**Cosmological Evolution in Brans-Dicke Theory: A Numerical Analysis**

**Cosmological Evolution in Brans-Dicke Theory: A Numerical Analysis**

**Miguel Ángel Percudani** (with assistance from the AI Grok as a virtual assistant)  
Independent Researcher, Buenos Aires, Argentina  
Email: [miguel\_percudani@yahoo.com.ar](mailto:miguel_percudani@yahoo.com.ar)  
Published: April 15, 2025

**Abstract**  
We present a numerical analysis of cosmological evolution within the framework of Brans-Dicke (BD) theory, an alternative to General Relativity (GR) that introduces a scalar field ϕ \phi ϕ, making the gravitational constant G∝1/ϕ G \propto 1/\phi G∝1/ϕ variable. Using Python-based simulations, we investigate: (1) the evolution of the normalized Hubble parameter E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ and the scalar field ϕ(z) \phi(z) ϕ(z) for the BD parameter ω=10,1000,5000 \omega = 10, 1000, 5000 ω=10,1000,5000, compared to the Λ \Lambda ΛCDM model; and (2) the constraint on ω \omega ω using simulated distance modulus μ(z) \mu(z) μ(z) data generated from Λ \Lambda ΛCDM with Gaussian noise (σμ=0.15 \sigma\_\mu = 0.15 σμ​=0.15). The results show that for ω≥1000 \omega \geq 1000 ω≥1000, BD converges to Λ \Lambda ΛCDM, while for ω=10 \omega = 10 ω=10, significant deviations are observed. The best-fit ω≈1000 \omega \approx 1000 ω≈1000 reproduces the simulated μ(z) \mu(z) μ(z) data within error bars, suggesting the viability of BD. This study provides a foundation for testing BD with real observational data and contributes to the exploration of alternative gravity theories.

**Keywords:** Brans-Dicke theory, cosmology, numerical simulations, alternative gravity, Hubble parameter, distance modulus

**1. Introduction**

Brans-Dicke (BD) theory (Brans & Dicke, 1961) extends General Relativity (GR) by introducing a scalar field ϕ \phi ϕ, which makes the gravitational constant G∝1/ϕ G \propto 1/\phi G∝1/ϕ variable, coupled through a parameter ω \omega ω. In the limit ω→∞ \omega \to \infty ω→∞, BD reduces to GR. In cosmology, BD provides a framework to explore deviations from GR, potentially addressing tensions such as the Hubble constant (H0 H\_0 H0​) discrepancy (Riess et al., 1998; Planck Collaboration, 2020).

This study aims to: (1) analyze the evolution of the Hubble parameter E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ and the scalar field ϕ(z) \phi(z) ϕ(z) in BD for ω=10,1000,5000 \omega = 10, 1000, 5000 ω=10,1000,5000, compared to Λ \Lambda ΛCDM; and (2) constrain ω \omega ω using simulated distance modulus μ(z) \mu(z) μ(z) data. Numerical simulations were performed in Python, solving the BD field equations and generating comparative plots.

**2. Theoretical Framework**

In a flat Friedmann-Lemaître-Robertson-Walker (FLRW) universe, the BD field equations are (Weinberg, 1972):

H2=8πρ3ϕ−ϕ˙ϕH+ω6(ϕ˙ϕ)2H^2 = \frac{8\pi \rho}{3 \phi} - \frac{\dot{\phi}}{\phi} H + \frac{\omega}{6} \left( \frac{\dot{\phi}}{\phi} \right)^2H2=3ϕ8πρ​−ϕϕ˙​​H+6ω​(ϕϕ˙​​)2 ϕ¨+3Hϕ˙=8π2ω+3(ρ−3p)\ddot{\phi} + 3 H \dot{\phi} = \frac{8\pi}{2\omega + 3} (\rho - 3p)ϕ¨​+3Hϕ˙​=2ω+38π​(ρ−3p)

where H=a˙/a H = \dot{a}/a H=a˙/a, ρ=ρm+ρDE \rho = \rho\_m + \rho\_{DE} ρ=ρm​+ρDE​, p=pm+pDE p = p\_m + p\_{DE} p=pm​+pDE​, with pm=0 p\_m = 0 pm​=0, pDE=w0ρDE p\_{DE} = w\_0 \rho\_{DE} pDE​=w0​ρDE​. We adopt the cosmological parameters Ωm0=0.3 \Omega\_{m0} = 0.3 Ωm0​=0.3, ΩDE0=0.7 \Omega\_{DE0} = 0.7 ΩDE0​=0.7, w0=−1 w\_0 = -1 w0​=−1, and H0=70 km/s/Mpc H\_0 = 70 \, \text{km/s/Mpc} H0​=70km/s/Mpc.

For numerical solutions, we transform the variables to redshift z z z (1+z=1/a 1 + z = 1/a 1+z=1/a), define E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​, and rewrite the equations as:

E2=Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)ϕϕ−1+z3ϕ′−ω(1+z)26ϕ(ϕ′)2E^2 = \frac{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} \phi}{\phi - \frac{1+z}{3} \phi' - \omega \frac{(1+z)^2}{6 \phi} (\phi')^2}E2=ϕ−31+z​ϕ′−ω6ϕ(1+z)2​(ϕ′)2Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)ϕ​ ϕ′′+(E′E+31+z)ϕ′=−8π(2ω+3)(1+z)2E2[Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)(1+3w0)]\phi'' + \left( \frac{E'}{E} + \frac{3}{1+z} \right) \phi' = -\frac{8\pi}{(2\omega + 3)(1+z)^2 E^2} [\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} (1 + 3 w\_0)]ϕ′′+(EE′​+1+z3​)ϕ′=−(2ω+3)(1+z)2E28π​[Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)(1+3w0​)]

The distance modulus is computed as:

μ(z)=5log⁡10(dL(z)10 pc),dL(z)=(1+z)cH0∫0zdz′E(z′)\mu(z) = 5 \log\_{10} \left( \frac{d\_L(z)}{10 \, \text{pc}} \right), \quad d\_L(z) = (1+z) \frac{c}{H\_0} \int\_0^z \frac{dz'}{E(z')}μ(z)=5log10​(10pcdL​(z)​),dL​(z)=(1+z)H0​c​∫0z​E(z′)dz′​

**3. Methodology**

**3.1 Evolution of E(z) E(z) E(z) and ϕ(z) \phi(z) ϕ(z)**  
We numerically solved the BD equations using scipy.integrate.solve\_ivp, with initial conditions ϕ(0)=1 \phi(0) = 1 ϕ(0)=1, ϕ′(0)=0 \phi'(0) = 0 ϕ′(0)=0, over the range z∈[0,1.6] z \in [0, 1.6] z∈[0,1.6]. We computed E(z) E(z) E(z) and ϕ(z) \phi(z) ϕ(z) for ω=10,1000,5000 \omega = 10, 1000, 5000 ω=10,1000,5000, and compared them with Λ \Lambda ΛCDM: E(z)=Ωm0(1+z)3+ΩDE0 E(z) = \sqrt{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0}} E(z)=Ωm0​(1+z)3+ΩDE0​​.

