

Computational Physics: Temperature of Earth Project Report

Stockholm University, Spring Term 2024

Max Maschke

Apr 04 2024

Contents

1	Introduction	2
2	Energy Balance Models	2
3	Numerical Formulation	3
3.1	Density calculation	3
3.2	Iterative EBM solution	3
3.3	Direct EBM solution	3
4	Implementation and Numerics	4
4.1	Performance	4
4.2	Numerics	4
5	Physical Predictions	6
5.1	Density	6
5.2	Sea level temperature predictions	7
5.3	Atmospheric temperature	10
6	Conclusion	10
7	References	11

1 Introduction

In this report, a simple energy balance model of Earth's atmosphere is presented, implemented and analysed. It is part of the coursework for the computational physics class held at Stockholm University in 2024.

2 Energy Balance Models

Energy balance models (EBMs) arguably represent the simplest class of climate models. They model the temperature of the Earth system using simple energy flux considerations only [1].

The simplest realisation of such a system is what's sometimes referred to as the zero-dimensional energy balance model: the temperature is calculated using the Stefan-Boltzmann law

$$\frac{P}{A} = \sigma T^4, \quad \sigma = \frac{2\pi^5 k_B^4}{15h^3 c^2} \approx 5.69 \frac{\text{W}}{\text{m}^2 \text{K}^4}, \quad (1)$$

assuming a steady state in which the Earth emits precisely the incoming flux at all times. Such a model predicts a surface temperature of -21°C , which is much too low.

The most important property of the Earth system that is neglected in the above model is the planet's atmosphere. Absorption in the atmosphere traps a portion of the incoming flux, which is known as the greenhouse effect.

In this analysis an energy balance model for a zero-dimensional planet with a one-dimensional atmosphere that is extended in the z -direction is studied, thus neglecting any convective effects and the water cycle. The incoming light is restricted to a single wavelength in the visible part of the spectrum and any thermally emitted light is in turn restricted to a single IR wavelength. This implies that there is only one absorption cross-section σ_i , $i = \text{IR}, \text{V}$, for both parts of the spectrum.

To make the model numerically tractable, the atmosphere is divided into n discrete layers of height dh . In any of these layers j , the energy balance is given by

$$E_j = A_j, \quad (2)$$

that is emission must equal absorption. All radiation incoming from neighbouring layers that is not transmitted is absorbed, so

$$E_j = A_j = \left(\frac{E_{j-1} + E_{j+1}}{2} + T_{j-1}^{\text{out}} + T_{j+1}^{\text{in, IR}} \right) \left(1 - e^{-\sigma_{\text{IR}} \rho_j dh} \right) + T_{j+1}^{\text{in, V}} \left(1 - e^{-\sigma_{\text{V}} \rho_j dh} \right). \quad (3)$$

The equations for the different transmission measures are omitted here for brevity and can be found in the course material. The lowest layer is assumed to be a black body, which is a first order approximation for the Earth's surface:

$$E_0 = A_0 = \frac{1}{2} E_1 + T_{j+1}^{\text{in, IR}} + T_{j+1}^{\text{in, V}} \quad (4)$$

It can be immediately seen that absorption depends on the density of the atmosphere at a given height $h_j = j \cdot dh$. From the ideal gas law

$$\frac{M}{V} = \rho = \frac{Pm}{k_B T} \quad (5)$$

where m is the molecular mass of air, and the hydrostatic condition considering the height-variation of the gravitational acceleration g

$$\frac{dP}{dh} = -\rho \cdot g(h), \quad (6)$$

we find

$$\frac{dP}{dh} = -\frac{Pm}{k_B T} \cdot g(h). \quad (7)$$

Assuming $T = \text{const.}$, this differential equation can be solved iteratively given a starting value for P . The density is then directly calculated from P using equation (5).

