# Manual for the Code to Compute Solutions and to Generate Figures for Flat Friedmann Differential Equation

Harri Ehtamo
Aalto University
harri.ehtamo@aalto.fi

Lauri Jokinen
Aalto University
lauri.jokinen@iki.fi

November 25, 2023

## Contents



# 1 Introduction

This manual describes the code that can be used to compute solutions and to generate figures for flat Friedmann differential equation with various parameter combinations; and with $\Lambda$-radiation term associated with cosmological constant. The mathematical model has been developed by Harri Ehtamo in [1] in detail. The code, designed by Lauri Jokinen, is available at GitHub [2]. We use Matlab R2021b.

The model includes a time-dependent density parameter arising as a consequence of a hypothetical radiation field. It is called $\Lambda$-radiation, and it is related to the cosmological constant term.

In Section 2 we define the Friedmann equation together with the $\Lambda$-radiation term. In Section 3 we define the parameters for the model and show how to compute unknown parameters from the flatness condition, together with other conditions. In particular, we fix the time unit of our model by defining a time unit for the Hubble constant appropriately. In Section 4 we discuss the computation of each of the 12 generated figures separately.

# 2 System of differential equations

Define, $a = a(t)$, and define $T$ by $a(T) = 1$. The $\Lambda$R-model is defined with the following equations,

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{H_T^2}{a^3}\Omega^{B+D} + \frac{H_T^2}{a^3}\alpha\Omega_t^{\Lambda\mathrm{R}} + H_T^2\Omega^{\Lambda}, \tag{1}$$

$$\Omega_t^{\Lambda\mathrm{R}} \triangleq \Omega^{\Lambda\mathrm{R}}(t) = \frac{H_T\sqrt{\Omega^{\Lambda}}}{a(t)}\int_0^t a(s)\, e^{-s\, H_T\sqrt{\Omega^{\Lambda}}}\ \mathrm{d}s, \tag{2}$$

with the initial condition, $a(0) = a_0 > 0$. Here, $\Omega^{B+D} = \Omega^B + \Omega^D$, with $\Omega^B \geq 0$, $\Omega^D \geq 0$ constants; and $H_T > 0$, $\Omega^{\Lambda} \geq 0$, $\alpha \geq 0$ constants. (Pure) Friedmann model corresponds to the case $\alpha = 0$. Equation (1) is equivalent to,

$$\dot{a} = H_T\sqrt{\frac{1}{a}\Omega^{B+D} + \frac{1}{a}\alpha\Omega_t^{\Lambda\mathrm{R}} + a^2\Omega^{\Lambda}}.$$

Then, define the integral in Equation (2) as, $b = b(t)$, which results in another ODE,

$$b = \int_0^t a(s)\, e^{-s\, H_T\sqrt{\Omega^{\Lambda}}}\ \mathrm{d}s \quad \Longleftrightarrow \quad \begin{cases} \dot{b} = a\, e^{-t\, H_T\sqrt{\Omega^{\Lambda}}}, \\ b(0) = 0, \end{cases} \tag{3}$$

and,

$$\Omega_t^{\Lambda\mathrm{R}} = \frac{H_T\sqrt{\Omega^{\Lambda}}}{a(t)}b(t).$$

Now, we have an ODE system which can be solved with, e.g., Matlab's ODE solver,

$$\begin{aligned} \dot{a} &= H_T\sqrt{\frac{1}{a}\Omega^{B+D} + \frac{1}{a}\alpha\left(\frac{H_T\sqrt{\Omega^{\Lambda}}}{a}b\right) + a^2\Omega^{\Lambda}}; \\ \dot{b} &= a\, e^{-t\, H_T\sqrt{\Omega^{\Lambda}}}; \\ a(0) &= a_0,\ b(0) = 0,\ t \geq 0. \end{aligned} \tag{4}$$

In the figures, we shift the time axis, s.t., the present time is at zero. This is done by replacing the time variable, $t$, by $t' = t - T$, where $a(T) = 1$. But for the purpose of this manual, it is sufficient to use $t$ instead of $t'$.