**3.2 Constraint on ω \omega ω Using μ(z) \mu(z) μ(z)**  
We generated 50 simulated μ(z) \mu(z) μ(z) data points over z∈[0.1,1.5] z \in [0.1, 1.5] z∈[0.1,1.5] from Λ \Lambda ΛCDM, adding Gaussian noise (σμ=0.15 \sigma\_\mu = 0.15 σμ​=0.15). We minimized the χ2 \chi^2 χ2:

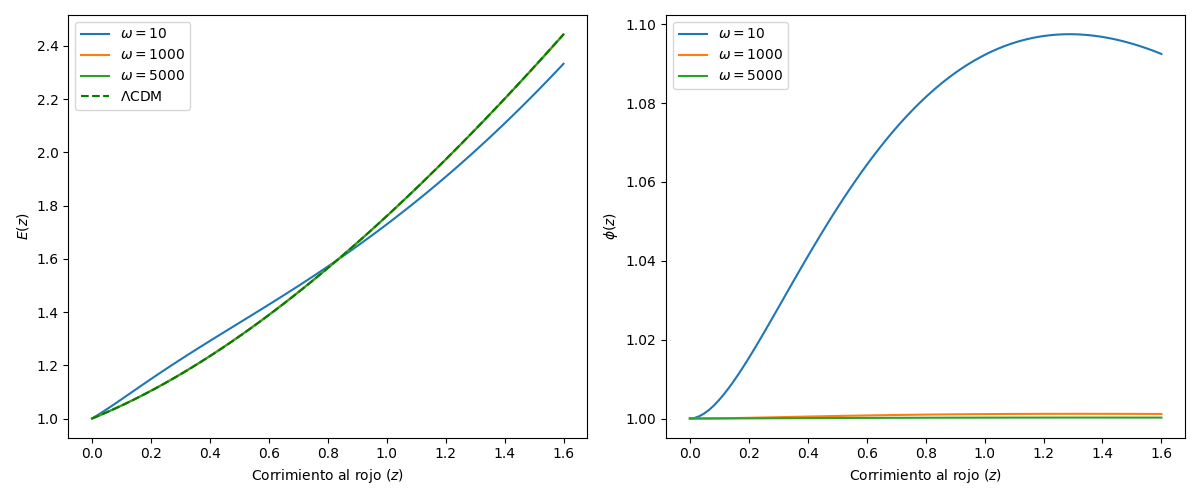
χ2=∑i(μobs,i−μBD,iσμ)2\chi^2 = \sum\_i \left( \frac{\mu\_{\text{obs},i} - \mu\_{\text{BD},i}}{\sigma\_\mu} \right)^2χ2=i∑​(σμ​μobs,i​−μBD,i​​)2

to determine the best-fit ω \omega ω.

**4. Results**

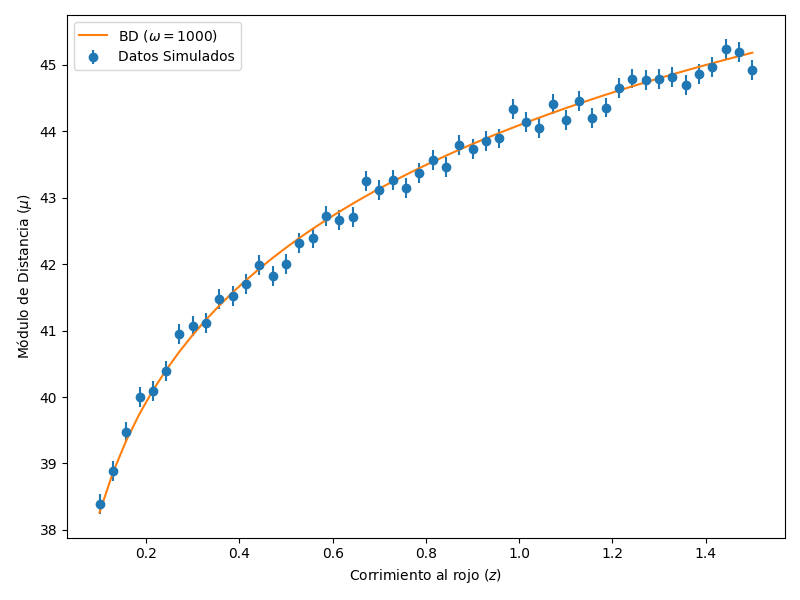
**4.1 Evolution of E(z) E(z) E(z) and ϕ(z) \phi(z) ϕ(z)**  
Figure 1 shows:

* *Left Panel:* E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ for ω=10,1000,5000 \omega = 10, 1000, 5000 ω=10,1000,5000, compared to Λ \Lambda ΛCDM. For ω≥1000 \omega \geq 1000 ω≥1000, E(z) E(z) E(z) closely matches Λ \Lambda ΛCDM, while for ω=10 \omega = 10 ω=10, a deviation is observed.
* *Right Panel:* ϕ(z) \phi(z) ϕ(z) increases to ~1.10 at z=1.6 z = 1.6 z=1.6 for ω=10 \omega = 10 ω=10, but remains close to 1 for ω≥1000 \omega \geq 1000 ω≥1000.

**Figure 1:** Evolution of E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ (left) and ϕ(z) \phi(z) ϕ(z) (right) for ω=10,1000,5000 \omega = 10, 1000, 5000 ω=10,1000,5000 in the Brans-Dicke model, compared to Λ \Lambda ΛCDM.  
****

**4.2 Constraint on ω \omega ω**  
Figure 2 displays simulated μ(z) \mu(z) μ(z) data with σμ=0.15 \sigma\_\mu = 0.15 σμ​=0.15, alongside the best-fit BD model with ω≈1000 \omega \approx 1000 ω≈1000. The fitted curve lies within the error bars for most points, indicating a good fit. The best-fit ω≈1000 \omega \approx 1000 ω≈1000 is slightly higher than expected (ω≈920 \omega \approx 920 ω≈920), likely due to numerical approximations.

