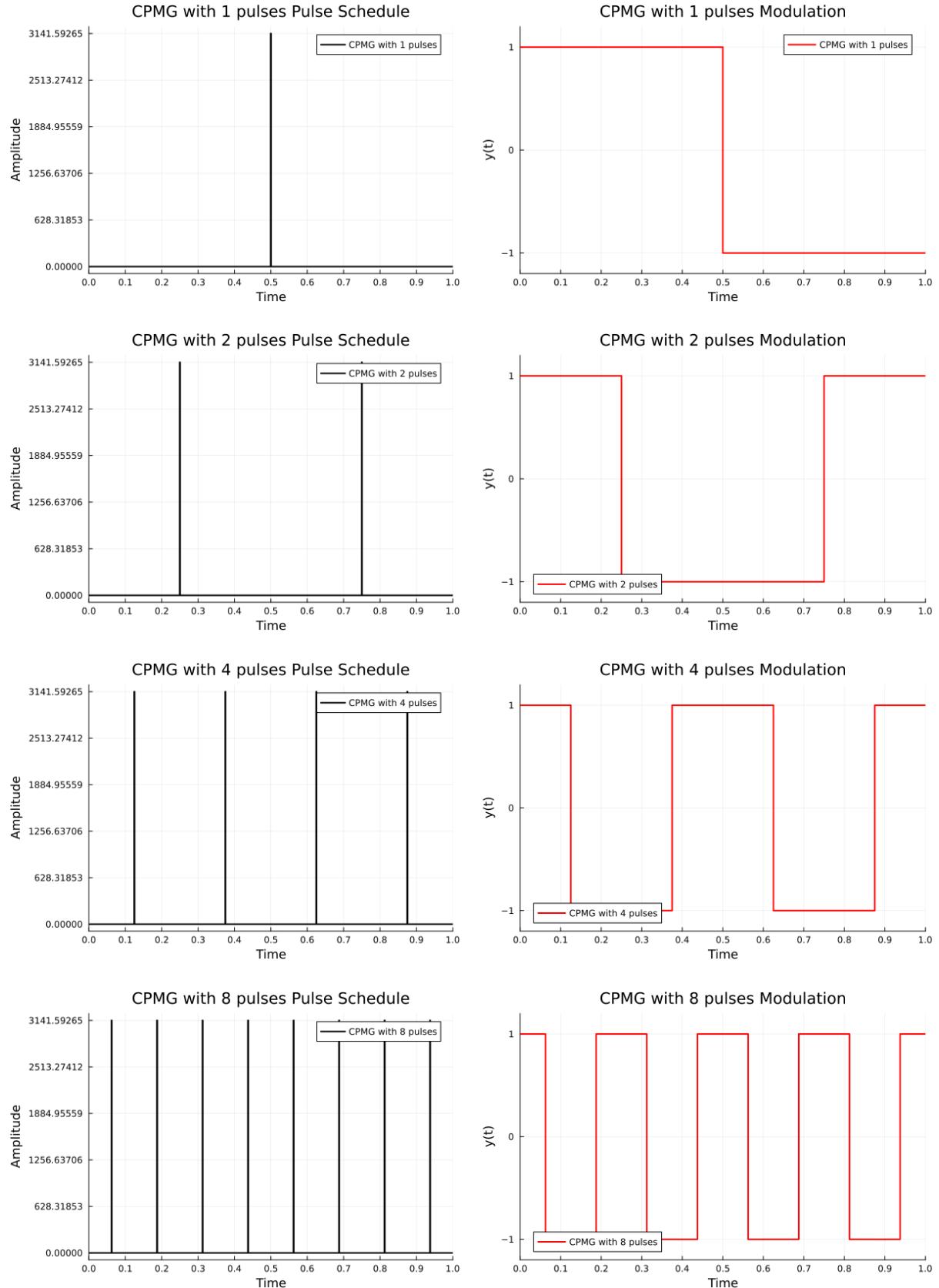


Figure 4.4: Pulse schedules and modulation functions for the CPMG sequence with increasing number of pulses  $N$ .

To better understand how the CPMG sequence manipulates noise sensitivity, we visualize both the pulse schedules and their corresponding modulation functions for different pulse counts  $N = 1, 2, 4, 8$ . Each  $\pi$ -pulse flips the sign of the modulation function  $y(t)$ , producing an alternating square-wave structure. As  $N$  increases, the modulation becomes more finely segmented, allowing the filter function to reject higher-frequency components more selectively. These visualizations complement the analytic filter function expressions and highlight how temporal structure influences spectral filtering. The resulting modulation patterns directly shape the filter function spectra derived in the next subsection.

### 4.3.2 CPMG Filter Function

The total integral becomes a sum of constant-sign segments

$$\begin{aligned} \int_0^T y_{\text{CPMG}}(t) e^{i\omega t} dt &= \sum_{k=1}^N (-1)^{k-1} \int_{t_{k-1}}^{t_k} e^{i\omega t} dt \\ &= \frac{1}{i\omega} \sum_{k=1}^N (-1)^{k-1} (e^{i\omega t_k} - e^{i\omega t_{k-1}}) \end{aligned}$$

Let us define the pulse spacing as  $\Delta t \triangleq T/N$  then  $t_k = k\Delta t$ ,

$$\Rightarrow \int_0^T y_{\text{CPMG}}(t) e^{i\omega t} dt = \frac{1}{i\omega} \sum_{k=1}^N (-1)^{k-1} (e^{i\omega k\Delta t} - e^{i\omega(k-1)\Delta t}) = \frac{e^{i\omega\Delta t} - 1}{i\omega} \sum_{k=1}^N (-e^{i\omega\Delta t})^{k-1}$$

This is a geometric series. Recall that  $\sum_{k=0}^{n-1} ar^k = a \frac{r^n - 1}{r - 1}$ . After re-indexing l=k-1, we have

$$\begin{aligned} \int_0^T y_{\text{CPMG}}(t) e^{i\omega t} dt &= \frac{e^{i\omega\Delta t} - 1}{i\omega} \sum_{l=0}^{N-1} (-e^{i\omega\Delta t})^l \\ &= \frac{e^{i\omega\Delta t} - 1}{i\omega} \cdot \frac{(-e^{i\omega\Delta t})^N - 1}{-e^{i\omega\Delta t} - 1} \\ &= \frac{e^{i\omega\Delta t} - 1}{i\omega(e^{i\omega\Delta t} + 1)} \cdot (1 - (-1)^N e^{i\omega\Delta t \cdot N}) \end{aligned}$$

we can write  $(-1)^N$  as  $e^{iN\pi}$

$$\begin{aligned} \int_0^T y_{\text{CPMG}}(t) e^{i\omega t} dt &= \frac{e^{i\omega\Delta t} - 1}{i\omega(e^{i\omega\Delta t} + 1)} \cdot (1 - e^{i\pi N} e^{i\omega\Delta t \cdot N}) \\ &= \frac{e^{i\omega\Delta t} - 1}{i\omega(e^{i\omega\Delta t} + 1)} \cdot (1 - e^{i(\omega\Delta t + \pi) \cdot N}) \end{aligned}$$

Squared magnitude of Numerator is

$$\begin{aligned}
 |\text{Numerator}|^2 &= |(e^{i\omega\Delta t} - 1) \cdot (1 - e^{i(\omega\Delta t + \pi) \cdot N})|^2 \\
 &= |e^{i\omega\Delta t} - 1|^2 \cdot |1 - e^{i(\omega\Delta t + \pi) \cdot N}|^2 \\
 \text{Now } |1 - e^{i\theta}|^2 &= |(1 - \cos \theta) - i(\sin \theta)|^2 \\
 &= (1 - \cos \theta)^2 + \sin^2 \theta \\
 &= 1 + \cos^2 \theta - 2 \cos \theta + \sin^2 \theta \\
 &= 2 - 2 \cos \theta = 4 \sin^2 \left( \frac{\theta}{2} \right) \\
 \therefore |\text{Numerator}|^2 &= 4 \sin^2 \left( \frac{\omega\Delta t}{2} \right) \cdot 4 \sin^2 \left( \frac{N}{2}(\omega\Delta t + \pi) \right) \\
 &= 16 \sin^2 \left( \frac{\omega\Delta t}{2} \right) \cdot \sin^2 \left( \frac{N\omega\Delta t}{2} \right)
 \end{aligned}$$

Squared magnitude of Denominator is

$$\begin{aligned}
 |\text{Denominator}|^2 &= |i\omega|^2 \cdot |1 + e^{i\omega\Delta t}|^2 = \omega^2 (2 + 2 \cos(\omega\Delta t)) \\
 &= 4\omega^2 \cos^2 \left( \frac{\omega\Delta t}{2} \right) \\
 \therefore F^{\text{CPMG}}(\omega, T) &= \frac{4}{\omega^2} \tan^2 \left( \frac{\omega\Delta t}{2} \right) \sin^2 \left( \frac{N\omega\Delta t}{2} \right) \quad (4.6)
 \end{aligned}$$

but  $\Delta t = T/N$  this implies the sine term simplifies and results in the following

$$F^{\text{CPMG}}(\omega, T) = \frac{4}{\omega^2} \tan^2 \left( \frac{\omega\Delta t}{2} \right) \sin^2 \left( \frac{\omega T}{2} \right)$$

For small values of  $\omega\Delta t$  we can approximate tangent with sine and we get the final expression

$F^{\text{CPMG}}(\omega, T) = \frac{4}{\omega^2} \sin^2 \left( \frac{\omega T}{2} \right) \cdot \sin^2 \left( \frac{\omega\Delta t}{2} \right)$

(4.7)

As the total duration  $T$  increases (with a fixed number of pulses), the filter function sharpens and develops a comb-like structure. This reflects increased spectral selectivity, as the sequence becomes more sensitive to noise at specific frequencies determined by pulse spacing  $\Delta t = T/N$ . The peaks correspond to frequencies where phase accumulation from noise is not canceled by the sequence. Refer to the figure 4.5

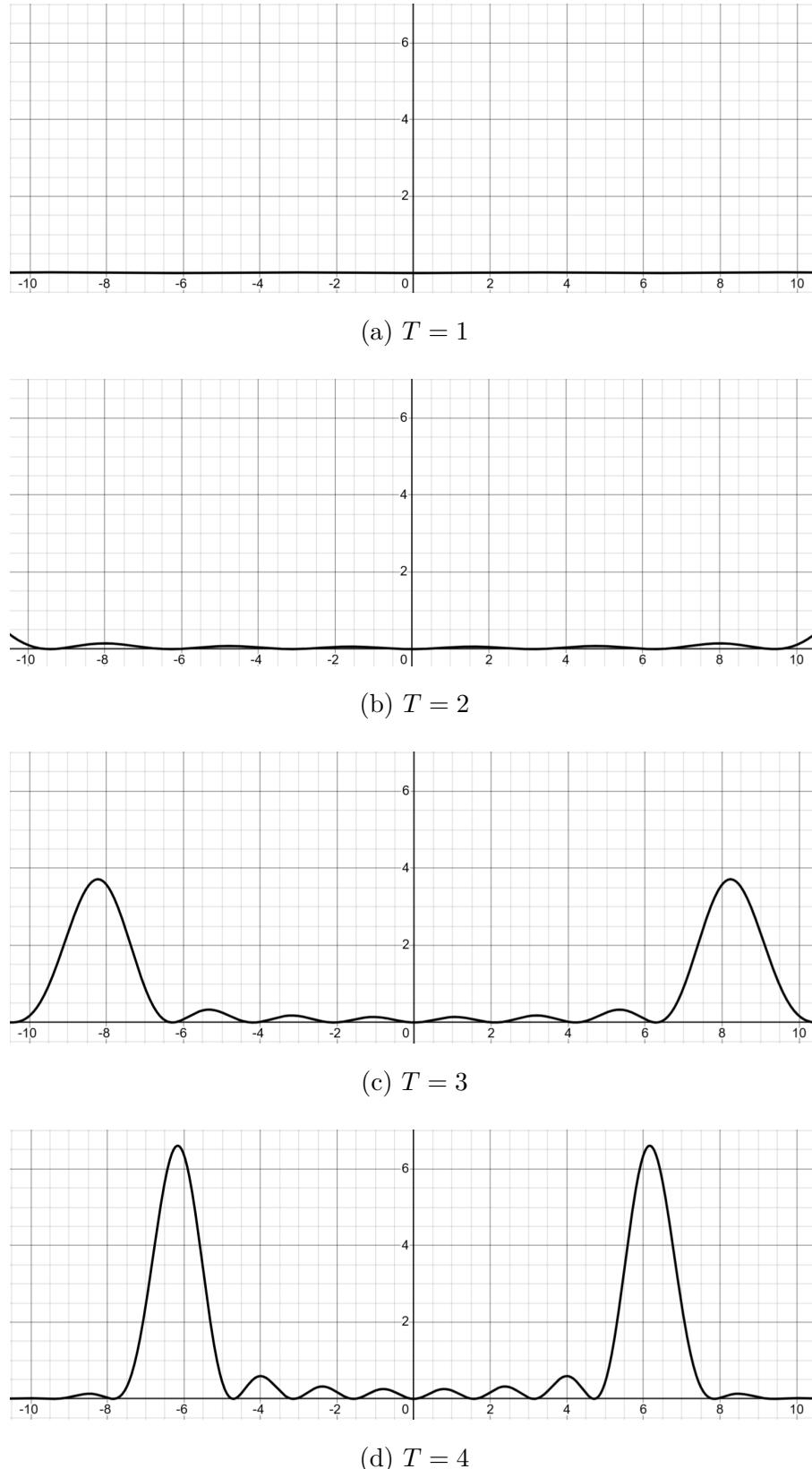


Figure 4.5: filter function plots for CPMG for fixed number of pulses  $N = 8$ , and increasing  $T$ . The horizontal axis represents frequency  $\omega$  and the vertical axis shows the filter function  $F(\omega, T)$ .

## 4.4 Periodic Dynamical Decoupling (PDD)

The Periodic Dynamical Decoupling (PDD) sequence is one of the simplest DD sequences. It consists of applying  $\pi$  pulses at equal time intervals throughout the total evolution period. PDD generalizes the Hahn Echo by allowing any number,  $N$ , of equally spaced  $\pi$  pulses. While PDD (Periodic Dynamical Decoupling) and CPMG (Carr–Purcell–Meiboom–Gill) both apply equally spaced  $\pi$  pulses to counteract decoherence, they differ subtly in design and application. PDD is a general-purpose decoupling scheme, applying pulses along a fixed axis (typically  $X$ ) without assumptions about the qubit’s initial state. CPMG, originally intended to preserve coherence for specific initial states (like along the  $Y$  axis), was designed with pulse imperfections in mind, hence the  $\pi$  pulses are traditionally applied along  $Y$  to correct systematic errors. However, in my simulations, we extend the CPMG sequence by allowing pulse axes to vary, enabling direct comparisons with PDD under identical noise conditions. This makes the distinction more about initial motivation than structural difference, as both now behave similarly under ideal pulses.

The modulation function for PDD alternates between  $+1$  and  $-1$  across  $N$  pulses spaced at times  $t_k = \frac{kT}{N+1}$ ,  $k = 1, 2, \dots, N$ .

$$y_{\text{PDD}} = \begin{cases} +1 & t \in [0, t_1) \cup [t_2, t_3) \cup \dots \\ -1 & t \in [t_1, t_2) \cup [t_3, t_4) \cup \dots \end{cases} \quad (4.8)$$

Notice that the form of the PDD modulation function looks exactly like CPMG modulation function, the difference is in the pulse timings  $t_k$ .