Above, $a(t)$, $t \geq 0$, is called the scale factor of the model. The parameters $\Omega^B$, $\Omega^D$ and $\Omega^{\Lambda}$ are the density parameters for baryonic matter, cold dark matter, and cosmological constant; the cosmological constant is defined by $\Lambda/3 \triangleq H_T^2\Omega^{\Lambda}$. The time-dependent density parameter $\Omega_t^{\Lambda\mathrm{R}}$ arises as a consequence of a radiation field and it is called $\Lambda$-radiation; the constant $\alpha$ is an extra parameter related to the amount of $\Lambda$-radiation. The constant $H_T$ is called the Hubble constant, and the Hubble parameter entering, e.g., in Figure 12, is defined by $H_t \triangleq H(t) = \dot{a}(t)/a(t)$, $t \geq 0$. Note that, since $a(T) = 1$, then $H_T = \dot{a}(T)$.

### 2.1 Solving the ODE system

Let's study the system with respect to the parameter $a_0 > 0$. Ideally, we would set $a_0 = 0$, however, the equation for $\dot{a}$ is singular at that point. However, Matlab's stiff ODE solver `ode23s` is able to solve the ODE with very small $a_0$. We shall use $a_0 = 10^{-16}$ which provides enough accuracy for our purposes.

For the sake of completeness, we should study this approximation carefully, since the system *may* behave rather differently if, say $a_0$, and $b(0) = b_0$ both are small numbers, instead of $b_0 = 0$. So, as an attempt to justify the above approximation, we simulate the system with randomly sampled initial conditions. The system's state variables are $(a, b, t)$, initial conditions are $(a_0, b_0, t_0)$, and the ideal initial conditions are $(0, 0, 0)$. In the Figure below, we have sampled random initial conditions under different upper bounds. We see that the curves, including the black curve, seem to converge as we approach the ideal initial conditions. This indicates that an ideal curve exists and the above approximation, $(10^{-16}, 0, 0)$, should indeed be close to the ideal curve.

The solver `ode23s` is versatile and provides various options. We use these options to increase the precision of the solver, see code for details. The function `ode23s` also provides a framework for finding user-defined *events* during integration. The solver then attempts to find these events with a required precision. As an example, define two such events.

Suppose, we want to solve $T$ from the equation $a(T) = 1$. This equation is trivial to define as an event. In Figure 3, we want to find the maximum point of $\Omega_t^{\Lambda\mathrm{R}}$. This is found by solving the equation,

$$\frac{d}{dt}\Omega_t^{\Lambda\mathrm{R}} = 0 \iff \frac{d}{dt}\left(\frac{b}{a}\right) = 0 \iff a\,\dot{b} - b\,\dot{a} = 0, \tag{5}$$

which we define as an event.

# 3 Defining parameters

The ODE system has parameters $H_T$, $\Omega^B$, $\Omega^D$, $\Omega^\Lambda$ and $\alpha$. Here, we provide different parameter definitions which are then later used in various Figures. To numerically solve equations in the code, we use Matlab's algorithm `fsolve`, and for unconstrained optimization we use `fminunc`. We increase the precision of these algorithms by using certain options, which are visible in the code.

