MATH 303, Mathematical Models

Computer Project #3: Energy Balance Models

DUE DATE: Friday, November 16, 5:00 pm

The goal of this project is for you to investigate the Earth's climate using some low-dimensional climate models called energy balance models. In the process of completing the lab, you will learn some important lessons about mathematical modeling (e.g., units, variables, parameters, constructing models from basic physical principles, refining models, tuning models, finding and analyzing equilibria, etc.). Although this project does not require sophisticated technology to complete—it could be done on a graphing calculator—another goal of the lab is for you to continue to experience using the software Matlab. This project is adopted from the 2012 teaching module "Energy Balance Models," by D. Flath, H. Kaper, F. Wattenberg, and E. Widiasih
(available at `http://dimacs.rutgers.edu/MPE/index.html`).

For this project, it is **required** that you work in a group of two or three people. Any help you receive from a source other than your lab partner(s) should be acknowledged in your report. For example, a textbook, website, another student, etc. should all be appropriately referenced at the end of your report. The project should be **typed** although you do not have to typeset your mathematical notation. For example, you can leave space for a graph, computations, tables, etc. and then write it in by hand later. You can also include graphs or computations in an appendix at the end of your report. Your presentation is important and I should be able to understand what you are saying. Spelling mistakes and sentence fragments, for example, should not occur. Only **one project per group** need be submitted.

Your report should provide coherent answers to each of the following exercises. Be sure to read carefully and answer all of the questions asked.

Note: Please turn in a printout of your Matlab file(s) (or send via email).

Mathematical Modeling

Part of the aim of this lab is to learn about the process of constructing and refining a mathematical model of a physical system. The system of interest for us is the Earth's climate system—a complex system with several components: the atmosphere, oceans, lakes and other bodies of water, snow and ice, land surface, all living things, and so on. The components interact and influence each other in ways that we don't always understand, so it is difficult to see how the system as a whole evolves, let alone why it evolves the way it does. For some complex systems it is possible to build a physical model and observe what happens if the environment changes. This is the case, for example, for a school of fish, whose behavior we can study in an aquarium. It is also true for certain aspects of human behavior, which we can study in a social network. But in climate science this is simply not possible; we have only one Earth, and we cannot perform a controlled real-life experiment. The best we can do if we want to gain insight into what might have happened to the Earth's climate system in the past, or what might happen to it in the future, is to build mathematical models and "play" with them. Mathematical models are the climate scientists' only experimental tools.

The modeling process—building and testing a series of imperfect models—is the most essential brick in the foundation of climate science and an indispensable tool to evaluate the arguments for or against climate change. Models are never perfect—at best, they provide some understanding and some ability to test "what-if" scenarios. Especially in an area as complex as the Earth's climate, we cannot and should not expect perfection. Recognizing and identifying imperfection and uncertainty are key parts of all modeling and, especially, climate modeling.

Mathematical models of the Earth's climate system come in many flavors. They can be simple enough that we can use them for back-of-the-envelope calculations, or they can be so complicated that we need a supercomputer to run them. Regardless of the type of model being used, we should always keep in mind that they are *simplified representations* of the real world, they are not the "real world," and they are made for a purpose, namely to better understand what is driving our climate system.

This project considers zero-dimensional **energy balance models**. They are the simplest possible description of the Earth's climate system. Nevertheless, as we will discover, they can provide insight into possible climate states of the planet. In these models, the state of the climate system is characterized by a single variable: the temperature of the Earth's surface, averaged over the entire globe. An energy balance equation is derived under the assumption that the temperature of the Earth increases if the Earth receives more energy from the Sun than it re-emits into space, and that it decreases if the opposite is the case. We will construct energy balance models by finding mathematical expressions for the incoming and outgoing energy. The models are tested against "real-world" data and improved in successive steps of the modeling process to better match the available data.