#### 4.4.1 PDD Pulse Scheduling and Modulation Plots

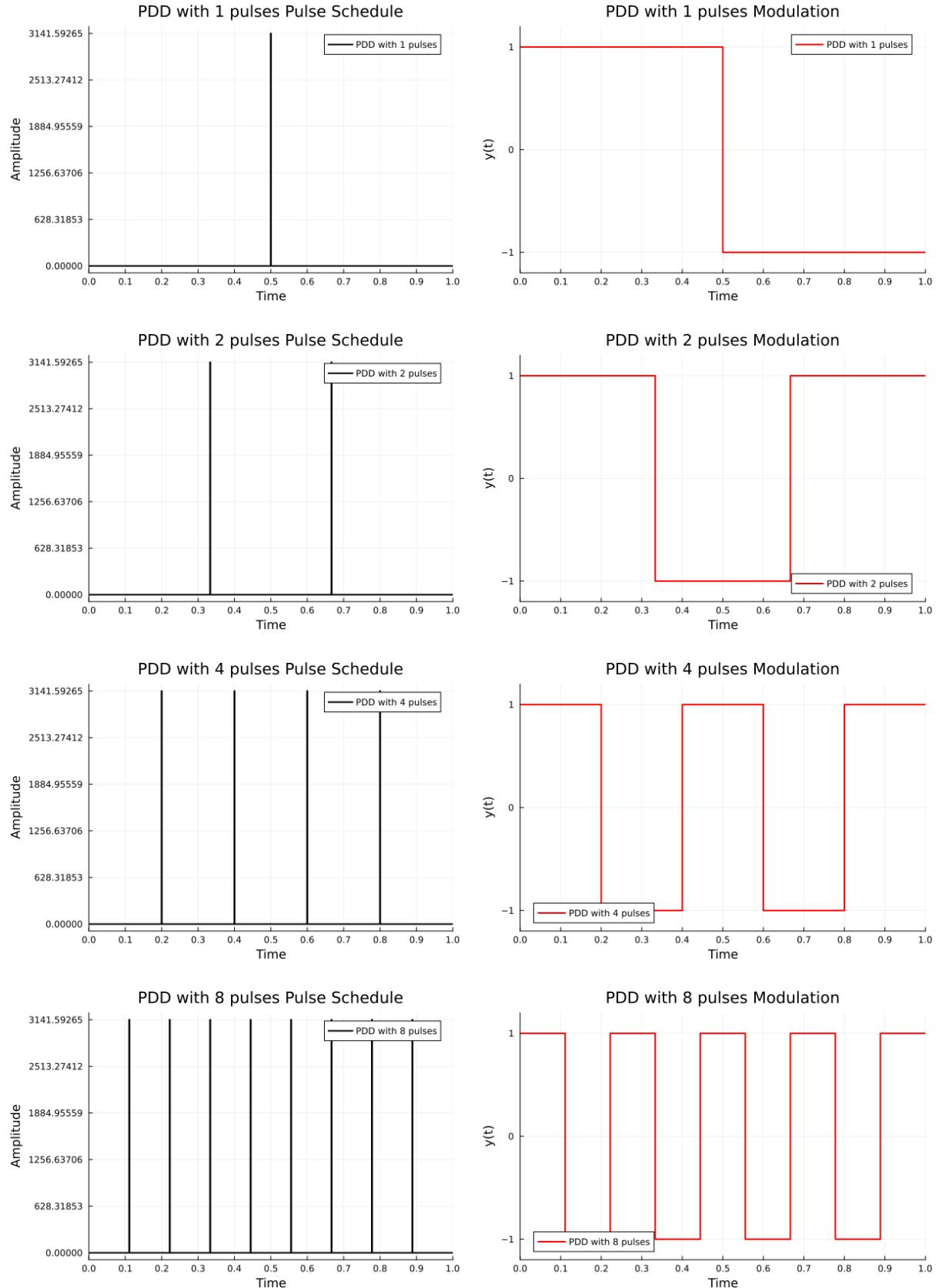


Figure 4.6: PDD Pulse Schedules and Modulation Functions for different pulse counts  $N$

#### 4.4.2 PDD Filter Function

The derivation follows exactly the same steps as in the CPMG case, and under the approximation  $\tan\left(\frac{\omega\Delta t}{2}\right) \approx \sin\left(\frac{\omega\Delta t}{2}\right)$  (valid when  $\Delta t$  is small), the resulting filter function takes the same form.

$$F^{\text{PDD}}(\omega, T) = \frac{4}{\omega^2} \sin^2\left(\frac{\omega T}{2}\right) \cdot \sin^2\left(\frac{\omega\Delta t}{2}\right) \quad (4.9)$$

However, a key distinction is that in PDD, the pulses are placed at intervals of  $\Delta t = \frac{T}{N+1}$ , unlike the CPMG sequence where  $\Delta t = \frac{T}{N}$ .

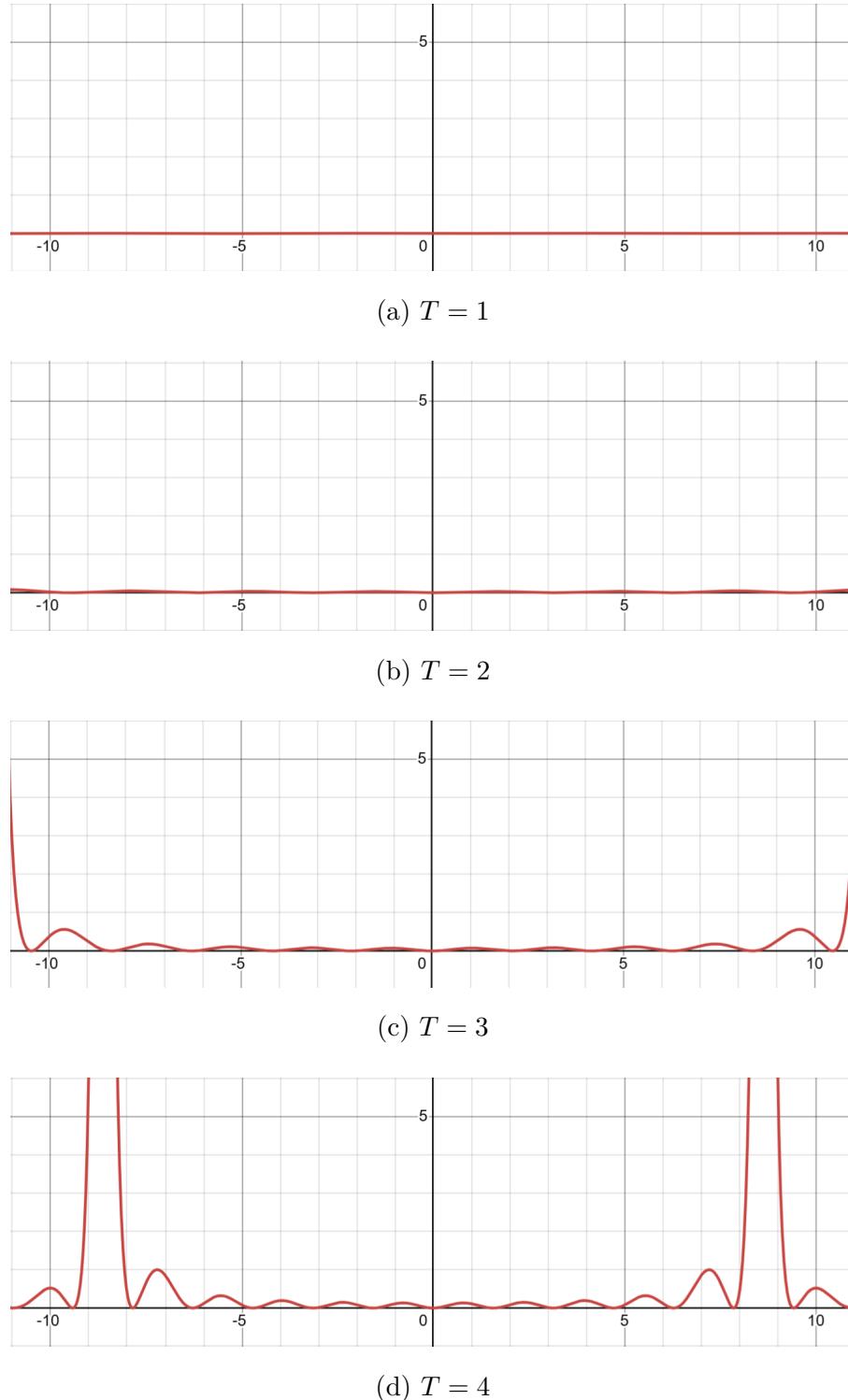


Figure 4.7: Actual PDD filter functions for increasing total durations  $T$ , with fixed number of pulses  $N = 10$ . These use the full filter function expression, not the small-angle approximation. The horizontal axis represents frequency  $\omega$  and the vertical axis shows the filter function  $F(\omega, T)$ .

## 4.5 Concatenated Dynamical Decoupling (CDD)

Concatenated Dynamical Decoupling (CDD) is a recursive technique that builds upon simpler DD sequences like PDD to achieve higher-order error suppression. The motivation is to suppress increasingly complex noise contributions by nesting control sequences within one another.

Concatenated Dynamical Decoupling (CDD) is a recursive scheme that builds increasingly robust sequences by nesting simpler sequences within one another. At its core, CDD can be understood as repeated applications of the Hahn Echo sequence on progressively smaller time intervals.

For example, the first level ( $\text{CDD}_1$ ) is just a Hahn Echo. The second level ( $\text{CDD}_2$ ) replaces the free evolution periods in  $\text{CDD}_1$  with complete  $\text{CDD}_1$  sequences, effectively applying Hahn Echo within Hahn Echo. This recursive nesting continues, with each level applying the previous level's pattern on shorter and shorter segments of time.

This hierarchical design allows CDD to suppress not just leading-order decoherence effects, but also higher-order contributions, making it powerful in environments with structured or broadband noise.

The pulse times depend on the level of CDD. The number of pulses in a  $\ell$ th level CDD is  $2^\ell - 1$ . The  $\ell$ th level modulation function is given by the following

For  $\ell = 0$ ,

$$y_0^{\text{CDD}}(t) = 1 \quad (4.10)$$

For level  $\ell \geq 1$ , the modulation alternates the sign of the previous level's modulation across segments

$$y_\ell(t) = \begin{cases} y_{\ell-1}(2t), & t \in \left[0, \frac{T}{2}\right) \\ -y_{\ell-1}\left(2t - \frac{T}{2}\right), & t \in \left[\frac{T}{2}, T\right] \end{cases} \quad (4.11)$$

### 4.5.1 CDD Pulse Scheduling and Modulation Plots

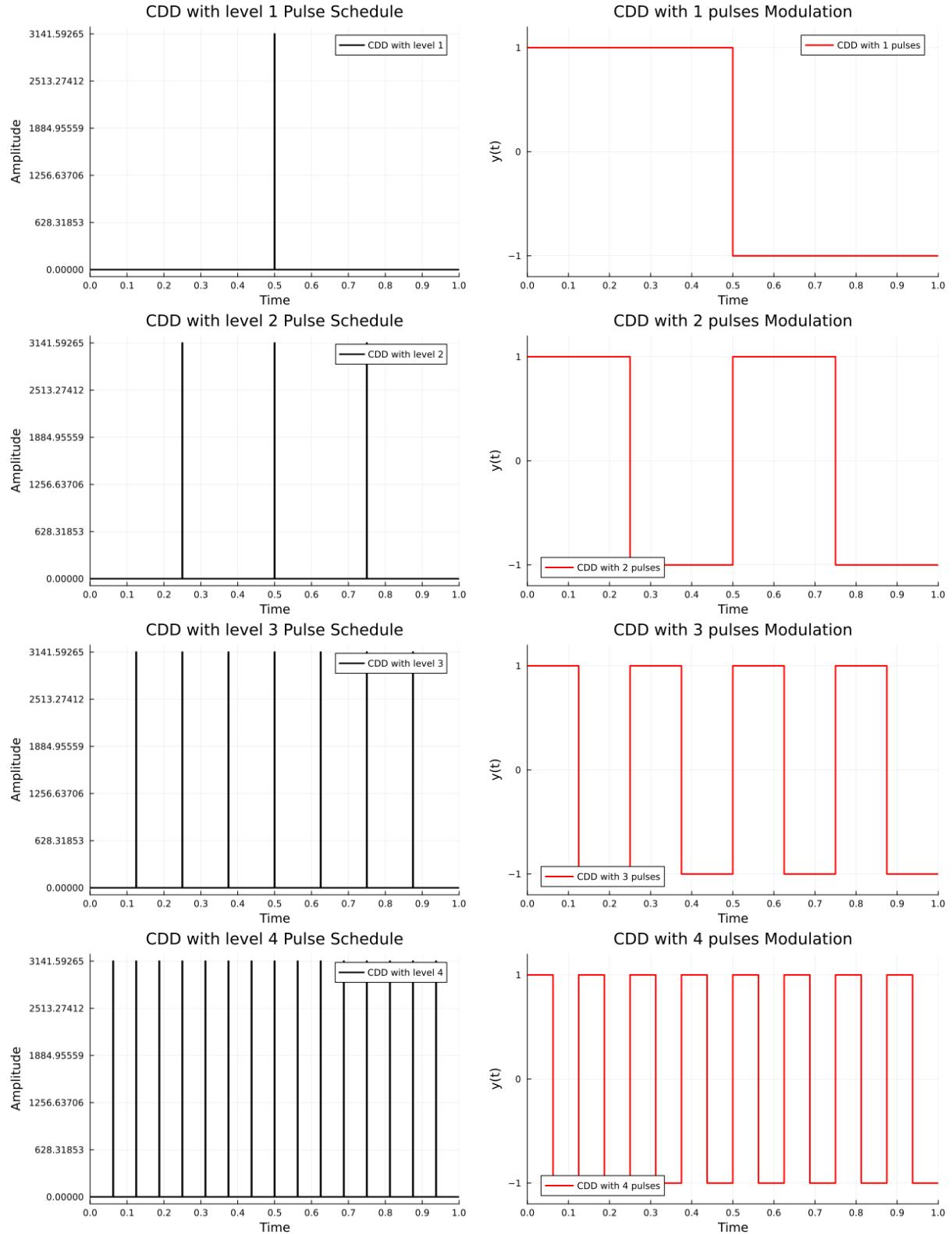


Figure 4.8: CDD Pulse Schedules and Corresponding Modulation Functions for Recursion Levels 1 to 4

### 4.5.2 $\ell$ th level CDD Filter Function

Level  $\ell = 0$  is free induction, so the fourier transform is given

$$y_0(t) = 1 \implies Y_0(\omega) = \int_0^T y_0(t) e^{i\omega t} dt = \frac{e^{i\omega T} - 1}{i\omega}.$$

then

$$F_0(\omega, T) = \frac{\sin^2(\omega T/2)}{(\omega/2)^2} \quad (4.12)$$

For  $\ell \geq 1$ , CDD is obtained by applying a Hahn echo to every half-segment of the previous level. Let's define the Fourier transform of  $\ell$ th level modulation function as  $Y_\ell(\omega) \triangleq \int_0^T y_\ell(t) e^{i\omega t} dt$ . We want to find a recursive relation between  $Y_\ell$  and  $Y_{\ell-1}$ .

$$\begin{aligned} Y_\ell &= \int_0^T y_\ell(t) e^{i\omega t} dt \\ &= \int_0^{\frac{T}{2}} y_{\ell-1}(2t) e^{i\omega t} dt - \int_{\frac{T}{2}}^T y_{\ell-1}(2t - T) e^{i\omega t} dt \end{aligned}$$

Let  $u = 2t \implies t = \frac{u}{2} \implies dt = \frac{du}{2}$  in the first term and let  $v = 2t - T \implies t = \frac{v+T}{2} \implies dt = \frac{dv}{2}$

$$\begin{aligned} \implies Y_\ell(\omega) &= \int_0^T y_{\ell-1}(u) e^{i\omega \frac{u}{2}} \frac{du}{2} - \int_0^T y_{\ell-1}(v) e^{i\omega \frac{v+T}{2}} \frac{dv}{2} \\ &= \frac{1}{2} \int_0^T y_{\ell-1}(u) \cdot e^{i\omega u/2} (1 - e^{i\omega T/2}) du \\ &= \frac{1 - e^{i\omega T/2}}{2} \int_0^T y_{\ell-1}(u) e^{i\omega u/2} du \\ &= \frac{1 - e^{i\omega T/2}}{2} \cdot Y_{\ell-1}\left(\frac{\omega}{2}\right) \\ &= -i \sin(i\omega T/4) \cdot e^{i\omega T/4} Y_{\ell-1}(\omega/2) \\ Y_\ell(\omega) &= -ie^{i\omega T/4} \sin\left(\frac{\omega T}{4}\right) \cdot Y_{\ell-1}\left(\frac{\omega}{2}\right) \end{aligned} \quad (4.13)$$

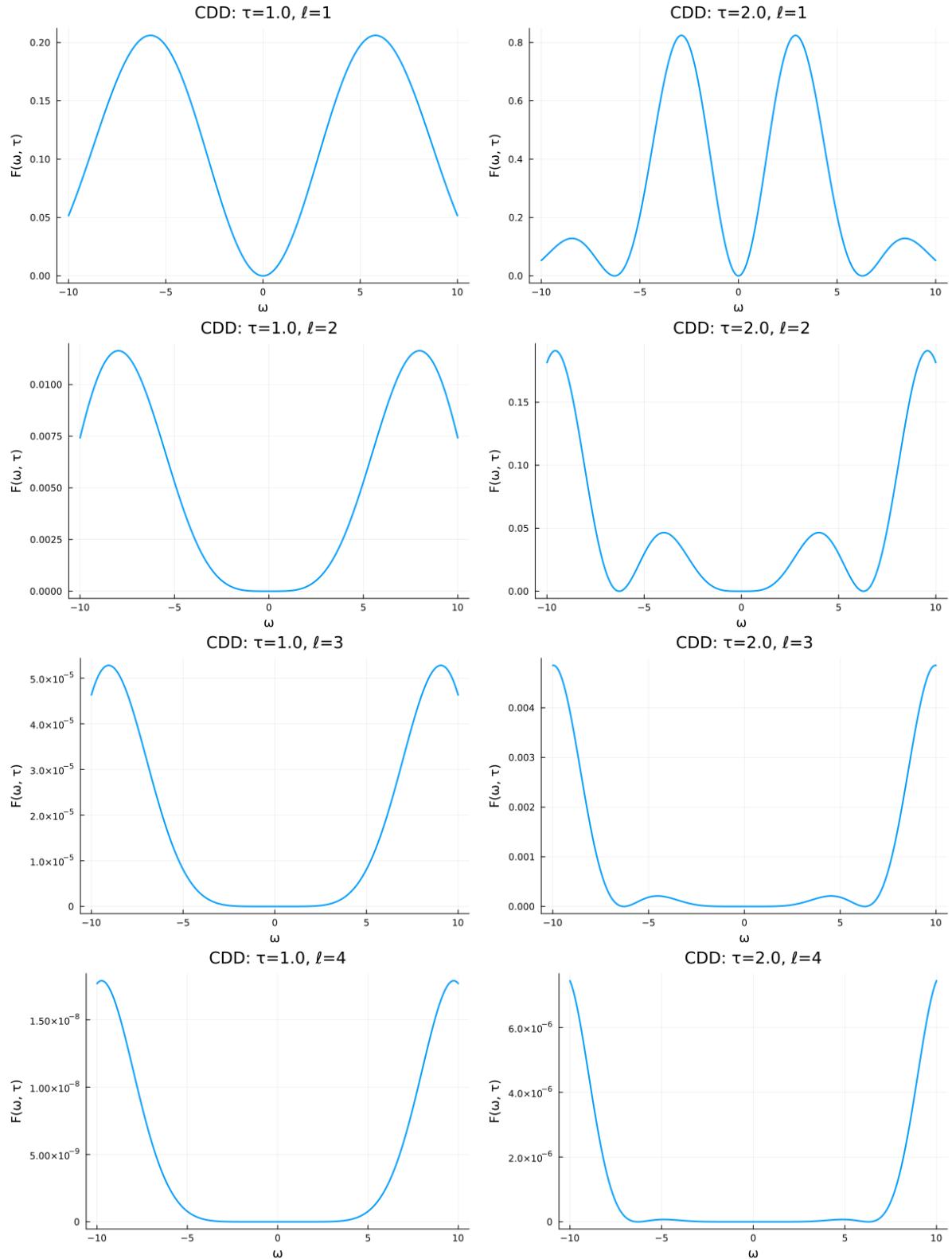
(The phase factor  $e^{i\omega T/4}$  drops out when the magnitude squared.)