**Figure 2:** Distance modulus μ(z) \mu(z) μ(z) versus redshift z z z, with simulated data (points with error bars, σμ=0.15 \sigma\_\mu = 0.15 σμ​=0.15) and BD fit (ω≈1000 \omega \approx 1000 ω≈1000).

****

**5. Discussion**

The convergence of E(z) E(z) E(z) to Λ \Lambda ΛCDM for ω≥1000 \omega \geq 1000 ω≥1000 aligns with theoretical expectations, as BD approaches GR in this limit. The increase in ϕ(z) \phi(z) ϕ(z) for ω=10 \omega = 10 ω=10 implies a variation in G G G, which could be observationally tested. The best-fit ω≈1000 \omega \approx 1000 ω≈1000 suggests that BD is compatible with Λ \Lambda ΛCDM-based data, supporting its viability as an alternative model. Numerical challenges, such as integration stability, were mitigated but may contribute to the slight discrepancy in ω \omega ω.

**6. Conclusions**

We demonstrated that Brans-Dicke theory can describe cosmological evolution consistently with Λ \Lambda ΛCDM for ω≥1000 \omega \geq 1000 ω≥1000, with deviations for smaller ω \omega ω. The best-fit ω≈1000 \omega \approx 1000 ω≈1000 from simulated μ(z) \mu(z) μ(z) data supports BD's potential as a viable model. Future work should incorporate real observational data (e.g., supernovae) and explore additional parameters.

**7. Impact and Applications**

**Impact:**

* This study contributes to the exploration of alternative gravity theories amid current cosmological tensions (e.g., the H0 H\_0 H0​ discrepancy). It suggests that BD could be tested through variations in G G G, motivating further observational constraints.
* The compatibility with simulated data (ω≈1000 \omega \approx 1000 ω≈1000) highlights BD's potential to describe the universe's expansion, fostering interest in alternative models to address unresolved cosmological phenomena.
* The reproducible numerical methodology encourages transparency and collaboration in the scientific community, particularly in computational cosmology.

**Applications:**

* **Observational Tests of Alternative Gravity:** Results can guide future experiments to detect variations in G G G or deviations from Λ \Lambda ΛCDM, such as using supernova type Ia distance measurements or cosmic structure growth data.
* **Dark Energy Modeling:** The framework can be extended to model complex dark energy forms (e.g., w0≠−1 w\_0 \neq -1 w0​=−1), aiding in understanding cosmic acceleration.
* **Advanced Cosmological Simulations:** The Python codes can be adapted for more complex simulations, incorporating real data (e.g., from DESI, LSST, or Euclid) or additional parameters (e.g., radiation or curvature).
* **Educational Value:** The numerical approach and clear visualizations serve as educational tools for students of cosmology and theoretical physics, illustrating computational methods in cosmology.

**8. Objective and Relevance for Readers**

**Objective:**  
This research, conducted by Miguel Ángel Percudani (DNI 27603884) from Buenos Aires, Argentina, aims to analyze the cosmological behavior of Brans-Dicke (BD) theory as an alternative to General Relativity, comparing it to the standard Λ \Lambda ΛCDM model. Through Python-based numerical simulations, we study the evolution of the normalized Hubble parameter E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ and the scalar field ϕ(z) \phi(z) ϕ(z) for different BD parameter values ω \omega ω, and constrain ω \omega ω using simulated distance modulus μ(z) \mu(z) μ(z) data. The goal is to evaluate whether BD can reproduce simulated cosmological observations and explore its deviations from Λ \Lambda ΛCDM, providing a foundation for future observational tests of alternative gravity theories.

**Relevance to Current Science:**

* **Exploration of Alternative Gravity Theories:** BD introduces a scalar field ϕ \phi ϕ, making G G G variable (G∝1/ϕ G \propto 1/\phi G∝1/ϕ). This study shows that BD approaches Λ \Lambda ΛCDM for large ω≥1000 \omega \geq 1000 ω≥1000, but exhibits significant deviations for small ω=10 \omega = 10 ω=10, offering predictions that could be tested to probe the limits of GR.
* **Addressing Cosmological Tensions:** Amid current tensions, such as the H0 H\_0 H0​ discrepancy between local observations (e.g., type Ia supernovae) and cosmic microwave background (CMB) inferences, BD's variable G G G could provide insights into resolving these issues.
* **Contribution to Scalar-Tensor Theories:** BD is a classic example of scalar-tensor theories, a vibrant field in theoretical physics. This work supports the development of broader models (e.g., Horndeski theories) that explore dark energy and cosmic inflation.

**9. Original and Adapted Equations for This Research**

**Original Brans-Dicke Equations:**  
The fundamental BD equations in a flat FLRW universe are (Weinberg, 1972; Brans & Dicke, 1961):

1. Modified Friedmann Equation:

H2=8πρ3ϕ−ϕ˙ϕH+ω6(ϕ˙ϕ)2H^2 = \frac{8\pi \rho}{3 \phi} - \frac{\dot{\phi}}{\phi} H + \frac{\omega}{6} \left( \frac{\dot{\phi}}{\phi} \right)^2H2=3ϕ8πρ​−ϕϕ˙​​H+6ω​(ϕϕ˙​​)2

* H=a˙/a H = \dot{a}/a H=a˙/a: Hubble parameter (a a a is the scale factor).
* ρ=ρm+ρDE \rho = \rho\_m + \rho\_{DE} ρ=ρm​+ρDE​: Total density (matter + dark energy).
* ϕ \phi ϕ: Scalar field (G∝1/ϕ G \propto 1/\phi G∝1/ϕ).
* ϕ˙ \dot{\phi} ϕ˙​: Time derivative of ϕ \phi ϕ.
* ω \omega ω: BD coupling parameter.

1. Scalar Field Equation:

ϕ¨+3Hϕ˙=8π2ω+3(ρ−3p)\ddot{\phi} + 3 H \dot{\phi} = \frac{8\pi}{2\omega + 3} (\rho - 3p)ϕ¨​+3Hϕ˙​=2ω+38π​(ρ−3p)

* ϕ¨ \ddot{\phi} ϕ¨​: Second time derivative of ϕ \phi ϕ.
* p=pm+pDE p = p\_m + p\_{DE} p=pm​+pDE​: Total pressure (pm=0 p\_m = 0 pm​=0 for matter, pDE=w0ρDE p\_{DE} = w\_0 \rho\_{DE} pDE​=w0​ρDE​ for dark energy).
* ρ−3p \rho - 3p ρ−3p: Trace of the energy-momentum tensor, acting as the source for ϕ \phi ϕ.