### 3.1 Unit conversion for $H_T$

The ODE system's only parameter whose unit depends on time is $H_T$, and in the figures, we use the time unit of Gyr, i.e., $10^9$ years. Since the value of $H_T$ is given in unit $\mathrm{kms^{-1}Mpc^{-1}}$, we convert it to $\mathrm{Gyrs^{-1}}$, where Gyrs is $10^9$ years in seconds. With a (Julian) year of 365.25 days $= 3.15576 \cdot 10^7$ seconds, and using 2012 and 2015 IAU exact definitions for astronomical unit, 1 au $= 1.495978707 \cdot 10^{11}$ m, and for parcek, 1 pc $= 6.48/\pi \cdot 10^5$ au [3],

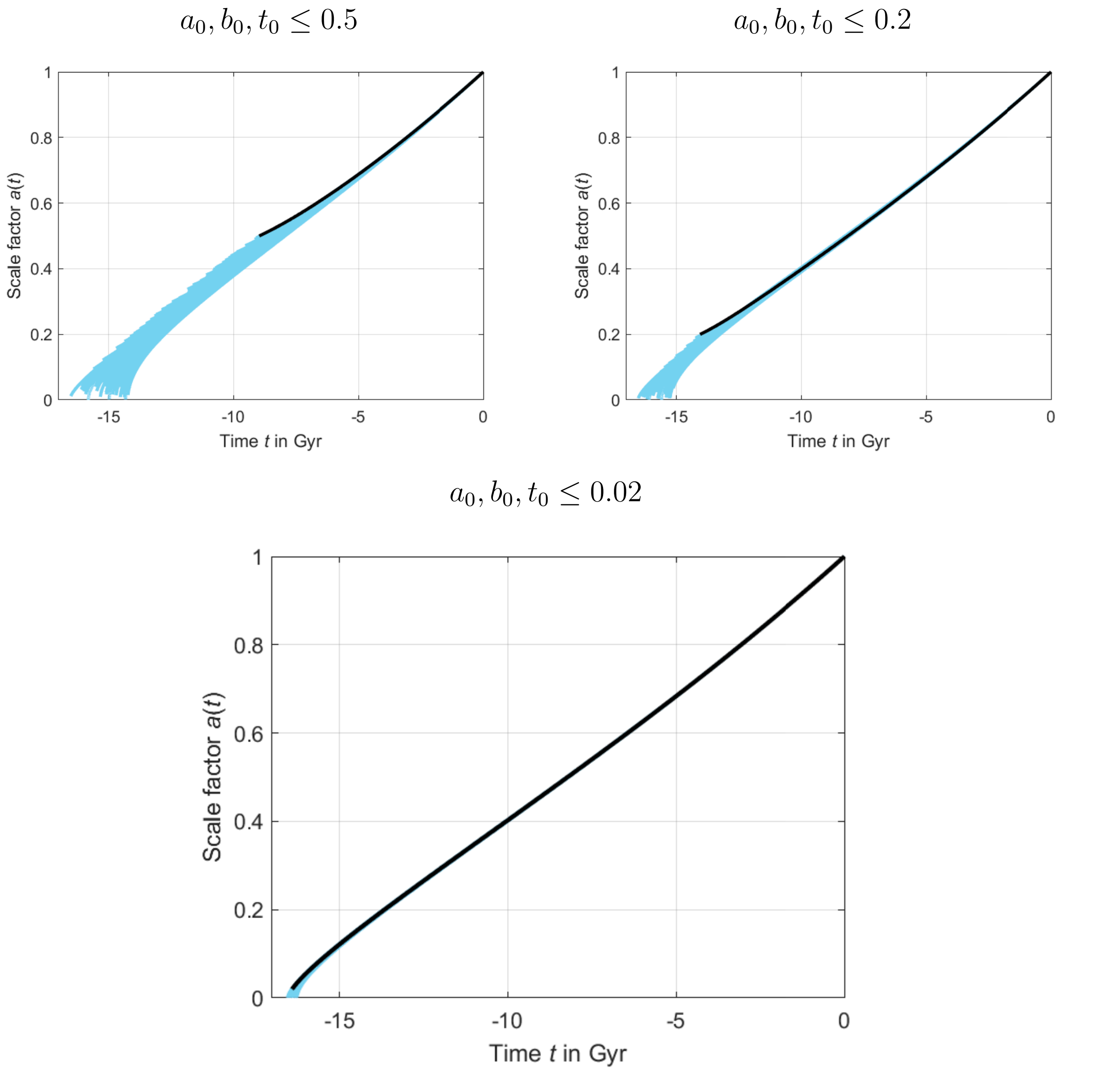


Figure. The ODE system solved with random positive initial conditions, namely, $(a_0, b_0, t_0)$. The curve in black color denotes the largest $a_0$ allowed by the constraint, and $b_0 = t_0 = 0$. The Figures are produced with parameters, $\Omega^{B+D} = 0.049$, $\alpha = 1$, $\Omega^{\Lambda} = 0.6881$, and, $H_T = 0.0688$.

we have an exactly defined conversion factor for the unit conversion (shown below). As an example, below we convert $H_T = 67.26\ \text{kms}^{-1}\text{Mpc}^{-1}$:

```
convertion_factor = 0.001 * 3.15576 / (1.495978707 * 6.48 / pi);
%%%               = 0.001022712165045695
H_T = 67.26 * convertion_factor;
%%% = 0.06878762022097345 (in unit 1/Gyrs)
```

This value for $H_T$ is used in Figures 1-10. In Figures 11-12, we also use other values for $H_T$.

## 3.2 Flatness equation

This section describes the function `flatness_solve_Omega_L` in the code. Since for given $H_T$ the equation $H_T = \dot{a}(T)$, or $a(T) = 1$, should hold, then the flatness equation,

$$\Omega^{B+D} + \Omega^{\Lambda} + \alpha\Omega_T^{\Lambda\mathrm{R}} = 1, \tag{6}$$