In this lab we will focus on the physics, but it is important to note that modeling the Earth's climate system is fundamentally an interdisciplinary activity. Understanding the Earth's climate requires knowledge, skills, and perspectives from multiple disciplines. For example, atmospheric chemistry explains why much of the incoming energy from the Sun (largely in the ultraviolet and visible regions of the spectrum) passes through the atmosphere and reaches the Earth's surface, but much of the black-body radiation emitted by the Earth (largely in the infrared regions of the spectrum) is trapped by greenhouse gases like water vapor and carbon dioxide. Similarly, the life sciences help us understand the part played by the biosphere in the Earth's climate system—the effects of the biosphere on the Earth's albedo and the interactions between atmospheric chemistry and plant and animal life.

Climate Model #1 : Calculating Thermal Equilibrium

We consider the Earth with its atmosphere, oceans, and all other components of the climate system as a homogeneous solid sphere, ignoring differences in the atmosphere's composition (e.g., clouds), differences among land and oceans, differences in topography (altitude), and many other things.

The climate system is powered by the Sun, which emits radiation in the ultraviolet (UV) regime (wavelength less than 0.4 μm). This energy reaches the Earth's surface, where it is converted by physical, chemical, and biological processes to radiation in the infrared (IR) regime (wavelength greater than 5 μm). This IR radiation is then re-emitted into space. If the Earth's climate is in **equilibrium** (a steady state), the average temperature of the Earth's surface does not change, so the amount of energy received must equal the amount of energy re-emitted.

Modeling

Units

- Length: meter (m), a μm is a micrometer = 0.001 mm
- Energy: watt (W). 1 watt = 1 joule per second = $1\frac{\text{kg}\cdot\text{m}^2}{\text{sec}^3}$. A joule is the amount of work done in applying a force of 1 Newton to move an object 1 meter.

- Temperature: kelvin (K). An object whose temperature is 0 K has no thermal energy; 0 K is *absolute zero*. Water freezes at 273.15 K = 0°C and boils at 373.15 K = 100°C. The Kelvin scale is basically the Celsius scale shifted up by 273.15. The magnitude of a degree in the Celsius scale is the same as the magnitude of a kelvin in the Kelvin scale, but the zero point is different. For the Celsius scale, the zero point is the temperature at which water freezes; for the Kelvin scale, the zero point is absolute zero.

Variables

- $T =$ the temperature of the Earth's surface averaged over the entire planet.

Physical Parameters

- $R =$ the radius of the Earth.
- $S =$ the **solar constant** (also referred to as the energy flux density). It is the amount of energy (W) flowing through a flat surface of area 1 m^2. From satellite observations we know that $S \approx 1368\ \text{W/m}^2$, although this value varies slightly based on sunspot activity.
- $\sigma = 5.67 \cdot 10^{-8}\ \text{W}/(\text{m}^2 \cdot \text{K}^4)$ (the Stefan-Boltzmann constant).

Building the Model

Viewed from the Sun, the Earth is really a disk with area πR^2. Since the energy flux density is S, the amount of incoming energy received by the Earth is

$$E_{\text{in}} \;=\; \pi R^2 S.$$

Note that the units of E_{in} are in watts (as expected) since the m^2 "cancel" in the above formula.

To determine the amount of energy radiated out by the Earth, we will regard the planet as a "black body" and use the Stefan-Boltzmann law: the energy flux equals σT^4, where σ is the Stefan-Boltzmann constant and T is the average temperature of the planet. Recall that the surface area of a sphere is $4\pi r^2$. Thus, the amount of energy radiated out by the Earth is

$$E_{\text{out}} \;=\; 4\pi R^2 \sigma T^4.$$

Once again, you should check that the units of E_{out} are also in watts (W).

Thermal Equilibrium

If the incoming energy is greater than the outgoing energy, the Earth's temperature increases. If the incoming energy is lower than the outgoing energy, the Earth's temperature decreases. If the incoming energy balances the outgoing energy, the Earth's temperature remains constant and the planet is said to be in **thermal equilibrium**. In order to have thermal equilibrium, the temperature T must satisfy $E_{\text{in}} = E_{\text{out}}$ or

$$\pi R^2 S \;=\; 4\pi R^2 \sigma T^4.$$

It is customary to introduce the constant $Q = S/4$ and work with Q instead of S. Some texts and articles treat Q as the solar constant. Thus, the previous equation becomes

$$Q \;=\; \sigma T^4. \tag{1}$$

Solving equation (1) for T, we find that the equilibrium temperature of the Earth is

$$T = (Q/\sigma)^{1/4}. \tag{2}$$

Exercise 1: Using the stated values of σ and S, use Matlab to compute the value of T in equation (2). Give your answer in both Kelvin and degrees Celsius. Look up the actual current mean temperature of the planet. How does your answer compare to the value of T from your model?