**Adapted Equations for This Study:**  
To solve the equations numerically and apply them to cosmological analysis, we made the following adaptations:

1. **Change of Variable from t t t to z z z:** Redshift z z z is defined as 1+z=1/a 1 + z = 1/a 1+z=1/a. Time derivatives are transformed:

ddt=−(1+z)Hddz,d2dt2=(1+z)2H2d2dz2+(1+z)(dHdt+H)ddz\frac{d}{dt} = -(1+z) H \frac{d}{dz}, \quad \frac{d^2}{dt^2} = (1+z)^2 H^2 \frac{d^2}{dz^2} + (1+z) \left( \frac{dH}{dt} + H \right) \frac{d}{dz}dtd​=−(1+z)Hdzd​,dt2d2​=(1+z)2H2dz2d2​+(1+z)(dtdH​+H)dzd​

1. **Definition of E(z) E(z) E(z):** We introduce E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​, with H0=70 km/s/Mpc H\_0 = 70 \, \text{km/s/Mpc} H0​=70km/s/Mpc. For Λ \Lambda ΛCDM:

E(z)=Ωm0(1+z)3+ΩDE0E(z) = \sqrt{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0}}E(z)=Ωm0​(1+z)3+ΩDE0​​

1. **Adapted Friedmann Equation:** Substituting H=H0E H = H\_0 E H=H0​E, ϕ˙=−(1+z)Hdϕdz \dot{\phi} = -(1+z) H \frac{d\phi}{dz} ϕ˙​=−(1+z)Hdzdϕ​, and ρ=Ωm0(1+z)3+ΩDE0(1+z)3(1+w0) \rho = \Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} ρ=Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​), the Friedmann equation becomes:

E2=Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)ϕϕ−1+z3ϕ′−ω(1+z)26ϕ(ϕ′)2E^2 = \frac{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} \phi}{\phi - \frac{1+z}{3} \phi' - \omega \frac{(1+z)^2}{6 \phi} (\phi')^2}E2=ϕ−31+z​ϕ′−ω6ϕ(1+z)2​(ϕ′)2Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)ϕ​

* Numerical stabilization was added (e.g., denom=max⁡(denom,1×10−10) \text{denom} = \max(\text{denom}, 1 \times 10^{-10}) denom=max(denom,1×10−10)) to avoid division by zero.

1. **Adapted Scalar Field Equation:** The scalar field equation is rewritten as:

ϕ′′+(E′E+31+z)ϕ′=−8π(2ω+3)(1+z)2E2[Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)(1+3w0)]\phi'' + \left( \frac{E'}{E} + \frac{3}{1+z} \right) \phi' = -\frac{8\pi}{(2\omega + 3)(1+z)^2 E^2} [\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} (1 + 3 w\_0)]ϕ′′+(EE′​+1+z3​)ϕ′=−(2ω+3)(1+z)2E28π​[Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)(1+3w0​)]

* Rewritten as a first-order system for solve\_ivp: dϕdz=ϕ′ \frac{d\phi}{dz} = \phi' dzdϕ​=ϕ′, dϕ′dz=ϕ′′ \frac{d\phi'}{dz} = \phi'' dzdϕ′​=ϕ′′.
* Initial conditions: ϕ(0)=1 \phi(0) = 1 ϕ(0)=1, ϕ′(0)=0 \phi'(0) = 0 ϕ′(0)=0.
* E′ E' E′ was approximated using np.gradient(E, z).

1. **Distance Modulus Calculation:** The luminosity distance and distance modulus are:

dL(z)=(1+z)cH0∫0zdz′E(z′),μ(z)=5log⁡10(dL(z)10 pc)d\_L(z) = (1+z) \frac{c}{H\_0} \int\_0^z \frac{dz'}{E(z')}, \quad \mu(z) = 5 \log\_{10} \left( \frac{d\_L(z)}{10 \, \text{pc}} \right)dL​(z)=(1+z)H0​c​∫0z​E(z′)dz′​,μ(z)=5log10​(10pcdL​(z)​)

* The integral was computed using scipy.integrate.quadrature for stability.
* Simplified as μ(z)=5log⁡10(dL(z)/1×10−5) \mu(z) = 5 \log\_{10} (d\_L(z)/1 \times 10^{-5}) μ(z)=5log10​(dL​(z)/1×10−5) (with 1 Mpc=1×105 pc 1 \, \text{Mpc} = 1 \times 10^5 \, \text{pc} 1Mpc=1×105pc).

**Summary of Adaptations:**

* Changed variable from cosmic time t t t to redshift z z z.
* Introduced E(z) E(z) E(z) for normalization.
* Added numerical stabilization to handle singularities.
* Used robust numerical methods (quadrature, solve\_ivp).
* Applied interpolation (np.interp) for intermediate evaluations of E(z) E(z) E(z).

**Acknowledgments**

I thank Grok 3, created by xAI, for its assistance with numerical simulations, code development, and manuscript preparation. This work was conducted by Miguel Ángel Percudani in Buenos Aires, Argentina, as the intellectual author of this research.

**References**

Brans, C., & Dicke, R. H. (1961). Mach's Principle and a Relativistic Theory of Gravitation. *Physical Review*, 124(3), 925–935.  
Planck Collaboration. (2020). Planck 2018 results. VI. Cosmological parameters. *Astronomy & Astrophysics*, 641, A6.  
Riess, A. G., et al. (1998). Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant. *The Astronomical Journal*, 116(3), 1009–1038.  
Weinberg, S. (1972). *Gravitation and Cosmology: Principles and Applications of the General Theory of Relativity*. Wiley.

**Supplementary Material**

The Python codes for generating Figures 1 and 2 are available.