should hold, that for given $\Omega^{B+D}$, $\alpha$ only depends on the parameter $\Omega^{\Lambda}$. This equation for $\Omega^{\Lambda}$ is solved iteratively by using a suitable iteration method. Note, that at every iteration step, the equations (4) are solved with the current value of $\Omega^{\Lambda}$ to get $\Omega_T^{\Lambda\mathrm{R}} \triangleq \Omega^{\Lambda\mathrm{R}}(T)$, and the left-hand side of (6).

E.g., for parameter values $\alpha = 1$, $\Omega^B = 0.049$ and $\Omega^D = 0$, the flatness equation implies, $\Omega^{\Lambda} \approx 0.6881$, $\Omega_T^{\Lambda\mathrm{R}} \approx 0.2629$.

## 3.3 Benchmark model's parameters

In the benchmark F-model, in the figures F-model b, the parameters are,

$$\Omega^B = 0.049, \quad \Omega^D = 0.268, \quad \Omega^{\Lambda} = 0.683, \quad \alpha = 0.$$

In other (pure) F-models, i.e., in models with $\alpha = 0$, also other $\Omega^B$, $\Omega^D$ and $\Omega^{\Lambda}$ parameters are used. Note, that for $t = T$, $a(T) = 1$, $H_T = \dot{a}(T)$, the flatness condition $\Omega^{B+D} + \Omega^{\Lambda} = 1$ should hold for the parameters. This model implies $T^b \approx 13.800$ Gyrs.

## 3.4 Optimal $\Omega^B$

This section describes the function `optimal_Omega_B`. Fix $\alpha = 1$ and $\Omega^D = 1$, which implies the following flatness equation, $\Omega^B + \Omega^{\Lambda} + \Omega_t^{\Lambda\mathrm{R}} = 1$. We then solve the following maximization problem,

$$\begin{aligned} &\max_{\Omega^B} \Omega_T^{\Lambda\mathrm{R}}, \\ \text{s.t.} \qquad &\Omega^B + \Omega^{\Lambda} + \Omega_T^{\Lambda\mathrm{R}} = 1, \end{aligned}$$

as in Figure 6. We use Matlab's unconstrained optimization algorithm, so the constraint needs to be satisfied at each iteration, i.e., we solve the flatness equation (6) as in 3.2 at each iteration step of the optimization method. This solving method is quite slow due to the layered iterations, but it is sufficient and accurate for the purpose.

