Final Projects

*Introduction to Numerical Analysis II*


Professor: Paul J. Atzberger

Due: Friday, March 20th

Turn in to TA’s Mailbox: (Pat Plunkett) [6th Floor South Hall by 5:00pm].

**Instructions:** In the final project you will apply the numerical methods developed in the class to a problem from physics or from finance (depending on your major and interests). For your assigned project you are to write a report which answers each of the stated questions. For the numerical solutions, you are to give the numerical results in the form of an appropriate table or clearly presented list which is clearly explained in the exposition of your report. All codes should be written in Matlab/Octave using the algorithms discussed in class. Your codes should all be submitted in an appendix to the report distinctly separated from your discussion of results. If you have questions please feel free to come to office hours or schedule an appointment for assistance.
A classical problem in mechanics is the study of the dynamical behaviors of pendulums, either single isolated pendulums, pendulums subject to forcing, or systems of multiple coupled pendulums. The simplest case is that of a single pendulum suspended on a rod from some fixed point, see Fig. (1). The equations of motion can be derived from the Lagrange Principle of Least Action. This principle states that the solutions of Newton’s equation can be equivalently expressed as those trajectories passing through a given point in phase space \( q_0, p_0 \) which minimizes the action

\[
A[q(t), p(t)] = \int_{t_0}^{T} L(q(t), p(t)) dt.
\]

The Lagrangian is given by \( L = T - V \), where \( T \) is the kinetic energy and \( V \) is the potential energy. It can be shown, under a few technical assumptions, that such a trajectory \((q(t), p(t))\) satisfies the differential equation:

\[
\frac{\partial L}{\partial q} - \frac{\partial}{\partial t} \left( \frac{\partial L}{\partial p} \right) = 0. \tag{1}
\]

For a single pendulum considered in a 2-dimensional space, see Fig. (1), the configurations can be described by a single angle \( \theta \) from the vertical, with \( x = \ell \sin(\theta) \), \( y = -\ell \cos(\theta) \). The kinetic energy is then given by \( T = \frac{1}{2} m (\dot{x}^2 + \dot{y}^2) = \frac{1}{2} m (\dot{\theta})^2 \) and the potential energy by \( V = g m y = -g m \ell \cos(\theta) \). By using Eq. (1) with \( q = \theta \) and \( p = \dot{\theta} \) we obtain the equations of motion:

\[
ml^2 \ddot{\theta} = -g m \ell \sin(\theta). \tag{2}
\]

The dynamics of pendulum systems in general give rise to coupled sets of non-linear ODE’s and studying all but the most basic cases using analytic methods is often quite challenging, but also quite interesting! We shall instead develop numerical approaches. In particular, we shall compare several different numerical methods for the basic case of a single pendulum and when there is coupling between two pendulums.

(a) Implement the following numerical methods to solve the second-order ODE given in Eq. (2): (i) Forward Euler Method, (ii) Modified Euler, (iii) Classical Fourth Order Runge-Kutta (version discussed in class). Consider the pendulum with \( \ell = 3m \), \( m = 10kg \), \( g = 9.8m/s^2 \). Now for each of the methods compute a trajectory of the pendulum from time 0s to time \( T = 10s \) for time step initially \( \Delta t = 0.1s \) and \( \theta(0) = 1 \), \( \dot{\theta}(0) = 0 \). Write a code which plots the angle \( \theta \) and angular velocity \( \dot{\theta} \) vs time, for example using \textit{subplot.m}. Also to help visualize the system write a code which can draw the configuration of the pendulum for each time step.

(b) For each of the methods recompute the trajectory for time steps obtained by successively halving the time step \( \Delta t = 0.4 \) at least 7 times. For the trajectories label the \( k^{th} \) time step used as \( h_k \) and denote the solution at time \( T \) by \( \theta^{[k]}(T) \). For the highest resolution trajectory computed label the corresponding time step \( h^* \) and denote the solution at time \( T \) by \( \theta^*(T) \). Compute for each of the methods the error estimate \( \tilde{e}(h_k) = |\theta^{[k]}(T) - \theta^*(T)| \). Make a table which gives these error estimates for each of the methods. Make a log-log plot of the error
estimates vs \( h \) for each of the methods and estimate the slope of the best fitting line for the 5 highest resolution time steps. Compare your estimate of the slope with the theoretical order of the method. Explain how these are related. Which method has the steepest slope? What does this indicate? Which method appears to perform the best for \( \Delta t = 0.1 \)?