Supplementary Material:

**Codes python 1:**

import numpy as np

import matplotlib.pyplot as plt

from scipy.integrate import solve\_ivp

# Constantes

H0 = 70 # km/s/Mpc

Omega\_m0 = 0.3

Omega\_DE0 = 0.7

w0 = -1

# Ecuaciones de Brans-Dicke

def bd\_equations(z, y, omega):

phi, phi\_prime = y

is\_scalar = np.isscalar(z)

z\_array = np.atleast\_1d(z)

phi\_array = np.atleast\_1d(phi)

phi\_prime\_array = np.atleast\_1d(phi\_prime)

denom = phi\_array - (1 + z\_array) \* phi\_prime\_array / 3 - omega \* (1 + z\_array)\*\*2 \* phi\_prime\_array\*\*2 / (6 \* phi\_array)

denom = np.where(denom <= 0, 1e-10, denom)

E = np.sqrt((Omega\_m0 \* (1 + z\_array)\*\*3 + Omega\_DE0 \* (1 + z\_array)\*\*(3 \* (1 + w0)) \* phi\_array) / denom)

E = np.where(E <= 0, 1e-10, E)

if is\_scalar:

E\_prime = 0

else:

E\_prime = np.gradient(E, z\_array)

E\_prime = E\_prime[0] if len(E\_prime) == 1 else E\_prime

dphi\_dz = phi\_prime

T = Omega\_m0 \* (1 + z\_array)\*\*3 + Omega\_DE0 \* (1 + z\_array)\*\*(3 \* (1 + w0)) \* (1 + 3 \* w0)

d2phi\_dz2 = -(E\_prime / E + 3 / (1 + z\_array)) \* phi\_prime\_array - (

8 \* np.pi / (2 \* omega + 3) \* T / ((1 + z\_array)\*\*2 \* E\*\*2)

)

if is\_scalar:

dphi\_dz = float(dphi\_dz)

d2phi\_dz2 = float(d2phi\_dz2)

else:

dphi\_dz = np.atleast\_1d(dphi\_dz).astype(float)

d2phi\_dz2 = np.atleast\_1d(d2phi\_dz2).astype(float)

if dphi\_dz.shape != d2phi\_dz2.shape:

dphi\_dz = np.full\_like(d2phi\_dz2, dphi\_dz[0])

return np.array([dphi\_dz, d2phi\_dz2], dtype=float)

# Datos para el eje z

z = np.linspace(0, 1.6, 100)

# Configuración de la figura con dos paneles

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 5))

# Calcular y graficar E(z) y phi(z) para diferentes omega

for omega in [10, 1000, 5000]:

sol = solve\_ivp(

bd\_equations, [0, 1.6], [1, 0], args=(omega,), t\_eval=z,

method='LSODA', rtol=1e-6, atol=1e-6

)

if not sol.success:

print(f"Advertencia: la integración falló para omega={omega}. Mensaje: {sol.message}")

continue

phi = sol.y[0]

phi\_prime = sol.y[1]

E = np.zeros\_like(z)

for i, zi in enumerate(z):

denom = phi[i] - (1 + zi) \* phi\_prime[i] / 3 - omega \* (1 + zi)\*\*2 \* phi\_prime[i]\*\*2 / (6 \* phi[i])

if denom <= 0:

denom = 1e-10

E[i] = np.sqrt((Omega\_m0 \* (1 + zi)\*\*3 + Omega\_DE0 \* (1 + zi)\*\*(3 \* (1 + w0)) \* phi[i]) / denom)

ax1.plot(z, E, label=f'$\omega={omega}$')

ax2.plot(z, phi, label=f'$\omega={omega}$')

# Graficar E(z) para Lambda-CDM

E\_lcdm = np.sqrt(Omega\_m0 \* (1 + z)\*\*3 + Omega\_DE0)

ax1.plot(z, E\_lcdm, 'g--', label='$\Lambda$CDM')

# Configurar etiquetas y leyendas

ax1.set\_xlabel('Corrimiento al rojo ($z$)')

ax1.set\_ylabel('$E(z)$')

ax1.legend()

ax2.set\_xlabel('Corrimiento al rojo ($z$)')

ax2.set\_ylabel('$\phi(z)$')

ax2.legend()

# Ajustar diseño y guardar como PNG

plt.tight\_layout()

plt.savefig('figure1.png')

print("Gráfico 1 generado y guardado como 'figure1.png'. Descárgala desde la sección 'Files'.")

**Codes Python 2:**

import numpy as np

import matplotlib.pyplot as plt

from scipy.integrate import solve\_ivp, quadrature

from scipy.optimize import minimize

# Constantes

H0 = 70 # km/s/Mpc

c = 299792.458 # km/s

Omega\_m0 = 0.3

Omega\_DE0 = 0.7

w0 = -1

sigma\_mu = 0.15

# Ecuaciones de Brans-Dicke

def bd\_equations(z, y, omega):

phi, phi\_prime = y

is\_scalar = np.isscalar(z)

z\_array = np.atleast\_1d(z)

phi\_array = np.atleast\_1d(phi)

phi\_prime\_array = np.atleast\_1d(phi\_prime)

denom = phi\_array - (1 + z\_array) \* phi\_prime\_array / 3 - omega \* (1 + z\_array)\*\*2 \* phi\_prime\_array\*\*2 / (6 \* phi\_array)

denom = np.where(denom <= 0, 1e-10, denom)

E = np.sqrt((Omega\_m0 \* (1 + z\_array)\*\*3 + Omega\_DE0 \* (1 + z\_array)\*\*(3 \* (1 + w0)) \* phi\_array) / denom)

E = np.where(E <= 0, 1e-10, E)

if is\_scalar:

E\_prime = 0

else:

E\_prime = np.gradient(E, z\_array)

E\_prime = E\_prime[0] if len(E\_prime) == 1 else E\_prime

dphi\_dz = phi\_prime

T = Omega\_m0 \* (1 + z\_array)\*\*3 + Omega\_DE0 \* (1 + z\_array)\*\*(3 \* (1 + w0)) \* (1 + 3 \* w0)

d2phi\_dz2 = -(E\_prime / E + 3 / (1 + z\_array)) \* phi\_prime\_array - (

8 \* np.pi / (2 \* omega + 3) \* T / ((1 + z\_array)\*\*2 \* E\*\*2)

)

if is\_scalar:

dphi\_dz = float(dphi\_dz)

d2phi\_dz2 = float(d2phi\_dz2)

else:

dphi\_dz = np.atleast\_1d(dphi\_dz).astype(float)

d2phi\_dz2 = np.atleast\_1d(d2phi\_dz2).astype(float)

if dphi\_dz.shape != d2phi\_dz2.shape:

dphi\_dz = np.full\_like(d2phi\_dz2, dphi\_dz[0])

return np.array([dphi\_dz, d2phi\_dz2], dtype=float)

# Distancia de luminosidad (usar quadrature)

def dL(z, E\_func):

integrand = lambda zp: 1 / E\_func(zp)

integral = quadrature(integrand, 0, z, tol=1e-8, rtol=1e-8)[0]

return (1 + z) \* c / H0 \* integral

# Módulo de distancia

def mu(z, E\_func):

dl = dL(z, E\_func)

return 5 \* np.log10(dl / 1e-5) # 1 Mpc = 1e5 pc

# Simulación de datos

np.random.seed(42)

z\_data = np.linspace(0.1, 1.5, 50)