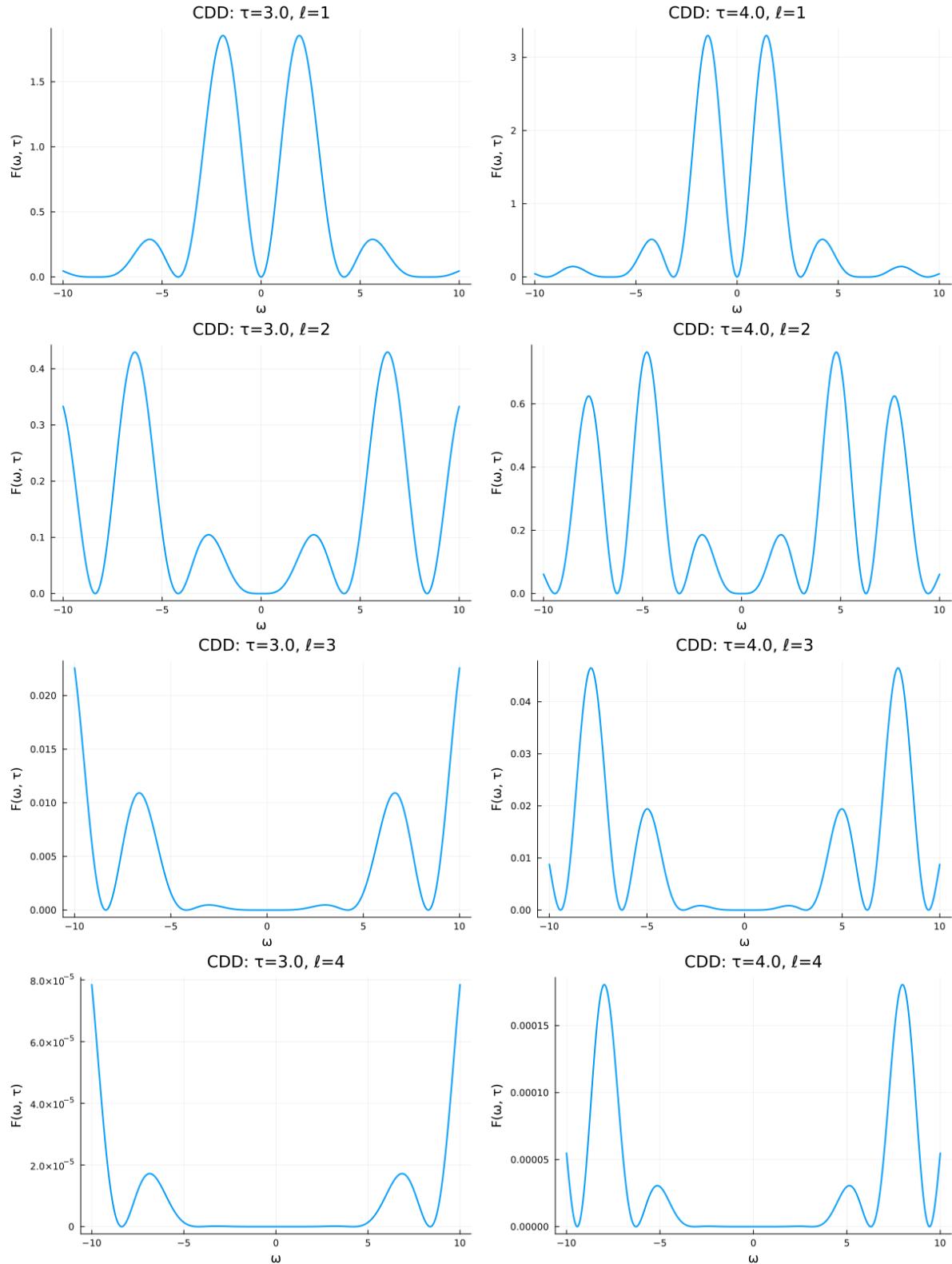
The CDD filter function is  $F_\ell(\omega, T) = |Y_\ell(\omega)|^2$ . Squaring the Eq (4.13) therefore yields

$$F_\ell(\omega, T) = \sin^2\left(\frac{\omega T}{4}\right) F_{\ell-1}\left(\frac{\omega}{2}, \frac{T}{2}\right)$$

(4.14)

The recursion makes the physical role of concatenation transparent, every new level inserts an extra  $\sin^2$  term that zeros-out noise at lower frequencies. In the simulations, I adopt the recursive formulation for constructing pulse sequences.

Figure 4.9: CDD Filter Function  $F_\ell(\omega, \tau)$  for  $\tau = 1, 2$  and  $\ell = 1$  to 4

Figure 4.10: CDD Filter Function  $F_\ell(\omega, \tau)$  for  $\tau = 3, 4$  and  $\ell = 1$  to 4

As the concatenation level  $\ell$  increases, we observe that the filter function  $F_\ell(\omega, \tau)$  becomes progressively flatter near  $\omega = 0$ , indicating an increased suppression of low-frequency noise. This is consistent with the fact that higher-level CDD sequences are designed to cancel higher-order contributions from the noise Hamiltonian. The filter function increasingly vanishes at  $\omega = 0$ , demonstrating that the system becomes less sensitive to slow fluctuations.

On the other hand, as the total evolution time  $\tau$  increases, the filter functions develop more oscillations and sharper peaks at higher frequencies. These peaks correspond to the frequency components that are less effectively suppressed. The presence of these side-lobes implies that while low-frequency noise is increasingly filtered out, the system becomes more sensitive to specific higher-frequency bands — a trade-off inherent to dynamical decoupling. Thus, careful tuning of  $\tau$  and  $\ell$  is essential to match the filter characteristics with the spectral density of the environmental noise.

## 4.6 Uhrig Dynamic Decoupling (UDD)

Uhrig Dynamical Decoupling (UDD) is a non-equidistant pulse sequence designed to optimally suppress pure dephasing under specific noise conditions<sup>1</sup>. Unlike all the DD sequences we have seen till now, which use uniformly spaced  $\pi$  pulses, UDD strategically places pulses to minimize decoherence order-by-order in a time-expansion of the fidelity decay.

For a total of  $N$  pulses applied over a time  $T$ , the  $j^{\text{th}}$  pulse is placed at

$$t_j = T \sin^2 \left( \frac{j\pi}{2(N+1)} \right), \quad j = 1, 2, \dots, N \quad (4.15)$$

The pulse timings are chosen such that they optimally suppress dephasing up to order  $N$  in time-dependent perturbation theory. Uhrig showed that this choice of pulse timings leads to cancellation of the first  $N$  terms in the expansion of the decoherence function under specific noise conditions (typically for Ohmic environments).

Unlike CPMG or PDD, UDD does not admit a simple closed-form expression for its filter function. Therefore, we compute  $F_{\text{UDD}}(\omega, T)$  numerically using the modulation function approach

$$F_{\text{UDD}}(\omega, T) = \left| \int_0^T y_{\text{UDD}}(t) e^{i\omega t} dt \right|^2 \quad (4.16)$$

---

<sup>1</sup>G. S. Uhrig, "Keeping a Quantum Bit Alive by Optimized  $\pi$ -Pulse Sequences," *Phys. Rev. Lett.* **98**, 100504 (2007).

### 4.6.1 UDD Pulse Scheduling and Modulation function

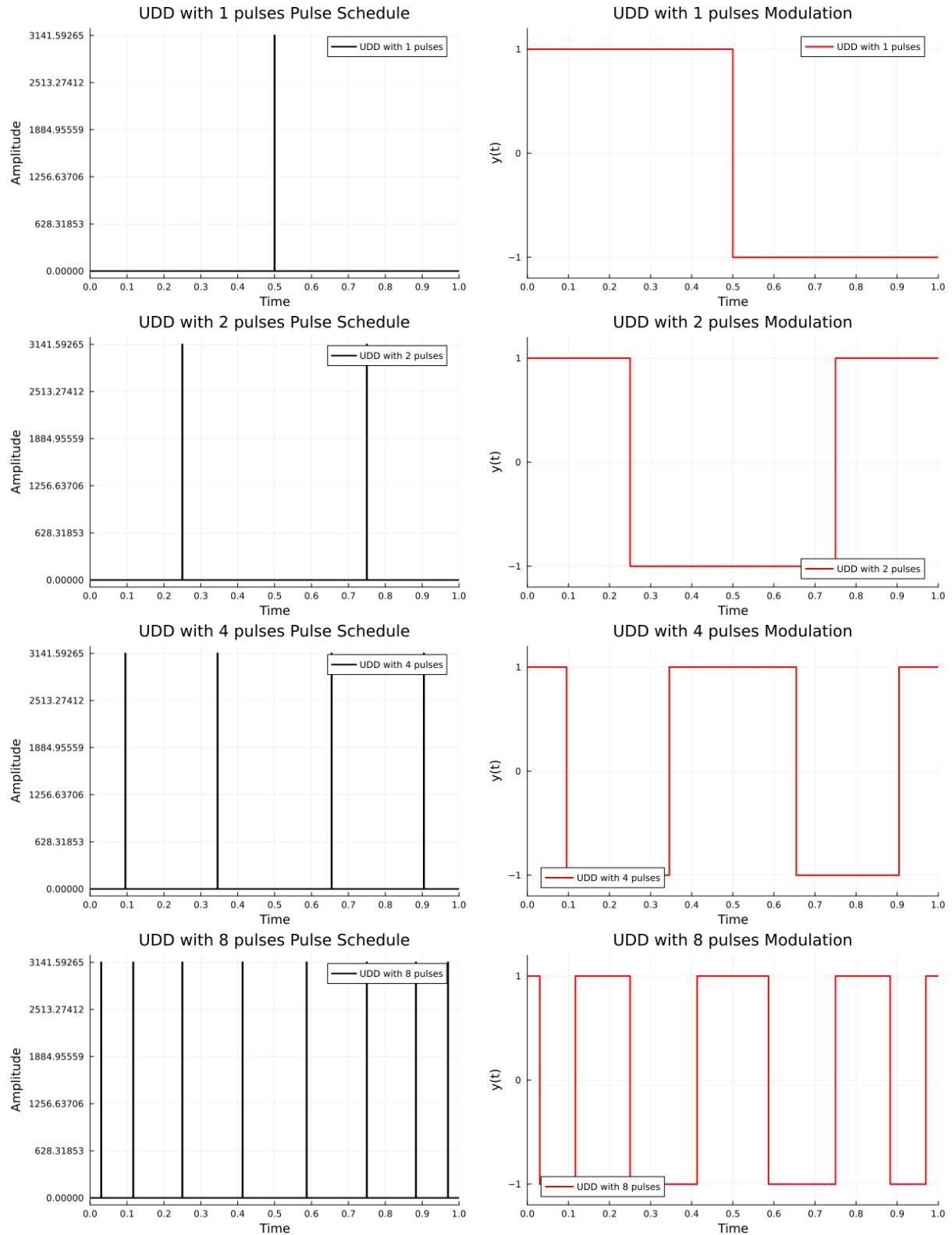


Figure 4.11: Pulse scheduling and modulation functions for UDD sequences at different pulse counts

To emphasize the non-linearity I am including more plots

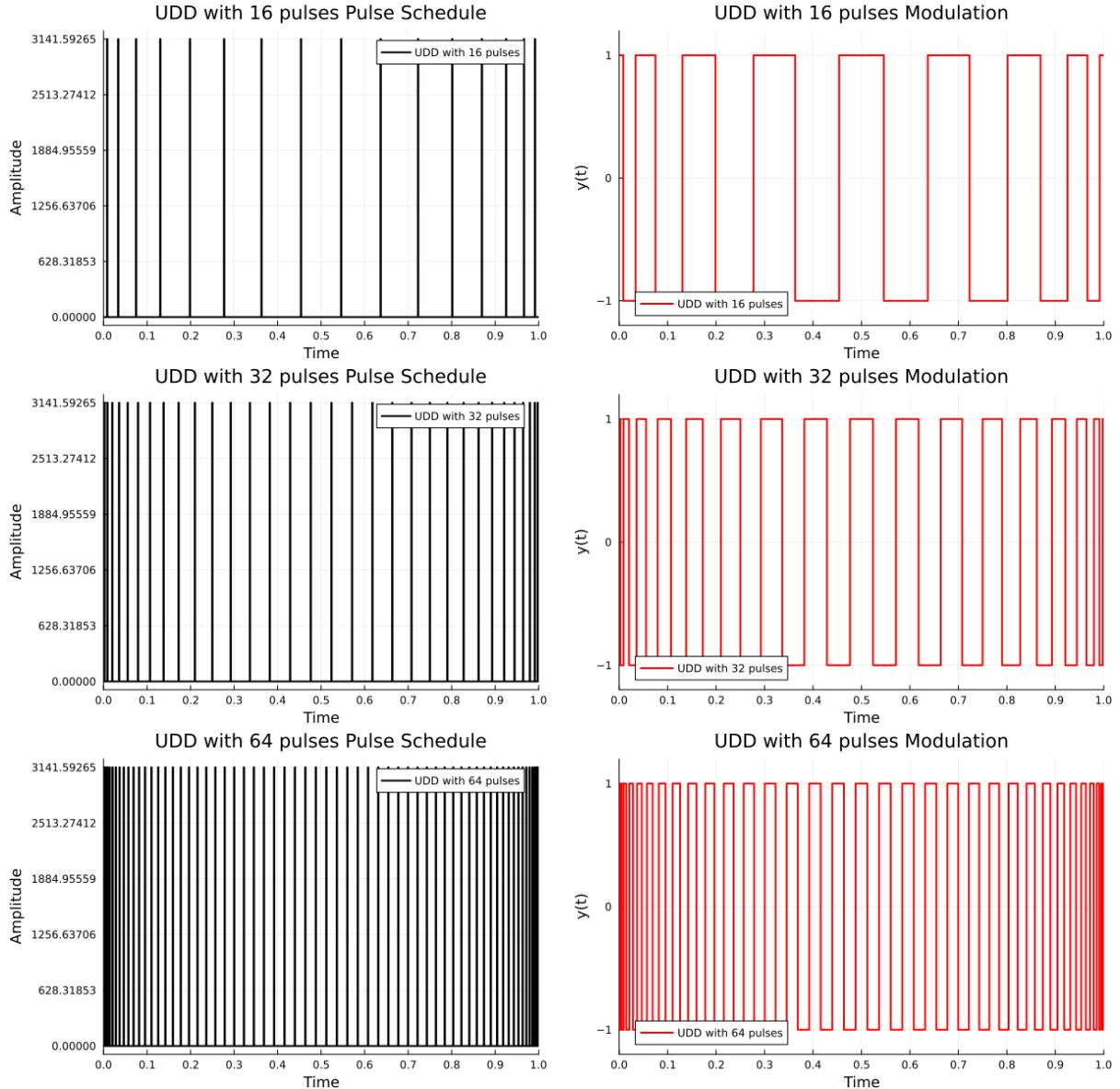


Figure 4.12: UDD pulse schedules and modulation functions showing strong nonlinearity as  $N$  increases. The pulses become increasingly concentrated toward the boundaries.

