Phase 10 – Part 1  
Linearization of ψ-Equations Around Stationary ψ Wells

Goal  
The purpose of this part is to establish the linearized framework of ψ-gravity around equilibrium states. By doing so, I can prepare the ground for deriving dispersion relations in the following parts. The method is to expand ψ near a stationary solution (a ψ well), introduce small perturbations, and study how those perturbations evolve under the governing dynamics.

Core Equation  
The upgraded ψ-gravity equation remains the anchor:

Plain text:  
Gravity(x, t) = (∇²[space(x) + current(x)²]) × ψ(x, t)

The force felt by a particle or fluctuation is:

Plain text:  
Force(x, t) = −∇[Gravity(x, t)]

Stationary ψ Well  
Consider a localized equilibrium profile, a “ψ well”:

Plain text:  
ψ₀(x)

This represents the stationary desert floor, shaped by the balance of space(x) and current(x). No net temporal evolution occurs for ψ₀(x) itself.

Perturbation Expansion  
Introduce a small perturbation:

Plain text:  
ψ(x, t) = ψ₀(x) + ε φ(x, t)

where ε ≪ 1, and φ(x, t) is the perturbation field.

Gravity then becomes:

Plain text:  
Gravity(x, t) = (∇²[space(x) + current(x)²]) (ψ₀(x) + ε φ(x, t))

Linearized Form  
Expand to first order in ε:

Plain text:  
Gravity(x, t) ≈ (∇²[space(x) + current(x)²]) ψ₀(x) + ε (∇²[space(x) + current(x)²]) φ(x, t)

The zeroth-order term corresponds to the equilibrium well.  
The first-order term governs perturbation dynamics.

Effective Perturbation Equation  
Define the curvature factor:

Plain text:  
C(x) = ∇²[space(x) + current(x)²]

Then:

Plain text:  
Gravity(x, t) ≈ C(x) ψ₀(x) + ε C(x) φ(x, t)

The force is:

Plain text:  
Force(x, t) ≈ −∇(C(x) ψ₀(x)) − ε ∇(C(x) φ(x, t))

Perturbation Dynamics  
Since test-particle motion follows Newtonian-like rules (acceleration = force), the perturbation field φ should evolve according to a wave-like equation. For simplicity, assume a homogeneous region where C(x) is approximately constant:

Plain text:  
C(x) ≈ C₀

Then:

Plain text:  
Force(x, t) ≈ −ε C₀ ∇φ(x, t)

This suggests that the perturbation behaves as a propagating mode modulated by C₀.

Ansatz for Perturbations  
Assume φ is wave-like:

Plain text:  
φ(x, t) = exp(i (k·x − ω t))

Substituting this ansatz in the linearized framework will yield dispersion relations, to be developed in Part 2.

Desert Analogy Translation

* The stationary ψ well (ψ₀) is the desert floor carved out by wind and sand balance.
* The perturbation (φ) is like ripples forming on the desert floor.
* The curvature factor (C(x)) is the way sand and wind interact locally.
* Linearization tells me how tiny ripples evolve without collapsing the entire dune structure.

Python Setup (Perturbation Linearization Test)

# simulations/phase10\_part1\_linearization.py  
import numpy as np  
import matplotlib.pyplot as plt  
  
# Define spatial grid  
x = np.linspace(-10, 10, 500)  
dx = x[1] - x[0]  
  
# Stationary psi well (Gaussian for example)  
psi0 = np.exp(-x\*\*2 / 4)  
  
# Curvature factor C(x) ~ Laplacian(space + current^2)  
# For now assume homogeneous constant curvature  
C0 = 1.0   
  
# Perturbation field (small sinusoidal ripple)  
k = 1.0  
phi = np.sin(k \* x)  
  
# Linearized gravity contribution  
gravity\_linear = C0 \* phi  
  
plt.plot(x, psi0, label="ψ₀(x)")  
plt.plot(x, phi, label="Perturbation φ(x)")  
plt.plot(x, gravity\_linear, label="Linearized Gravity")  
plt.legend()  
plt.title("Phase 10 Part 1: Linearization Around ψ Well")  
plt.show()