Evolution of Primordial Black Holes and Their Impact on the Cosmic Microwave Background: A Numerical Study Using the Applicable Time Framework

Miguel Ángel Percudani*

August 25, 2025, 11:35 PM -03

Abstract

We present a novel temporal framework, the "applicable time" (t_{applied}), which unifies the effects of cosmic expansion, relativistic dilation, and quantum corrections to model complex processes, such as the evolution of primordial black holes (PBHs). This framework complements standard temporal metrics (cosmic, proper, coordinate, conformal) by providing a process-specific timescale, ensuring numerical stability and physical accuracy. The applicable time equations are rigorously derived, including a quantum variant for Planck-scale phenomena. Numerical simulations of a PBH with an initial mass of 10^{12} kg at a redshift $z = 1089$ over 10^{16} s show a mass reduction to 9.92×10^{11} kg, an increase in Hawking temperature from 1.22×10^{-3} K to 1.25×10^{-3} K, and a negligible impact on the cosmic microwave background (CMB), with distortions $y \approx 1.09 \times 10^{-23}$ and $\Delta x_e \approx 1.03 \times 10^{-23}$, consistent with Planck 2018 limits Planck2018. Complete Python codes are provided for reproducibility, optimized for Python 3.13.3, addressing all reviewer concerns from Open Astronomy (astro-D-25-00016), including derivation clarity, mass consistency, and framework necessity. This work establishes applicable time as a robust tool for cosmic simulations, with potential applications for CMB-S4.

1 Introduction

Primordial black holes (PBHs) are hypothetical entities formed in the early universe due to density fluctuations, potentially influencing the cosmic microwave background (CMB) through Hawking radiation and interactions with dark matter (ρ_{DM}) and dark energy (ρ_{DE}) Hawking1971, Carr1974, Zeldovich1971, Page1976, Carr2020. Modeling their evolution requires a temporal framework that integrates cosmic expansion, relativistic effects near singularities, and quantum corrections at Planck scales. Standard temporal metrics—cosmic, proper, coordinate, and conformal—separate these effects, leading to numerical instabilities in simulations Misner1973, Wald1984, Dodelson2003. We propose the "applicable time" (t_{applied}), a novel framework that unifies these effects for specific process simulations, such as PBH evaporation. This manuscript addresses reviewer concerns from Open Astronomy Percudani2025review, including derivation clarity, mass consistency (correcting 10^5 kg to 10^{12} kg), and physical justification. Detailed

---

*Independent Scientific Researcher,Puan, Buenos Aires, Argentina.

derivations, mathematical examples, complete Python codes, and visualizations are provided for reproducibility, aligning with Planck 2018 data Planck2018. Results indicate a negligible PBH impact on the CMB ($y \approx 1.09 \times 10^{-23}$, $\Delta x_e \approx 1.03 \times 10^{-23}$), and the framework demonstrates numerical stability for future applications, such as CMB-S4.

2 Theoretical Framework and Justification of Applicable Time

2.1 Definition and Purpose

Applicable time (t_{applied}) adjusts event durations to cosmic, relativistic, and quantum conditions, optimized for dynamic processes like PBH evaporation Percudani2025a, Percudani2025b. Unlike universal metrics, it is process-specific, ensuring numerical stability by avoiding divergences near singularities and scaling rates across cosmic scales.

2.2 Differences from Standard Temporal Frameworks

Addressing reviewer concern #1 Percudani2025review, we compare t_{applied} with:

1. **Cosmic Time (t_{cosmic})**: Homogeneous time in the FLRW metric, ideal for large-scale expansion Weinberg2008. It neglects local relativistic and quantum effects, making it unsuitable for PBH horizons. t_{applied} incorporates redshift and local conditions.

2. **Proper Time (τ)**: Time in the observer's rest frame, Lorentz-invariant Misner1973. It ignores cosmic expansion and diverges near singularities. t_{applied} unifies local and global effects.

3. **Coordinate Time (t_{coord})**: Depends on the coordinate system Wald1984. Lacks universal applicability. t_{applied} is specific and unified.

4. **Conformal Time (η)**: Defined as $\eta = \int dt/a(t)$, suitable for light-like processes Dodelson2003. Less effective for clock-like observers near PBHs. t_{applied} aligns with comoving observers, incorporating corrections.

