Physics 305 – Computational Physics, Fall 2020
Term Project
Full project submission Due Date: Tuesday December 15, 5pm
Presentation Phase: November 30 - December 11

The program in your term project can be either submitted as a python program or ipython notebook, where the latter is preferred. The program, an explanation of what the program does, along with answers to **all** questions asked should be uploaded to d2l.

You are expected to write a term paper (in word or Latex) on your project that discusses the problem you are trying to solve, the basic equations that govern the problem, includes plots that show the solutions, and describes the solution and the numerical method involved. In addition, you must demonstrate that your solution is correct by showing that the code converges at the expected order. If your code does not converge at the expected order you should try to identify potential reasons for why this is the case. You are expected to work on your term project by yourself..

Your term project will receive full credit **only** if: (a) the program runs successfully without errors using python 3, (b) the programs have explanatory comments and variable names that identify with the problem equations you are trying to solve, (c) give the correct output, and (d) demonstrate the validity of the solution through convergence plots. No credit will be given to late term projects.

The term paper is as important as the code (50% of the term project credit will go to the code and the other 50% to the paper). Answers to the questions and analysis requested below should be elaborated in the report. Plots should be clearly labeled and be properly described in the report, and not just shown. You will need to explain what each and every plot demonstrates. A polished paper written in word or LaTex (preferred, e.g. please try overleaf) is expected to get full credit.

Note: Before you present results from numerical integrations that answer the questions in the project, it is critical to *first* perform the convergence tests for one case, and to estimate errors. This will tell you how small a step size is necessary for accurate solutions. Only after errors are estimated, does it make sense to run your code for producing results that help you learn more about the system you study.

I. SIMULATING CRUISE CONTROL

One of the most important examples of control theory is the automobile cruise control. You will be studying how this works using simple ordinary differential equations. The equation of motion of a car moving up an inclined road is given by

$$\frac{dv}{dt} + \frac{c}{m}v = \frac{F}{m} - g\theta \tag{1}$$

where we have made an approximation that the slope is small so that $\sin\theta \approx \theta$ and g is the gravitational acceleration which we will set $g = 10$ in SI. The term cv describes the momentum loss due to air resistance and rolling. F is the force generated by the engine, and we will assume that F/m is proportional to the signal u that goes to the throttle. If we introduce parameters for a typical car the equation becomes

$$\frac{dv}{dt} = u - 10\theta - 0.02v. \tag{2}$$

where here the control signal u is normalized, i.e., $0 \leq u \leq 1$, with $u = 0$ meaning no throttle, and $u = 1$ meaning full throttle. Notice that this equation on a 0 slope, implies that the car achieves terminal velocity $0.01v = 1$, i.e., $v = 1/0.02 = 50\text{m/s} = 180\text{km/h}$. If the slope is 10% or larger, the right-hand-side of Eq. (2) becomes $dv/dt = u - 10\theta - 0.02v = 1 - 10 \times 0.1 - 0.02v = -0.02v < 0$, i.e., the car decelerates. In other words, the car would not be able to climb a slope of 10% or larger based on this model.

Now, we want to introduce a controller that regulates the signal u if the slope changes within the capabilities of the car. A PI (Proportional-Integral) controller is the usual choice, for which the control signal u is

$$u = k(v_r - v) + k_i \int_0^t (v_r - v(\tau))d\tau. \tag{3}$$

The reason why this controller is called PI is because the first term is proportional to the error $e = v_r - v$ between the target velocity v_r and the current velocity v, while the second term is proportional to the (integrated) history of this error until the current time (t). We will next derive the full closed control system which shows how the error e evolves under the control system. Assuming v_r is constant, since this is the target velocity, we have

$$\frac{dv}{dt} = -\frac{de}{dt}, \frac{d^2v}{dt^2} = -\frac{d^2e}{dt^2}. \tag{4}$$

If we differentiate Eq. (3) (to avoid dealing with integrals and derivative at the same time) we obtain

$$\frac{du}{dt} = k\frac{de}{dt} + k_i e \tag{5}$$

If we differentiate Eq. (2) and use Eqs. (5) and (4) we arrive at

$$\frac{d^2e}{dt^2} = -(k + 0.02)\frac{de}{dt} - k_i e + 10\frac{d\theta}{dt}. \tag{6}$$

This equation states that if the car is traveling at the target speed and there is no change in the slope, then e is 0, and the control signal is 0. But, if the slope is changing then the error evolves. The equation governing the error evolution is very similar to a damped harmonic oscillator, which is given by

$$\frac{d^2x}{dt^2} + 2\zeta\omega_0\frac{dx}{dt} + \omega_0^2 x = 0, \tag{7}$$

where $\omega_0 = \sqrt{k/m}$ is the undamped harmonic oscillation angular frequency, and $\zeta = c/2\sqrt{mk}$ (with c the contant multiplying the term proportional the velocity of the oscillator) is called the damping ratio. Recall that when $\zeta > 1$ we have overdamping, $\zeta = 1$ we have critical damping (the system returns to steady state as quickly as possible), and $\zeta < 1$ we have underdamping. Thus, the undamped harmonic frequency in Eq. (2) is $k_i = \omega_0^2$, and the damping ratio is given by $k = 2\zeta\omega_0 - 0.02$. We would like to achieve critical damping so we would like $\zeta = 1$, i.e., the control signal parameters will be

$$k = 2\omega_0 - 0.02, \quad k_i = \omega_0^2. \tag{8}$$

Thus, the control system is regulated by one parameter ω_0 which you will be varying in your analysis to determine its influence on the system.