3 Numerical Formulation

3.1 Density calculation

Since the density in principle only has to be calculated once for all the other calculations, a more expensive but more accurate solver than a simple Euler method

$$P_{i+1} = P_i + \frac{dP}{dh}(h_i) \cdot dh \quad (8)$$

can be used for increased accuracy. We chose to apply a three-step Adams–Bashforth method:

$$P_{i+1} = P_i + \left(\frac{23}{12} \frac{dP}{dh}(h_i) - \frac{16}{12} \frac{dP}{dh}(h_{i-1}) + \frac{5}{12} \frac{dP}{dh}(h_{i-2}) \right) \cdot dh \quad (9)$$

Note that the first two steps have to be taken using Euler steps or similar. This method has error of order $O(dh^3)$ instead of the Euler method's $O(dh)$ [2].

3.2 Iterative EBM solution

The set of equations describing the model, namely (3) etc., are coupled and must thus be solved consistently. One approach to this is to initialise $T_n^{\text{in}, V} = 1$ and then propagate the quantities $T^{\text{in}, V}$, $T^{\text{in}, \text{IR}}$ and E downwards. Subsequently, starting from E_0 , T^{out} can be propagated upwards. This procedure can be repeated as many times as required until a steady solution is found.

This scheme is advantageous since it is very quick, requiring only $2n \cdot n_{\text{sweep}}$ iterations where n_{sweep} is the number of full sweeps done. However, some boundary conditions that we might want to impose on the model are not explicitly enforced. For example, the outgoing flux $E_n/2 + T_n^{\text{out}}$ should be equal to the incoming visible flux, but the iteration does not guarantee this so it must be checked.

3.3 Direct EBM solution

Another approach is to realise that the equations describing the model constitute a linear system $Ax = b$ with $4n + 4$ unknowns if we include an $n + 1$ -th layer representing space where

the density has dropped to zero. If A is non-singular, the system has a unique solution. To ensure A has full rank, the following additional boundary conditions have to be imposed:

$$\begin{aligned} T_{n+1}^{\text{in}, V} &= 1 \\ T_{n+1}^{\text{in}, \text{IR}} &= 0 \\ T_0^{\text{out}} &= E_0 \\ T_{n+1}^{\text{out}} &= 1 \\ E_{n+1} &= 0 \end{aligned} \tag{10}$$

In English, these mean that, a), the incoming visible flux is normalised, b), no IR flux is incoming from the sun, c), the ground emits as a black body, d), the outgoing IR radiation must be equal in energy flux to the incoming visible flux and e), space does not absorb or emit.

From here, the methods discussed last week can be applied to solve the linear system directly. The advantage is that the solution will obey the boundary conditions exactly. However, calculating the LU -decomposition for a system of size $4n + 4 \times 4n + 4$ becomes costly quite quickly as n is increased. As will be seen later, this method is thus not very efficient. Another problem with the interpretability of the directly obtained solution was also encountered later, namely that E_0 , from which the temperature is obtained, came out to be negative, which is unphysical.

4 Implementation and Numerics

The calculation of the density and the solving of the EBM are implemented in Python. The code is enclosed with this report and also available at [3]. To speed up the iteration which would otherwise be performed in the interpreter, the code is just-in-time compiled using `Numba` [4]. The LSE is solved using `Numpy's linalg.solve` function [5], which internally calls on `LAPACK` [6] to compute the LU -decomposition.

Values of the important physical constants and measurements such as the sea level pressure are taken from [7], cf. the head of `physics.py` [3]. The cut-off for the atmosphere is chosen to be $h_{\text{max}} = 100$ km, which implies that if, e.g., 10000 layers are used, $dh = 10$ m. The Earth's albedo is fixed at 0.33 and a solar flux of 344 W/m^2 is used. Due to limited time, the influence of the mean surface reflectivity could not be implemented nor studied. The absorption in the visible bands was fixed at $\alpha_V = 5 \cdot 10^{-5} / \text{m}$

4.1 Performance

The measured runtimes of solving the EBM using both approaches is shown in figure 1. It can be seen that the iteration, which used 15 full sweeps, outperforms the LSE solver by orders of magnitude and thus allows the study of much finer layers. It can also be seen that the iteration is $O(n \cdot n_{\text{sweep}}) = O(n)$ while the LSE solver is $O((4n + 4)^3)$.

4.2 Numerics

More important than the performance of any numerical method are its stability and convergence. To investigate this for both methods, their behaviour as the main numerical parameter n is tuned must be observed.