### 4.6.2 UDD Filter Function plots

The Filter function plots are given below for  $T = \{1, 2, 4\}$  and varying the number of pulses from  $N = \{1, 2, 4, 6, 8, 10\}$

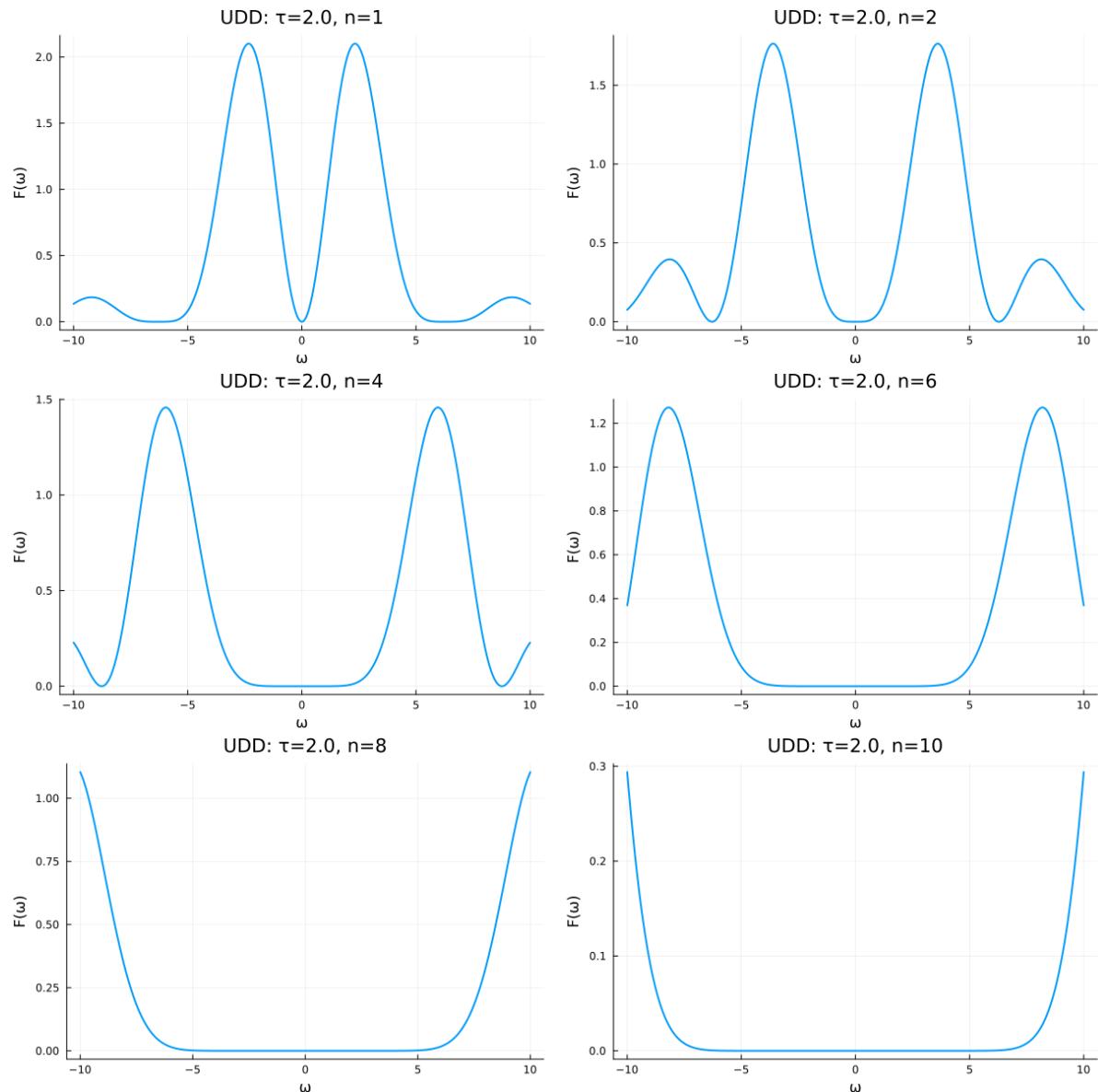
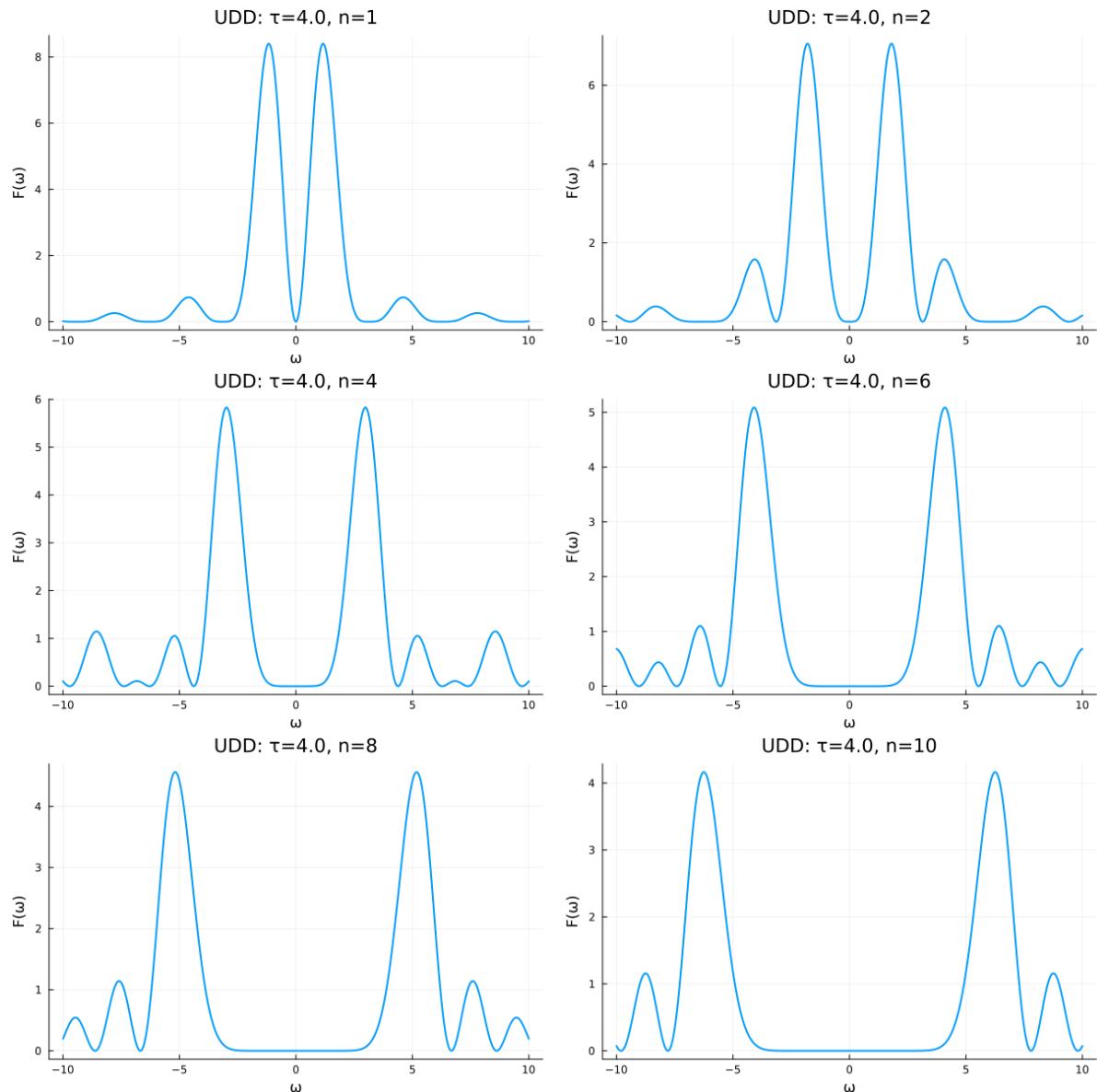


Figure 4.13: UDD filter functions for  $T = 2$  with varying pulse numbers.

Figure 4.14: UDD filter functions for  $T = 4$  with varying pulse numbers.

## 4.7 Multiaxial Dynamical Decoupling Sequences

The DD sequences discussed so far such as Hahn echo, CPMG, PDD, and UDD, primarily operate with control pulses along a single axis (typically  $x$ -axis). While such sequences effectively suppress pure dephasing noise (that is, noise coupling through  $\sigma_z$ ), they are limited in scenarios where the noise couples to the system via multiple axes (i.e.,  $\sigma_x$ ,  $\sigma_y$ , and  $\sigma_z$ ).

Multiaxial DD sequences overcome this limitation by interleaving pulses along multiple control axes<sup>2</sup>, thus averaging out general decoherence processes more effectively. These sequences are particularly relevant in experimental regimes where qubits experience transverse noise (from charge fluctuations, magnetic fields, etc.) in addition to longitudinal noise ( $z$ -noise).

In real qubit systems, noise is often not limited to a single axis. Environmental couplings can occur along all three Pauli axes  $\sigma_x$ ,  $\sigma_y$ , and  $\sigma_z$ . To mitigate such general decoherence, DD sequences that apply control pulses along multiple axes, referred to as *multiaxial dynamical decoupling* are employed.

The most well-known among these is the XY family of sequences<sup>3</sup>, which alternate  $\pi$  pulses along the  $X$  and  $Y$  axes to suppress transverse noise components. These sequences are widely used in experiments, particularly in contexts where both  $\sigma_z$  dephasing and  $\sigma_x/\sigma_y$  noise are significant.

In this project, we have simulated multiaxial noise models and evaluated the performance of such sequences numerically. The control pulses were applied in both  $X$  and  $Y$  directions, and their modulation effect was included in the system Hamiltonian directly.

### 4.7.1 The XY Family of DD Sequences

The **XY family** of DD sequences is the most widely studied class of multiaxial DD protocols. These sequences apply  $\pi$  pulses alternately along the  $X$  and  $Y$  axes to eliminate systematic pulse errors and to symmetrically suppress both transverse and longitudinal noise.

The simplest members of this family include

- **XY-4:** Consists of four  $\pi$  pulses in the sequence  $X-Y-X-Y$ , equally spaced over total evolution time  $T$ .
- **XY-8:** Concatenates XY-4 and its time-reversed version to cancel out even higher-order error terms  $X-Y-X-Y-Y-X-X-Y-X$ .

---

<sup>2</sup>See Biercuk et al., "Optimized dynamical decoupling in a model quantum memory," *Nature* **458**, 996–1000 (2009); and Khodjasteh and Lidar, "Fault-tolerant quantum dynamical decoupling," *Phys. Rev. Lett.* **95**, 180501 (2005).

<sup>3</sup>See Gullion et al., "New, Compensated Carr-Purcell Sequences," *J. Magn. Reson.* **89**, 479–484 (1990), introducing XY-4 and its variants.

- **XY-16:** Repeats the XY-8 sequence with phase-inverted pulses.

These pulse arrangements are designed not only to suppress general decoherence but also to compensate for certain systematic pulse errors. While a full analytical filter function treatment of these sequences is possible, in this project we rely on numerical simulations to evaluate their performance under multiaxial noise models. The results of these simulations, including fidelity decay curves and comparative performance with uniaxial sequences, are presented in next chapters.

### 4.7.2 Other Notable Multiaxial DD Sequences

While the XY family remains the most widely used and experimentally accessible multiaxial sequence, several other constructions have been proposed to further improve noise suppression or compensate for pulse imperfections. The **Knill Dynamical Decoupling (KDD)** sequence, for instance, uses carefully phased composite  $\pi$  pulses to correct for both dephasing and control errors. **Concatenated Dynamical Decoupling (CDD)** recursively nests multi-axial sequences like XY4 to achieve suppression to arbitrarily high order, though at the cost of exponential pulse overhead. **Quadratic DD (QDD)** combines the spectral selectivity of UDD along one axis with additional UDD layers along a perpendicular axis, providing balanced suppression of both longitudinal and transverse noise.

Although these sequences offer attractive theoretical properties, our project focuses on XY-based sequences due to their simplicity and practical relevance. A more detailed treatment of advanced sequences like KDD and QDD is left for future work.

# Chapter 5

## Simulation Framework and Workflow

This chapter describes the simulation framework developed for analyzing qubit coherence under various dynamical decoupling (DD) sequences and noise models. The goal of these simulations is to validate the filter function formalism, compare the performance of different DD strategies, and investigate their effectiveness under realistic multiaxial noise environments.

### 5.1 Overview of the Simulation Framework

The simulations were implemented in the `Julia` programming language using a modular structure. The codebase is divided into several components corresponding to key functionalities

- **Noise Generation:** Time-domain noise traces  $\beta(t)$  were synthesized from user-defined spectral density functions  $S(\omega)$  using inverse Fourier techniques.
- **Filter Function Evaluation:** Theoretical filter functions  $F(\omega)$  were computed for each DD sequence using numerical integration of modulation functions.
- **Time Evolution:** The qubit was evolved under a time-dependent Hamiltonian with both control terms and stochastic noise, using piecewise constant evolution operators.
- **Fidelity Calculation:** Coherence decay was quantified using average fidelity metrics across multiple noise realizations.

A detailed discussion of each module is provided in the following sections, including its role, mathematical formulation, and implementation details.

## 5.2 Assumptions and Their Justification

Before describing the simulation methods in detail, we list the key assumptions made in this project:

- **Classical Noise Approximation:** The environment is modeled as a classical stochastic field. While quantum noise models exist, classical models are sufficient to study Gaussian dephasing and general noise suppression.
- **Stationary and Zero-Mean Noise:** All noise processes  $\beta_\mu(t)$  are assumed to be stationary and zero-mean, ensuring that they are fully characterized by their power spectral densities  $S_\mu(\omega)$ <sup>1</sup>.
- **Uncorrelated Noise Across Axes:** Unless otherwise stated, noise along different axes (X, Y, Z) is assumed to be uncorrelated, simplifying the coherence functional to a sum over independent filter functions.
- **Piecewise-Constant Control:** Control pulses are modeled as ideal (or shaped)  $\pi$  rotations with finite duration but no internal modulation. Time evolution is approximated using piecewise-constant Hamiltonians.
- **Single-Qubit System:** All simulations were performed for a single qubit. The extension to two or more qubits is left for future work.<sup>2</sup>

These assumptions are standard in the dynamical decoupling literature and represent a balance between physical relevance and computational tractability. Their validity is discussed further in the concluding chapter.

## 5.3 Noise Generation

A key component of the simulation framework is the generation of time-domain stochastic noise traces  $\beta(t)$  corresponding to a given spectral density  $S(\omega)$ . This is essential for modeling the environment-induced fluctuations in the qubit Hamiltonian.

### 5.3.1 Spectral Density Specification

The noise is modeled as a stationary classical random process, fully characterized by its power spectral density (PSD)  $S(\omega)$ . To enable flexible simulations, a unified function  $S(\cdot)$  was implemented, allowing a wide variety of noise profiles to be constructed through

---

<sup>1</sup>For stationary zero-mean noise, the Wiener–Khinchin theorem ensures that the noise is fully characterized by its spectral density. See S. M. Kay, *Fundamentals of Statistical Signal Processing*, Vol. I (Prentice Hall, 1993).

<sup>2</sup>See L. Viola and E. Knill, "Robust Dynamical Decoupling of Quantum Systems with Bounded Controls," *Phys. Rev. Lett.* **94**, 060502 (2005) [KDD]; K. Khodjasteh and D. A. Lidar, "Fault-tolerant quantum dynamical decoupling," *Phys. Rev. Lett.* **95**, 180501 (2005) [CDD]; and J. West, B. H. Fong, and D. A. Lidar, "Near-optimal dynamical decoupling of a qubit," *Phys. Rev. Lett.* **104**, 130501 (2010) [QDD].

adjustable parameters. The general form of the spectral density is given by

$$S(\omega) = \left[ \frac{a}{\omega^2 + \varepsilon} + \frac{b}{(\omega + \varepsilon)^\alpha} + c + d\omega + e\omega^2 + f\omega^3 + A \exp\left(-\frac{(\omega - \mu)^2}{2\sigma^2}\right) \right] \cdot \exp(-\omega/\omega_c),$$

where  $\varepsilon \sim 10^{-6}$  is introduced to regularize divergences at low frequencies. The optional exponential cutoff with decay scale  $\omega_c$  allows simulation of physically realistic spectra that fall off at high frequencies<sup>3</sup>. To simulate without a cutoff,  $\omega_c$  may be set to `Inf`.

This allows synthesis of several common noise types:

- **White noise:**  $S(\omega) \propto 1$
- **$1/f$  noise:**  $S(\omega) \propto 1/\omega$
- **Ornstein-Uhlenbeck (OU):**  $S(\omega) \propto 1/(\omega^2 + \gamma^2)$
- **Quasi-static or composite noise:** Linear combinations with small  $a, b, c$

Named presets such as "white", "1/f", "OU", "quasi-static", and "composite" are mapped to parameter sets via a utility function `get_preset_params()`.

Parameter	Effect on $S(\omega)$
$a$	Controls low-frequency Lorentzian component ( $\propto 1/(\omega^2 + \varepsilon)$ ); mimics quasi-static noise or OU-type noise.
$b$	Controls $1/f^\alpha$ component; for $\alpha = 1$ , gives $1/f$ noise.
$\alpha$	Power-law exponent for colored noise; $\alpha = 0$ (white), $\alpha = 1$ ( $1/f$ ), $\alpha > 1$ (sub-Ohmic).
$c$	Adds constant (white) noise floor across all frequencies.
$d$	Linear ramp; models linearly growing spectral noise with frequency.
$e$	Quadratic term; enhances high-frequency noise beyond linear growth.
$f$	Cubic component; allows for sharper spectral growth or tailoring specific behaviors.
$A$	Amplitude of localized Gaussian peak in $S(\omega)$ .
$\mu$	Center frequency of the Gaussian peak; simulates resonant noise features.
$\sigma$	Width of the Gaussian peak; broader values model smoother resonances.
$\omega_c$	High-frequency exponential cutoff; controls spectral decay. Use <code>Inf</code> for no cutoff.
$\varepsilon$	Small constant added to prevent divergence at $\omega = 0$ ; set to $10^{-6}$ .

Table 5.1: Interpretation of parameters in the general spectral density function  $S(\omega)$

<sup>3</sup>High-frequency roll-off matches experimental flux-noise spectra; see Bylander *et al.*, “Noise spectroscopy through dynamical decoupling with a superconducting flux qubit,” *Nat. Phys.* **7**, 565 (2011).

### 5.3.2 Time-Domain Synthesis of $\beta(t)$

To simulate a real-valued time trace  $\beta(t)$  with a given PSD  $S(\omega)$  and target standard deviation  $\sigma$ , the inverse Fourier synthesis was employed  $S(\omega)$ .<sup>4</sup>

1. A frequency grid  $\omega_n$  is constructed using the FFT frequency convention over a long window of duration  $T_{\text{long}} = kT$  (with  $k \sim 10$ ).
2. The desired spectral shape  $S(\omega)$  is evaluated and normalized such that

$$\int_0^\infty S(\omega) d\omega \approx \sigma^2,$$

ensuring that the resulting  $\beta(t)$  has approximately the desired variance. This normalization is performed numerically using Riemann sums.

3. A complex-valued random spectrum is constructed as

$$\tilde{\beta}(\omega_n) = \sqrt{S(\omega_n)\Delta\omega} \cdot e^{i\phi_n},$$

where  $\phi_n$  are independent phases uniformly sampled from  $[0, 2\pi)$ .