Climate Model #2: Including Albedo

If you did the previous exercise correctly, your equilibrium temperature should have been around 10°C below the actual current temperature of the planet. Our model predicts too cold a planet. What went wrong?

One factor we neglected to include involves reflection—some of the incoming energy from the Sun is reflected back out into space. Snow, ice, and clouds, for example, reflect a great deal of the incoming light from the Sun. We use the term **albedo** to measure the Earth's reflectivity.

Refining the Model

Let us introduce a new parameter α that measures the amount of energy from the Sun that is reflected back into space before it reaches the surface of the Earth. A typical value for α averaged over the whole planet is $\alpha = 0.3$, meaning that about 70% of the incoming energy is absorbed by the Earth's surface. The value of α can depend greatly on latitude and/or temperature, as we would expect a higher albedo near the poles since ice reflects more sunlight than land does. It is important to note that α is a **dimensionless parameter**, that is, it has no units. This is because it represents a *percentage* or fraction of a given quantity (energy), rather than a specific physical object.

Our model now has a new value for incoming energy:

$$E_{\text{in}} = \pi R^2 S(1-\alpha).$$

The $1-\alpha$ term is used (as opposed to just α) because α represents the proportion reflected, so $1-\alpha$ is the fraction absorbed. The outgoing energy E_{out} is unchanged. Simplifying $E_{\text{in}} = E_{\text{out}}$, our new equilibrium temperature is obtained by solving the equation

$$Q(1-\alpha) = \sigma T^4 \tag{3}$$

for T.

Exercise 2: Assuming $\alpha = 0.3$, use Matlab to compute the value of T from equation (3). Give your answer in both kelvin and degrees Celsius. How does our new equilibrium temperature compare to the value obtained for the previous model? Is our new model an improvement? Why or why not?

Climate Model #3: Tuning

If you did the previous exercise correctly, you should have found that the new model does an even worse job at predicting the mean temperature of the Earth than the first model did. We included more physics (the albedo), yet the results were worse. This is ok! Modeling is an iterative process and sometimes what seems like an improvement to the model turns out to make things worse. To fix the issue, we do not discard albedo, we include something new—**the greenhouse effect**.

Greenhouse gases like carbon dioxide, methane, and water, as well as dust and aerosols, have a significant effect on the properties of the atmosphere. The effect on the outgoing radiation is difficult to model, but the simplest approach is to reduce the Stefan-Boltzmann law by some factor.

Let us introduce the dimensionless parameter ϵ representing the amount by which the outgoing energy is reduced due to the greenhouse effect. We will assume $0 < \epsilon < 1$. A priori, the value of ϵ is unknown. The new value for the outgoing energy radiated out by the Earth is now

$$E_{\text{out}} \;=\; 4\pi R^2 \epsilon\sigma T^4.$$

To calculate our new thermal equilibrium, we once again solve $E_{\text{in}} = E_{\text{out}}$, which leads to

$$Q(1-\alpha) \;=\; \epsilon\sigma T^4. \tag{4}$$

Now we are going to do something that might seem like cheating. We can't solve equation (4) to find the equilibrium value T because we don't have a specific value for ϵ. But we can go the other direction. Let's use the current average temperature of the Earth $T^* \approx 287.7$ K to find the value of ϵ that makes equation (4) satisifed. In modeling jargon this is called **tuning** the model. The point here is that we are adjusting our model so that the equilibrium temperature of our model matches that of the planet.

Exercise 3:

(a) Assuming $T = 287.7$, use Matlab to compute the value of ϵ from equation (4). You should use the same parameter values for α, S, etc. as before. Give your answer to four decimal places. Denote this particular value as ϵ^*.

(b) Suppose that you decrease the value of ϵ below ϵ^*. What will happen to the equilibrium temperature computed from equation (4)? Does this make sense? Explain in terms of physics and the greenhouse effect.

