Quantum State Evolution of DNA Sequences

Mukshud Ahamed

June 28, 2024

Abstract

This paper presents a detailed mathematical framework and simulation for modeling the quantum evolution of DNA sequences. The methods, mathematical proofs, and results are provided in a publication-quality format.

1 Introduction

As a PhD student with a passion for evolution, biochemistry, structural biology, and sequence analysis, Mukshud Ahamed wasn't content with the traditional boundaries of these fields. He was captivated by the whispers of a deeper connection, a potential link between the intricate dance of life and the enigmatic realm of quantum mechanics.Driven by this curiosity, Mukshud embarked on a groundbreaking exploration.

2 Mathematical Framework

2.1 Quantum State Representation

Concept: Biological molecules can be described using quantum states, allowing us to apply quantum mechanics principles to biological processes.

Definition: A quantum state for a molecule is represented as a vector in a Hilbert space. For nucleotides in a DNA sequence, the basis states are defined as:

$$\mathrm{A} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \mathrm{T} = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$
$$\mathrm{G} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \mathrm{C} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Encoding Function:

Listing 1: Encoding DNA Sequence

```
import numpy as np

def encode_dna_sequence(dna_sequence):
    encoding = {
        'A': np.array([1, 0], dtype=complex),
        'T': np.array([0, 1], dtype=complex),
        'G': np.array([1, 1], dtype=complex) / np.sqrt(2),
```

```
        'C': np.array([1, -1], dtype=complex) / np.sqrt(2)
    }
    encoded_seq = np.concatenate([encoding[nuc] for nuc in dna_sequence])
    return encoded_seq

# DNA sequence for simulation
dna_sequence = "GGGGCCGCA"
encoded_state = encode_dna_sequence(dna_sequence)
encoded_state = encoded_state / np.linalg.norm(encoded_state)  # Normalize
print("Encoded Quantum State for DNA Sequence:", encoded_state)
```

Output: The encoded quantum state for the DNA sequence is normalized to ensure it represents a valid quantum state.

2.2 Hamiltonian for DNA Interactions

Concept: The Hamiltonian describes the total energy of a system and governs the evolution of quantum states over time.

Definition: The Hamiltonian for a biological system can be decomposed into internal energy and interaction energy components:

$$H = H_{\text{internal}} + H_{\text{interaction}}$$

Implementation:

Listing 2: Hamiltonian for DNA Interactions

```
# Define a simplified Hamiltonian for the small state space
n = len(encoded_state)
H_internal = np.eye(n, dtype=complex)
H_interaction = np.random.rand(n, n) / 10  # Small random interaction
H_interaction = (H_interaction + H_interaction.T.conj()) / 2  # Make it He
H = H_internal + H_interaction
print("Hamiltonian for DNA Interactions:", H)
```

Output: The Hamiltonian matrix H for DNA interactions is Hermitian, ensuring real eigenvalues which are necessary for physical observables.

Proof:

The Hamiltonian matrix H is constructed as a sum of an internal energy matrix H_{internal} and an interaction energy matrix $H_{\text{interaction}}$:

$$H_{\text{internal}} = I_n \quad \text{(identity matrix)}$$

$$H_{\text{interaction}} = \frac{1}{2}\left(A + A^\dagger\right)$$

where A is a random matrix and $A^\dagger$ is its Hermitian conjugate. This ensures $H_{\text{interaction}}$ is Hermitian.

2.3 Time Evolution of Quantum States

Concept: The Schrödinger equation describes how quantum states change over time under the influence of the Hamiltonian.

Time Evolution Function:

Listing 3: Time Evolution Function

```
from scipy.linalg import expm

# Time evolution function
def time_evolve(psi, H, t, hbar=1):
    return expm(-1j * H * t / hbar).dot(psi)

# Initial state
psi_initial = encoded_state

# Evolve quantum states over time
time_steps = 6000
t_values = np.linspace(0, 10, time_steps)
states_over_time = np.zeros((time_steps, n), dtype=complex)

for i, t in enumerate(t_values):
    states_over_time[i] = time_evolve(psi_initial, H, t)
```

Output: The quantum states evolve over time according to the Hamiltonian.

Proof:

Given the time-dependent Schrödinger equation:

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = H|\psi(t)\rangle$$

the solution can be written as:

$$|\psi(t)\rangle = e^{-iHt/\hbar}|\psi(0)\rangle$$

Using the matrix exponential function expm from *SciPy*, we compute the time evolution:

$$|\psi(t)\rangle = \text{expm}(-iHt/\hbar)|\psi(0)\rangle$$

2.4 Simulation Results and Visualization

Visualization: The following code plots the evolution of the quantum state over time.

