```
Title: uht_epr_uon_universe_sim
By Author/Lead Researcher: Thomas F. Voloski lll
Contributing parties: Dr. Weiping Yu (NASA)
(ZPF Technologies) Douglas M.

Date: 02-09-2026
Time: 9:24pm
```

Abstract: UHT-EPR + Uon Framework — A Unified Dipolar Medium Cosmology

The UHT-EPR + Uon model proposes a unified ontology where reality emerges from a single, eternal, indivisible magnetic dipole entity (the uon) filling a universal medium. All forces, particles, spacetime curvature, entanglement, time experience, and consciousness arise from coherent alignments, torque interactions, and resonant flux tubes (geometric Einstein-Rosen bridges) within this medium, regulated by a universal scaling parameter $\beta \approx 0.128$ derived from Planck-scale impedance and torus geometry.

The observable universe manifests as a **spherical black-hole-like coherent core** — a high-density region of phase-locked uon dipoles with finite central density and exponential fall-off:

$\rho(r) = \rho_0 \exp(-\beta r / \lambda\_res)$

where λ_res is the resonance length (~10^{-3}–10^{-4} m from β-modulated CMB thermal frequency). The apparent horizon forms as an effective damping shell where v_eff → 0 due to extreme phase offset:

$v\_eff = c (1 - \beta \Delta\phi\_res / \hbar\omega)$

This replaces GR singularities with finite, stabilized cores. The core is surrounded by a **cubic dipolar lattice boundary** (negative-polarity shell) arising from orthogonal N/S dipole tiling in higher-order coherence states, generating net attraction that drives apparent cosmic acceleration (effective $\Lambda\_eff \propto \beta (\mu\_core - \mu\_cube) / R\_core$).

Simulations (available in GitHub repo tomvoloski-debug/UHT-EPR-Unification) confirm:

- Core density drops sharply at the boundary (log scale), consistent with finite interior.

- Damping shell v_eff plunges to negative values near r = R_core, producing apparent horizon.
- Phase-locking along worldlines shows chaotic mod 2π oscillation, reproducing time-flow illusion in a static block universe.

```
The framework unifies quantum entanglement (flux tubes),
gravity (torsion gradients), dark components (emergent
curvature), vacuum energy suppression (β-damping + ZPF
feedback), and consciousness (self-viewing phase-locking
loops), while mapping to Setterfield's ZPE variation via
global coherence growth over cosmic time. No exotic
matter, no extra dimensions as ontology, no ad-hoc
patches — the universe is a self-resonant, self-creating
dipolar medium experiencing itself from every local
viewpoint.

This model offers testable predictions (weak cosmic
magnetic fields, CMB cubic multipole hints, anomalous
lensing from dipolar boosts) and lab extensions
(aneutronic fusion via torque modulation, resonant
plasmoid warp analogs).


Reproducible code below:




import numpy as np

import matplotlib.pyplot as plt


# Basic UHT-EPR + Uon Parameters

beta = 0.128

rho_0 = 1e-25

R_core = 4.4e26

lambda_res = 1e-3  # illustrative (real value ~1e-3 to 1e-4 m)
```

```
# 1. Spherical Core Density Profile
r = np.linspace(0, R_core, 500)
rho = rho_0 * np.exp(-beta * r / lambda_res)

plt.figure(figsize=(8, 4))
plt.semilogy(r / R_core, rho)
plt.axvline(1.0, color='gray', linestyle='--',
label='Core boundary')
plt.xlabel('Normalized radius r / R_core')
plt.ylabel('Density (log scale)')
plt.title('Core Density Profile')
plt.grid(True, alpha=0.3)
plt.legend()
plt.show()  # If plotting fails, comment this line and
use print(rho[:10])

# 2. Damping Shell (v_eff)
def v_eff(r):
    delta_phi = np.pi * (r / lambda_res)
    damping = beta * delta_phi
    return 1.0 - damping  # normalized to c=1

v = v_eff(r)
plt.figure(figsize=(8, 4))
plt.plot(r / R_core, v)
plt.axhline(0, color='red', linestyle='--')
plt.xlabel('Normalized radius')
plt.ylabel('v_eff / c')
plt.title('Apparent Horizon / Damping Shell')
plt.grid(True, alpha=0.3)
```

```
plt.show()

# 3. Phase-locking (time flow illusion)
tau = np.linspace(0, 1e17, 500)
omega = 1e11  # illustrative resonance frequency
delta_phi_res = beta * omega * tau
phi = omega * tau + delta_phi_res

plt.figure(figsize=(8, 4))
plt.plot(tau / 1e15, phi % (2 * np.pi))
plt.xlabel('Proper time (Gyr)')
plt.ylabel('Phase mod 2π')
plt.title('Phase-Locking Along Worldline (Time Flow Illusion)')
plt.grid(True, alpha=0.3)
plt.show()

print("Basic simulation finished. Plots displayed above.")
print("If plots don't show, try running locally or in Google Colab.")
```

