PURDUE UNIVERSITY · CS 31400
NUMERICAL ANALYSIS

HOMEWORK
David F. Gleich
April 30, 2021

Please answer the following questions in complete sentences in a typed manuscript and submit the solution on Gradescope by - April 30th at 5am if you want it graded before the final - May 3rd at noon if you want it graded after the final. (No penalty, just no feedback before the exam.)

Problem 0: Homework checklist

- Please identify anyone, whether or not they are in the class, with whom you discussed your homework. This problem is worth 1 point, but on a multiplicative scale.
- Make sure you have included your source-code and prepared your solution according to the most recent Piazza note on homework submissions.

Problem 1: (Ascher and Greif, Problem 16.5)

Consider the ODE

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), 0 \le t \le b$$

with $b \gg 1$.

1. Apply the stretching transformation $t = \tau b$ to obtain the equivalent ODE

$$\frac{d\mathbf{z}}{d\tau} = b\mathbf{f}(\tau b, \mathbf{z}), 0 \le \tau \le 1.$$

   And show the relationship between $\mathbf{y}(t)$ and $\mathbf{z}(\tau)$.

2. Show that applying the forward Euler method to the ODE for $\mathbf{y}$ in t with step size h_t is equivalent to applying the same method to the ODE for $\mathbf{z}$ in τ with step size h_τ that satisfies $h_t = bh_\tau$.

Problem 2: (Ascher and Greif, Example 16.20)

Consider the following system that describes the behavior of two masses in the presence of a third mass of (small) size. The masses are $\mu = 0.012277471$ and $\hat{\mu} = 1 - \mu$ (earth and sun):

$$\begin{aligned}
u_1'' &= u_1 + 2u_2' - \hat{\mu}\frac{u_1 + \mu}{D_1} - \mu\frac{u_1 - \hat{\mu}}{D_2}, \\
u_2'' &= u_2 - 2u_1' - \hat{\mu}\frac{u_2}{D_1} - \mu\frac{u_2}{D_2}, \\
D_1 &= ((u_1 + \mu)^2 + u_2^2)^{3/2}, \\
D_2 &= ((u_1 - \hat{\mu})^2 + u_2^2)^{3/2}.
\end{aligned}$$

The initial conditions are

$$\begin{aligned}
&u_1(0) = 0.994, u_2(0) = 0, \\
&u_1'(0) = 0, u_2'(0) = -2.00158510637908252240537862224.
\end{aligned}$$

Let $T = 17.1$.

1. Implement a 4th order RK method for this problem.

2. Show what happens for 1000, 5000 and 10000, 50000, 100000 steps. Discuss any interesting observations.
3. Use a standard integration software package such as `ode45` or one of the Julia ODE packages or an equivalent tool to study this problem.

Problem 3: (Gautschi Machine Exercise 5.5)

Newton's equations for the motion of a particle on a planar orbit (with eccentricity ε, $0 < \varepsilon < 1$) are

$$\begin{aligned}
&x'' = -\frac{x}{r^3}, x(0) = 1 - \varepsilon, x'(0) = 0,\\
&y'' = -\frac{y}{r^3}, y(0) = 0, y'(0) = \sqrt{\frac{1+\varepsilon}{1-\varepsilon}},\\
&r^2 = x^2 + y^2,\\
&\quad 0 \le t \le 1.
\end{aligned}$$

1. Verify that the solution can be written in the form of

$$\begin{aligned}
x(t) &= \cos u(t) - \varepsilon,\\
y(t) &= \sqrt{1-\varepsilon^2}\sin u(t),\\
u(t) &- \varepsilon \sin u(t) - t = 0.
\end{aligned}$$

2. Write a program to solve the nonlinear equation at each step using Newton's method and plot the exact solution for $\varepsilon = 0.3, 0.5, 0.7$.
3. Write a forward and backwards Euler solver for this problem.
4. Evaluate the error with $N = 50, 120, 200, 500$, and 1000 time-steps using the same values of ε.

Problem 4: Raptors redux

Recall the Raptor problem from the midterm.

```
function simulate_raptors(angle)
    vhuman=6.0
    vraptor0=10.0 # the slow raptor velocity in m/s
    vraptor=15.0 #

    raptor_distance = 20.0

    raptor_min_distance = 0.2 # a raptor within 20 cm can attack
    tmax=10.0 # the maximum time in seconds
    nsteps=1000

    # initial positions
    h = [0.0,0.0]
    r0 = [1.0,0.0]*raptor_distance
    r1 = [-0.5,sqrt(3.)/2.]*raptor_distance
    r2 = [-0.5,-sqrt(3.)/2.]*raptor_distance

    # how much time el
    dt = tmax/nsteps
    t = 0.0

    hhist = zeros(2,nsteps+1)
    r0hist = zeros(2,nsteps+1)
```

```
    r1hist = zeros(2,nsteps+2)
    r2hist = zeros(2,nsteps+2)

    hhist[:,1] = h
    r0hist[:,1] = r0
    r1hist[:,1] = r1
    r2hist[:,1] = r2

    tend = tmax

    """
    This function will compute the derivatives of the
    positions of the human and the raptors
    """
    function compute_derivatives(angle,h,r0,r1,r2)
        dh = [cos(angle),sin(angle)]*vhuman
        dr0 = (h-r0)/norm(h-r0)*vraptor0
        dr1 = (h-r1)/norm(h-r1)*vraptor
        dr2 = (h-r2)/norm(h-r2)*vraptor
        return dh, dr0, dr1, dr2
    end

    for i=1:nsteps
        dh, dr0, dr1, dr2 = compute_derivatives(angle,h,r0,r1,r2)
        h += dh*dt
        r0 += dr0*dt
        r1 += dr1*dt
        r2 += dr2*dt
        t += dt

        hhist[:,i+1] = h
        r0hist[:,i+1] = r0
        r1hist[:,i+1] = r1
        r2hist[:,i+1] = r2

        if norm(r0-h) <= raptor_min_distance ||
            norm(r1-h) <= raptor_min_distance ||
            norm(r2-h) <= raptor_min_distance

            # truncate the history
            hhist = hhist[:,1:i+1]
            r0hist = r0hist[:,1:i+1]
            r1hist = r1hist[:,1:i+1]
            r2hist = r2hist[:,1:i+1]
            tend = t
            break
        end
    end
    return tend
end
```

1. Modify this code to use Heun's method instead of forward Euler and comment on how the results change with angle.
2. **Not assigned in the end** (Tentative, not currently assigned.) Something using Backward Euler. This involves solving the nonlinear system which is more like the previous problem.