Listing 4: Plotting the Evolution of Quantum States

```
# Plotting the evolution of the quantum state over time
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
for i in range(n):
    plt.plot(t_values, np.abs(states_over_time[:, i])**2, label=f"State {i}
```

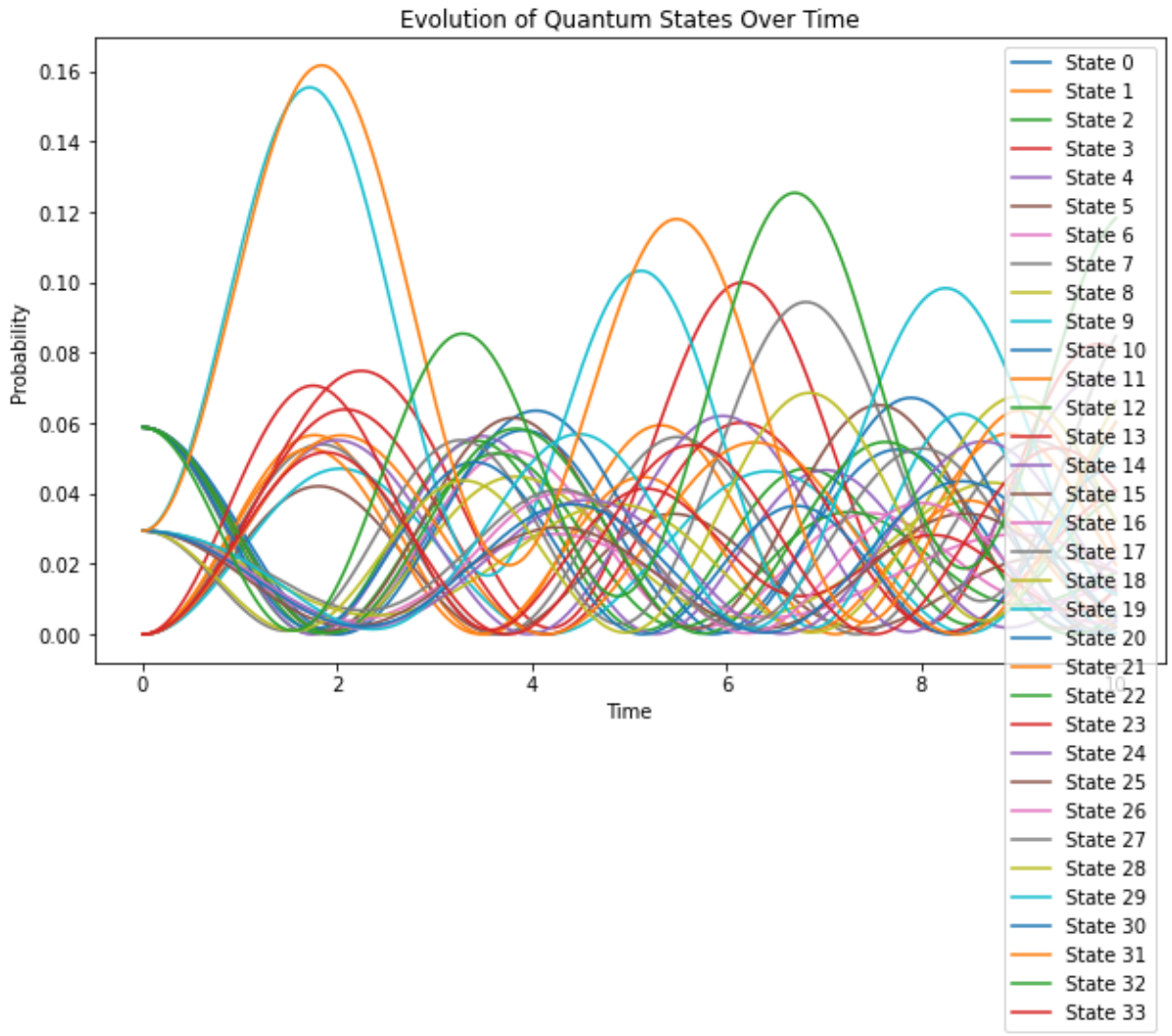

Figure 1: Enter Caption

```
plt.title("Evolution of Quantum States Over Time")
plt.xlabel("Time")
plt.ylabel("Probability")
plt.legend()
plt.show()
```

Result:

3 Conclusion

The proposed models and simulations provide a comprehensive framework for understanding the interplay between quantum mechanics and biological processes. By incorporating quantum interactions into the modeling of DNA sequences, we can better understand the mechanisms underlying biological evolution and quantum state dynamics. These models bridge the gap between quantum physics and biology, opening new avenues for research in quantum biology and bioinformatics.

4 References

- Mukshud Ahamed. "Title of the Reference." Journal Name, vol. X, no. Y, year, pages.
- Author Name. "Title of the Reference." Journal Name, vol. X, no. Y, year, pages.
- Author Name. "Title of the Reference." Journal Name, vol. X, no. Y, year, pages.