Climate Model #4: A Differential Equation

Up until now, our climate models are basically static because there is no mechanism for the temperature T to change over time. To address this, we now make the assumption that the temperature changes at a rate proportional to the difference between the incoming and outgoing energy. This leads to the following ODE:

$$C\frac{dT}{dt} \;=\; (1-\alpha)Q - \epsilon\sigma T^4, \tag{5}$$

which builds off our previous model. The right-hand side of our ODE is $E_{\text{in}} - E_{\text{out}}$ divided by $4\pi R^2$, the surface area of the Earth. On the left-hand side, we have a positive constant C that represents the **heat capacity** of the planet (the amount of energy needed to raise the temperature of the planet by 1 K), and a derivative that measures the rate of change of the temperature (with time t typically measured in years).

As a basic check on our model, note that if the incoming energy is greater than the outgoing energy, then the right-hand side of equation (5) is positive so that $dT/dt > 0$. Thus, our model predicts a gain in energy to cause the temperature of the planet to increase, as expected. Similarly, if the outgoing energy is greater than the incoming energy, then the right-hand side of equation (5) is negative and hence $dT/dt < 0$. In this case our model predicts a loss in planetary energy to cause the temperature to decrease. Finally, if $E_{\text{in}} = E_{\text{out}}$, the right-hand side of equation (5) is zero and $dT/dt = 0$, which means the temperature is constant (thermal equilibrium).

Define the right-hand side of equation (5) to be the function $f(T)$, that is,

$$f(T) \;=\; (1-\alpha)Q - \epsilon\sigma T^4.$$

Although it looks complicated, f is just a function of one variable T, since the other variables are all parameters (constants).

Exercise 4: Let $\epsilon = \epsilon^*$, the value you found in Exercise 3, part (a). Use Matlab to plot the graph of $f(T)$ where $200 < T < 400$ (remember that T is measured in kelvin).

(a) What does the vertical axis in your plot represent in the physical world? What are its units?

(b) What is the root (or zero) of f between 200 K and 400 K? Where have you seen this value before? What does it represent?

(c) Suppose the current temperature is 300 K. Do you expect the temperature to increase, decrease, or stay the same? According to the model, where is the temperature heading over time? Explain.

(d) Suppose the current temperature is 250 K. Do you expect the temperature to increase, decrease, or stay the same? According to the model, where is the temperature heading over time? Explain.

Exercise 5: Now let $\epsilon = 0.5$ and make a new plot of $f(T)$. Answer the same questions as in Exercise 4 (skip part **(a)**) and compare your answers. Be sure to include both figures in your final report.

Solving equations in Matlab

Solving equations in Matlab is pretty straight-forward using the `vpasolve` command. For instance, to find a solution to $\cos x = x$, type

```
syms x
vpasolve(cos(x) == x, x, 1.0)
```

at the command prompt. The syntax for the `vpasolve` command is to list the equation first (don't forget the double equal sign!), then the variable, then an initial guess. You can use this to solve part (b) in the previous two exercises.

Graphing in Matlab

To plot a function, you will first need to create a vector of input values. For example, to create a vector of T-values between 0 and 10, type

```
T = 0:0.1:10;
```

This will create a vector called T containing values from 0 to 10 in increments of length 0.1. So the vector starts as $0, 0.1, 0.2, 0.3, \ldots$ and ends with $9.7, 9.8, 9.9, 10$. Then we need to define a vector that contains the function values $f(T)$. This is the tricky part. To handle a function such as $f(T) = 5T^4$, you need to type

```
y = 5*T.^4;
```

The all-important period before the carrot indicates that we wish to raise each entry in the vector T to the fourth power. The expression `T^4` will return an error (try it) as Matlab will interpret this as raising the *entire* vector as a unit to the fourth power. This only works for square matrices, not vectors.

The other issue to address is how to add a constant to our function. Here we need to make use of the `ones` command. For example, to define the output values of $f(T) = 5T^4 + 7$, you type

```
y = 5*T.^4 + 7*ones(size(T));
```

The command `size(T)` gives the length of the vector, and then the `ones` command will produce a vector of all ones with the correct length. Multiplying this vector by 7 gives a vector of all 7's with the same length as T. Then we can add this to the `5*T.^4` command to obtain the correct vector of function values.