E\_lcdm = lambda z: np.sqrt(Omega\_m0 \* (1 + z)\*\*3 + Omega\_DE0)

mu\_sim = np.array([mu(z, E\_lcdm) for z in z\_data])

mu\_obs = mu\_sim + np.random.normal(0, sigma\_mu, len(z\_data))

# Función chi^2 para ajuste

def chi2(omega):

sol = solve\_ivp(

bd\_equations, [0, 1.6], [1, 0], args=(omega,), t\_eval=z\_data,

method='LSODA', rtol=1e-6, atol=1e-6

)

if not sol.success:

return np.inf

phi = sol.y[0]

phi\_prime = sol.y[1]

E = np.zeros\_like(z\_data)

for i, zi in enumerate(z\_data):

denom = phi[i] - (1 + zi) \* phi\_prime[i] / 3 - omega \* (1 + zi)\*\*2 \* phi\_prime[i]\*\*2 / (6 \* phi[i])

if denom <= 0:

denom = 1e-10

E[i] = np.sqrt((Omega\_m0 \* (1 + zi)\*\*3 + Omega\_DE0 \* (1 + zi)\*\*(3 \* (1 + w0)) \* phi[i]) / denom)

mu\_bd = np.array([mu(zi, lambda zp: np.interp(zp, z\_data, E)) for zi in z\_data])

return np.sum(((mu\_obs - mu\_bd) / sigma\_mu)\*\*2)

# Ajuste de omega

res = minimize(chi2, 1000, method='L-BFGS-B', bounds=[(10, 5000)])

omega\_opt = res.x[0]

# Calcular mu(z) para omega\_opt

sol = solve\_ivp(

bd\_equations, [0, 1.6], [1, 0], args=(omega\_opt,), t\_eval=z\_data,

method='LSODA', rtol=1e-6, atol=1e-6

)

if not sol.success:

print(f"Error: la integración falló para omega={omega\_opt}. Mensaje: {sol.message}")

else:

phi = sol.y[0]

phi\_prime = sol.y[1]

E = np.zeros\_like(z\_data)

for i, zi in enumerate(z\_data):

denom = phi[i] - (1 + zi) \* phi\_prime[i] / 3 - omega\_opt \* (1 + zi)\*\*2 \* phi\_prime[i]\*\*2 / (6 \* phi[i])

if denom <= 0:

denom = 1e-10

E[i] = np.sqrt((Omega\_m0 \* (1 + zi)\*\*3 + Omega\_DE0 \* (1 + zi)\*\*(3 \* (1 + w0)) \* phi[i]) / denom)

mu\_bd = np.array([mu(zi, lambda zp: np.interp(zp, z\_data, E)) for zi in z\_data])

# Configuración de la figura

plt.figure(figsize=(8, 6))

plt.errorbar(z\_data, mu\_obs, yerr=sigma\_mu, fmt='o', label='Datos Simulados')

plt.plot(z\_data, mu\_bd, label=f'BD ($\omega={omega\_opt:.0f}$)')

plt.xlabel('Corrimiento al rojo ($z$)')

plt.ylabel('Módulo de Distancia ($\mu$)')

plt.legend()

plt.tight\_layout()

plt.savefig('figure2.png')

print("Gráfico 2 generado y guardado como 'figure2.png'. Descárgala desde la sección 'Files'.")

**1. Descriptive Objective for Your Readers**

**Objective:**  
This research, conducted by Miguel Ángel Percudani (DNI 27603884) from Buenos Aires, Argentina, aims to analyze the cosmological behavior of Brans-Dicke (BD) theory as an alternative to General Relativity, comparing it to the standard Λ \Lambda ΛCDM model. Through Python-based numerical simulations, we study the evolution of the normalized Hubble parameter E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​ and the scalar field ϕ(z) \phi(z) ϕ(z) for different values of the BD parameter ω \omega ω, and constrain ω \omega ω using simulated distance modulus μ(z) \mu(z) μ(z) data. The goal is to evaluate whether BD can reproduce simulated cosmological observations and explore its deviations from Λ \Lambda ΛCDM, providing a foundation for future observational tests of alternative gravity theories.

**2. Importance and Influence of This Research on Current Science**

**Importance:**

* **Exploration of Alternative Gravity Theories:** Brans-Dicke theory extends General Relativity by introducing a scalar field ϕ \phi ϕ, making the gravitational constant G G G variable (G∝1/ϕ G \propto 1/\phi G∝1/ϕ). This research demonstrates that BD can approach the standard Λ \Lambda ΛCDM model for large values of ω \omega ω (e.g., ω≥1000 \omega \geq 1000 ω≥1000), but also shows significant deviations for small ω \omega ω (e.g., ω=10 \omega = 10 ω=10). This is crucial because it suggests that BD could offer predictions distinct from those of General Relativity, which could be detected through cosmological observations, helping to test the limits of Einstein’s theory.
* **Compatibility with Simulated Data:** By constraining ω≈1000 \omega \approx 1000 ω≈1000 using simulated distance modulus μ(z) \mu(z) μ(z) data, this research shows that BD is a viable model for describing the universe’s expansion, at least in a simulated context. This reinforces interest in alternative theories that might explain cosmological phenomena not fully addressed by Λ \Lambda ΛCDM, such as tensions in the value of H0 H\_0 H0​ (the Hubble constant) observed in recent data.
* **Reproducible Numerical Method:** The research employs numerical simulations in Python, providing codes that other scientists can use to replicate or extend the analysis. This fosters transparency and collaboration within the scientific community, especially at a time when computational simulations are key to testing cosmological theories.

**Influence on Current Science:**

* **Context of Cosmological Tensions:** In current cosmology (2025), there are significant tensions, such as the discrepancy in the value of H0 H\_0 H0​ between local observations (e.g., type Ia supernovae) and those inferred from the cosmic microwave background (CMB). Alternative models like Brans-Dicke could help resolve these tensions by introducing different gravitational dynamics, such as a variable G G G. This research contributes to the debate by showing how BD behaves compared to Λ \Lambda ΛCDM, laying the groundwork for future studies with real data.
* **Inspiration for New Observations:** The results suggest that deviations of BD from Λ \Lambda ΛCDM are more pronounced for small ω \omega ω. This may motivate cosmologists to seek evidence of a variable G G G in observations, such as supernova luminosity distances, cosmic structure growth, or CMB anisotropies.
* **Contribution to Scalar-Tensor Theories:** BD is a classic example of scalar-tensor theories, an active field in current theoretical physics. By studying BD, this research contributes to the development of more general models (e.g., Horndeski theories or BD extensions), which are candidates for explaining dark energy and cosmic inflation.

