ReadSpyn: Quantum Dot Readout Simulator

Theoretical Foundations and Model Assumptions

Jan A. Krzywda

August 31, 2025

Abstract

This document provides a comprehensive explanation of the ReadSpyn package, a simulator for quantum dot readout systems with realistic noise models and RLC resonator sensors. We detail the theoretical foundations, physical assumptions, and mathematical models underlying the simulation framework, including quantum dot systems, RLC resonator circuits, noise processes, and signal processing techniques.

Contents

1 Introduction

ReadSpyn is a comprehensive simulator for quantum dot readout systems that enables researchers to simulate and analyze the performance of quantum dot readout systems under various noise conditions. The package models realistic quantum dot systems with capacitive coupling, RLC resonator-based sensors, and advanced noise models including Ornstein-Uhlenbeck processes, 1/f noise, and telegraph noise.

2 Quantum Dot System Model

2.1 Physical Description

Quantum dots are nanoscale semiconductor structures that confine electrons in all three spatial dimensions, creating discrete energy levels. In the context of quantum computing, quantum dots serve as qubits where the charge state (number of electrons) or spin state can encode quantum information.

2.2 Capacitive Coupling Model

The ReadSpyn package models quantum dot systems using a capacitive coupling approach, where the interaction between dots and sensors is described by capacitance matrices.

2.2.1 Dot-Dot Capacitance Matrix (C_{dd})

The dot-dot capacitance matrix C_{dd} describes the mutual capacitive coupling between quantum dots:

$$C_{dd} = \begin{pmatrix} C_{11} & C_{12} & \cdots & C_{1N} \\ C_{21} & C_{22} & \cdots & C_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ C_{N1} & C_{N2} & \cdots & C_{NN} \end{pmatrix} \tag{1}$$

where C_{ii} represents the self-capacitance of dot i and C_{ij} (for $i \neq j$) represents the mutual capacitance between dots i and j.

2.2.2 Dot-Sensor Capacitance Matrix (C_{ds})

The dot-sensor capacitance matrix C_{ds} describes the coupling between quantum dots and readout sensors:

$$C_{ds} = \begin{pmatrix} C_{1s_1} & C_{1s_2} & \cdots & C_{1s_M} \\ C_{2s_1} & C_{2s_2} & \cdots & C_{2s_M} \\ \vdots & \vdots & \ddots & \vdots \\ C_{Ns_1} & C_{Ns_2} & \cdots & C_{Ns_M} \end{pmatrix} \tag{2}$$

where C_{is_j} represents the capacitance between dot i and sensor j.

2.3 Energy Calculation

The energy of each quantum dot is calculated using the inverse capacitance matrix:

$$E_i = \sum_{j=1}^{N} (C_{dd}^{-1})_{ij} \left(Q_j + \sum_{k=1}^{M} C_{ds,jk} V_{s_k} \right) + \epsilon_0 \tag{3}$$

where:

- Q_j is the charge state of dot j
- V_{s_k} is the voltage applied to sensor k
- ϵ_0 is a common gate voltage offset

2.4 Energy Offset for Sensors

The energy offset for each sensor, which determines the conductance through the sensor, is calculated as:

$$\epsilon_{sensor} = \sum_{i=1}^{N} \sum_{j=1}^{N} C^T_{ds,ji} (C^{-1}_{dd})_{ij} \left(Q_j + \sum_{k=1}^{M} C_{ds,jk} V_{s_k} \right) + \epsilon_0 \tag{4}$$

3 RLC Resonator Sensor Model

3.1 Circuit Description

The RLC resonator sensor consists of an inductor (L_c), parasitic capacitance (C_p), load resistance (R_L), and coupling resistance (R_c) connected to a quantum dot system through a conductance $G_s(t)$.