Now we are ready to plot. Typing `plot(T,y)` should plot the function $f(T) = 5T^4 + 7$ over the domain $T \in [0, 10]$. Your graph will appear in a separate window. You can then click on the square icon in the control panel (furthest to the right) to add things like grid lines and labels to your graph. Try this particular example before attempting to plot the functions in Exercises 4 and 5.

Climate Model #5: Temperature Dependent Albedo

Our final climate model will incorporate the idea that albedo changes depending on temperature. For colder temperatures, more ice forms, and the albedo α should be higher because ice reflects more of the suns rays. On the other hand, for warmer temperatures we expect a lower albedo since water absorbs more of the sun than ice does. One possible formula for our new albedo is

$$\alpha(T) \;=\; 0.7 - 0.4\frac{e^{(T-265)/5}}{1 + e^{(T-265)/5}}, \tag{6}$$

which satisfies $\alpha(T) \approx 0.7$ for $T < 250$ and $\alpha(T) \approx 0.3$ for $T > 280$. This is a monotonically decreasing function of T, as desired. Using this new albedo, our model becomes

$$C\frac{dT}{dt} \;=\; (1 - \alpha(T))Q - \epsilon\sigma T^4, \tag{7}$$

where α is no longer constant, but dependent on temperature T as indicated in equation (6).

Let $g(T)$ represent the function on the right-hand side of the ODE in equation (7). This function is much more complicated than $f(T)$ used in the previous model. Finding the equilibria is not possible analytically; it must be done numerically.

Exercise 6: Let $\epsilon = \epsilon^*$, the value you found in Exercise 3, part (a).

(a) Use Matlab to find the equilibrium points for the ODE model described in equation (7). There are three values (not one as with the previous models). Describe the kind of climate on the planet for each equilibrium. What would it be like to live on Earth in each case?

(b) One way to visualize the different equilibria is to plot E_{in} and E_{out} on the same graph and see where they intersect. To do this in Matlab you will need to define two different output vectors, say y1 and y2, that represent the functions corresponding to E_{in} and E_{out}, respectively. Then use the command `plot(T,y1,'b',T,y2,'r')` to produce both functions on the same plot. The first will be in blue, while the second is in red.

(c) Draw the phase line for the ODE in equation (7) and classify each equilibrium point as a sink, source, or node.

Exercise 7: Let ϵ be a changing parameter in the ODE model described in equation (7).

(a) As ϵ decreases below ϵ^*, a saddle-node bifurcation occurs at a special value ϵ_h. Find this value (to seven decimal places) and describe the behavior of the system before, at, and after the bifurcation. What is the climate of the Earth like after the bifurcation?

(b) As ϵ increases above ϵ^*, a saddle-node bifurcation occurs at a special value ϵ_{sb}. Find this value (to seven decimal places) and describe the behavior of the system before, at, and after the bifurcation. What is the climate of the Earth like after the bifurcation?

(c) Combining your answers to parts **(a)** and **(b)**, sketch the bifurcation diagram for the ODE model as the parameter ϵ varies (put ϵ on the horizontal axis and T on the vertical axis).

Note: To find the bifurcation values to a high degree of accuracy, you will need to solve a system of two equations in two unknowns using Matlab. Below is some sample code you can use as a guide:

```
syms x y
eqn1 = (x+1)*cos(x*y);
eqn2 = sin(x - 2) + tan(y);
Sols = vpasolve([eqn1 == 0, eqn2 == 0], [x y],[0.1,1.5]);
```

These commands are used to solve the system $(x+1)\cos(xy) = 0$ and $\sin(x-2) + \tan y = 0$. The solution is computed with the `vpasolve` command. The values `[0.1,1.5]` are initial guesses of x and y respectively for the solution (you need to make an educated guess). The solutions are stored in the matrix `Sols`. To see the results, type `Sols.x` and `Sols.y` and hit return.

```
Sols.x = -1
Sols.y = 0.14019425002806923303467047539108
```