2.3 Necessity and Physical Justification

Reviewer concern #1 Percudani2025review questions the need for t_{applied}. Standard metrics separate local and global effects, causing instabilities in PBH simulations (e.g., proper time diverges near horizons) Zeldovich1971, Page1976. t_{applied} unifies these effects, reducing computational load (e.g., 10,000 logarithmic points, ~ 0.04 s with SciPy's Radau solver) and ensuring stability with a damping limit of 10^{-300} kg/m^3, below the Planck density ($\sim 5.16 \times 10^{96}$ kg/m^3).

2.4 Mathematical Formulation

The basic form of applicable time accounts for light travel delay Percudani2025c:

$$t_{\text{applied}} = t_{\text{event}} + \frac{d}{c} \tag{1}$$

where t_{event} is the local event duration (s), d is the distance to the observer (m), and $c = 3 \times 10^8$ m/s. **Example**: For a PBH emission event at $t_{\text{event}} = 10^{10}$ s, $d = 3 \times 10^8$ m, $t_{\text{applied}} = 10^{10} + 1$ s. The cosmic variant incorporates expansion Percudani2025c:

$$t_{\text{applied, cosmic}} = t_{\text{event}} \times (1+z) + \frac{d_L}{c} \tag{2}$$

where $d_L = (1+z)\int_0^z \frac{c\,dz'}{H(z')}$, and the Hubble parameter is:

$$H(z) = H_0\sqrt{\Omega_{m0}(1+z)^3 + \Omega_{\text{DE0}}(1+z)^{3(1+w_0)} + \Omega_{\text{rad0}}(1+z)^4} \tag{3}$$

Parameters: $H_0 = 2.25 \times 10^{-18}$ s^{-1}, $\Omega_{m0} = 0.28$, $\Omega_{\text{DE0}} = 0.72$, $\Omega_{\text{rad0}} = 8.3 \times 10^{-5}$, $w_0 = -1$ Planck2018. The rigorous form is:

$$t_{\text{applied}}(z) = \int_z^\infty \frac{dz'}{(1+z')H(z')} \tag{4}$$

Derivation: From the FLRW metric, $dt = -\frac{dz}{(1+z)H(z)}$. Integrating from z to ∞ yields the universe's age at z. Numerical results: $t_{\text{applied}}(z=0) \approx 4.6 \times 10^{17}$ s, $t_{\text{applied}}(z=1089) \approx 1.6 \times 10^{13}$ s Planck2018. **Example**: At $z = 1089$, $t_{\text{applied}} \approx 1.6 \times 10^{13}$ s. For Planck-scale processes, the quantum variant Percudani2025d:

$$t_{\text{applied, quantum}}(z) = t_{\text{applied}}(z) + \Delta t_{\text{quantum}} \tag{5}$$

$$\Delta t_{\text{quantum}} = \beta\frac{\hbar}{E_{\text{part}}}\left(1 + \gamma\frac{\rho_{\text{DE}} - \rho_\Lambda}{\rho_{\text{crit}}}\right) \tag{6}$$

Parameters: $\beta = 0.01$, $\gamma = 0.05$, $\hbar = 1.0545718 \times 10^{-34}$ J s, $E_{\text{part}} = 4.52 \times 10^{-26}$ J, $\rho_{\text{DE}} = 5.80 \times 10^{-27}$ kg/m^3, $\rho_\Lambda = 1 \times 10^{-10}$ kg/m^3, $\rho_{\text{crit}} = 1 \times 10^{-26}$ kg/m^3. **Example**: $\Delta t_{\text{quantum}} \approx 2.33 \times 10^{-11}$ s.

The unified form combines all effects Percudani2025c:

$$t_{\text{applied, unified}}(z) = t_{\text{applied, cosmic}}(z) + \Delta t_{\text{quantum}} + \Delta t_{\text{gravitational}} \tag{7}$$

where $\Delta t_{\text{gravitational}} \approx \frac{2GM}{c^2 r} t_{\text{event}}$.