(c) An important consideration when using numerical methods is to determine when (and even if at all) they converge to the desired solution of the differential equations. Developing a general theory for numerical methods when applied to nonlinear systems is often difficult. One basic empirical approach which can be used to help assess convergence is to use the fact that associated with solutions of Newton’s equation are a number of quantities which are conserved along each solution trajectory in phase space. For example the total energy, linear momentum, and angular momentum of a system are conserved in the absence of dissipation (friction) and external forcing. This feature of the system can be used to help assess if the numerical method is doing a good job in approximating the analytic solution. For instance, poor conservation would strongly indicate that the numerical solution is inaccurate in approximating the solution of the differential equations. This can occur even though the results may look qualitatively correct.

For each of the trajectories in part (a) compute the total energy of the system \( E = T + V \). Discuss how well each of the methods conserves energy. In particular, discuss how well the energy is conserved by each of the methods for large time steps, intermediate time steps, and small time steps. One measure of how well energy is conserved is to compute the mean energy value over the entire run \( \mu = \frac{1}{N} \sum_{i=1}^{N} E_i \) and compute the variance estimate \( \sigma^2 = \frac{1}{N} \sum_{i=1}^{N} (E_i - \mu)^2 \), smaller \( \sigma^2 \) indicates better conservation. Give a plot of the energy vs time for methods (i), (ii), (iii). Comparing (i), (ii), (iii) which performs better? Which methods overall appear to perform the best with respect to energy conservation?

(d) We shall now consider a system of two pendulums which are linked together, see Fig. 2. This is referred to as the double pendulum problem and as we shall soon see can have quite rich dynamics. The first pendulum has configuration \( x_1 = \ell_1 \sin(\theta_1), y_1 = -\ell_1 \cos(\theta_1) \) and the second has \( x_2 = \ell_1 \sin(\theta_1) + \ell_2 \sin(\theta_2), y_2 = -\ell_1 \cos(\theta_1) - \ell_2 \cos(\theta_2) \). The kinetic energy
is then $T = \frac{1}{2}m_1(x_1^2 + \dot{y}_1^2) + \frac{1}{2}m_2(x_2^2 + \dot{y}_2^2)$ and the potential energy is $V = gm_1y_1 + gm_2y_2$. From the Lagrangian action principle with $q_1 = \theta_1, q_2 = \theta_2, p_1 = \dot{\theta}_1, p_2 = \dot{\theta}_2$ we obtain the equations of motion:

\begin{align*}
(m_1 + m_2)\ell_1\ddot{\theta}_1 + m_2\ell_2\ddot{\theta}_2 \cos(\theta_1 - \theta_2) + m_2\ell_2\dot{\theta}_2^2 \sin(\theta_1 - \theta_2) + g(m_1 + m_2) \sin(\theta_1) &= 0 \\
m_2\ell_2\ddot{\theta}_2 + m_2\ell_1\dot{\theta}_1 \cos(\theta_1 - \theta_2) - m_2\ell_1\dot{\theta}_1^2 \sin(\theta_1 - \theta_2) + m_2g \sin(\theta_2) &= 0.
\end{align*}

With a little algebra this can be expressed as:

\[
\begin{bmatrix}
\ddot{\theta}_1 \\
\ddot{\theta}_2
\end{bmatrix} = A \mathbf{g}
\]

where

\[
A(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) = \frac{1}{m_2\ell_1\ell_2(m_1 + m_2 \sin^2(\theta_1 - \theta_2))} \begin{bmatrix}
m_2\ell_2 & -m_2\ell_2 \cos(\theta_1 - \theta_2) \\
-m_2\ell_1 \cos(\theta_1 - \theta_2) & (m_1 + m_2)\ell_1
\end{bmatrix}
\]

and

\[
\mathbf{g}(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2) = \begin{bmatrix}
-m_2\ell_2\dot{\theta}_2^2 \sin(\theta_1 - \theta_2) - g(m_1 + m_2) \sin(\theta_1) \\
m_2\ell_1\dot{\theta}_1^2 \sin(\theta_1 - \theta_2) - m_2g \sin(\theta_2)
\end{bmatrix}.
\]

For each of the numerical methods implement a code to integrate these equations. Given the somewhat intimidating look of the above expressions, we pause to remark that you need not multiply out the above by hand. In your implementation you should instead compute the values of $A, \mathbf{g}$ and numerically evaluate the product to obtain the right-hand side of the equation. Also, you should write your codes generically so that solving the above system will only require implementation of one additional appropriately defined function for the right-hand side which is called by all of the methods. Also, write a code which can draw the configuration of the pendulum for each time step. Write a code which draws the trace of the second pendulum in the $xy$-plane for all time steps.