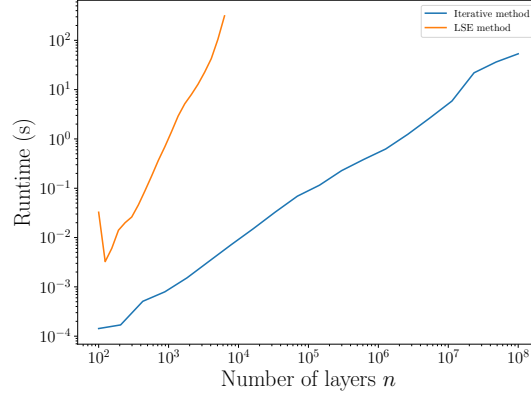
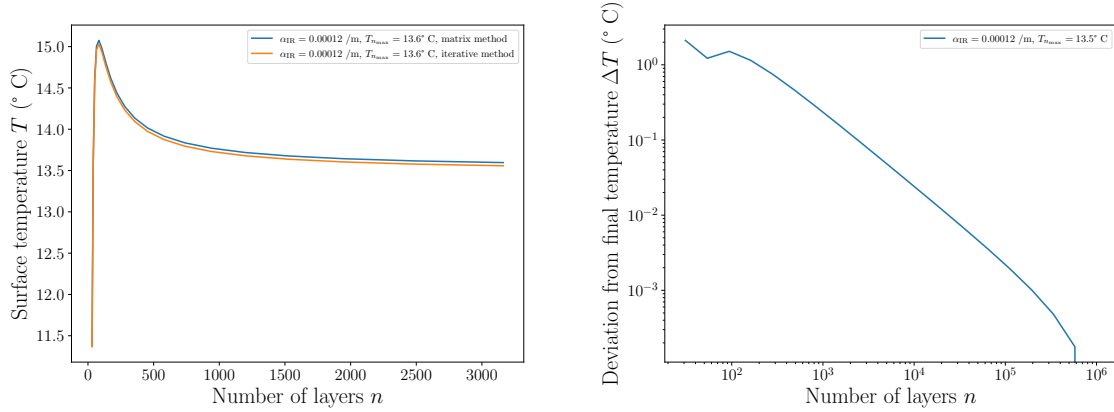


Figure 1: Measured runtimes of solving the EBM using the iterative and the LSE based approaches. The iteration, which used 15 full sweeps, outperforms the LSE solver by orders of magnitude and thus allows the study of much finer layers. Both curves have different slopes in the logarithmic plot, indicating that the iteration is $O(n \cdot n_{\text{sweep}}) = O(n)$ while the LSE solver is $O((4n + 4)^3)$.



- (a) Comparison of the predicted surface temperature for varying n for both methods, using $\alpha_{\text{IR}} = 1.2 \cdot 10^{-4}/\text{m}$. Both methods converge and agree up to the first decimal.
- (b) Convergence towards the final value for the iterative method using $\alpha_{\text{IR}} = 1.2 \cdot 10^{-4}/\text{m}$. To achieve three decimals of numerical accuracy, 10^5 layers are needed.

Figure 2: Numerics of increasing n for both approaches. Both methods are seen to be stable for increasing n in the parameter ranges studied.