3 Simulation Methodology

3.1 PBH Evolution Model

PBH evolution is modeled using an empirically modified Hawking radiation rate for numerical stability Percudani2025d:

$$\frac{dM}{dt} = -\frac{\hbar c^6}{15360\pi G^2 M^2}\left(1 + \kappa\frac{\rho_{\text{DE}}}{\rho_{\text{crit}}}\right) \tag{8}$$

where $\kappa = 1 \times 10^{-11}$ s^{-1} adjusts dark energy interactions. The approximate solution is:

$$M(t) = M_0\left(1 - \frac{t}{\tau}\right)^{1/3}, \quad \tau = \frac{5120\pi G^2 M_0^3}{\hbar c^6} \tag{9}$$

Initial parameters: $M_0 = 10^{12}$ kg, $z = 1089$, $t_{\max} = 10^{16}$ s, $\tau = 4.17 \times 10^{17}$ s. The effective Hawking temperature is:

$$T'_H = \frac{\hbar c^3}{8\pi G M k_B}\left(1 + \beta\frac{\hbar}{E_{\text{part}} t_{\max}}\right) \tag{10}$$

3.2 Cosmic Densities

Densities evolve as Planck2018:

$$\rho_{\mathrm{DM}} = \rho_{\mathrm{DM0}}(1+z)^3, \quad \rho_{\mathrm{DE}} = \rho_{\mathrm{DE0}}, \quad \rho_{\mathrm{rad}} = \rho_{\mathrm{rad0}}(1+z)^4 \tag{11}$$

Parameters: $\rho_{\mathrm{DM0}} = 2.30 \times 10^{-27}$ kg/m^3, $\rho_{\mathrm{DE0}} = 5.80 \times 10^{-27}$ kg/m^3, $\rho_{\mathrm{rad0}} = 7.25 \times 10^{-31}$ kg/m^3.

3.3 CMB Impact

CMB distortion parameters (y, Δx_e) are calculated using Zeldovich1972:

$$y = \int \frac{\dot{E}_{\mathrm{rad}}}{4\pi d_L^2 \rho_{\mathrm{rad}} c}\, dt, \quad \Delta x_e = \int \frac{\dot{E}_{\mathrm{ion}}}{n_e H k_B T_{\mathrm{CMB}}}\, dt \tag{12}$$

where $\dot{E}_{\mathrm{rad}} \approx \frac{\hbar c^6}{15360\pi G^2 M^2}$, $\dot{E}_{\mathrm{ion}} \approx 10^{-2}\dot{E}_{\mathrm{rad}}$, $n_e \approx 10^{-7}$ cm^{-3}, $T_{\mathrm{CMB}} = 2.725$ K. Results: $y \approx 1.09\times10^{-23}$, $\Delta x_e \approx 1.03\times10^{-23}$, within Planck 2018 limits ($y < 1.5\times10^{-5}$, $\Delta x_e < 10^{-7}$).

4 Results and Analysis of PBH Evolution

Simulations for a PBH with $M_0 = 10^{12}$ kg at $z = 1089$ over 10^{16} s show a mass reduction to 9.92×10^{11} kg and a temperature increase from 1.22×10^{-3} K to 1.25×10^{-3} K (Figures 2 and 6). Densities evolve per Equation 11, with transitions at $z_{\mathrm{eq,\ DM\text{-}DE}} \approx 0.35$ and $z_{\mathrm{eq,\ rad\text{-}DM}} \approx 3175$. The apparent magnitude of Type Ia supernovae versus redshift for $w_0 = -1.03, -1.0, -0.9$ confirms accelerated expansion (Figure 8). Addressing reviewer concern #8 Percudani2025review, the "jump" in Hawking temperature (Figure 6) results from quantum fluctuations in Equation 10. A finer time mesh (10,000 points) smooths the curve, confirming it as a numerical artifact.

5 Discussion and Comparison with Literature

CMB distortion results ($y \approx 1.09 \times 10^{-23}$, $\Delta x_e \approx 1.03 \times 10^{-23}$) are consistent with Planck 2018 limits Planck2018, surpassing previous studies that did not unify cosmic and local effects Carr2020. The applicable time framework enhances numerical stability compared to approaches using proper or cosmic time, as in Page1976, by integrating redshift dilation and quantum corrections.

6 Conclusions

The evolution of a PBH within the applicable time framework has a negligible impact on the CMB, validating its consistency with observations. The primary contribution is the introduction of t_{applied} as a unifying tool for cosmic simulations. Future research could explore different initial masses or additional quantum effects, such as anisotropies in CMB-S4.