Now using each of the methods compute trajectories of the double pendulum system with the parameters $T = 3.4s, \Delta t = 0.0034s, \theta_1 = -0.5, \theta_2 = 0.01, \dot{\theta}_1 = 0, \dot{\theta}_2 = 0, m_1 = 10\text{kg}, m_2 = 0.01\text{kg}, \ell_1 = 3\text{m}, \ell_2 = 1.5\text{m}, g = 9.8\text{m/s}^2$. Compare the traces obtained from each of the methods. Do they agree? Give a plot of the trace the second pendulum for method...
(i), (ii), and (iii). Now halve the time step a few times. Which of the methods appears to converge most rapidly?

Now using each of the methods compute trajectories of the double pendulum system with the parameters $T = 6.5\text{s}$, $\Delta t = 0.0065\text{s}$, $\theta_1 = -0.1$, $\theta_2 = \pi$, $\dot{\theta}_1 = -0.01$, $\dot{\theta}_2 = 0$, $m_1 = 10\text{kg}$, $m_2 = 0.01\text{kg}$, $\ell_1 = 3\text{m}$, $\ell_2 = 1.5\text{m}$, $g = 9.8\text{m/s}^2$. Compare the traces obtained from each of the methods. Do they agree? Give a plot of the trace of the second pendulum for method (i), (ii), and (iii). Now halve the time step a few times. Which of the methods appears to converge most rapidly?

Discuss at least one way you might try to validate that your computed trajectories are indeed a feature of the physical system and not simply a manifestation of numerical errors.
Project 2: Black-Scholes-Merton PDE

In this project we discuss the pricing of derivatives contracts using the Black-Scholes-Merton theory. A derivative contract depends on the price of some underlying asset, such as stock or bond. One such commonly traded derivative is a call option, which gives the holder the right to buy an asset at a specified price at some specified time in the future. At this agreed upon time the contract is said to have “matured.” What should be emphasized is that the holder has the right, but is not required to buy the asset at the time of maturity. This is why such contracts are referred to as “options.” The value of such a contract depends on the current price of the underlying asset (referred to as the ‘‘spot price’’) and the amount of time to maturity of the contract. From the assumption of no arbitrage in the market place the price $V$ of a derivative contract can be shown to satisfy the PDE:

$$\frac{\partial V}{\partial t} + rs \frac{\partial V}{\partial s} + \frac{1}{2} \sigma^2 s^2 \frac{\partial^2 V}{\partial s^2} - rV = 0, \text{ for } t < T$$

$$V(s, T) = \phi(s), \text{ for } t = T$$

where $t$ is time, $s$ is the market spot price of the underlying asset, $r$ is the “risk-free” interest rate, $\sigma^2$ is the volatility of the market price of the underlying asset. The maturity time is denoted by $T$ and the amount of payoff of the contract at maturity if the market price of the underlying asset is $s$ is given by $\phi(s)$.

a) An important feature of the Black-Sholes-Merton PDE useful in practice is that its solutions can be related to solutions of the Heat Equation under an appropriate change of variable. Show that by letting

$$s = Ke^x$$
$$t = T - \frac{2\tau}{\sigma^2}$$

and $v(x, \tau) = KV(s, t)$ the Black-Sholes-Merton PDE given by equation 3 becomes

$$\frac{\partial v}{\partial \tau} = \frac{\partial^2 v}{\partial x^2} + (c - 1) \frac{\partial v}{\partial x} - cv$$

where $c = \frac{2r}{\sigma^2}$. By letting $v(x, \tau) = e^{-\frac{1}{2}(c-1)x^2 - \frac{1}{4}(c+1)^2 \tau} u(x, \tau)$ show that $u(x, \tau)$ defined in this manner satisfies the Heat Equation

$$\frac{\partial u}{\partial \tau} = \frac{\partial^2 u}{\partial x^2}$$

$$u(x, 0) = \psi(x).$$

Also, give the explicit expression of $\psi(x)$ in terms of $\phi(s)$. This shows that a solution $u(x, \tau)$ of the Heat Equation given in equation 4 can be used to construct a solutions $V(s, t)$ of the Black-Scholes-Merton PDE given by equation 3 provided appropriate boundary conditions are chosen.
b) Many approaches can be used to discretize PDE’s. We shall discuss a finite difference approach. To make use of the theory developed in class we shall initially discretize the problem in space to approximate the PDE given by equation 4 by a coupled system of ODE’s. A more thorough treatment of discretizations of PDE’s will be given next quarter. We shall use the following spatial discretization for the Heat Equation

\[ \frac{\partial u_m}{\partial \tau} = \frac{u_{m+1} - 2u_m + u_{m-1}}{h^2}. \]  

(5)

where \( u_m(t) \) denotes the approximate value of \( u \) at time \( t \) and mesh site \( m \) with \( m = 0,1, \ldots, M \), and \( h \) is the mesh spacing in the uniform spatial discretization \( h = x_{m+1} - x_m \).