3.2 Circuit Equations

The RLC circuit is described by a system of ordinary differential equations:

$$\frac{dV_{C_p}}{dt} = \frac{I_L - V_{C_p} G_s(t)}{C_p(t)} \tag{5}$$

$$\frac{dI_L}{dt} = \frac{V_s(t) - R_L I_L - V_A}{L_c} \tag{6}$$

where:

- V_{C_p} is the voltage across the parasitic capacitance
- I_L is the current through the inductor
- $V_s(t)$ is the source voltage (typically sinusoidal)
- V_A is the voltage at node A (junction between R_c and G_s)

3.3 Conductance Model

The conductance $G_s(t)$ through the quantum dot system is modeled using a Fermi-Dirac-like function:

$$G_s(t) = \frac{2}{\cosh^2(2\epsilon(t)/\epsilon_w)} \cdot \frac{1}{R_0} \tag{7}$$

where:

- $\epsilon(t)$ is the time-dependent energy offset
- ϵ_w is the energy width of the Coulomb peak
- R_0 is the base resistance

3.4 Resonant Frequency

The resonant frequency of the RLC circuit is given by:

$$\omega_0 = \frac{1}{\sqrt{L_c C_{total}}} \tag{8}$$

where $C_{total} = C_p + C_{self}$ includes both parasitic and self-capacitances.

4 Noise Models

4.1 Ornstein-Uhlenbeck (OU) Noise

The OU noise model implements a continuous-time Markov process with exponential autocorrelation:

$$dx = -\gamma x dt + \sigma \sqrt{2\gamma} dW \tag{9}$$

where:

- γ is the correlation rate (Hz)
- σ is the noise amplitude
- dW is a Wiener process increment

The autocorrelation function is:

$$\langle x(t)x(t+\tau)\rangle = \sigma^2 e^{-\gamma|\tau|} \tag{10}$$

4.2 1/f Noise Model

The 1/f noise is implemented using multiple fluctuators with different switching rates:

$$S(f) = \frac{S_1}{f} \tag{11}$$

where S_1 is the 1/f noise amplitude. The total noise is constructed by summing contributions from individual fluctuators with log-uniformly distributed frequencies.

4.3 Telegraph Noise

Telegraph noise models two-level fluctuators that randomly switch between two states:

$$P(\text{switch in } \Delta t) = \frac{1}{2} - \frac{1}{2} e^{-2\gamma \Delta t} \tag{12}$$

where γ is the switching rate.

5 Signal Processing

5.1 IQ Demodulation

The reflected signal is processed using IQ demodulation:

$$V_{refl}^{phasor}(t) = \mathcal{H}[V_{refl}(t)] \cdot e^{-j\omega_0 t} \tag{13}$$

where $\mathcal{H}[\cdot]$ is the Hilbert transform. The in-phase (I) and quadrature (Q) components are:

$$I(t) = \mathrm{Re}[V_{refl}^{phasor}(t)] \tag{14}$$

$$Q(t) = \mathrm{Im}[V_{refl}^{phasor}(t)] \tag{15}$$

5.2 Signal-to-Noise Ratio

The meaningful SNR is calculated based on the conductance difference between charge states:

$$\mathrm{SNR} = \frac{|G(\mathrm{state}_1) - G(\mathrm{state}_2)|}{\mathrm{mean}(G(\mathrm{state}_1), G(\mathrm{state}_2))} \tag{16}$$

6 Model Assumptions

6.1 Quantum Dot Assumptions

1. **Classical Charge States**: The model assumes classical charge states rather than quantum superposition states.
2. **Capacitive Coupling**: Interactions are purely capacitive, neglecting quantum tunneling effects.
3. **Linear Response**: The system operates in the linear response regime.
4. **Constant Capacitances**: Capacitance matrices are assumed to be constant in time.
5. **No Spin Effects**: The model does not include spin-related effects or spin-charge coupling.

6.2 RLC Circuit Assumptions

1. **Lumped Element Model**: The circuit is modeled using lumped elements rather than distributed parameters.
2. **Linear Components**: All circuit components (L, C, R) are assumed to be linear.
3. **No Parasitic Inductance**: Parasitic inductance effects are neglected.
4. **Constant Resonant Frequency**: The resonant frequency is assumed to be constant during readout.
5. **Perfect Matching**: The circuit is assumed to be perfectly matched to the transmission line.

