BEE 4750 Homework 2: Systems Modeling and Simulation

2024-07-24

> **Due Date**
>
> Thursday, 09/19/24, 9:00pm

Overview

Instructions

- Problem 1 asks you to derive a model for water quality in a river system and use this model to check for regulatory compliance.
- Problem 2 asks you to discretize a simple climate model and use it to simulate global mean temperatures under a future emissions scenario.
- Problem 3 (5750 only) asks you to modify the lake eutrophication example from Lecture 04 to account for atmospheric deposition.

Load Environment

The following code loads the environment and makes sure all needed packages are installed. This should be at the start of most Julia scripts.

```julia
import Pkg
Pkg.activate(@__DIR__)
Pkg.instantiate()
```

```
using Plots
using LaTeXStrings
using CSV
using DataFrames
```

Problems (Total: 50/60 Points)

Problem 1 (25 points)

A river which flows at 10 km/d is receiving discharges of wastewater contaminated with CRUD from two sources which are 15 km apart, as shown in the Figure below. CRUD decays exponentially in the river at a rate of 0.36 d^{-1}.
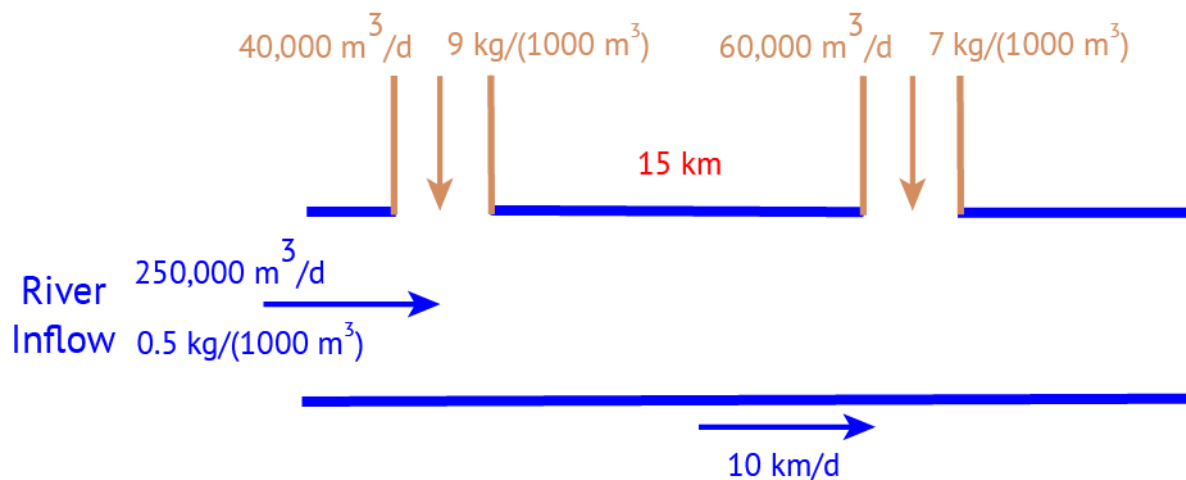

Figure 1: Schematic of the river system in Problem 1

In this problem:

- Assuming steady-state conditions, derive a model for the concentration of CRUD downriver by solving the appropriate differential equation(s) analytically.
- Determine if the system in compliance with a regulatory limit of 2.5 kg/(1000 m^3).

> 💡 Tip
>
> Your solution will need to be in terms of distance downriver.

Problem 2 (25 points)

The simplest climate model involves capturing changes to the Earth's energy budget (it is commonly called the *energy balance model*, or EBM). These changes are also called *radiative forcings* (RF), and can result from several causes, including greenhouse gas emissions, volcanic eruptions, and changes to the solar cycle. The EBM treats the Earth as a 0-dimensional sphere covered with water, which absorbs heat in response to radiative forcings. Chanwith global temperature changes resulting from imbalances in the average (over the entire surface area) heat flux.

The EBM equations are:

$$\underbrace{\frac{dH}{dt}}_{\text{change in heat}} = \underbrace{\hat{F}}_{\substack{\text{radiative} \\ \text{forcing}}} - \underbrace{\hat{\lambda}\hat{T}}_{\substack{\text{change in} \\ \text{temperature}}}$$

$$\underbrace{C}_{\substack{\text{ocean heat} \\ \text{capacity}}} \frac{dT}{dt} = F - \lambda T$$

$$c \underbrace{d}_{\substack{\text{ocean} \\ \text{mixing depth}}} \frac{dT}{dt} = F - \lambda T,$$

where $c = 4.184 \times 10^6 \,\text{J/K/m}^2$ is the specific heat of water per area, d is the depth of the ocean mixed layer (we'll assume $d = 86\text{m}$), and λ is the **climate feedback factor** and controls how much the Earth warms in response to increased radiative forcing (assume $\lambda = 2.1$). The total radiative forcing $F = F_{\text{non-aerosol}} + \alpha F_{\text{aerosol}}$, where α is an uncertain scaling factor reflecting aerosol-cloud feedbacks (we'll assume $\alpha = 0.8$).

