H. Ammari
B. Davies
A. Vanel

Spring Term 2020

Numerical Analysis II

ETH Zürich
D-MATH

Final Exam Winter 2020

Dice mark difficulty of corresponding problem. ⚀ stands for the easiest, and ⚅ stands for the hardest.

Problem 1 [40 points]

Consider the system of Lotka-Volterra's ODEs given by

$$\begin{cases} \dfrac{\mathrm{d}u}{\mathrm{d}t} = u(v-2), \\ \dfrac{\mathrm{d}v}{\mathrm{d}t} = v(1-u). \end{cases} \tag{1.1}$$

where $u, v : [0, +\infty) \to \mathbb{R}$.

We would like to apply Explicit Euler method, Implicit Euler method and Symplectic Implicit Euler method on equation system (1.1) and check their symplecticity. Since (1.1) is not a Hamiltonian system, a transformation is needed.

(1a) ⚁ Apply $p = \ln u$ and $q = \ln v$ to system (1.1). Verify the resulting system is Hamiltonian with Hamiltonian function $I(p,q) = p - e^p + 2q - e^q$.

Now consider the following three numerical schemes:

Explicit Euler Method for symplectic system:

$$\begin{cases} p^{k+1} = p^k - \Delta t \, \dfrac{\partial I}{\partial q}(p^k, q^k), \\ q^{k+1} = q^k + \Delta t \, \dfrac{\partial I}{\partial p}(p^k, q^k). \end{cases} \tag{1.2}$$

Implicit Euler Method for symplectic system:

$$\begin{cases} p^{k+1} = p^k - \Delta t \, \dfrac{\partial I}{\partial q}(p^{k+1}, q^{k+1}), \\ q^{k+1} = q^k + \Delta t \, \dfrac{\partial I}{\partial p}(p^{k+1}, q^{k+1}). \end{cases} \tag{1.3}$$

Implicit Symplectic Method for symplectic system:

$$\begin{cases} p^{k+1} = p^k - \Delta t \, \dfrac{\partial I}{\partial q}(p^{k+1}, q^k), \\ q^{k+1} = q^k + \Delta t \, \dfrac{\partial I}{\partial p}(p^{k+1}, q^k), \end{cases} \tag{1.4}$$

We apply these three methods to modified system from **(1a)**.

(1b)

Implement (1.4) in time span $T = [0, 12]$ with step size $h = 0.05$ by filling in templates `ImpEulerSymplecticSolve.m` and `LV_symplectic.m`. For the initial condition, use $p(0) = 2$ and $q(0) = 2$.

For Explicit Euler and Implicit Euler schemes, the codes are already filled in and given in the template folder, they are `EulerStep.m` `EulerSolve.m`, `ImpEulerStep.m` and `ImpEulerSolve.m`.

Plot the graph of invariance $I(p, q)$ to all the three methods. Is $I(p, q)$ preserved by any of these methods?

(1c) Are the methods (1.2), (1.3) and (1.4) symplectic? Justify.

(1d) Prove that (1.4) is of order 1. Since this is a multi-dimensional case, one has to prove that each element in the truncation error vector is of order 1.

Problem 2 [40 points]

Consider the system of equations

$$\begin{cases} \dfrac{\mathrm{d}p}{\mathrm{d}t} = q, & t \geq 0, \\ \dfrac{\mathrm{d}q}{\mathrm{d}t} = \cos p, \\ p(0) = p_0 \in \mathbb{R},\ q(0) = q_0 \in \mathbb{R}. \end{cases} \tag{2.1}$$

(2a) Is (2.1) a Hamiltonian system?

(2b) Rewrite (2.1) in the form

$$\begin{cases} \dfrac{\mathrm{d}x}{\mathrm{d}t} = f(x), & \text{where } x = (p, q)^T \\ x(0) = x_0 = (p_0, q_0)^T. \end{cases} \tag{2.2}$$

(2c) Consider the midpoint scheme

$$\begin{cases} x^{k+1} = x^k + \Delta t f\Big(\dfrac{x^k + x^{k+1}}{2}\Big), \\ x^0 = x_0. \end{cases} \tag{2.3}$$

for solving (2.2) where Δt is the step size. Prove that the method is symplectic.

(2d) Suppose that there exists a matrix $S \in \mathbb{R}^{2\times 2}$ (symmetric and positive definite) and a positive constant c such that

$$x(t)^T S x(t) = c, \quad \forall t \geq 0, \quad \text{where } x \text{ is the solution to (2.2).} \tag{2.4}$$

Prove that $(x^k)^T S x^k = c$ for $k \geq 0$, where x^k is from (2.3).

(2e) Let us denote $x^k = (p^k, q^k)$. Does the midpoint scheme (2.3) preserve the quantity

$$-\frac{1}{2} q_k^2 + \sin p_k \ ?$$

(2f)

- Implement the midpoint scheme (2.3) using templates `ImpMidPointSolve.m` and `ImpMidPointStep.m`. `y1=ImpMidPointStep(f,y0,t0,h,tol)` should return the value of one-step midpoint scheme (2.3), and `y=ImpMidPointSolve(f,y0,T,h,tol)` should return the approximate value of IVP at the end time T. In `ImpMidPointStep.m`, we use the fixed point iteration as the method for root-finding problem in implicit method. The parameters `f`, `y0`, `T`, `h`, `tol` stand for the right-hand side of IVP, initial value of IVP, ending time, step size and error tolerance, correspondingly.

- Complete the template `RunPrb2.m` to solve (2.2) on time interval $[0, 8]$ using your codes `ImpMidPointSolve.m` and `ImpMidPointStep.m`. Set the initial value as $x(0) = [\pi/2, 0]^T$. Set the time step `h` as $h = 0.25$ and set the error tolerance `tol` as $tol = 10^{-6}$.

- At the end of `RunPrb2.m`, template code will plot the computed numerical solution.

Problem 3 **[20 points]**

Let us consider the following Runge-Kutta method:

$$\begin{aligned} y^{n+1} &= y^n + h\left(\frac{1}{6}k_1 + \frac{2}{3}k_2 + \frac{1}{6}k_3\right) \\ k_1 &= f(t_n, y_n) \\ k_2 &= f\left(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1\right) \\ k_3 &= f(t_n + h, y_n - hk_1 + 2hk_2) \end{aligned}$$

(3a) Show that the method is consistent.

Consider the problem

$$\begin{cases} y' = \cos(t) + y, & t \in (0, 1], \\ y(0) = -1. \end{cases} \tag{3.1}$$

(3b) Using the method of integrating factors, show that (3.1) has solution

$$y(t) = \frac{1}{2}(-\exp(t) + \sin(t) - \cos(t)).$$

(3c) Implement in Matlab a program for the method using template `rk3.m` and verify with `RKmethodscript.m` that the order of accuracy is 3 with respect to h for (2.2).