The solution to this problem is $\Omega^B_{\mathrm{opt}} \approx 0.0458$, and, $\Omega^{\Lambda\mathrm{R}}_{T,\mathrm{opt}} \approx 0.2629$.

## 3.5 Age-optimal $\Omega^D$ and $\alpha$: case 1

E.g., in Figure 8, we have two curves with $\alpha\Omega_T^{\Lambda\mathrm{R}} = c$, $c = 0.06$, or $0.16$ and the age of the universe is matched to the benchmark model's age, namely, $T^b \approx 13.800$ Gyrs, see definition in 3.3. In addition, $\Omega^B = 0.049$ is fixed. In mathematical terms, we find the parameters $\Omega^D$, $\alpha$, and $\Omega^{\Lambda}$ so that the following system of equations,

$$\begin{aligned} \alpha\Omega_T^{\Lambda\mathrm{R}} &= c, \\ T &= T^b, \\ \Omega^{B+D} + \Omega^{\Lambda} + \alpha\Omega_T^{\Lambda\mathrm{R}} &= 1, \end{aligned}$$

is satisfied.

### 3.6 Age-optimal $\Omega^D$ and $\alpha$: case 2

This case describes, e.g., the optimal $\Lambda$R-model in Figure 9, and the function `optimal_Omega_D_and_alpha` in the code. We aim to find a model with the following property,

$$\frac{\Omega^D + \alpha\Omega_T^{\Lambda\mathrm{R}}}{\Omega^B} = \frac{\Omega_{T,\mathrm{opt}}^{\Lambda\mathrm{R}}}{\Omega_{\mathrm{opt}}^B} \approx 5.7380,$$

where $\Omega_{T,\mathrm{opt}}^{\Lambda\mathrm{R}}$, $\Omega_{\mathrm{opt}}^B$ are as in 3.4. As described above, we match the age of the universe to the benchmark model's age and fix $\Omega^B = 0.049$. In mathematical terms, we solve the parameters $\Omega^D$ and $\alpha$ from the following system,

$$\begin{aligned} \frac{\Omega^D + \alpha\Omega_T^{\Lambda\mathrm{R}}}{\Omega^B} &= \frac{\Omega_{T,\mathrm{opt}}^{\Lambda\mathrm{R}}}{\Omega_{\mathrm{opt}}^B}, \\ T &= T^b, \\ \Omega^{B+D} + \Omega^\Lambda + \alpha\Omega_T^{\Lambda\mathrm{R}} &= 1, \end{aligned}$$

when $a(T) = 1$. We further replace the third equation by combining the first and third equations. This yields an equivalent system,

$$\begin{aligned} \frac{\Omega^D + \alpha\Omega_T^{\Lambda\mathrm{R}}}{\Omega^B} &= \frac{\Omega_{T,\mathrm{opt}}^{\Lambda\mathrm{R}}}{\Omega_{\mathrm{opt}}^B}, \\ T &= T^b, \\ \Omega^\Lambda &= 1 - \left(\frac{\Omega_{\mathrm{opt}}^{\Lambda\mathrm{R}}}{\Omega_{\mathrm{opt}}^B} + 1\right)\Omega^B. \end{aligned}$$

With this system, we no longer need to solve the flatness equation iteratively, since the value of $\Omega^\Lambda$ is fixed. The solution to this system is , $\Omega^D \approx 0.2589$, $\alpha \approx 0.0832$, and $\alpha\Omega_T^{\Lambda\mathrm{R}} \approx 0.0223$.

# 4 Figures

Here, we describe the contents of each figure separately.

### 4.1 Figures 1 and 2

In general, to compute the solution trajectories, we use the parameter values shown in the text boxes in the figures. The computation of the $\Lambda$R-model is discussed in 3.2.