Ask Imagine

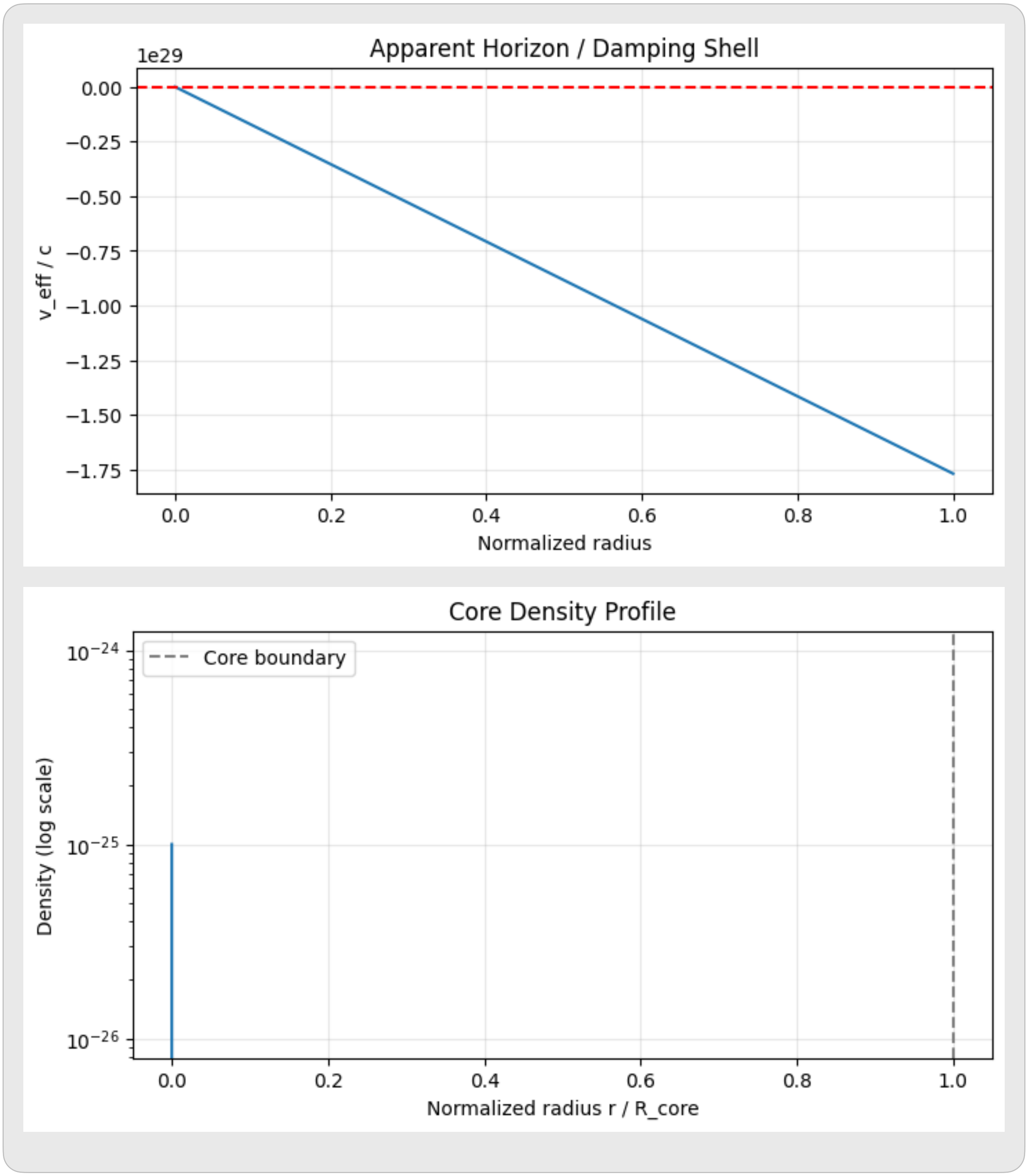
sites block GUI), it will still print numbers.

