Python Codes

Joshua

September 2025

1 Exact SFT Prediction for $R_o(t)/R_{\mathrm{UCZ}}$

import numpy as np import matplotlib.pyplot as plt

— Parameters — H0$_kmsMpc = 67.77 Hubbleconstant[km/s/Mpc] H0 = H0_kmsMpc * 1000.0/3.0857e22 convertto1/s Omega_m = 0.317 Omega_L = 0.683$

R$_{ot}oday = 4.4e26 Presentobservableradius[m] R_ucz = 1.4466e27 UCZradius[m] ratio_Ro_to_Rucz = R_{ot}oday/R_ucz$

sec$_per_Gyr = 3.15576e16$

— Integration grid — a$_min, a_max, N = 1e{-}8, 50.0, 20000 a = np.logspace(np.log10(a_min), np.log10(a_max),$

E = np.sqrt(Omega$_m * a**(-3) + Omega_L) integrand = 1.0/(a*E)$

cum = np.cumsum(0.5 * (integrand[1:] + integrand[:-1]) * (a[1:] - a[:-1])) t$_sec = np.empty_like(a) t_sec[0] = 0.0 t_sec[1:] = (1.0/H0)*cum t_Gyr = t_sec/sec_per_Gyr$

Invert to get a(t) t$_plot = np.linspace(0.01, 40.0, 800) a_of_t = np.interp(t_plot, t_Gyr, a, left = a[0], right = a[-1])$

Ro$_norm = ratio_Ro_to_Rucz * a_of_t$

Present age and crossing t0 = float(np.interp(1.0, a, t$_Gyr)) idx_star = np.argmax(Ro_norm >= 1.0) t_star = t_plot[idx_star] if np.any(Ro_norm >= 1.0) else None$

— Plot — plt.figure(figsize=(10,6)) plt.plot(t$_plot, Ro_norm, color =' orange', lw = 2, label = r'R_o(t)/R_{UCZ}$ (SFT exact)') plt.axhline(1.0, color='red', ls='–', lw=1.5, label='UCZ boundary') if t$_star : plt.axvline(t_star, color =' gray', ls =':', label = fr't_* = t_star : .2f$ Gyr') plt.scatter([t0], [ratio$_Ro_to_Rucz], color =' black', marker =' x', s = 80, label = fr't_0 = t0 : .2f$ Gyr')

plt.xlabel("Cosmic time t (Gyr)") plt.ylabel("Normalized radius R$_o(t)/R_UCZ") plt.title("ExactSFTpredic$
$R_o(t)/R_UCZ(Friedmannintegral)") plt.legend() plt.grid(True) plt.tight_layout() plt.savefig("SFT_Ro_over_Rucz$
300)$plt.show()$