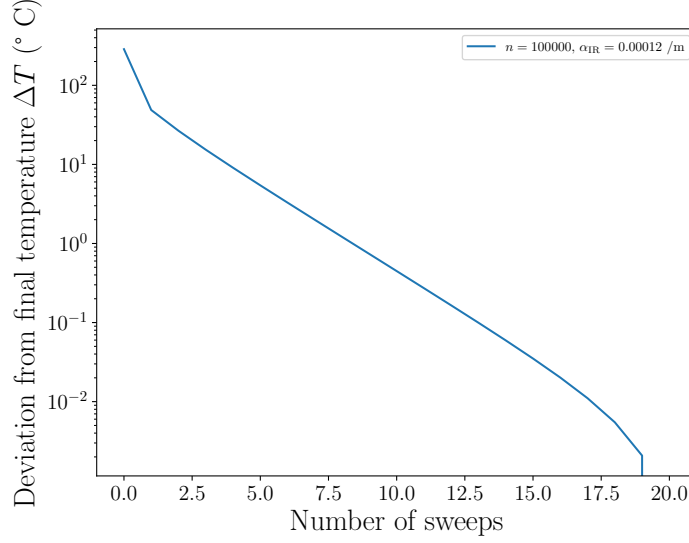


Figure 3: Numerics of increasing the number of sweeps for the iterative method, using $\alpha_{\text{IR}} = 1.2 \cdot 10^{-4}/\text{m}$ and 10^5 layers. To achieve two decimals of convergence, 15 sweeps are needed.

Figure 2 shows how both models respond to increasing n . Not only do both of them show numerical convergence, they also agree up to the first decimal for $n = 3000$ layers. Much finer layers are numerically intractable for the LSE method, but with the iterative method, we can go up to $n = 10^8$. The additional precision gained by this is sure to be offset by systematic sources of error in the choice of physical parameters and approximations of the model, so such a high number of discretisations is not needed in practice, which is why $n = 105$ was used consistently later on.

For the iterative method, there is another numerical parameter, namely the number of sweeps n_{sweeps} . Figure 3 shows the numerical convergence of increasing this quantity. It can be clearly seen that at least a handful of sweeps are required for the temperature to settle. We estimate that the two decimals of convergence given by using 15 sweeps is enough precision for the rest of this analysis and thus used this amount henceforth.

It is important to note that the iteration did not exactly achieve $T_{n+1}^{\text{in}, V} = T_{n+1}^{\text{out}} = 1$. The latter fluctuated in a typical range of 0.5 to 2. Unfortunately, there was not enough time left to investigate this further, but it is surprising given that the LSE and the iteration agree in their predictions for surface temperature and the LSE enforces this condition strictly.

5 Physical Predictions

In this section, the physical predictions of the model are to be discussed.

5.1 Density

Before the temperatures are looked at, the validity of the ODE solution giving the density is to be confirmed. It has to be reiterated here that the density curve depends on a guess of

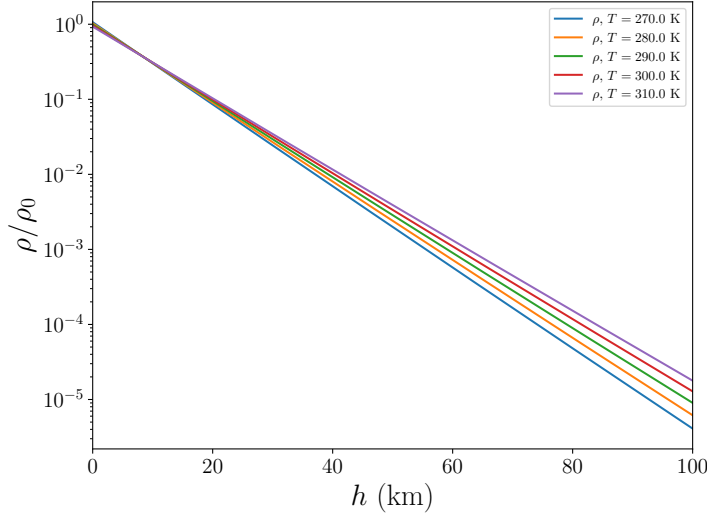


Figure 4: Density profile of the atmosphere obtained from numerically solving equation (7) for different temperature guesses. The choice of temperature more strongly affects the densities at higher heights than at sea level.

temperature that is assumed to be constant while the density is calculated. This has only a slight influence on the results, though, as long as this guess for T is kept in a reasonable range. We used $T = 15^\circ \text{C}$ henceforth to ensure that the density at $h = 0$ which is derived from the pressure at sea level is as close to the literature value for sea level density as possible, see figure 4.

The approximately linear drop on the lin-log plot implies that the drop in density is roughly exponential, which is what our first order expectation would be considering the barometric formula known from introductory physics classes.