### 4.2 Figure 3

The function $\Omega^{\Lambda\mathrm{R}}(t)$ is defined by equation (2). The blue dotted graph is the $\Lambda$R-model in Figures 1 and 2 but integrated up to 50 Gyr. The red graph is computed by putting $\Omega^\Lambda \approx 0.6881$ in 2, and $a(t)$ the benchmark model trajectory shown in Figure 2. To find the maximum value of $\Omega^{\Lambda\mathrm{R}}(t)$ we use (5).

## 4.3 Figure 4

We have three cases: $\Omega^{B+D} \in \{0.2,\ 0.27,\ 0.3\}$. In all cases, we have $\Omega^B = 0.049$ and $\Omega^\Lambda \approx 0.6881$. The parameter $\alpha$ is solved from the flatness equation.

Note that in the pure F-model, i.e., $\alpha = 0$, we now have $\Omega^B = 0.049$, $\Omega^D = 0.2629$, $\Omega^\Lambda = 0.6881$.

## 4.4 Figure 5

In each graph, we have $\Omega^D = 0$, $\alpha = 1$ and $\Omega^\Lambda$ is solved from the flatness equation. We let $\Omega^B$ take a few certain values including $\Omega^B_{\text{opt}}$, see 3.4.

## 4.5 Figures 6 and 7

We have $\Omega^D = 0$ and $\alpha = 1$, and we let $\Omega^B$ vary to produce the curves. We solve the flatness equation at each point on the curve and record the $\Omega^{\Lambda\text{R}}_T$-values. In the linear approximation curve, we use $a(t) = H_T\, t$, $t \geq 0$; and integrate (2) analytically to the upper bound $t = T = 1/H_T$, which gives the flatness equation in the following form:

$$\Omega^B + \frac{1 - \left(1 + \sqrt{\Omega^\Lambda}\right) \exp\left(-\sqrt{\Omega^\Lambda}\right)}{\sqrt{\Omega^\Lambda}} + \Omega^\Lambda = 1.$$

Models used to generate these graphs are also used to form the numbers in Tables 1 and 2.

## 4.6 Figure 8

Note that one dotted curve (the last one in the box) is defined by,

$$\Omega^{B+D} = 0.2, \quad \alpha = 0.68, \quad \Omega^B = 0.049,$$

and another dotted curve by,

$$\Omega^{B+D} = 0.27, \quad \alpha = 0.3, \quad \Omega^B = 0.049.$$

In both, $\Omega^\Lambda$ is given by the flatness equation.

In the two remaining graphs, the parameters are as in 3.5, where we fix $\alpha\Omega^{\Lambda\text{R}}_T = 0.06$, or 0.16.

## 4.7 Figures 9 and 10

There are three curves with $\alpha\Omega^{\Lambda\text{R}}_T \in \{0.01, 0.02, 0.03\}$. These models are computed as in 3.5. The optimal ΛR-model is as in Figure 9.

## 4.8 Figures 11 and 12

Define $H_t = \dot{a}(t)/a(t)$, $t \geq 0$. Here, we have different values for $H_T$, all of which need to be converted to Gyrs, see 3.1. For the pure F-models, we have $\alpha = 0$, and all other required parameters are shown.

We choose the ΛR-model's parameter $\alpha$, s.t., its age of the universe matches with the F-model 1, while $\Omega^{B+D} = 0.307$, $\Omega^B = 0.049$, and $\Omega^\Lambda$ is solved from the flatness equation.

# References


[1] Harri Ehtamo. *A Coincidence Problem Related to the Λ-CDM Cosmological Model*. Private Manuscript, July 2019.

[2] Lauri Jokinen. *Code to Compute Solutions and to Generate Figures for Flat Friedmann Differential Equation*. 2023. URL: https://github.com/lauri-jokinen/Lambda-R-model.

[3] Wikipedia. *Parsec*. visited in 2023. URL: https://en.wikipedia.org/wiki/Parsec.