4. The time-domain noise  $\beta(t)$  is obtained as the inverse FFT of this spectrum

$$\beta(t) = \text{Re} [\text{IFFT}(\tilde{\beta})].$$

5. To avoid edge artifacts, a segment of length  $T$  is extracted from the middle of the longer trace.

While individual realizations of  $\beta(t)$  may exhibit deviations, the method produces noise whose expected autocorrelation function corresponds to the inverse Fourier transform of  $S(\omega)$ . The implementation allows optional DC offsets, random seeds for reproducibility, and margin buffers to prevent spurious correlations due to FFT periodicity.

### 5.3.3 Validation

For all generated  $\beta(t)$ 's, noise realisations, the empirical standard deviation on average was verified to match the target value within numerical error. Spectral density reconstructions via FFT were also compared to the input  $S(\omega)$  to confirm correct shaping.

---

<sup>4</sup>A fast way to generate a real, stationary Gaussian process with an arbitrary PSD is the inverse-FFT method of Timmer & Koenig, “On generating power-law noise,” *Astron. & Astrophys.* **300**, 707 (1995).

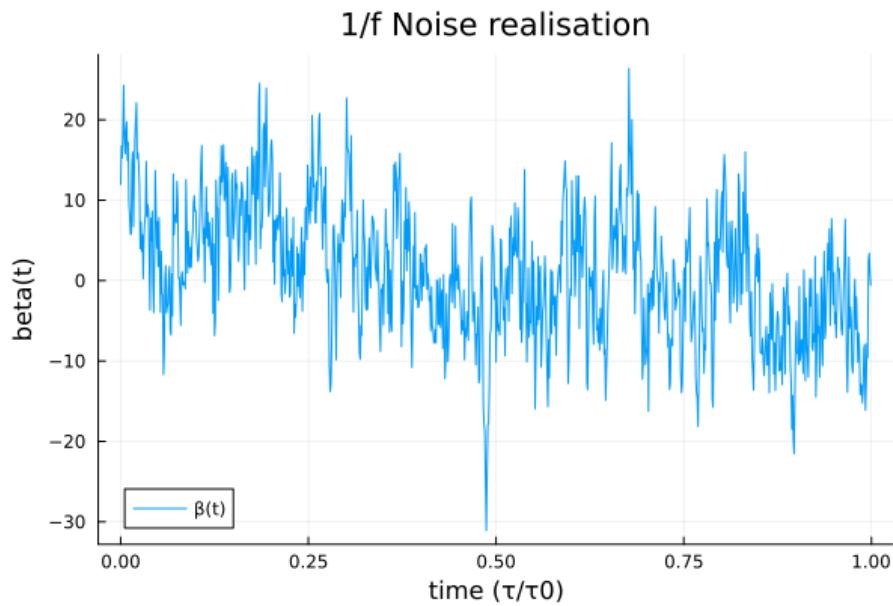
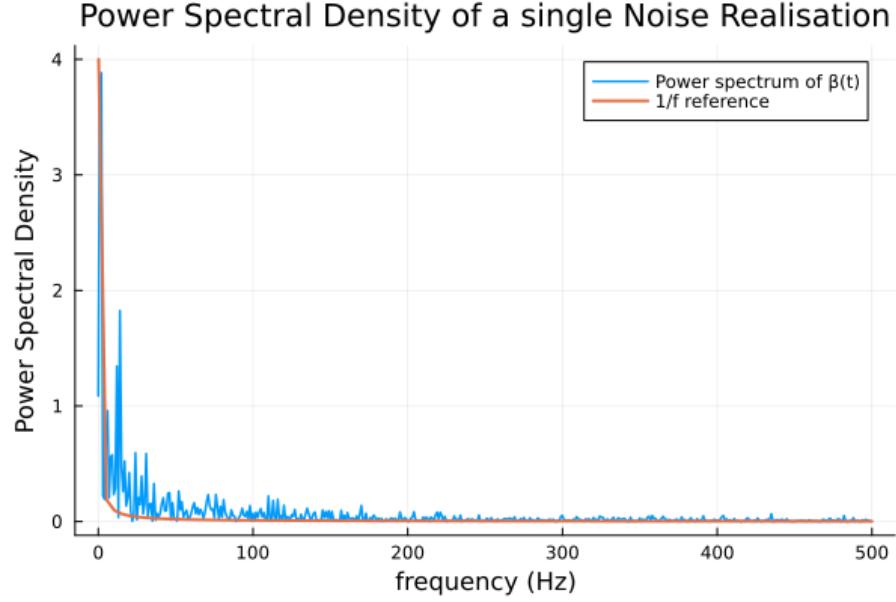
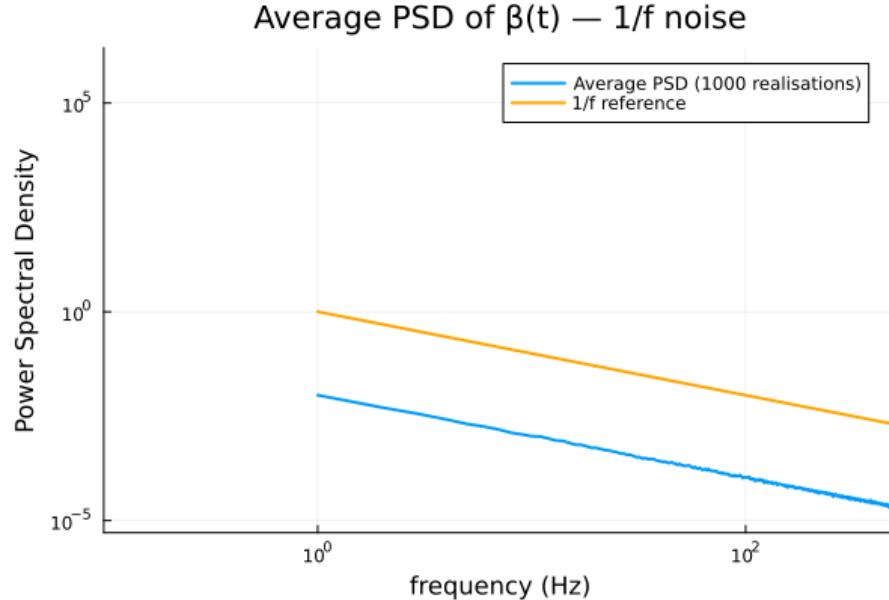


Figure 5.1: A single realization  $\beta(t)$  generated using the target  $1/f$  spectral density.

(a) Power spectral density of one realization compared to  $1/f$  reference.

(b) Average power spectral density over 1000 realizations.

Figure 5.2: Validation plots for noise generation. Each trace reflects the intended  $1/f$  spectral behavior, with averaging improving spectral accuracy.

While the average power spectral density (PSD) of the noise realizations closely matches the target  $1/f$  spectrum in linear scale, small deviations become more pronounced in log-scale error metrics, particularly near the edges of the frequency band. These discrepancies can be attributed to factors such as finite length of beta generated called trace length, spectral leakage, and the stochastic nature of individual realizations. Log-scale metrics like log-MSE are especially sensitive to near-zero values and outliers, further amplifying these effects. Nonetheless, since all dynamical decoupling simulations in this work are performed using noise generated via the same method, the relative comparisons remain internally consistent and the results valid for evaluating sequence performance.

## 5.4 Control Pulse Specification and Scheduling

In order to simulate the effect of dynamical decoupling (DD) sequences, we must specify the control Hamiltonians applied to the qubit over time. This requires a systematic way of generating and shaping pulses, scheduling them at appropriate times, and converting them into time-dependent control terms that can be provided as inputs to the solver.

The control infrastructure in this simulation is modularly organized into the following components:

- **Pulse Definition (`pulse.jl`):** Each control pulse is defined by a `Pulse` struct containing its start and stop times, the axis of rotation ( $X$ ,  $Y$ , or  $Z$ ), and the pulse shape (e.g., square or Gaussian). Functions such as `get_pulse_times` and `get_shaped_pulses` determine when and how pulses are placed based on the selected DD sequence (e.g., CPMG, UDD, Hahn, or CDD).
- **Control Term Construction (`control.jl`):** Once the pulses are scheduled, `make_control_terms` and `make_composite_control_terms` convert them into a list of operator-function pairs ( $\hat{H}_i, f_i(t)$ ), suitable as control Hamiltonian terms in `mesolve`. The function `control_function` defines the shape of the pulse envelope, and composite terms group pulses along the same axis to improve simulation efficiency.
- **Module Integration (`QuantumDD.jl`):** The `QuantumDD.jl` module aggregates all necessary components—filter functions, pulse scheduling, and control generation—into a unified interface. Importing this module ensures compatibility between simulation routines and control structures.

This design ensures flexibility across different sequences, pulse shapes, and axes, enabling straightforward experimentation and comparison of DD protocols.

## 5.5 Simulation Setup

To simulate the decoherence of a qubit under classical noise and dynamical decoupling sequences, we employed a modular framework built atop `QuantumToolbox.jl`, utilizing its `mesolve()` function for time-dependent Schrödinger evolution. The key function responsible for simulation is `run_simulation()`, which accepts a set of time-dependent control and noise terms and returns the time evolution and fidelity with respect to the initial state.

Noise was introduced as a classical stochastic process  $\beta(t)$  modulating Pauli operators via Hamiltonian terms like  $\beta(t)\sigma_z$ . The function `generate_beta()` was used to generate these noise traces with a specified power spectral density  $S(\omega)$ . Each  $\beta(t)$  was interpolated and supplied to the simulation as a function of time.

Control terms were provided as shaped pulse schedules (discussed in the previous section), represented as time-dependent amplitude functions multiplying the appropriate Pauli operator. These were passed as a list of (`Operator`, `Function`) tuples to the simulator.

The fidelity decay was computed for each realization by evolving an initial state (typically the equal superposition state) under the combined noise and control Hamiltonian. Multiple such realizations were averaged using functions like `simulate_shaped_control_fidelity()`, `simulate_modulated_noise_fidelity()`, and `fast_average_fidelity_vs_time()`, depending on the scenario and control sequence, with the most universal being `simulate_shaped_control_fidelity()`.

All simulations used consistent parameters (such as time resolution `dt`, maximum simulation time `T_max`, and number of realizations) to ensure internal comparability between different DD sequences and noise types.

The central observable in our simulations is the fidelity of the evolved qubit state with respect to its initial state. Specifically, at each time point  $t$ , we compute the fidelity

$$\mathcal{F}(t) = |\langle \psi_0 | \psi(t) \rangle|^2,$$

where  $|\psi_0\rangle$  is the initial pure state and  $|\psi(t)\rangle$  is the state after evolution under the combined effect of noise and control Hamiltonian.

The function `run_simulation()` returns both the time-evolved quantum states and their corresponding fidelities. To account for the stochastic nature of the noise, the simulation is repeated over a large number of independent noise realizations (typically  $N = 1000$ ). The fidelity values are then averaged across all realizations to obtain a smooth ensemble-averaged coherence curve:

$$\langle \mathcal{F}(t) \rangle = \frac{1}{N} \sum_{i=1}^N |\langle \psi_0 | \psi_i(t) \rangle|^2.$$

This averaged fidelity is the main observable used to evaluate the effectiveness of different dynamical decoupling sequences under various noise environments. While theoretical decay functions can be estimated using filter function formalism, we restrict our attention in this report to numerically obtained results based on direct simulation to the most part.

## 5.6 Filter Functions and Theoretical Coherence Estimation

Filter functions provide a compact way to capture how a dynamical decoupling (DD) sequence reshapes environmental noise in frequency space. In this project, numerical routines were implemented for the standard protocols FID, CPMG, UDD, CDD, and PDD. Each routine returns a numerical approximation of  $F(\omega, \tau)$ , obtained either from an analytic expression (FID and CPMG) or from a discrete Fourier transform of the modulation function  $y(t)$  (UDD, CDD, PDD).

For illustration,

$$F_{\text{FID}}(\omega, \tau) = \frac{4 \sin^2(\omega\tau/2)}{\omega^2}, \quad F_{\text{CPMG}}(\omega, \tau) \simeq \frac{4 \sin^2(\omega\tau/2) \sin^2(\omega\Delta t/2)}{\omega^2},$$

where  $\Delta t = \tau/n$  for a sequence of  $n$  equally spaced  $\pi$ -pulses.

Given a noise spectral density  $S(\omega)$ , the filter function formalism predicts the dephasing integral

$$\chi(\tau) = \frac{1}{\pi} \int_0^\infty S(\omega) F(\omega, \tau) d\omega, \quad \mathcal{L}(\tau) = e^{-\chi(\tau)},$$

which yields an analytic expectation for coherence decay under Gaussian pure-dephasing noise.

Although the present work focuses on numerically averaged fidelities obtained from full time-domain simulations, having  $F(\omega, \tau)$  available enables rapid back-of-the-envelope estimates and provides theoretical context for interpreting the numerical results.

### 5.6.1 Simulation Workflow: CPMG under White Noise in Longitudinal axis and OU Noise in the Transverse axes

To systematically evaluate the performance of various dynamical decoupling (DD) sequences under realistic noise, a unified simulation pipeline was developed using the custom Julia module `QuantumDD.jl`. The workflow is designed to simulate the fidelity decay of a qubit subject to multiaxial classical noise, using pulse sequences applied via time-dependent control Hamiltonians.

The simulations are conducted using the `simulate_shaped_control_fidelity()` function, which integrates the system's evolution via `mesolve()` from `QuantumToolbox.jl`. The key steps in the workflow are:

1. **Parameter Setup:** Total simulation time  $T$ , time step  $\Delta t$ , number of realizations, and noise strengths along  $x$ ,  $y$ , and  $z$  axes ( $\sigma_x, \sigma_y, \sigma_z$ ) are specified.
2. **Noise Modeling:** Longitudinal ( $\sigma_z$ ) noise is sampled from a white noise spectral density, while transverse ( $\sigma_{x,y}$ ) noise is drawn from an Ornstein-Uhlenbeck (OU) process, each defined via a flexible spectral density function  $S(\omega)$ .
3. **Pulse Sequence Configuration:** A DD sequence (e.g., CPMG) and pulse shape (e.g., square) are selected. Pulse durations are matched to the time step.
4. **Fidelity Evaluation:** For each noise realization, the system is evolved under the time-dependent Hamiltonian, and fidelity with respect to the initial state is computed at each time point.
5. **Averaging and Fitting:** The fidelity values are averaged over all realizations, and a stretched exponential model

$$\text{Fit}(t) = A + (1 - A) \exp(-(t/\lambda)^\beta)$$

is fit to extract effective decay times ( $T_{2,\text{eff}} = 1/\lambda$ ) and stretch factors ( $\beta$ ).

**Note:** While  $T_{2,\text{eff}}$  provides a useful effective timescale for coherence decay, it should not be conflated with the strict definition of  $T_2$  in quantum process tomography or Lindbladian dynamics. The fitted  $T_{2,\text{eff}}$  depends on the specific noise model, pulse sequence, and fitting window used, and serves primarily as a comparative metric across different simulation scenarios.

This pipeline allows consistent simulation and comparative analysis of different sequences under various realistic noise conditions.

### 5.6.2 Note on Noise Strength Calibration

To enable meaningful comparisons across different noise models, noise strengths along each axis were calibrated so that the effective decoherence timescale (e.g.  $T_2^*$  under FID) lies within a comparable range across simulations. For example, a white-noise spectral density may require a standard deviation of  $\sigma = 0.2$  to drive the fidelity down to  $1/e$  in 0.1 a.u., whereas an Ornstein–Uhlenbeck (OU) noise model with the same  $\sigma$  could produce a markedly different  $T_2^*$ . Accordingly, the parameters  $\sigma_z$ ,  $\sigma_x$ , and  $\sigma_y$  were tuned independently for each simulation so that all decay profiles fall within similar timescales.

For multiaxial simulations, the transverse noise strengths were set to  $\sigma_x = \sigma_y = \sigma_z/10$  unless otherwise noted. This maintains the dominance of longitudinal dephasing (as is typical in many physical qubit systems) while allowing transverse noise effects to be meaningfully probed.

**Note:** The decoherence window was empirically chosen to be around 0.1 (a.u.)—short enough for fidelity decay to be clearly observable, yet long enough that the impact of different DD pulse counts remains visible. This makes the simulations both informative and computationally manageable.

# Chapter 6

## Results and Discussion

This section presents the numerical fidelity-decay results obtained with the simulation workflow outlined in Section 5.6.1. Unless otherwise stated, each data set corresponds to  $N = 1000$  independent noise realisations, a time step  $\Delta t = 10^{-3}$ , and an empirically calibrated decoherence window of 0.1 a.u. (cf. Noise-strength note).

Throughout, the ensemble-averaged fidelity

$$\langle \mathcal{F}(t) \rangle = \frac{1}{N} \sum_{i=1}^N |\langle \psi_0 \rangle \psi_i(t)|^2$$

is fitted to a stretched-exponential model

$$\text{Fit}(t) = A + (1 - A) \exp(-( \lambda t)^\beta), \quad T_{2,\text{eff}} = \lambda^{-1},$$

yielding the effective coherence time  $T_{2,\text{eff}}$  and stretch factor  $\beta$  for each protocol.

### Note on Saturation Fidelity and Stretch Exponents.

Across most simulations, the baseline parameter  $A$  (corresponding to the long-time saturation fidelity) remains close to 0.5, consistent with the qubit uniformly decohering into a maximally mixed state along the Bloch sphere equator under pure dephasing. This behavior is particularly evident in free induction decay (FID) and poorly performing DD sequences under strong noise.

Interestingly, in simulations where dynamical decoupling is highly effective — such as UDD applied against Ornstein–Uhlenbeck (OU) noise with high pulse counts — the fitted  $A$  rises above 0.5. This indicates that the DD sequence is not merely delaying decoherence but actively preserving higher coherence even in the long-time limit. The system does not saturate to complete randomness within the simulated window, implying robust protection.