5.2 Sea level temperature predictions

The predicted sea level temperature of the model strongly depends on the absorption in the IR bands, parametrised by

$$\alpha_{\text{IR}} = \sigma_{\text{IR}} \cdot \rho_0 \quad (11)$$

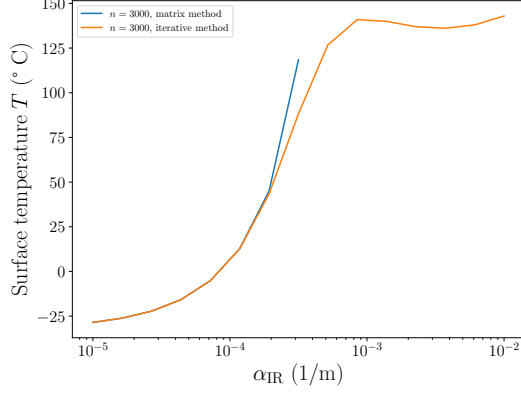
where ρ_0 is the sea level density. Given more time, it would have been prudent to determine a reasonable range for this parameter from known concentrations of greenhouse gases in the atmosphere. Instead, we are forced to limit our analysis to varying α_{IR} with the goal of predicting the average temperature of Earth, $T_{\text{E}} \approx 15^\circ \text{C}$.

The predicted sea level temperature for a large range of values of α_{IR} is shown in figure 5a for both the iterative and the LSE method, using $n = 3000$ layers. We see that at low α_{IR} , they are in excellent agreement, however, beyond a certain point, the LSE method breaks down. This is because the predicted surface emission becomes negative, which is unphysical. The reason for this could not be definitely determined but may have to do with the condition of the matrix as α_{IR} is increased. By design, the iterative method does not depend on the rank of the coefficient matrix A , so it does not show this problem. For the iterative method, the

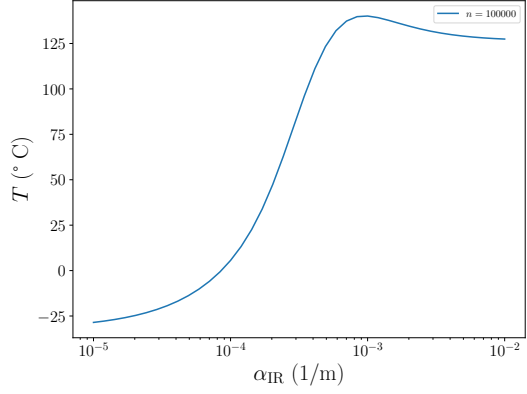
plot is repeated in figure 5b for a higher n of 10^5 , showing minimal differences except for very large α_{IR} .

Physically, we see that our model shows the greenhouse effect as desired. In the absence of IR-absorbing gases, i.e. as $\alpha_{\text{IR}} \rightarrow 0$, the predicted temperature is quite cold and close to that of the zero-dimensional EBM. Note that it is slightly lower than the 21°C quoted above since visible absorption is present in the model which reduces the flux that reaches the surface.

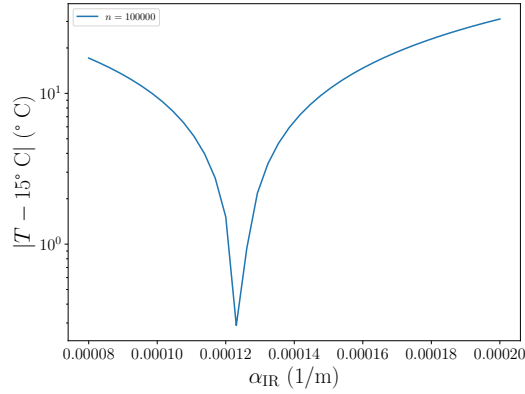
Conversely, as α_{IR} is increased, the temperature rises and, at very high and possibly unphysical α_{IR} , plateaus. Figure 5c shows that for an α_{IR} of ca. $1.2 \cdot 10^{-4}/\text{m}$, the predicted temperature is approximately 15°C which is the roughly expected real value.



(a) Predicted sea level temperature for a large range of values of α_{IR} , both methods, $n = 3000$. For large α_{IR} , the LSE method breaks down.