To summarize, our final equations describing the cruise control system, i.e., determine the control signal, are

$$\begin{aligned}\frac{du}{dt} =& k\frac{de}{dt} + k_i e \\ \frac{d^2e}{dt^2} =& -(k + 0.02)\frac{de}{dt} - k_i e + 10\frac{d\theta}{dt}.\end{aligned} \tag{9}$$

To solve this system of ODEs, we need initial conditions, and a rate of change in the slope. We will assume that the car is initially traveling at the target velocity on a flat road, and that the normalized control signal u is $u = 0.5$, i.e., the car has $dv/dt = 0$ and from Eq. (2) the car is traveling at $0.5 = 0.02v \Rightarrow v = 25$m/s. At $t = 0$ there will be a sudden change in the slope from 0 to 5%. This means that we will not have a changing slope, because $\lim_{t\to 0+} d\theta/dt = 0$. The sudden change will only affect the initial deceleration of the car, which from Eq. (2), it becomes $dv/dt = -10 \times 5\% = -0.5\text{m/s}^2$. Since, $\frac{dv}{dt} = -\frac{de}{dt}$, we obtain the initial velocity error derivative is $\frac{de}{dt} = 0.5$. Thus, our initial conditions for Eq. (9) are

$$u(0) = 0.5, \quad e(0) = 0, \frac{de}{dt}\Big|_{t=0} = 0.5. \tag{10}$$

Armed with the initial conditions, the next important step is to cast the second equation in Eq. (9) to first-order form. Do so, and show your work in the term paper.

1. Use RK4, to integrate numerically the system of ODEs (9) from $t = 0$ and evolve long enough until you reach near 0 error $|e|$. Note that e is an absolute eror, and 0 absolute error does not mean much. Think about how you need to normalize $|e|$ to have a meaningful measure of how close the error is to 0.

   Use your judgement as to how small a step size you need to solve this system accurately. If you cannot figure this out from pure thought, experiment with different step sizes and evolve up to a given time t_1. Then determine how small the step size has to be in order for the solution $u(t_1)$, $e(t_1)$ to not change appreciably.

2. Use different values of $\omega_0 = 0.05, 1.0, 1.5, 2$ and determine its impact on the response of the control system. How quickly does the signal u respond as ω_0 increases? Show in separate plots the evolution of u vs t and of e vs t. Does the control signal asymptote to the same terminal value independently of ω_0? You will need one plot for u vs t and one for e vs t, and each of these two plots should contain 4 curves corresponding to the different values of ω_0.

3. What is the final velocity of the car on the sloped road? What is the final value of the control signal u? What is the final value of the velocity error e? Try to justify these values.

4. **Self-convergence**: Use a number of step sizes, and the value of $\omega_0 = 1$, and make a plot to demonstrate that the code solution for u and e at the final time you will integrate to self-converges. Show your convergence plot. Does the solution converge at the order you expect? If not, try to explain why.

5. Using the order of convergence you determined, employ Richardson extrapolation to determine an error for the solution for $u(t)$ at a time of your choosing.

6. After you have solved the previous system consider a case where the slope is changing. We will express

$$\frac{d\theta}{dt} = \frac{d\theta}{dx}\frac{dx}{dt} = \frac{d\theta}{dx}v = \frac{d\theta}{dx}(v_r - e). \tag{11}$$

Then the complete set of equations becomes

$$\begin{aligned} \frac{du}{dt} &= k\frac{de}{dt} + k_i e \\ \frac{d^2e}{dt^2} &= -(k+0.02)\frac{de}{dt} - k_i e + 10\frac{d\theta}{dx}(v_r - e). \end{aligned} \tag{12}$$

with v_r the target velocity. To solve this system of ODEs, we need initial conditions, and a rate of change in the slope. We will assume that the car is initially traveling at the target velocity on a flat road, and that the normalized control signal u is $u = 0.5$, i.e., the car has $dv/dt = 0$ and from Eq. (2) the car is traveling at $0.5 = 0.02v \Rightarrow v = v_r = 25\text{m/s}$. At $t = 0$ there will be a sudden change in the slope from 0 to 1% and the change will remain constant to $\frac{d\theta}{dx} = 0.0001$. This means now we will have a changing slope, because $d\theta/dt \neq 0$. The sudden change will affect the initial deceleration of the car, which from Eq. (2), it becomes $dv/dt = -10 \times 1\% = -0.1\text{m/s}^2$. Since, $\frac{dv}{dt} = -\frac{de}{dt}$, we obtain the initial velocity error derivative is $\frac{de}{dt} = 0.1$. Thus, our initial conditions for Eq. (12) are

$$u(0) = 0.5, \quad e(0) = 0, \frac{de}{dt}\Big|_{t=0} = 0.1. \tag{13}$$

Integrate Eqs. (12), using different values of $\omega_0 = 0.05, 1.0, 1.5, 2$ and determine its impact on the response of the control system. Show plots of the signal u and of the error e vs t. Does the control signal asymptote to the same terminal value independently of ω_0? Integrate tha above equations only up to the point where the control signal becomes $u = 1$, beyond that the car cannot climb the uphill road.

7. **Self-convergence**: Use a number of step sizes, and the value of $\omega_0 = 2$ in the case of changing slope, and make a plot to demonstrate that the code solution for u and e at the final time you will integrate to self-converges. Show your convergence plot. Does the solution converge at the order you expect? If not, try to explain why.