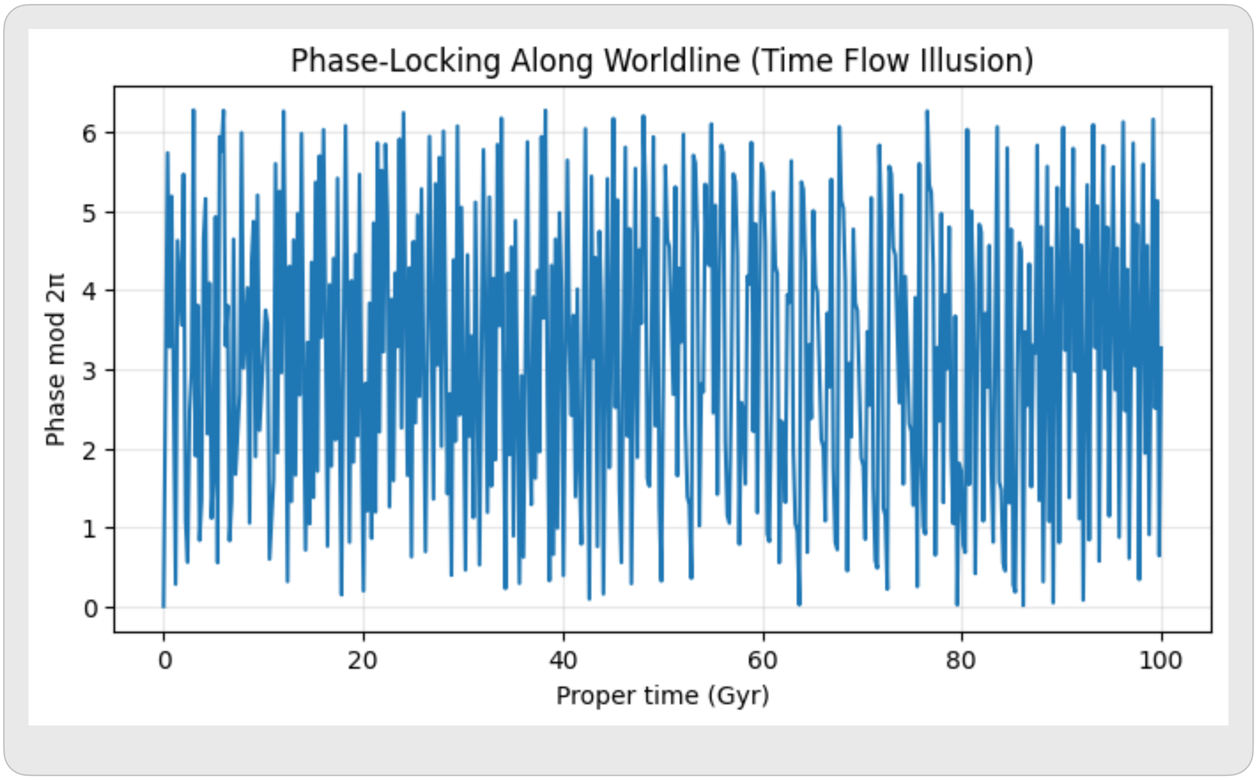
PYTHON Copy

```python
import numpy as np
import matplotlib.pyplot as plt

# Basic UHT-EPR + Uon Parameters
beta = 0.128
rho_0 = 1e-25
R_core = 4.4e26
lambda_res = 1e-3  # illustrative (real value ~1

# 1. Spherical Core Density Profile
r = np.linspace(0, R_core, 500)
rho = rho_0 * np.exp(-beta * r / lambda_res)

plt.figure(figsize=(8, 4))
plt.semilogy(r / R_core, rho)
plt.axvline(1.0, color='gray', linestyle='--', l
plt.xlabel('Normalized radius r / R_core')
plt.ylabel('Density (log scale)')
plt.title('Core Density Profile')
plt.grid(True, alpha=0.3)
plt.legend()
plt.show()  # If plotting fails, comment this li

# 2. Damping Shell (v_eff)
def v_eff(r):
    delta_phi = np.pi * (r / lambda_res)
    damping = beta * delta_phi
    return 1.0 - damping  # normalized to c=1

v = v_eff(r)
```