Furthermore, the fitted stretch exponent  $\beta$  serves as a strong qualitative indicator of the nature of the decoherence. A value of  $\beta \approx 1$  corresponds to exponential decay, typical for memoryless (Markovian) environments. In contrast, values approaching

or exceeding  $\beta = 2$  indicate a crossover into Gaussian decay regimes, which generally arise when noise correlations are strongly suppressed, a signature of effective DD.

While a higher  $\beta$  cannot be directly equated to longer coherence times without also considering  $\lambda$ , it reflects a more gradual initial decay and is often desirable. In particular, larger  $\beta$  values:

- Imply enhanced short-time coherence,
- Indicate suppression of low-frequency noise contributions,
- Suggest that the dominant noise process is being efficiently filtered out by the pulse sequence.

Hence, the emergence of  $\beta \gtrsim 2$  in some simulations should be interpreted as a positive outcome, even if  $T_{2,\text{eff}} = 1/\lambda$  is modest.

In this work, simulations were performed to evaluate the effectiveness of various dynamical decoupling (DD) sequences under classical noise. The majority of these simulations involved uniaxial dephasing noise (typically along the  $Z$  axis) and uniaxial control pulses (typically along  $X$ ). Specifically, uniaxial noise + uniaxial control simulations were conducted for CPMG, UDD, and CDD sequences.

Preliminary simulations were also conducted for the Periodic Dynamical Decoupling (PDD) sequence, but these were not pursued further since the results were found to be qualitatively very similar to those of CPMG.

Multiaxial noise simulations (i.e., noise applied along all three  $X$ ,  $Y$ , and  $Z$  axes) were carried out only for the CPMG sequence. This was not a design choice but rather a consequence of time constraints toward the end of the project. Given additional time, similar simulations could be extended to UDD, CDD, and other sequences for a more comprehensive comparison.

Nonetheless, the CPMG simulations under multiaxial noise serve as a meaningful demonstration of how transverse noise affects DD performance, and how such scenarios can be tackled within the developed simulation framework.

The results below are organized by pulse sequence. For each case, the fidelity decay under different noise types and pulse counts is presented, along with fits to stretched exponential decay models to extract effective decoherence times and stretch exponents.

## 6.1 Simulation Results

### 6.1.1 Hahn Echo

To establish a baseline for DD performance, we begin by examining the Hahn echo sequence — the simplest form of dynamical decoupling involving a single  $\pi$  pulse at the midpoint of evolution. Simulations were performed under various noise models applied along the longitudinal ( $Z$ ) axis. The results reveal that the Hahn echo sequence offers almost no improvement, compared to FID, in coherence under common stochastic noise models such as white noise,  $1/f$  noise, and Ornstein–Uhlenbeck (OU) noise. This is expected, as these models lack a dominant DC (zero-frequency) component, which the Hahn echo is specifically designed to cancel.

However, when the noise includes a significant quasi-static or low-frequency (DC) component, the Hahn echo proves effective in partially reversing dephasing. This is consistent with its theoretical role in refocusing dephasing caused by time-independent or slowly varying fluctuations in the environment.

These observations highlight the limited utility of Hahn echo in suppressing high-frequency or broadband noise, and motivate the use of more sophisticated sequences (e.g., CPMG, UDD) that better target such spectra.

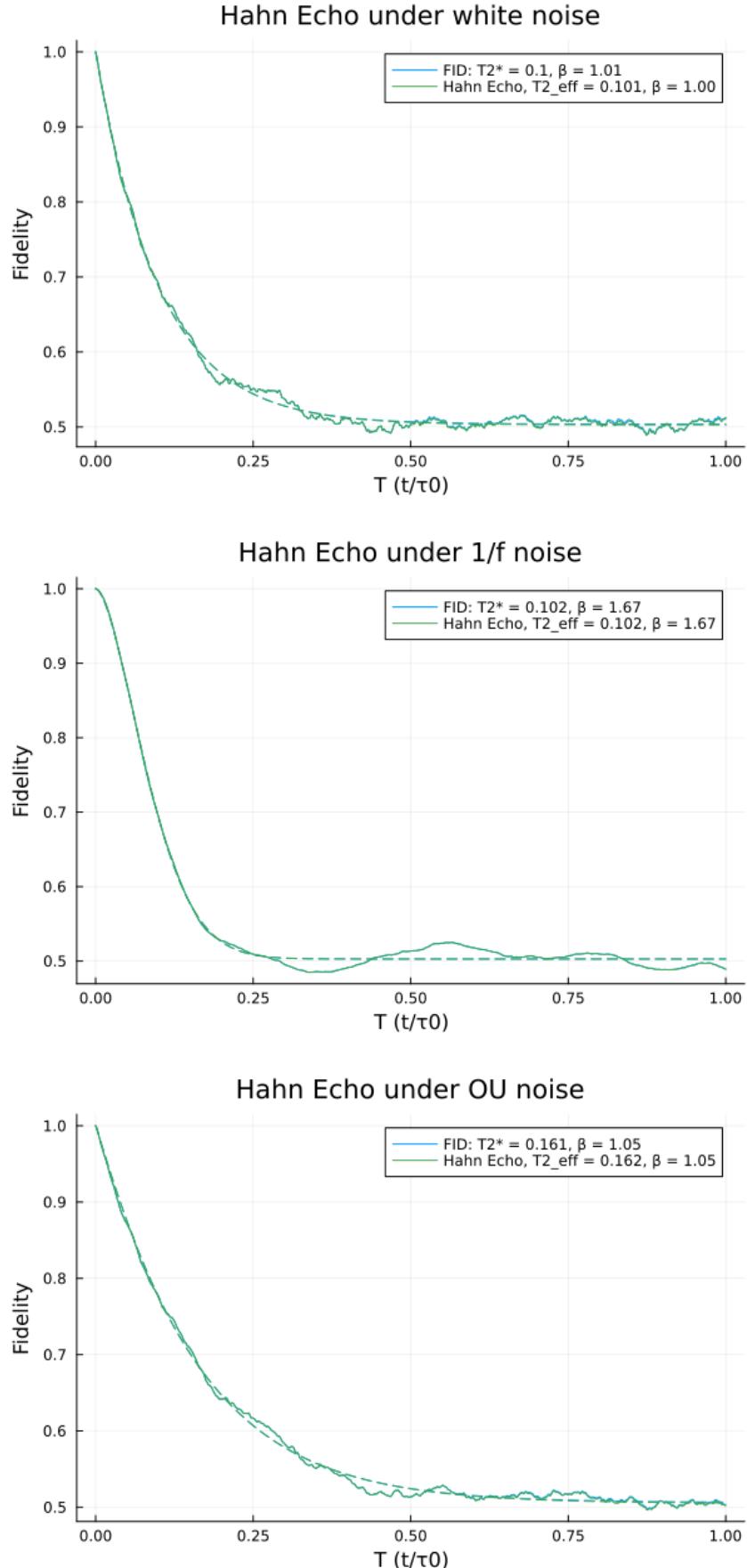
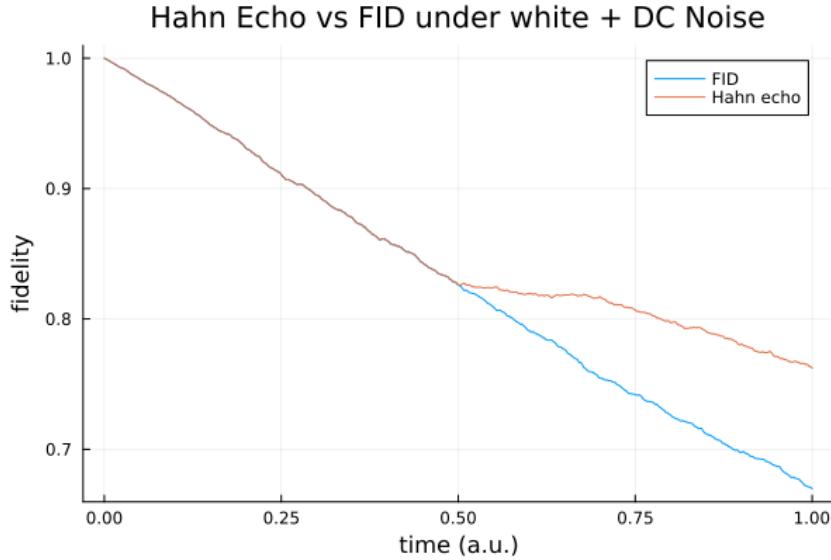
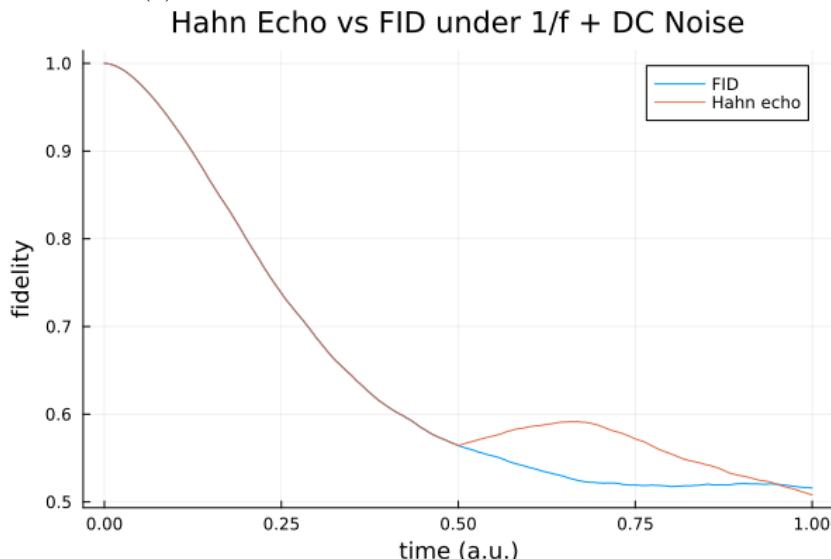
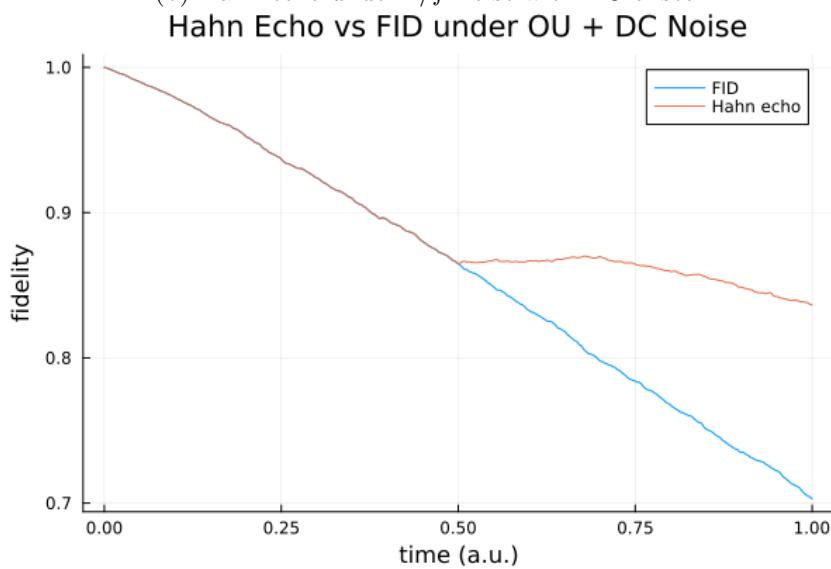


Figure 6.1: Hahn echo applied to different noise models. The FID plots are almost completely overlapped by the Hahn echo curves, demonstrating that Hahn echo offers negligible improvement under white,  $1/f$ , or OU noise.



(a) Hahn echo under White noise with DC offset.

(b) Hahn echo under  $1/f$  noise with DC offset.

(c) Hahn echo under OU noise with DC offset.

Figure 6.2: Comparison of Hahn echo performance under different noise types with added DC component. Unlike the pure stochastic cases, here a clear recovery is observed due to cancellation of quasi-static detuning.

### 6.1.2 CPMG Sequence

Table 6.1: Relative increase in effective coherence time  $T_{2,\text{eff}}/T_2^*$  for CPMG under various noise models.

Number of Pulses	White Noise	$1/f$ Noise	OU Noise
200	1.36	1.30	1.46
400	2.05	1.82	2.54
600	4.70	2.91	5.76
800	22.04	6.54	23.33

**Discussion of Results:** From Table 6.1, we observe that CPMG sequences significantly enhance the effective coherence time  $T_{2,\text{eff}}$  compared to free induction decay (FID) across all noise types. The extent of improvement varies with the noise spectrum and the number of applied pulses.

Under white noise, the increase in  $T_{2,\text{eff}}$  remains modest for low pulse counts but grows rapidly with higher pulse numbers, reaching more than 22 times the original  $T_2^*$  at 800 pulses. This is consistent with the broadband nature of white noise, where increased pulse density effectively filters high-frequency components.

For  $1/f$  noise, the improvements are more gradual. While still substantial, the gains saturate earlier compared to white noise, reflecting the more challenging low-frequency-dominated nature of  $1/f$  spectra.

In contrast, Ornstein–Uhlenbeck (OU) noise responds exceptionally well to CPMG. The observed  $T_{2,\text{eff}}$  increases over 23-fold by 800 pulses. This dramatic enhancement is a hallmark of the finite correlation time of OU noise, which CPMG can exploit to suppress decoherence more effectively.

Overall, these results affirm that CPMG performs robustly across noise types, with particularly strong benefits in suppressing noise with finite correlation structure. The relative increase in  $T_{2,\text{eff}}$  is a clear indicator of DD effectiveness in each case.

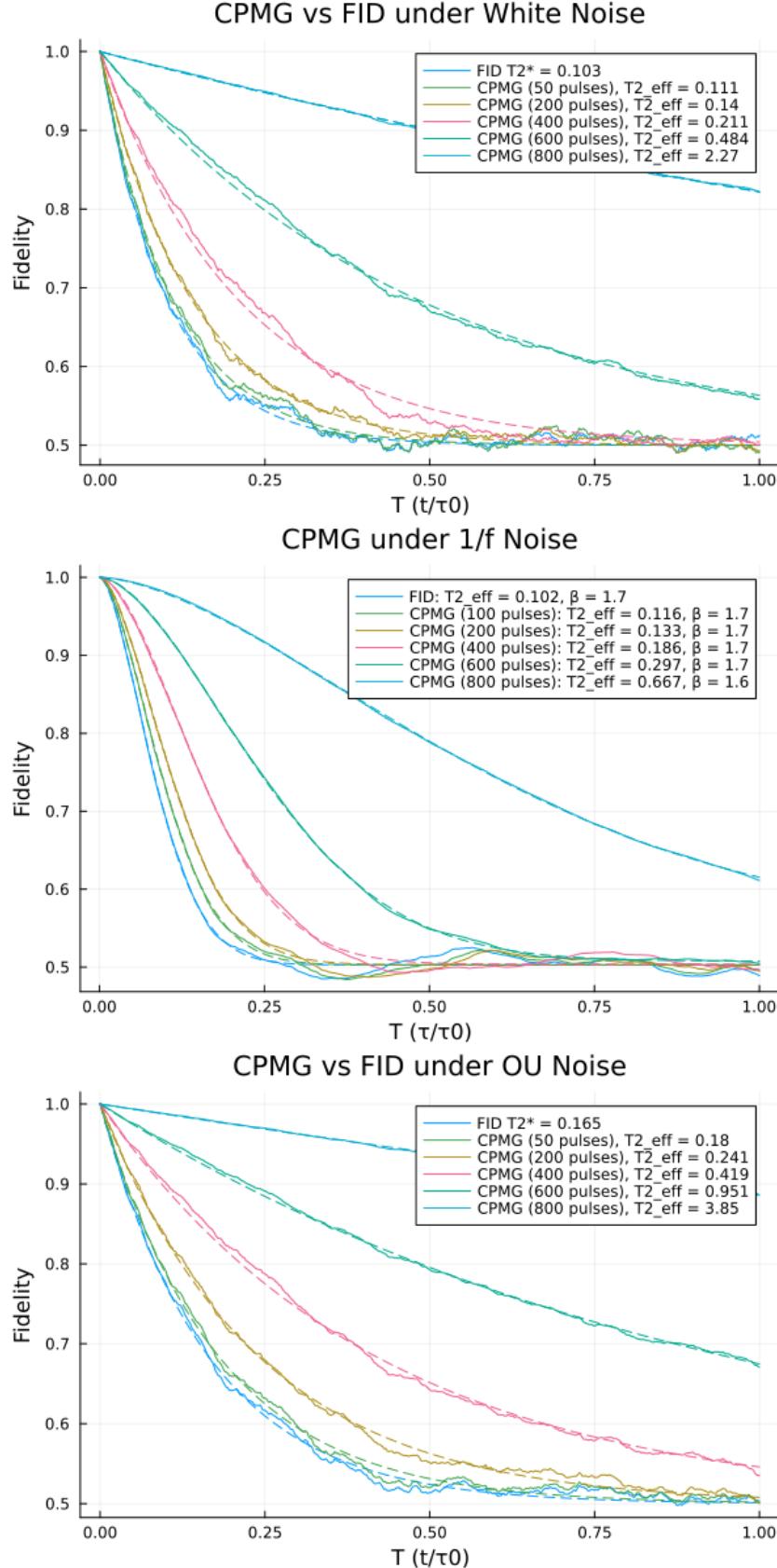


Figure 6.3: Ensemble-averaged fidelity for CPMG sequence subjected to three representative noise spectra. Each panel shows FID and CPMG curves for  $N = 100\text{--}800$  pulses, together with stretched-exponential fits.