6.3 Noise Assumptions

1. **Stationary Processes**: All noise processes are assumed to be stationary.
2. **Gaussian Statistics**: OU noise assumes Gaussian statistics.
3. **Independent Fluctuators**: In 1/f noise, individual fluctuators are assumed to be independent.
4. **Markovian Processes**: OU and telegraph noise are assumed to be Markovian.
5. **No Cross-Correlations**: Different noise sources are assumed to be uncorrelated.

6.4 Signal Processing Assumptions

1. **Linear Demodulation**: IQ demodulation is assumed to be linear.
2. **Perfect Phase Reference**: The demodulation assumes perfect phase reference.
3. **No Aliasing**: Sampling is assumed to be sufficient to avoid aliasing.
4. **Stationary Statistics**: Signal statistics are assumed to be stationary during readout.

7 Implementation Details

7.1 Time Integration

The RLC circuit equations are solved using the Radau method from SciPy's `solve_ivp` function, which is suitable for stiff differential equations.

7.2 Noise Generation

Noise trajectories are pre-generated for efficiency, using vectorized operations where possible. The Numba JIT compiler is used to optimize performance-critical functions.

7.3 Memory Management

The simulator uses efficient data structures to minimize memory usage, especially important for long simulations with multiple sensors and charge states.

8 Limitations and Future Improvements

8.1 Current Limitations

1. **No Quantum Effects**: The model does not include quantum tunneling or quantum interference effects.
2. **Single-Particle Picture**: The model assumes single-particle physics rather than many-body effects.
3. **No Temperature Effects**: Temperature-dependent effects are not included.
4. **Simplified Noise Models**: Real quantum dot systems may exhibit more complex noise characteristics.
5. **No Feedback Effects**: The model does not include feedback from the readout on the quantum dot system.

8.2 Potential Improvements

1. **Quantum Tunneling**: Include quantum tunneling effects in the conductance model.
2. **Temperature Dependence**: Add temperature-dependent effects to noise models.
3. **Many-Body Effects**: Include electron-electron interactions and correlation effects.
4. **Advanced Noise Models**: Implement more sophisticated noise models including cross-correlations.
5. **Feedback Effects**: Include feedback effects from readout on the quantum dot system.

6. **Distributed Circuit Models**: Use distributed circuit models for more accurate high-frequency behavior.

9 Conclusion

The ReadSpyn package provides a comprehensive framework for simulating quantum dot readout systems with realistic noise models and RLC resonator sensors. While the current model makes several simplifying assumptions, it captures the essential physics of quantum dot readout and provides a solid foundation for understanding and optimizing readout performance.

The modular design of the package allows for easy extension and modification, making it suitable for both educational purposes and research applications. Future improvements can address the limitations identified in this document, leading to more accurate and comprehensive simulations of quantum dot readout systems.

A Parameter Ranges

Typical parameter ranges used in ReadSpyn simulations:

Parameter	Symbol	Typical Range	Units
Inductance	L_c	10^{-9} - 10^{-6}	H
Capacitance	C_p	10^{-15} - 10^{-12}	F
Load Resistance	R_L	10 - 100	Ω
Coupling Resistance	R_c	10^{6} - 10^{8}	Ω
Energy Width	ϵ_w	10^{-4} - 10^{-3}	eV
Base Resistance	R_0	10^{4} - 10^{6}	Ω
Resonant Frequency	f_0	10^{6} - 10^{9}	Hz

Table 1: Typical parameter ranges for RLC resonator sensors

B Noise Parameters

Noise Type	Parameter	Typical Range	Units
OU Noise	σ	10^{-15} - 10^{-12}	eV
OU Noise	γ	10^{3} - 10^{6}	Hz
1/f Noise	S_1	10^{-6} - 10^{-3}	eV^2/Hz
Telegraph	σ	10^{-12} - 10^{-9}	eV
Telegraph	γ	10^{4} - 10^{7}	Hz

Table 2: Typical noise parameter ranges