(b) Predicted sea level temperature for a large range of values of α_{IR} , iterative method, $n = 10^5$. The iterative method is stable for all α_{IR} .



(c) Predicted sea level temperature for a narrow range of values of α_{IR} , iterative method, $n = 10^5$. An α_{IR} of ca. $1.2 \cdot 10^{-4}/\text{m}$ reproduces the expected value.

Figure 5: Sea level temperature predictions of the EBM for varying α_{IR} .

5.3 Atmospheric temperature

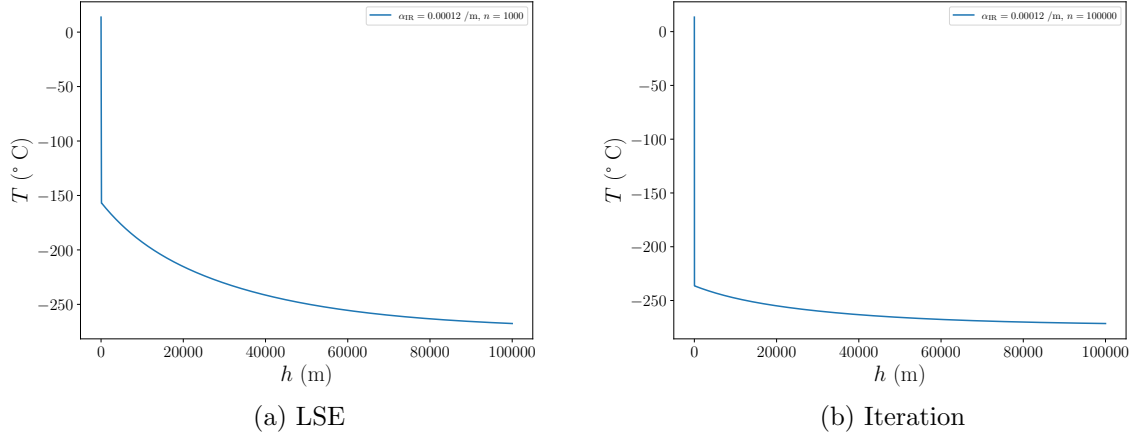


Figure 6: Temperature profile of the atmosphere predicted using the EBM.

As a last step, we want to see how the temperature of the atmosphere varies with height. This is shown in figure 6. It appears that while the ground layer is at high temperature, the atmosphere it is supposed to be in thermal equilibrium with has much too low temperature even in the first layer above the ground. This could either be because the way the temperatures were calculated using the Stefan-Boltzmann law is wrong for the atmosphere, or there is a previously undiscovered mistake somewhere else in the model. The latter would be discouraging because it would mean that the model does not make physically meaningful predictions but is instead just fitted such that the surface temperature matches expectation.

Due to a lack of time, we can sadly not investigate this further.

6 Conclusion

In this report, we presented our implementation of the EBM described and discuss its numerics and physical predictions. Unfortunately, in the last step, a large problem emerged that we could not solve in time, which makes any interpretation of the results discussed before dubious.

References

- [1] Gerald R. North and Mark J. Stevens. “Energy-balance climate models”. In: *Frontiers of Climate Modeling*. Ed. by J. T. Kiehl and V.Editors Ramanathan. Cambridge University Press, 2006, pp. 52–72.
- [2] Wikipedia. *Linear multistep method*. URL: https://en.wikipedia.org/wiki/Linear_multistep_method (visited on 04/03/2024).
- [3] Max Maschke. *compphys*. URL: <https://github.com/max-mas/compphys> (visited on 04/03/2024).
- [4] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. “Numba: a LLVM-based Python JIT compiler”. In: *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*. LLVM ’15. Austin, Texas: Association for Computing Machinery, 2015. DOI: 10.1145/2833157.2833162.
- [5] Charles R. Harris et al. “Array programming with NumPy”. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2.
- [6] E. Anderson et al. *LAPACK Users’ Guide*. Third. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999.
- [7] Inc. Wolfram Research. *Mathematica, Version 14.0*. Champaign, IL, 2024. URL: <https://www.wolfram.com/wolfram-alpha-notebook-edition>.