### 6.1.3 UDD Sequence

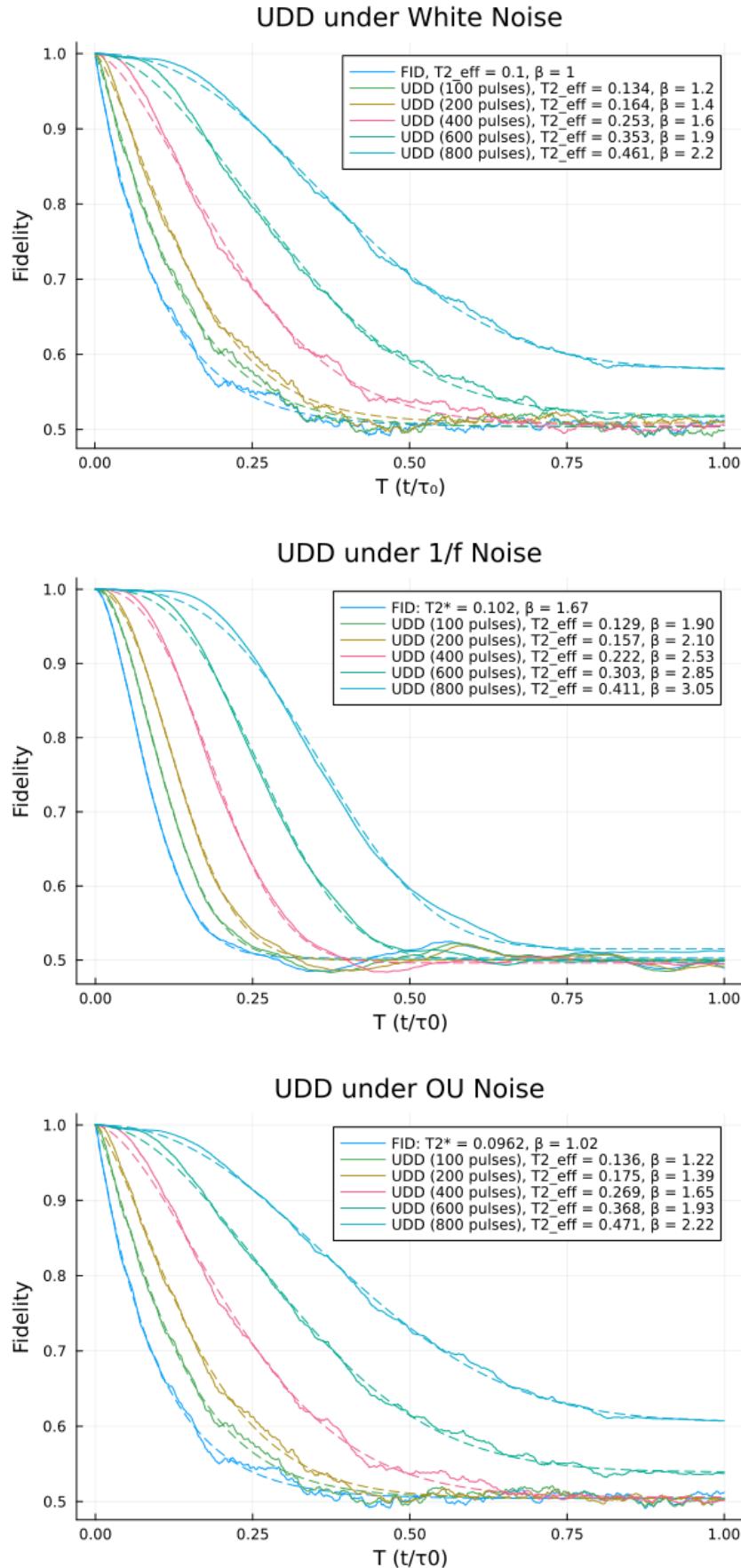


Figure 6.4: Performance of UDD sequences against different noise types at high pulse counts.

Table 6.2: Comparison of  $T_{2,\text{eff}}/T_2^*$  and stretch exponent  $\beta$  for UDD under different noise models

<b>Pulses</b>	<b>White Noise</b>		<b>1/f Noise</b>		<b>OU Noise</b>	
	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$
FID	1.00	1.01	1.00	1.67	1.00	1.02
100	1.34	1.18	1.26	1.90	1.42	1.22
200	1.64	1.38	1.54	2.10	1.82	1.39
400	2.53	1.60	2.18	2.53	2.80	1.65
600	3.53	1.89	2.97	2.85	3.83	1.93
800	4.61	2.24	4.03	3.05	4.90	2.22

Across all three noise models—white,  $1/f$ , and Ornstein–Uhlenbeck (OU)—we observe a clear trend of increasing  $T_{2,\text{eff}}$  and stretch exponent  $\beta$  with the number of pulses in the UDD sequence. This reflects the progressively stronger suppression of low-frequency noise components and correlated fluctuations.

Notably,

- $\beta$  **increases significantly** with pulse count, especially under  $1/f$  noise, reaching values beyond 3 for 800 pulses. This transition suggests a shift from exponential-like decay toward *Gaussian decay*, which is often associated with more coherent dynamics and suppressed memoryless noise.
- **The long-time saturation fidelity  $A$**  (not shown in the table but evident from the fitted plots) begins to exceed 0.5 at high pulse counts, often approaching 0.6. This indicates that decoherence is not just delayed, but fundamentally altered—portions of the quantum state remain preserved even in the long-time limit.
- Under white and OU noise, this saturation effect is especially pronounced, with UDD providing remarkable protection at high pulse counts. This is consistent with UDD’s broad high-frequency filtering, which aligns well with the spectral features of these noise types.

These results emphasize UDD’s ability to **not only extend coherence times** but also qualitatively **improve the nature of the decay**, hinting at more robust preservation of quantum information.

#### 6.1.4 CDD Sequence

The results at **CDD level 9** involve  $2^9 - 1 = 511$  pulses, which is comparable in magnitude to the 400–600 pulse range used for CPMG simulations. This explains the similarity in the achieved  $T_{2,\text{eff}}$  and  $\beta$  values between the two protocols at high control complexity. Such comparisons validate that while CDD grows exponentially in the number of pulses with sequence level, its performance becomes comparable to CPMG when matched in total pulse count.

Both CPMG and CDD sequences exhibit a stretch exponent  $\beta \approx 1.7$  at high pulse counts or nesting levels under  $1/f$  noise, indicating that the decay profile transitions

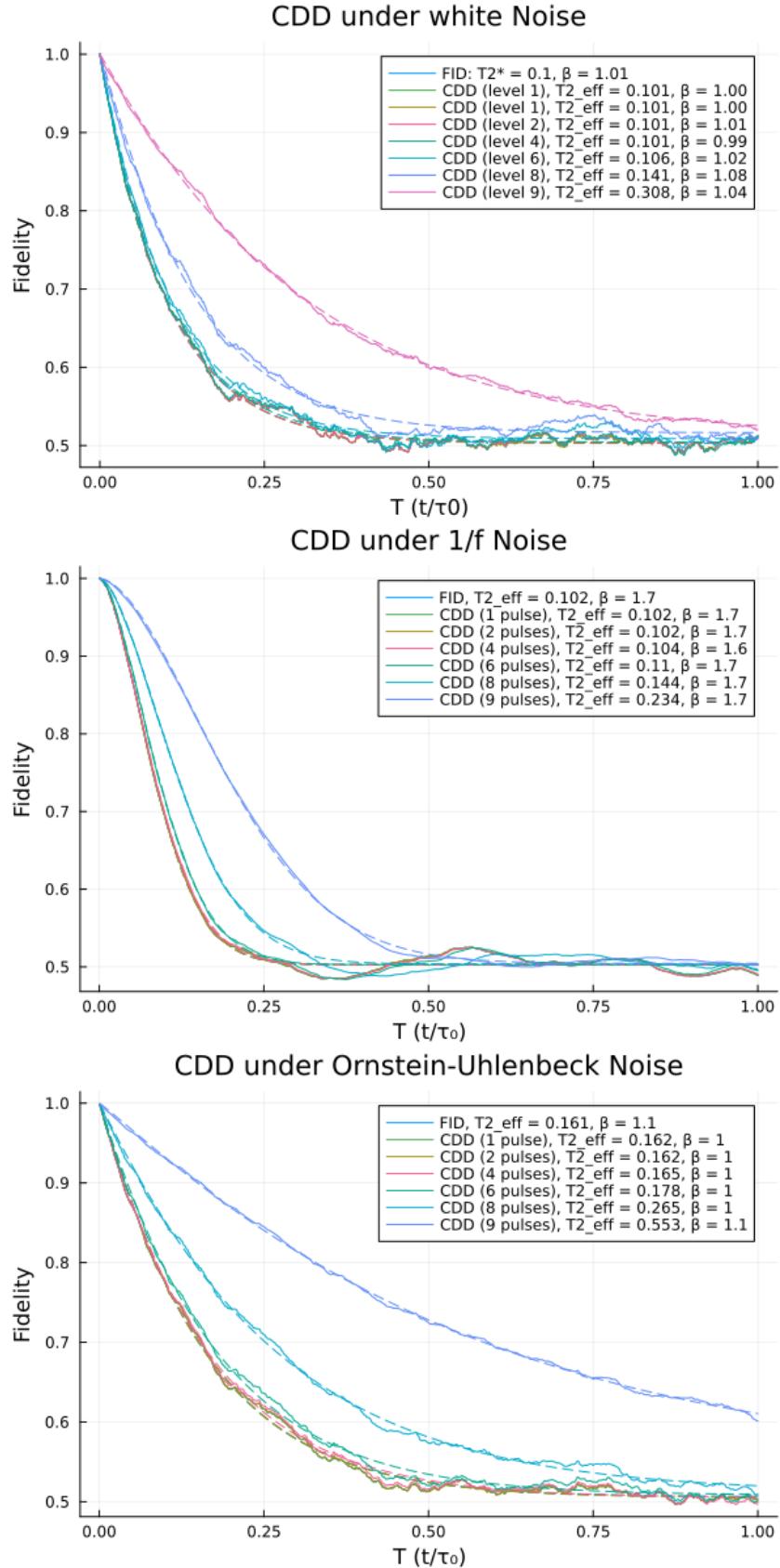


Figure 6.5: Performance of CDD under different noise models: white, 1/f, and OU.

Table 6.3: Comparison of  $T_{2,\text{eff}}/T_2^*$  and stretch exponent  $\beta$  for CDD under different noise models

Level	White Noise		1/f Noise		OU Noise	
	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$
FID	1.00	1.01	1.00	1.67	1.00	1.05
1	1.01	1.00	1.00	1.67	1.01	1.05
2	1.01	1.01	1.00	1.67	1.01	1.05
4	1.01	0.99	1.02	1.65	1.03	1.03
6	1.06	1.02	1.08	1.66	1.10	1.04
8	1.41	1.08	1.41	1.68	1.65	1.05
9	3.08	1.04	2.30	1.74	3.43	1.07

from exponential to nearly Gaussian in both cases. This reflects effective suppression of low-frequency noise components, which dominate the 1/f spectrum.

It appears as if these two DD sequences—although formulated quite differently—result in remarkably similar behavior under all the noises simulated. The similarity in  $\beta$  values suggests comparable filtering characteristics in the relevant spectral band.

### 6.1.5 CPMG under Longitudinal White Noise with Transverse Noise

From Tables 6.5 and 6.6, we observe a consistent trend: the introduction of transverse noise—even when weaker than the longitudinal component—leads to a notable degradation in coherence performance as measured by  $T_{2,\text{eff}}$ . This effect is particularly pronounced for the combinations with transverse 1/f and OU noise, where up to 38% reduction in  $T_{2,\text{eff}}$  is observed at high pulse counts.

Interestingly, the **white+white** case exhibits slightly *better* performance at 400 pulses, with a marginal increase in  $T_{2,\text{eff}}$  over the uniaxial case. This may be attributed to stochastic fluctuations in the realization ensemble or minor constructive interference in the control-noise interplay, though the deviation is small enough that it is not expected to be significant.

At low pulse counts (FID to 200), the degradation is relatively modest, suggesting that CPMG’s filtering capability is still effective in suppressing slow transverse fluctuations. However, as the number of pulses increases and the filter bandwidth widens, transverse noise becomes a more dominant contributor to decoherence.

Overall, these results highlight that while CPMG is effective even in multiaxial noise environments, its performance is not immune to the addition of transverse noise. This underscores the importance of multiaxial noise modeling in dynamical decoupling simulations and the potential need for more robust DD sequences that suppress decoherence across all axes simultaneously.

Table 6.4: Comparison of  $T_{2,\text{eff}}/T_2^*$  and stretch exponent  $\beta$  for CPMG under longitudinal white noise with varied transverse noise

<b>Pulses</b>	<b>Transverse White</b>		<b>Transverse 1/f</b>		<b>Transverse OU</b>	
	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$
FID	1.00	1.00	1.00	1.30	1.00	1.02
100	1.19	0.95	1.13	1.30	1.18	0.96
200	1.35	1.00	1.35	1.30	1.35	1.04
400	2.19	1.10	2.14	1.20	2.18	1.10
600	4.40	1.10	4.33	1.20	4.37	1.11
800	14.4	1.10	21.7	1.10	14.5	1.05

Table 6.5: Comparison of  $T_{2,\text{eff}}$  for CPMG under uniaxial white noise vs white noise combined with transverse components. Only common pulse counts are compared.

<b>Pulses</b>	<b>White Only</b>	<b>White + White</b>	<b>White + 1/f</b>	<b>White + OU</b>
FID	0.103	0.099	0.064	0.097
200	0.140	0.134	0.087	0.132
400	0.211	0.217	0.138	0.213
600	0.484	0.436	0.279	0.425
800	2.27	1.43	1.40	1.41

Table 6.6: Percentage decrease in  $T_{2,\text{eff}}$  compared to uniaxial white noise case for CPMG simulations. Here, W = White, 1/f = 1/f noise, and OU = Ornstein–Uhlenbeck noise.

<b>Pulses</b>	<b>only W</b>	<b>W+W</b>	<b>% ↓</b>	<b>W+1/f</b>	<b>% ↓</b>	<b>W+OU</b>	<b>% ↓</b>
FID	0.103	0.099	3.88% ↓	0.064	37.86% ↓	0.097	5.83% ↓
200	0.140	0.134	4.29% ↓	0.087	37.86% ↓	0.132	5.71% ↓
400	0.211	0.217	-2.84% ↑	0.138	34.60% ↓	0.213	-0.95% ↑
600	0.484	0.436	9.92% ↓	0.279	42.36% ↓	0.425	12.19% ↓
800	2.27	1.43	37.00% ↓	1.40	38.33% ↓	1.41	37.89% ↓

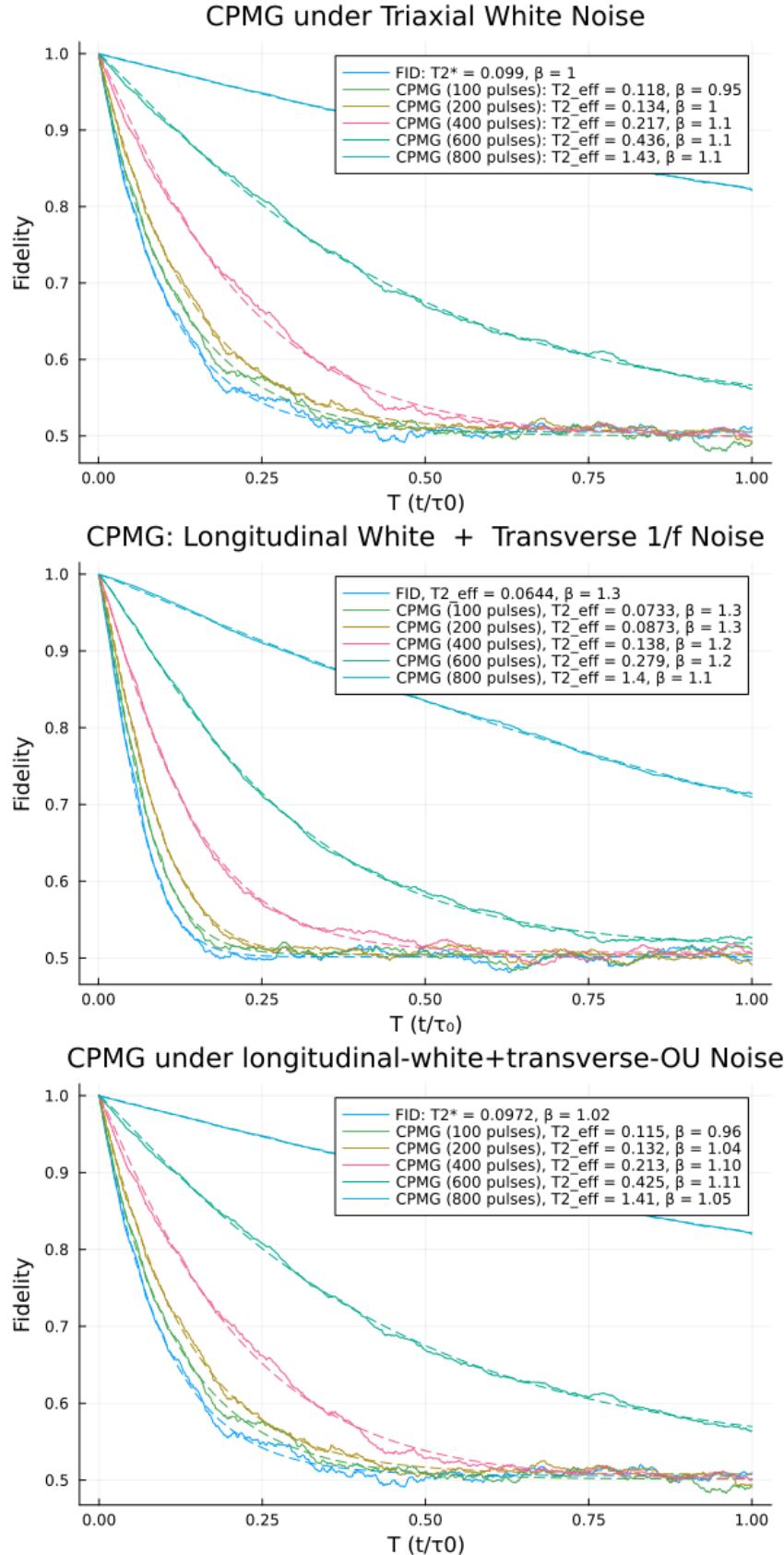


Figure 6.6: CPMG under multiaxial noise with white noise along the longitudinal axis and varying transverse noise models: (Top) white, (Middle)  $1/f$ , (Bottom) Ornstein–Uhlenbeck (OU).