The code below loads historical and projected radiative forcings (under the SSP5-8.5 future emissions scenario, which is the most extreme of the scenarios used to project climate change impacts) from `data/ERF_ssp585_1750-2500.csv` into a `DataFrame` object and calculates the non-aerosol and aerosol components of those forcings.

> 💡 **Tip**
>
> Look closely at and experiment with the code below: `DataFrames` are a common Julia datatype for tabular data, and you may work more with them later in the semester or beyond! They are broadly similar to `DataFrames` from `pandas` in Python.

```
# Dataset from https://zenodo.org/record/3973015
# The CSV is read into a DataFrame object, and we specify that it is comma
↪  delimited
```

```
forcings_all = CSV.read("data/ERF_ssp585_1750-2500.csv", DataFrame,
↪   delim=",")

# Separate out the individual components
# Get total aerosol forcings
forcing_aerosol_rad = forcings_all[!,"aerosol-radiation_interactions"]
forcing_aerosol_cloud = forcings_all[!,"aerosol-cloud_interactions"]
forcing_aerosol = forcing_aerosol_rad + forcing_aerosol_cloud
# Calculate non-aerosol forcings from the total.
forcing_total = forcings_all[!,"total"]
forcing_non_aerosol = forcing_total - forcing_aerosol
```
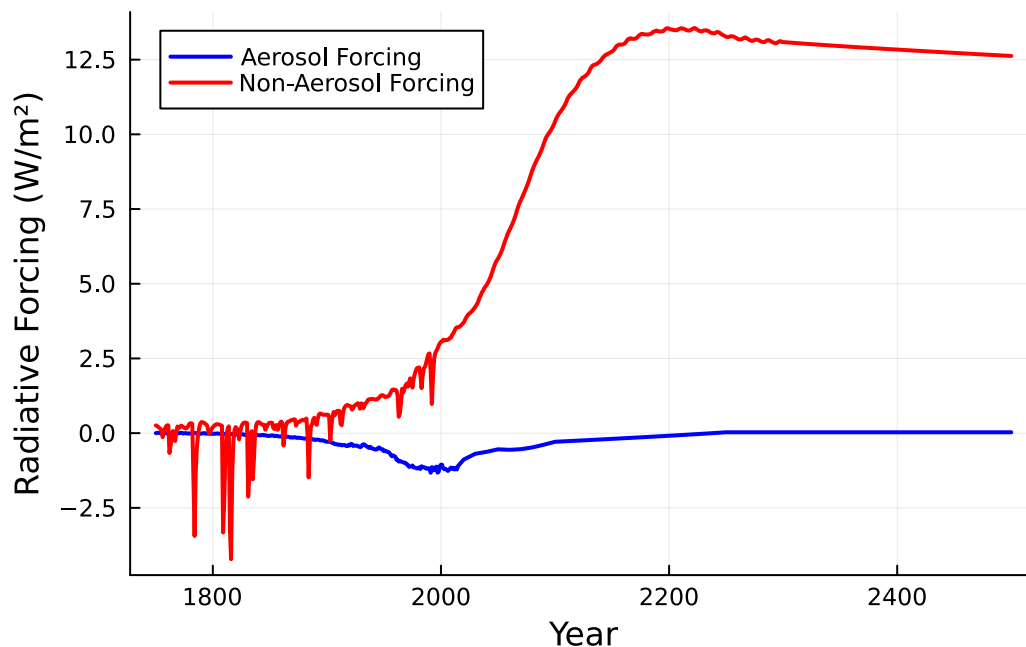
We can plot the aerosol and non-aerosol forcings below.

```
t = Int64.(forcings_all[!,"year"]) # Ensure that years are interpreted as
↪   integers
p_forcing = plot(; xlabel="Year", ylabel="Radiative Forcing (W/m²)")
plot!(p_forcing, t, forcing_aerosol, label="Aerosol Forcing", color=:blue,
↪   linewidth=2)
plot!(p_forcing, t, forcing_non_aerosol, label="Non-Aerosol Forcing",
↪   color=:red, linewidth=2)
```

In this problem:

- Discretize the EBM to produce a simulation model of global mean temperatures T over time as a result of total radiative forcings F.
- Simulate global mean temperature anomalies (in $^\circ C$ relative to 1750) from your model using the historical and SSP5-8.5 radiative forcing data. Use an annual time step for the simulation (in seconds: $\Delta t = 31,558,152$s). You can assume $T(0) = 0^\circ C$.
- Plot your simulated global mean temperature anomalies from 1750 through 2100.

Problem 3 (10 points)

This problem is only required for students in BEE 5750.

Consider the lake eutrophication example from Lecture 04. Suppose that phosphorous is also atmospherically deposited onto the lake surface at a rate of $1.6 \times 10^{-4} \text{kg}/(\text{yr} \cdot \text{m}^2)$, which is then instantly mixed into the lake. Derive a model for the lake phosphorous concentration and find the maximum allowable point source phosphorous loading if the goal is to keep lake concentrations below 0.02 mg/L.

References

List any external references consulted, including classmates.