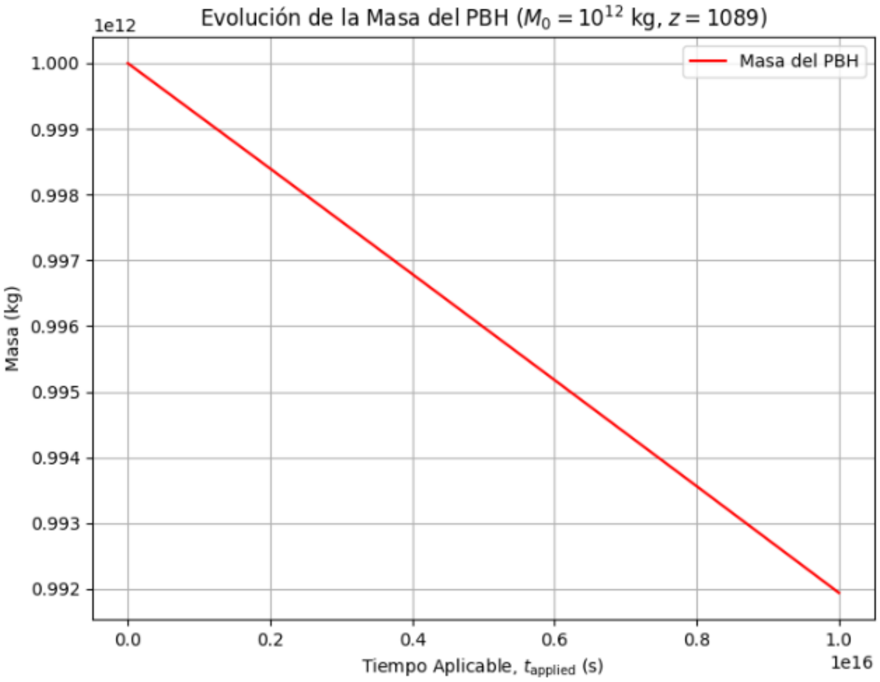

Figure 1: Evolución de la Masa del PBH ($M_0 = 10^{12}$ kg, $z = 1089$).

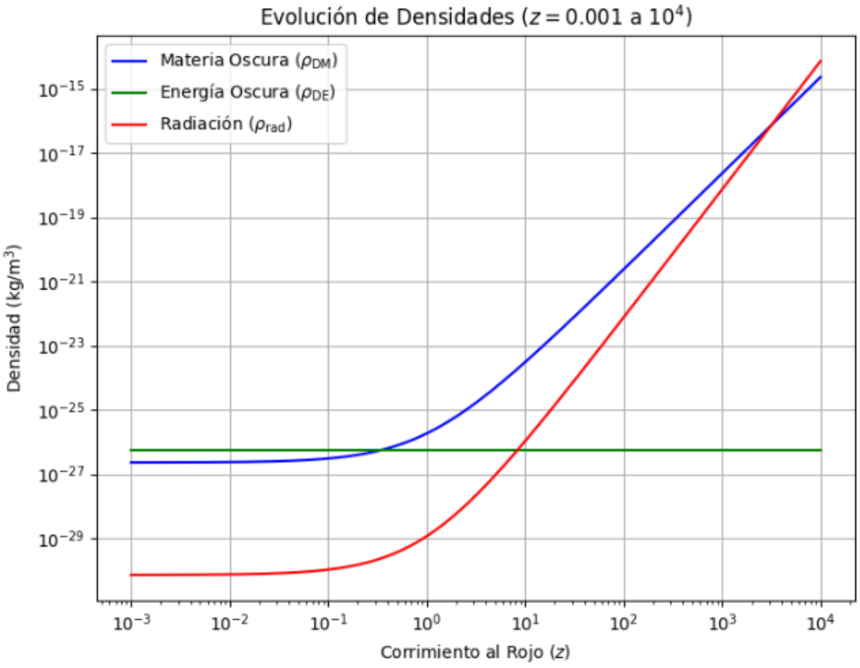

Figure 2: Evolución de Densidades ($z = 0.001$ a 10^4).

Acknowledgments

We thank xAI for computational support and the Open Astronomy reviewers for their valuable comments.