### 6.1.6 CPMG under Longitudinal 1/f Noise with Transverse Noise

In retrospect, the near-identical results across all simulations involving longitudinal 1/*f* noise, regardless of the transverse noise type, suggest that the transverse components were too weak to influence the coherence dynamics. This is likely due to the choice of  $\sigma_x = \sigma_y = \sigma_z/10$ , which rendered the transverse fluctuations nearly negligible compared to the dominant 1/*f* noise along the *z*-axis. Ideally, this effect should have been anticipated earlier, and a more balanced noise strength could have been explored. Nonetheless, this outcome reinforces the conclusion that in the present setup, decoherence is governed almost entirely by the longitudinal 1/*f* noise.

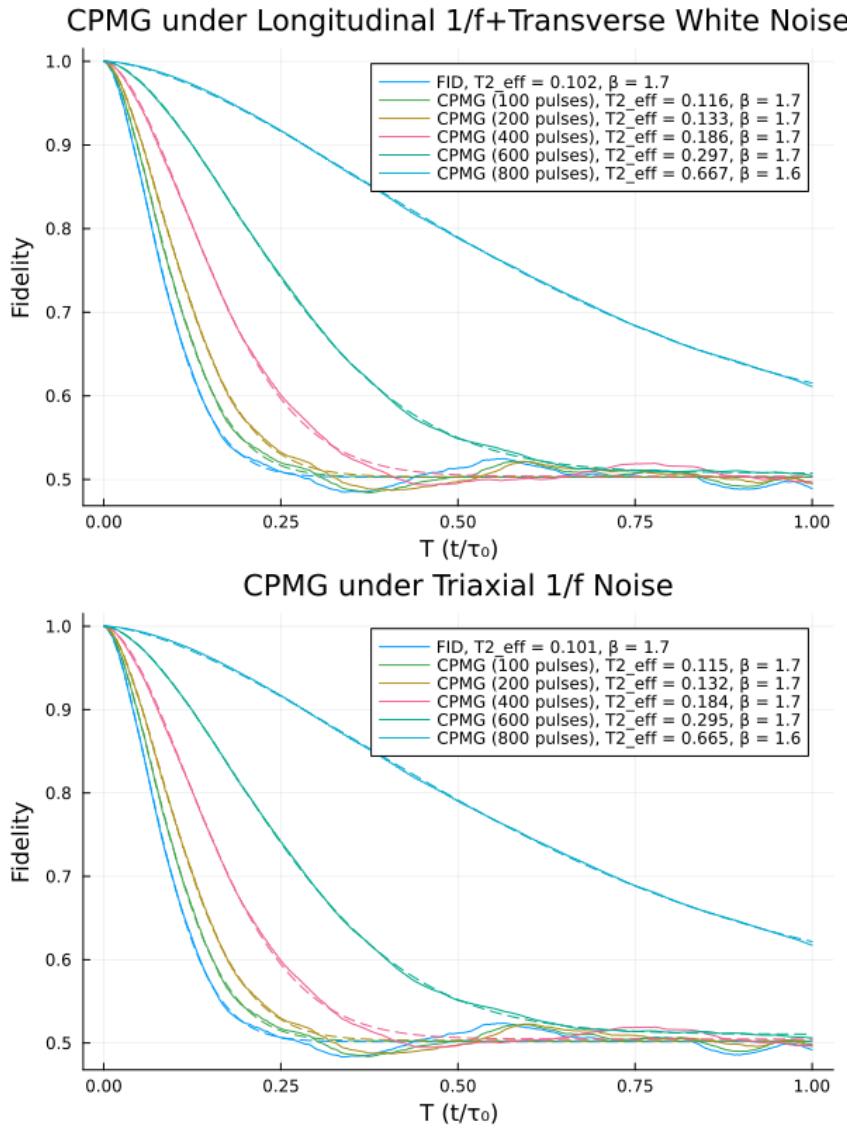


Figure 6.7: CPMG simulations under longitudinal 1/*f* noise combined with different transverse noise models: (top) transverse white, (bottom) transverse 1/*f*,

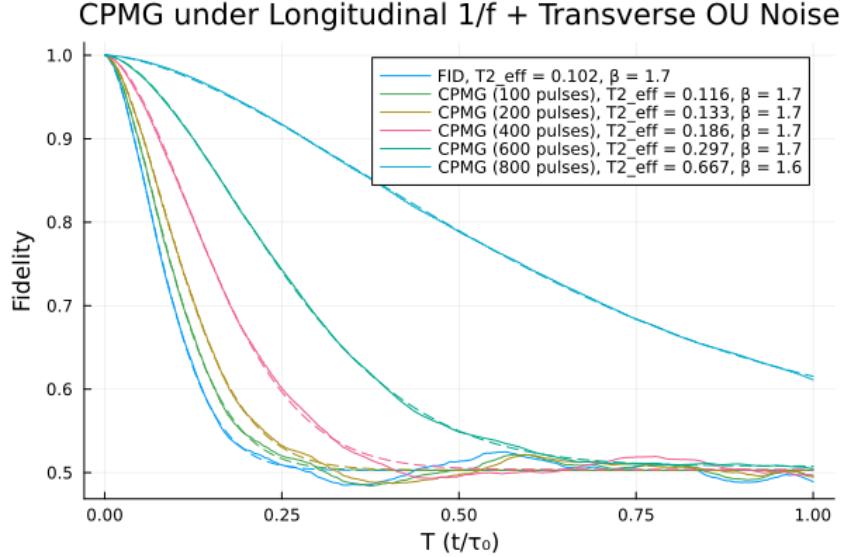


Figure 6.8: CPMG simulations under longitudinal  $1/f$  noise combined with transverse OU noise.

### 6.1.7 CPMG under Longitudinal OU Noise with Transverse Noise

- Transverse Noise Degrades Performance:** Adding transverse noise components (white,  $1/f$ , or OU) to longitudinal OU noise consistently reduces  $T_{2,\text{eff}}$ , particularly at higher pulse counts. This indicates that the efficiency of CPMG is diminished when decoherence arises from multiple spatial directions.
- $1/f$  Transverse Noise is Most Detrimental:** The addition of transverse  $1/f$  noise causes the largest drop in  $T_{2,\text{eff}}$ —over 50% at 800 pulses—suggesting that low-frequency transverse noise evades suppression by CPMG. This aligns with expectations, as  $1/f$  noise has long temporal correlations that are poorly handled by sequences designed for white or weakly correlated noise.

Table 6.7: Comparison of  $T_{2,\text{eff}}/T_2^*$  and stretch exponent  $\beta$  for CPMG under longitudinal OU noise with different transverse noise components.

Pulses	OU + White		OU + $1/f$		OU + OU	
	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$	$T_{2,\text{eff}}/T_2^*$	$\beta$
FID	1	1.05	1	1.21	1	1.05
100	1.23	1.01	1.18	1.18	1.24	1.01
200	1.42	1.07	1.42	1.21	1.42	1.07
400	2.43	1.10	2.29	1.21	2.44	1.10
600	4.63	1.10	4.76	1.14	4.68	1.09
800	9.44	1.09	15.12	1.11	9.62	1.09

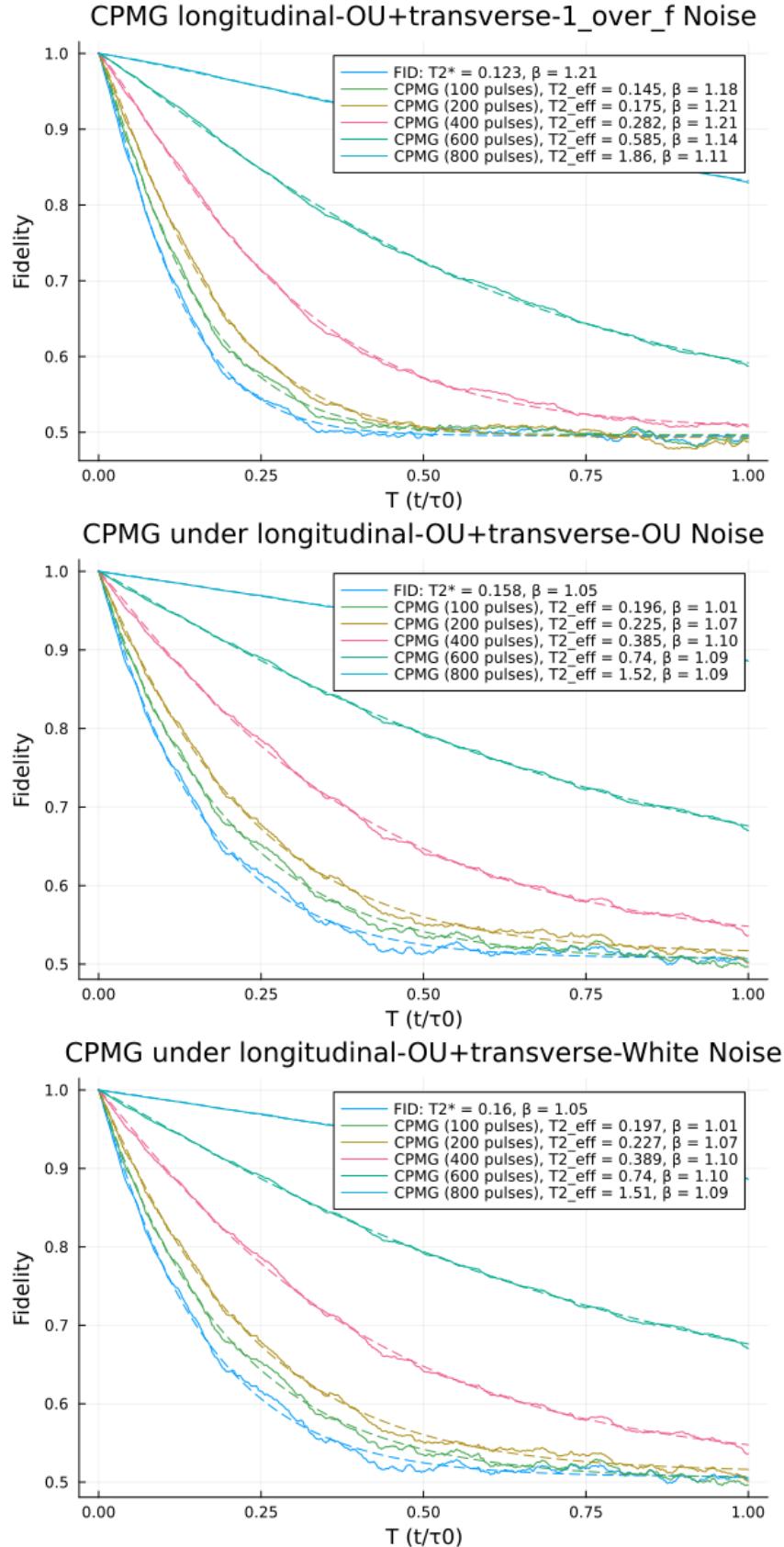


Figure 6.9: CPMG performance under longitudinal OU noise combined with transverse white noise (top), transverse  $1/f$  noise (middle), and transverse OU noise (bottom).

Table 6.8: Percentage decrease in  $T_{2,\text{eff}}$  compared to uniaxial OU noise case for CPMG simulations. Transverse noise models include white,  $1/f$ , and OU.

Pulses	OU Only	OU+W	% ↓	OU+1/f	% ↓	OU+OU	% ↓
FID	0.165	0.160	3.03% ↓	0.123	25.45% ↓	0.158	4.24% ↓
200	0.241	0.227	5.81% ↓	0.175	27.38% ↓	0.225	6.64% ↓
400	0.419	0.389	7.16% ↓	0.282	32.69% ↓	0.385	8.11% ↓
600	0.951	0.740	22.18% ↓	0.585	38.48% ↓	0.740	22.18% ↓
800	3.85	1.51	60.78% ↓	1.86	51.69% ↓	1.52	60.52% ↓

- **White and OU Transverse Noise Have Comparable Impact:** The degradation in performance due to transverse white and OU noise is more modest compared to  $1/f$ , and their effects are nearly identical at high pulse counts. This suggests that CPMG can handle uncorrelated or weakly correlated transverse noise to some extent, but struggles as correlations grow.
- **Suppression Weakens with Multiaxial Complexity:** As the spatial complexity of the noise increases, the suppression offered by CPMG diminishes. This highlights a limitation of single-axis dynamical decoupling sequences in multiaxial noise environments and motivates the need for sequences such as the XY-family.
- **No Improvement in Stretch Exponent  $\beta$ :** Unlike in pure dephasing scenarios where  $\beta$  increases with pulse count, the inclusion of transverse components does not lead to significant changes in  $\beta$ . In many cases,  $\beta$  remains close to 1, suggesting that the decoherence remains exponential or becomes more Markovian in nature.

In summary, while CPMG provides strong protection under uniaxial OU noise, its effectiveness drops significantly when multiaxial noise is present—especially when low-frequency transverse components dominate. This behavior underscores the need for more sophisticated decoupling strategies in realistic, anisotropic noise environments.

### 6.1.8 Final Remarks on Results and Limitations

With the above analyses, all simulation results currently available have been presented. While I had originally intended to extend the multiaxial noise analysis to other DD sequences like CDD and UDD, time constraints limited the scope to CPMG alone. Additionally, a direct comparison between the simulated coherence decay and the theoretical predictions obtained from the filter function formalism could not be completed in the final report.

Nonetheless, this comparison was performed during the preliminary stages for selected cases, and the simulated results were found to be in close agreement with the theoretical predictions, with only minor numerical discrepancies. This preliminary validation, although not reported here, supports the correctness of the simulation framework and builds confidence in the trends observed throughout the study.

Although not exhaustive, the results provide clear insights into the performance of CPMG under varied noise environments, both uniaxial and multiaxial, and lay the foundation for future work that incorporates theoretical validation more comprehensively and explores

other DD sequences.

## 6.2 Conclusion

In this report, a comprehensive simulation-based analysis was conducted to investigate the effectiveness of dynamical decoupling (DD) sequences, with a primary focus on the CPMG sequence, under various classical dephasing noise models. The study considered both uniaxial and multiaxial noise environments, incorporating white noise,  $1/f$  noise, and Ornstein–Uhlenbeck (OU) processes in different combinations along the  $x$ ,  $y$ , and  $z$  axes.

The results demonstrate that CPMG is highly effective in mitigating uniaxial white and OU noise, with  $T_{2,\text{eff}}$  showing significant improvements as pulse count increases. However, in the presence of  $1/f$  noise, which contains low-frequency components that are harder to suppress, the improvement in coherence time is more modest. The stretch exponent  $\beta$  values in these cases also reflect a more complex non-exponential decay behavior, characteristic of slow noise.

In multiaxial scenarios, the addition of transverse noise components revealed nuanced impacts on  $T_{2,\text{eff}}$ . When the longitudinal noise was white, the influence of added transverse noise was clearly visible, particularly for transverse  $1/f$  and OU noise, which caused a notable reduction in coherence time. In contrast, when the longitudinal noise itself was  $1/f$ , the addition of transverse components had a negligible impact—likely due to the dominance of the  $z$ -axis noise and the relatively low strength assigned to the  $x$  and  $y$  noise components.

While multiaxial noise analysis was restricted to the CPMG sequence due to time constraints, simulations were in fact carried out for all considered DD sequences—CPMG, UDD, and CDD—under uniaxial noise models. This broader coverage enabled a more complete understanding of how different DD strategies perform in idealized single-axis noise scenarios. A preliminary validation of the simulation framework against theoretical coherence decay curves derived from the filter function formalism (in the uniaxial case) further supports the accuracy and reliability of the results obtained, even though these comparisons were not formally included in the report.

**In summary**, the simulation study provides strong empirical evidence supporting the practical utility of DD sequences like CPMG in real-world qubit environments, particularly when tailored to the spectral properties of the noise. The insights gained here can inform more effective design of noise-resilient qubit control strategies in quantum computing experiments.

## Scope for Future Work

This project focused on simulating coherence decay in single-qubit systems under classical, uncorrelated, and isotropic noise using various dynamical decoupling (DD) sequences. While the framework developed here provides a robust foundation, several natural and important extensions remain:

- **Multiaxial Control Optimization:** One promising direction is to apply different DD sequences along distinct axes (e.g., CPMG along  $x$ , UDD along  $y$ ) to explore whether this improves suppression of anisotropic noise.
- **Noise Spectroscopy Inversion:** Although this project used known spectral densities as simulation input, implementing the full inversion procedure — reconstructing  $S(\omega)$  from simulated coherence decay — would complete the loop and test the feasibility of practical noise spectroscopy.
- **Advanced Multiaxial Sequences:** Implementing and comparing advanced sequences such as KDD, QDD, and adaptive XY-based protocols under multiaxial noise models could reveal performance gains over basic XY sequences.
- **Experimental Implementation:** Ultimately, validating these simulations against real quantum hardware would test both the assumptions and accuracy of the noise models and filter-function approach.

These extensions would deepen the understanding of dynamical decoupling performance under realistic noise environments and potentially guide the design of more effective control protocols for near-term quantum devices.