Now, analyze the spatial discretization (expression on the right-hand-side) for any fixed time. Show that the spatial discretization is consistent with \( \partial^2 u / \partial x^2 \). Determine the order of the truncation error of the spatial discretization.

c) We shall now discuss a discretization in time. A very stable approach to approximate solutions of the PDE given by equation 4 is to use the Crank-Nicolson implicit method given by

\[ \frac{u_{m+1}^{n+1} - u_m^n}{k} = \frac{1}{2} \frac{u_{m+1}^{n+1} - 2u_m^{n+1} + u_{m-1}^{n+1}}{h^2} + \frac{1}{2} \frac{u_{m+1}^n - 2u_m^n + u_{m-1}^n}{h^2} \]  

(6)

where \( u_m^n \) denotes the approximate value of \( u \) at time \( t_n \) at mesh site \( m \) with \( m = 0,1, \ldots, M, \) \( n = 0,1, \ldots, N, \) and the time step for the uniform temporal discretization is denoted by \( k = t_{n+1} - t_n \). Since values of \( u_m^{n+1} \) appear on both sides of equation 6, in order to advance a time step of this numerical method requires that a linear system be solved for the values of \( u \) at time \( n + 1 \). This can be expressed as

\[ (I - \frac{1}{2} \lambda A) u^{n+1} = (I + \frac{1}{2} \lambda A) u^n \]  

(7)

where \( u^n \) denotes the composite vector of values at time \( n \) over all spatial mesh sites \( m = 0,1, \ldots, M, \) \( A_{i,j} = \delta_{i,j+1} - 2\delta_{i,j} + \delta_{i,j-1} \) (\( \delta_{a,b} = 1 \), if \( a = b, \delta_{a,b} = 0 \), otherwise) and \( \lambda = k/h^2 \). We remark that in practice the values of \( u_m^n \) will be known from the previous time step or from the initial condition \( (u_m^0 = \psi(x_m)) \). Verify the expression 7 and the entries of \( A \) obtained from the numerical scheme given by equation 6.

d) To solve the linear system for \( u^{n+1} \) given by equation 7 implement from scratch a Gaussian elimination procedure with complete pivoting. Treat the values outside the range of \([0, \ldots, M]\) as zero, so that \( u_{-1} = 0 = u_{M+1} \). Use the scheme given by equation 7 to find the solution of the heat equation when the initial condition is \( u_m^0 = \psi(x_m) = \frac{1}{h}\delta_{m,M/2} \). The mesh should be centered about the origin extending from \([-L, L]\) for \( L \) large, say \( L = 50 \). Using this convention the initial data \( u_m^0 \) corresponds to the mesh site at the origin being \( 1/h \) while the other mesh sites are zero. This serves as an approximation of the Dirac \( \delta \)-function. Show that your solution agrees well with the analytic solution

\[ \rho(x, \tau) = \frac{1}{\sqrt{4\pi \tau}} e^{-x^2/4\tau}. \]  

(8)
at time $\tau = 1$. Show a plot of your approximate solution $u_{m}^{N}$ and $\rho(x_{m}, 1)$. For what time step $k$ and spatial step $h$ do you see good agreement for this test problem.

e) We shall now discuss how this solver can be used to price a call option. Recall that a call option gives the holder the right to sell an asset at the strike price $K$ at maturity time $T$. The “payoff” function for a call option is $\phi(s) = s - K$, if $s > K$, and $\phi(s) = 0$, if $s \leq K$. Use your change of variable formulas and state the appropriate initial condition $\psi(x)$ for the call option.

For the numerical values associated with the market place and the call option use $T = 10$, $r = 0.05$, $K = 1$, $\sigma^2 = 0.2$. Solve the Heat Equation with the initial condition $\psi$ until $\tau = 25$. From the numerical solution $u_{m}^{n+1}$ using your change of variable determine the value of the call option. Note that for a given $(m, n)$ this determines the option value $V(Ke^{x_{m}}, T - 2\tau_{n}/\sigma^2)$. Give a surface plot for the values $V$ of the call option as a function of the spot price $s$ and the time to maturity $T - t$ (see matlab surf and mesh). We remark that the boundaries introduced by making the mesh finite for numerical purposes will introduce some artifacts in the valuation of the option. As such, exercise some caution when interpreting the option values for spot prices in the vicinity of the boundaries.

f) If the spot price is the same as the strike price $s = K$ does a call option’s value become more or less valuable as the duration to maturity time increases? Think first about the case when $T - t = 0$ and then what happens as this time to maturity increases. What if the spot price is slightly above the strike price, say $s = 1.25K$? Note that both of these questions can be addressed by looking at the cross section of your surface plot for a fixed $s$. Present plots supporting your answers.