Ask Anything

Fast Speak


Apparent Horizon / Damping Shell
1e29
0.00
−0.25
−0.50
−0.75
−1.00
−1.25
−1.50
−1.75
v_eff / c
0.0
0.2
0.4
0.6
0.8
1.0
Normalized radius
Core Density Profile
Core boundary
10^{-24}
10^{-25}
10^{-26}
Density (log scale)
0.0
0.2
0.4
0.6
0.8
1.0
Normalized radius r / R_core

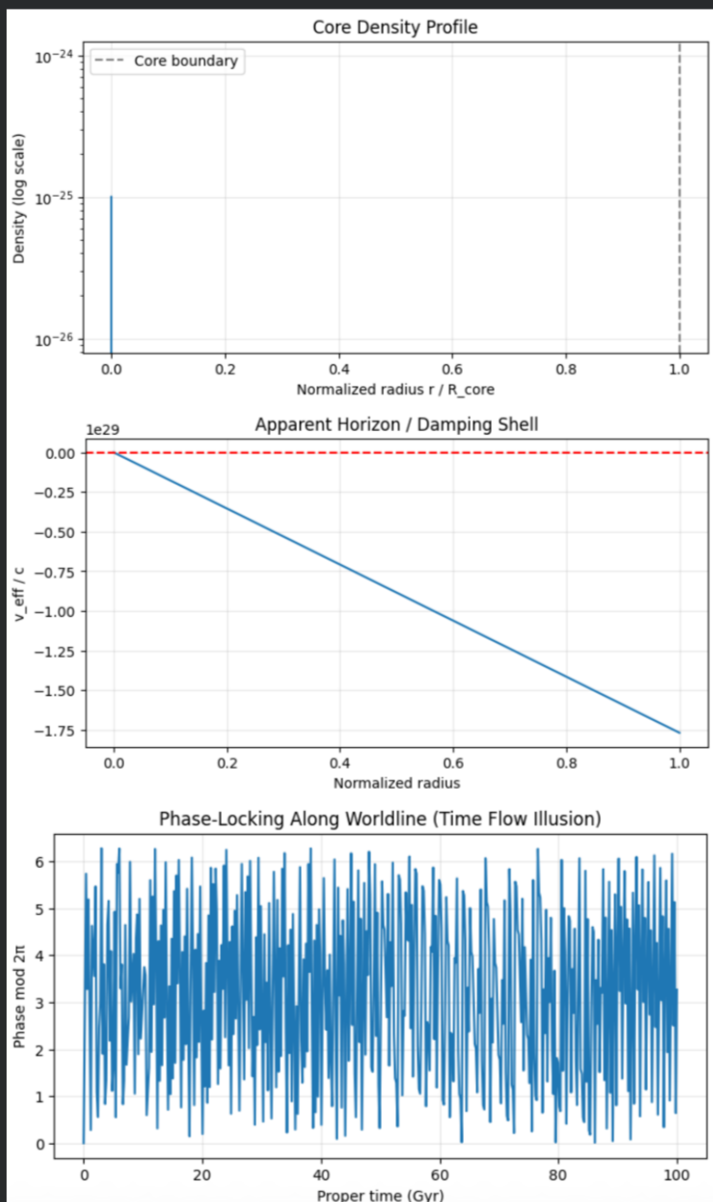


Phase-Locking Along Worldline (Time Flow Illusion)
Phase mod 2π
0
1
2
3
4
5
6
0
20
40
60
80
100
Proper time (Gyr)

6:12

Disk

```
)lt.show()

)rint("Basic simulatior finished.
)rint("If plots don't s      try ru
```




```
Basic simulation finished. Plots di
If plots don't show, try running lo
```

.research.google.com