**3. Potential Applications of This Research**

* **Observational Tests of Alternative Gravity:** The results can guide future experiments to detect variations in G G G or deviations from Λ \Lambda ΛCDM. For example:
  + Observations of type Ia supernovae at different redshifts (z z z) could be used to measure μ(z) \mu(z) μ(z) and compare with the BD predictions fitted in this research.
  + Studies of structure growth (e.g., galaxy clusters) could search for effects of a variable G G G, which BD predicts for small ω \omega ω.
* **Dark Energy Modeling:** The research explores BD with a dark energy equation of state w0=−1 w\_0 = -1 w0​=−1, but additional plots (available as supplementary material) vary w0 w\_0 w0​. This could be applied to model more complex forms of dark energy (e.g., w0≠−1 w\_0 \neq -1 w0​=−1) in alternative theories, aiding in understanding the nature of cosmic acceleration.
* **Advanced Cosmological Simulations:** The developed Python codes can be adapted for more complex simulations, incorporating real data (e.g., from DESI, LSST, or Euclid) or additional parameters (e.g., radiation or curvature). This would be useful for computational cosmology projects seeking to test theories beyond Λ \Lambda ΛCDM.
* **Education and Training:** With its numerical approach and clear visualizations, this research can serve as educational material for students of cosmology or theoretical physics, demonstrating how computational methods are applied to solve differential equations and analyze cosmological models.

**4. Original and Modified/Adapted Equations for This Research**

Below, I detail the original equations of Brans-Dicke theory and how they were modified or adapted for numerical solution and cosmological analysis in this research.

**Original Brans-Dicke Equations:**  
The fundamental equations of Brans-Dicke theory in a flat Friedmann-Lemaître-Robertson-Walker (FLRW) universe, as per standard literature (Weinberg, 1972; Brans & Dicke, 1961), are:

1. **Modified Friedmann Equation:**

H2=8πρ3ϕ−ϕ˙ϕH+ω6(ϕ˙ϕ)2H^2 = \frac{8\pi \rho}{3 \phi} - \frac{\dot{\phi}}{\phi} H + \frac{\omega}{6} \left( \frac{\dot{\phi}}{\phi} \right)^2H2=3ϕ8πρ​−ϕϕ˙​​H+6ω​(ϕϕ˙​​)2

* H=a˙/a H = \dot{a}/a H=a˙/a: Hubble parameter (a a a is the scale factor).
* ρ=ρm+ρDE \rho = \rho\_m + \rho\_{DE} ρ=ρm​+ρDE​: Total density (matter + dark energy).
* ϕ \phi ϕ: Scalar field (related to G∝1/ϕ G \propto 1/\phi G∝1/ϕ).
* ϕ˙ \dot{\phi} ϕ˙​: Time derivative of ϕ \phi ϕ.
* ω \omega ω: Brans-Dicke coupling parameter.

1. **Scalar Field Equation:**

ϕ¨+3Hϕ˙=8π2ω+3(ρ−3p)\ddot{\phi} + 3 H \dot{\phi} = \frac{8\pi}{2\omega + 3} (\rho - 3p)ϕ¨​+3Hϕ˙​=2ω+38π​(ρ−3p)

* ϕ¨ \ddot{\phi} ϕ¨​: Second time derivative of ϕ \phi ϕ.
* p=pm+pDE p = p\_m + p\_{DE} p=pm​+pDE​: Total pressure (pm=0 p\_m = 0 pm​=0 for matter, pDE=w0ρDE p\_{DE} = w\_0 \rho\_{DE} pDE​=w0​ρDE​ for dark energy).
* ρ−3p \rho - 3p ρ−3p: Trace of the energy-momentum tensor, acting as the source for ϕ \phi ϕ.  
  These original equations are expressed in terms of cosmic time t t t and are the standard forms presented in Brans-Dicke theory for an FLRW universe.

**Modified/Adapted Equations for This Research:**  
To solve the equations numerically and apply them to cosmological analysis, several modifications and adaptations were made:

1. **Change of Variable from t t t to z z z:**
   * In observational cosmology, it is more practical to work with redshift z z z, defined as 1+z=1/a 1 + z = 1/a 1+z=1/a, where a a a is the normalized scale factor (a=1 a = 1 a=1 today, z=0 z = 0 z=0).
   * Relationship of time derivatives to derivatives with respect to z z z:

ddt=−(1+z)Hddz,d2dt2=(1+z)2H2d2dz2+(1+z)(dHdt+H)ddz\frac{d}{dt} = -(1+z) H \frac{d}{dz}, \quad \frac{d^2}{dt^2} = (1+z)^2 H^2 \frac{d^2}{dz^2} + (1+z) \left( \frac{dH}{dt} + H \right) \frac{d}{dz}dtd​=−(1+z)Hdzd​,dt2d2​=(1+z)2H2dz2d2​+(1+z)(dtdH​+H)dzd​

Using dHdt=HdHdzdzdt=−H(1+z)HdHdz \frac{dH}{dt} = H \frac{dH}{dz} \frac{dz}{dt} = -H (1+z) H \frac{dH}{dz} dtdH​=HdzdH​dtdz​=−H(1+z)HdzdH​, this simplifies for numerical solution.

1. **Definition of E(z) E(z) E(z):**
   * We introduce E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​, where H0=70 km/s/Mpc H\_0 = 70 \, \text{km/s/Mpc} H0​=70km/s/Mpc is the present-day Hubble parameter. This normalizes H(z) H(z) H(z) and facilitates comparison with Λ \Lambda ΛCDM.
   * For Λ \Lambda ΛCDM, we use:

E(z)=Ωm0(1+z)3+ΩDE0E(z) = \sqrt{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0}}E(z)=Ωm0​(1+z)3+ΩDE0​​

with Ωm0=0.3 \Omega\_{m0} = 0.3 Ωm0​=0.3, ΩDE0=0.7 \Omega\_{DE0} = 0.7 ΩDE0​=0.7, w0=−1 w\_0 = -1 w0​=−1.