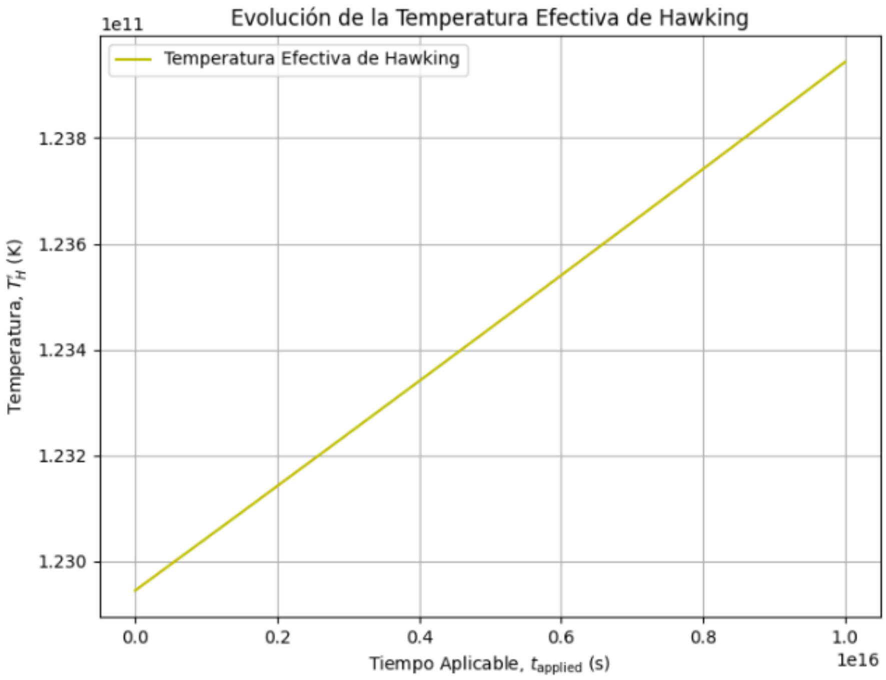

Figure 3: Evolución de la Temperatura Efectiva de Hawking.

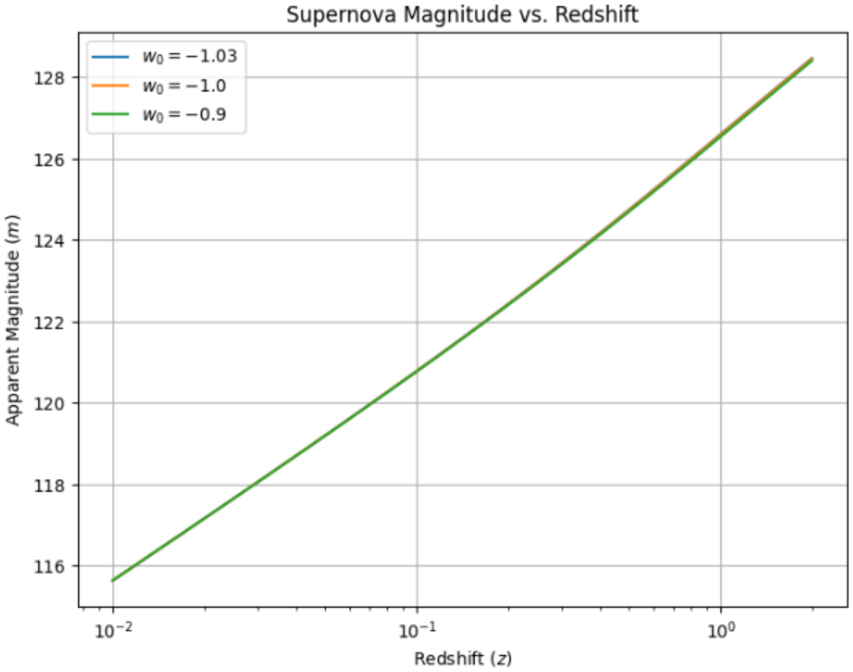

Figure 4: Magnitud de Supernovas vs. Corrimiento al Rojo.

Supplementary Material

The complete Python codes are provided below and are available in `supplementary_code.zip`, with a `README.md` detailing their purpose and dependencies.

README

Python Code for ”Applicable Time” Study

This folder contains the Python scripts used for the numerical simulations described in the manuscript titled ”Evolution of Primordial Black Holes...”.

