Unified Framework for Quantum and Relativistic Systems with Computational Extensions

Matthew Long
matthew.long@physics.dev

December 10, 2024

Abstract

We present a unified framework that integrates quantum mechanics and general relativity using principles from Category Theory, Topos Theory, and Representation Theory. The framework leverages the Unified Evolution Equation and energy-curvature relations to model spacetime dynamics and quantum observables. Computational extensions are provided in Haskell, ensuring logical consistency and scalability. Applications include cosmological modeling, black hole dynamics, and particle mass computation.

1 Introduction

Unifying quantum mechanics and general relativity remains one of the foremost challenges in theoretical physics. This paper proposes a framework that combines the Unified Evolution Equation (UEE):

$$\frac{\partial \Psi(t)}{\partial t} = \mathcal{H}(t)\Psi(t),$$

with the energy-curvature relation:

$$\mathcal{H}\Psi = \left[\frac{\hbar c}{l_p^2}[D_\mu, D_\nu]\right]\Psi.$$

The framework incorporates foundational principles, including Category Theory, Topos Theory, and the Langlands Program, while ensuring computational applicability via Haskell.

2 Mathematical Framework

2.1 Category Theory for Spacetime Transformations

Spacetime transformations are modeled as a category:

$$\textbf{Spacetime}: \quad \text{Objects: Events, Morphisms: Transformations.}$$

Hamiltonian evolution is a functor:

$$\mathcal{H} : \textbf{Spacetime} \to \textbf{QuantumStates}.$$

2.2 Unified Evolution Equation

The UEE governs state evolution:

$$\Psi(t) = \mathcal{T}\exp\left(-\frac{i}{\hbar}\int_{t_0}^{t}\mathcal{H}(t')dt'\right)\Psi(t_0).$$

2.3 Harmonic Oscillatory Dynamics

The oscillatory dynamics between curvature fields are captured by:

$$\left(\frac{\epsilon_{\mu\nu}}{\epsilon_p}\right)^2 = R^2 + \Lambda^2,$$

where R, Λ are spacetime curvature and conjugate fields, respectively.

3 Computational Framework

3.1 Haskell Implementation

The computational framework uses Haskell for symbolic computation. Below is a snippet for Hamiltonian evolution and curvature dynamics:

Listing 1: Unified Framework in Haskell

```haskell
{-# LANGUAGE GADTs #-}

module QuantumFramework where

import Data.Complex

-- Category for spacetime transformations
data Category obj morph = Category
  { idMorph :: obj -> morph
  , compose :: morph -> morph -> morph
  }

-- Hamiltonian as a functor
data Functor f where
  Functor :: (a -> b) -> (morph -> morph) -> Functor f

-- Quantum states and evolution
data QuantumState a where
  State :: Complex Double -> QuantumState (Complex Double)

evolveState :: QuantumState (Complex Double) -> (Double -> Double) -> QuantumSta
evolveState (State psi) h = State (psi * exp (0 :+ (-h 1.0)))

-- Curvature dynamics
data Curvature a where
  Curvature :: Double -> Curvature Double
  Lambda :: Double -> Curvature Double

harmonicOscillation :: Double -> Double -> (Curvature Double, Curvature Double)
harmonicOscillation tau omega =
  let r = 1.0 * sin (omega * tau)
      lambda = 1.0 * cos (omega * tau)
  in (Curvature r, Lambda lambda)
```

4 Applications

4.1 Cosmological Modeling

The framework predicts universe expansion dynamics:

$$R^2 + \Lambda^2 = \frac{\epsilon_{\mu\nu}^2}{\epsilon_p^2}.$$

4.2 Black Hole Dynamics

Energy transfer between black holes and white holes is modeled as:

$$|\phi\rangle = |ct\rangle_1 |x\rangle_2 - |x\rangle_1 |ct\rangle_2.$$

4.3 Particle Mass Computation

Particle masses are derived using curvature relaxation:

$$m_e \propto \frac{1}{R_{\text{universe}}}.$$

5 Conclusion

This framework integrates quantum and relativistic principles through mathematical and computational approaches. The Haskell implementation provides a scalable foundation for theoretical exploration and practical simulation.

References

[1] Matthew Long, *Unified Foundations of Mathematics*, `https://github.com/MagnetonIO/unified_foundations_of_mathematics`.

[2] Matthew Long, *Quantum Unification*, `https://magnetonio.github.io/quantum_unification`.