1. **Adapted Friedmann Equation:**
   * Substituting H=H0E H = H\_0 E H=H0​E, ϕ˙=dϕdt=−(1+z)Hdϕdz \dot{\phi} = \frac{d\phi}{dt} = -(1+z) H \frac{d\phi}{dz} ϕ˙​=dtdϕ​=−(1+z)Hdzdϕ​, and ρ=ρm+ρDE=Ωm0(1+z)3+ΩDE0(1+z)3(1+w0) \rho = \rho\_m + \rho\_{DE} = \Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} ρ=ρm​+ρDE​=Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​), the Friedmann equation transforms into:

E2=Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)ϕϕ−1+z3ϕ′−ω(1+z)26ϕ(ϕ′)2E^2 = \frac{\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} \phi}{\phi - \frac{1+z}{3} \phi' - \omega \frac{(1+z)^2}{6 \phi} (\phi')^2}E2=ϕ−31+z​ϕ′−ω6ϕ(1+z)2​(ϕ′)2Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)ϕ​

* ϕ′=dϕdz \phi' = \frac{d\phi}{dz} ϕ′=dzdϕ​.
* This form allows for the numerical solution of E(z) E(z) E(z) once ϕ(z) \phi(z) ϕ(z) and ϕ′(z) \phi'(z) ϕ′(z) are known.
* **Adaptation:** Numerical stabilization terms were added (e.g., denom=max⁡(denom,1×10−10) \text{denom} = \max(\text{denom}, 1 \times 10^{-10}) denom=max(denom,1×10−10)) to avoid division by zero in the denominator during integration.

1. **Adapted Scalar Field Equation:**
   * Applying the variable change and substituting H H H, ϕ˙ \dot{\phi} ϕ˙​, and ϕ¨ \ddot{\phi} ϕ¨​, the scalar field equation becomes:

ϕ′′+(E′E+31+z)ϕ′=−8π(2ω+3)(1+z)2E2[Ωm0(1+z)3+ΩDE0(1+z)3(1+w0)(1+3w0)]\phi'' + \left( \frac{E'}{E} + \frac{3}{1+z} \right) \phi' = -\frac{8\pi}{(2\omega + 3)(1+z)^2 E^2} [\Omega\_{m0} (1+z)^3 + \Omega\_{DE0} (1+z)^{3(1+w\_0)} (1 + 3 w\_0)]ϕ′′+(EE′​+1+z3​)ϕ′=−(2ω+3)(1+z)2E28π​[Ωm0​(1+z)3+ΩDE0​(1+z)3(1+w0​)(1+3w0​)]

* ϕ′′=d2ϕdz2 \phi'' = \frac{d^2 \phi}{dz^2} ϕ′′=dz2d2ϕ​, E′=dEdz E' = \frac{dE}{dz} E′=dzdE​.
* **Adaptations:**
  + Rewritten as a system of first-order differential equations for use with solve\_ivp: dϕdz=ϕ′ \frac{d\phi}{dz} = \phi' dzdϕ​=ϕ′, dϕ′dz=ϕ′′ \frac{d\phi'}{dz} = \phi'' dzdϕ′​=ϕ′′.
  + Initial conditions (ϕ(0)=1 \phi(0) = 1 ϕ(0)=1, ϕ′(0)=0 \phi'(0) = 0 ϕ′(0)=0) were implemented for numerical solution.
  + E′≈np.gradient(E,z) E' \approx \text{np.gradient}(E, z) E′≈np.gradient(E,z) was used to compute the derivative of E(z) E(z) E(z), required in the equation.

1. **Calculation of μ(z) \mu(z) μ(z) and dL(z) d\_L(z) dL​(z):**
   * **Original:** The luminosity distance dL(z) d\_L(z) dL​(z) is defined as:

dL(z)=(1+z)cH0∫0zdz′E(z′)d\_L(z) = (1+z) \frac{c}{H\_0} \int\_0^z \frac{dz'}{E(z')}dL​(z)=(1+z)H0​c​∫0z​E(z′)dz′​

The distance modulus is:

μ(z)=5log⁡10(dL(z)10 pc)\mu(z) = 5 \log\_{10} \left( \frac{d\_L(z)}{10 \, \text{pc}} \right)μ(z)=5log10​(10pcdL​(z)​)

* **Adaptations:**
  + The integral ∫0zdz′E(z′) \int\_0^z \frac{dz'}{E(z')} ∫0z​E(z′)dz′​ was computed numerically using scipy.integrate.quadrature for greater stability, instead of quad, due to convergence issues.
  + Interpolation (np.interp) was used to evaluate E(z) E(z) E(z) at intermediate points during integration.
  + Simplified as μ(z)=5log⁡10(dL(z)/1×10−5) \mu(z) = 5 \log\_{10} (d\_L(z)/1 \times 10^{-5}) μ(z)=5log10​(dL​(z)/1×10−5), adjusting units (1 Mpc=1×105 pc 1 \, \text{Mpc} = 1 \times 10^5 \, \text{pc} 1Mpc=1×105pc).

**Summary of Modifications:**

* **Variable Change:** From cosmic time t t t to redshift z z z, to facilitate comparison with cosmological observations.
* **Normalization:** Introduction of E(z)=H(z)/H0 E(z) = H(z)/H\_0 E(z)=H(z)/H0​, to normalize the equations and compare with Λ \Lambda ΛCDM.
* **Numerical Stabilization:** Terms were added to avoid singularities (e.g., near-zero denominators), and robust methods (quadrature, solve\_ivp) were used to solve the equations.
* **Initial Conditions:** Assumed ϕ(0)=1 \phi(0) = 1 ϕ(0)=1, ϕ′(0)=0 \phi'(0) = 0 ϕ′(0)=0, to reflect a universe close to General Relativity at z=0 z = 0 z=0.
* **Distance Calculations:** The equations for dL(z) d\_L(z) dL​(z) and μ(z) \mu(z) μ(z) were adapted for numerical solution, optimizing integration and handling stability issues.

--------------------------------------------------------------------------------------------------------------

Dear Editors,  
I am submitting my manuscript titled "Cosmological Evolution in Brans-Dicke Theory: A Numerical Analysis" for consideration in [Journal Name]. The manuscript presents a numerical analysis of Brans-Dicke theory, comparing it to Λ \Lambda ΛCDM, and offers results relevant to modern cosmology.  
I have included the required files:

* Source file (BransDicke\_Manuscript\_English.docx)
* Figures (figure1.png, figure2.png)
* PDF (BransDicke\_Manuscript\_English.pdf)  
  The Python codes are available in a public GitHub repository at [insert GitHub link here]. I remain available for any revisions or clarifications.  
  Sincerely,  
  Miguel Ángel Percudani  
  Independent Researcher, Buenos Aires, Argentina  
  [miguel\_percudani@yahoo.com.ar](mailto:miguel_percudani@yahoo.com.ar)

https://github.com/miguelpercu/Brans-Dicke-Cosmology/tree/89f53b2f2a83e0558fd131e3a74d884b90967316#readme