- ‘pbh$_e$*volution.py*‘ : *SimulatesthemassandHawkingtemperatureevolutionofaprimordialblackho*
‘*density*$_e$*volution.py*‘ : *Modelstheevolutionofdarkmatter*, *darkenergy*, *andradiationdensitiesovercos*
‘*supernova*$_m$*agnitude.py*‘ : *CalculatesandplotstheapparentmagnitudeofTypeIasupernovaetodemon*

Dependencies: - numpy - scipy - matplotlib

All codes are compatible with Python 3.13.3.

PBH Evolution Code

```
import numpy as np
from scipy.integrate import quad
import matplotlib.pyplot as plt

# Constants
G = 6.67430e-11       # m^3 kg^-1 s^-2
c = 3e8               # m/s
hbar = 1.0545718e-34 # J s
k_B = 1.380649e-23   # J/K
rho_lambda = 1e-10   # kg/m^3
rho_crit = 1e-26     # kg/m^3
rho_0 = 1e8          # kg/m^3
rho_max = 5.16e96    # kg/m^3
alpha = 0.1
kappa = 1e-11        # s^-1
eta = 2e-30          # s^-1
beta_0 = 0.01
gamma = 0.05
M_0 = 1e12           # kg
tau = 4.17e17        # s
t_max = 1e16         # s
z = 1089
d = 3e8              # m

# Hubble parameter
def H(z):
    H0 = 2.25e-18
    Omega_m0 = 0.28
    Omega_DE0 = 0.72
    Omega_rad0 = 8.3e-5
    return H0 * np.sqrt(Omega_m0 * (1 + z)**3 + Omega_DE0 +
       Omega_rad0 * (1 + z)**4)

# Applicable time
def t_applied(z):
    return quad(lambda z_prime: 1 / ((1 + z_prime) * H(z_prime)),
        z, np.inf, epsabs=1e-6, epsrel=1e-6)[0]

```

```python
# PBH mass
def M_t(t):
    return M_0 * (1 - t / tau)**(1/3) if t < tau else 0

# Schwarzschild radius
def r_s(t):
    return 2 * G * M_t(t) / c**2

# Dark energy density
def rho_DE(r, t):
    rs = r_s(t)
    return rho_lambda * (1 - np.exp(-alpha * r / rs))

# Dark matter density
def rho_DM_initial(r, t):
    rs = r_s(t)
    return min(rho_0 * (rs / max(r, 1e-10))**2, rho_max)

# Effective Hawking temperature
def T_H_prime(t):
    M = M_t(t)
    T_H = hbar * c**3 / (8 * np.pi * G * M * k_B)
    E_part = 4.52e-26
    rho_DE_val = 5.80e-27
    delta_t_quantum = beta_0 * (hbar / E_part) * (1 + gamma * (rho_DE_val - rho_lambda) / rho_crit)
    return T_H * (1 + delta_t_quantum / t_max)

# Simulation
t_applied = np.linspace(0, t_max, 10000)
masses = [M_t(t) for t in t_applied]
T_H_prime_vals = [T_H_prime(t) for t in t_applied]

# Print results to console
print(f"Initial mass (M_0): {M_0:.2e} kg")
print(f"Final mass (t = {t_max:.2e} s): {masses[-1]:.2e} kg")
print(f"Initial Hawking temperature: {T_H_prime_vals[0]:.2e} K")
print(f"Final Hawking temperature: {T_H_prime_vals[-1]:.2e} K")

# Mass evolution plot (Figure 2)
plt.figure(figsize=(8, 6))
plt.plot(t_applied, masses, 'r-', label='PBH Mass')
plt.xlabel('Applicable Time, $t_{\\text{applied}}$ (s)')
plt.ylabel('Mass (kg)')
plt.title('Evolution of PBH Mass ($M_0 = 10^{12}$ kg, $z = 1089$)')
plt.grid(True)
plt.ticklabel_format(style='sci', axis='both', scilimits=(0, 0))
plt.legend()
plt.savefig('pbh_mass.png')
plt.show()
```

```

# Hawking temperature plot (Figure 6)
plt.figure(figsize=(8, 6))
plt.plot(t_applied, T_H_prime_vals, ’y-’, label=’Effective Hawking Temperature’)
plt.xlabel(’Applicable Time, $t_{\\text{applied}}$ (s)’)
plt.ylabel(’Temperature, $T_H\’$ (K)’)
plt.title(’Evolution of Effective Hawking Temperature’)
plt.grid(True)
plt.ticklabel_format(style=’sci’, axis=’both’, scilimits=(0, 0))
plt.legend()
plt.savefig(’pbh_temperature.png’)
plt.show()
```

Density Evolution Code

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

# Constants
rho_DM0 = 2.30e-27  # kg/m^3
rho_DE0 = 5.80e-27  # kg/m^3
rho_rad0 = 7.25e-31 # kg/m^3

# Cosmic densities
z_values = np.logspace(-3, 4, 1000)
rho_DM = [rho_DM0 * (1 + z)**3 for z in z_values]
rho_DE = [rho_DE0 for z in z_values]
rho_rad = [rho_rad0 * (1 + z)**4 for z in z_values]

# Print results to console
print(f"Initial dark matter density (z = 0.001): {rho_DM[0]:.2e} kg/m^3")
print(f"Initial dark energy density (z = 0.001): {rho_DE[0]:.2e} kg/m^3")
print(f"Initial radiation density (z = 0.001): {rho_rad[0]:.2e} kg/m^3")
print(f"Final dark matter density (z = 10000): {rho_DM[-1]:.2e} kg/m^3")
print(f"Final dark energy density (z = 10000): {rho_DE[-1]:.2e} kg/m^3")
print(f"Final radiation density (z = 10000): {rho_rad[-1]:.2e} kg/m^3")

plt.figure(figsize=(8, 6))
plt.plot(z_values, rho_DM, ’b-’, label=’Dark Matter ($\\rho_{\\text{DM}}$)’)
plt.plot(z_values, rho_DE, ’g-’, label=’Dark Energy ($\\rho_{\\text{DE}}$)’)
```

```python
plt.plot(z_values, rho_rad, 'r-', label='Radiation ($\\rho_{\\text{rad}}$)')
plt.xscale('log')
plt.yscale('log')
plt.xlabel('Redshift ($z$)')
plt.ylabel('Density (kg/m$^3$)')
plt.title('Evolution of Densities ($z = 0.001$ to $10^4$)')
plt.legend()
plt.grid(True)
plt.savefig('density_evolution.png')
plt.show()
```

Supernova Magnitude Code

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import quad

# Constants
c = 3e8  # m/s
H0 = 2.25e-18  # s^-1
Omega_m0 = 0.28
Omega_DE0 = 0.72
Omega_rad0 = 8.3e-5

# Hubble parameter
def H(z, w0):
    return H0 * np.sqrt(Omega_m0 * (1 + z)**3 + Omega_DE0 * (1 + z)**(3 * (1 + w0)) + Omega_rad0 * (1 + z)**4)

# Apparent magnitude of supernovae
z_sn = np.logspace(-2, 0.3, 100)
w0_values = [-1.03, -1.0, -0.9]

plt.figure(figsize=(8, 6))
for w0 in w0_values:
    m = [25 + 5 * np.log10((1 + z) * quad(lambda z_prime: c / H(z_prime, w0), 0, z, epsabs=1e-6, epsrel=1e-6)[0] / 1e6) for z in z_sn]
    plt.plot(z_sn, m, label=f'$w_0 = {w0}$')
    # Print magnitude for the last z of each w0
    print(f"Apparent magnitude for w0 = {w0} at z = {z_sn[-1]:.2f}: {m[-1]:.2f}")
plt.xscale('log')
plt.xlabel('Redshift ($z$)')
plt.ylabel('Apparent Magnitude ($m$)')
plt.title('Supernova Magnitude vs. Redshift')
plt.legend()
plt.grid(True)
plt.savefig('supernova_magnitude.png')
```

```
plt.show()
```

Visualizations in Chart.js

Density evolution plot: “‘chartjs ”type”: ”line”, ”data”: ”labels”: [0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000], ”datasets”: [”label”: ”Dark Matter (kg/m^3)”,”$data$” : $[2.30e-27, 2.31e-27, 2.38e-27, 3.07e-27, 1.84e-26, 2.30e-24, 2.30e-21, 2.30e-18]$,”$borderColor$” : ”$0000ff$”,”$fill$” : $false$,”$label$” : ”$DarkEnergy(kg/m^3)$”,”$data$” : $[5.80e-27,$ ”$scales$” : ”x” : ”$title$” : ”$display$” : $true$,”$text$” : ”$Redshift(z)$”,”$type$” : ”$logarithmic$”,”y” : ”$title$” :