Temple Math 5044 **Introduction to Numerical Analysis II** **Spring 2019**

Problem Set 3

(Out Mon 02/04/2019, Due Mon 02/11/2019)

Problem 3

Consider again our beloved Lotka-Volterra predator-prey model

$$\frac{d}{dt}\vec{u}(t) = \vec{f}(\vec{u}(t)) \ , \tag{1}$$

where $\vec{u} = \begin{pmatrix} u \\ v \end{pmatrix}$ and $\vec{f}(\vec{u}) = \begin{pmatrix} f_1(u,v) \\ f_2(u,v) \end{pmatrix} = \begin{pmatrix} u - 4uv \\ -v + 5uv \end{pmatrix}$, and use it to perform a numerical error analysis of various numerical methods. To that end, consider $t \in [0, 31]$, and initial conditions $\vec{u}(0) = \begin{pmatrix} 0.73 \\ 0.25 \end{pmatrix}$.

(a) First, obtain a reference solution[1] for the final state $\vec{u}(31)$, by using RK4 with step size $h = 10^{-4}$ (or smaller if you like). Call this vector $\vec{u}_{\text{true}}$.

(b) For a sequence of time steps $k = 10^{\beta}$, where $\beta \in \{-3.0, -2.75, -2.5, -2.25, -2.0, -1.75, -1.5, -1.25, -1.0\}$, approximate[2] the final state $\vec{u}(31)$ by using the following numerical schemes: (i) forward Euler, (ii) RK4, (iii) Crank-Nicolson, and (iv) the two-stage Gauss-Legendre method.[3]

(c) For each of the 4×9 results for the final state that you obtained in (b), compute the error $\|\vec{u}(31) - \vec{u}_{\text{true}}\|_2$, and plot the sequence (with respect to h) of errors in four curves in log–log scale. Explain what the slopes of these curves tell you, and what their abscissae tell you.

(d) In two new figures, plot the approximate solution curves $(u(t), v(t))$, obtained with (i) RK4 and (ii) Crank-Nicolson for system (1), with $t \in [0, 30000]$ and time step size $k = 0.3$.[4] What do you observe? Explain your observations.

Instructions

For each problem set, you need to submit one document, either in class or via email to the course instructor, that contains plots and explanations (hand-written or typed). If you decide to email the document, name it `yourfamilyname_problemset1.pdf`, where 1 stands for the number of the problem set.

In addition, for each programming task, email your respective program to the course instructor, under the filename `yourfamilyname_problem1a.m`, where `1` stands for the problem number and `a` for the sub-problem letter.

---

[1] If an analytical solution is unavailable, one can use a highly resolved computational approximation in lieu of it.

[2] Don't forget to round k so that you do an integer number of time steps.

[3] Use three Newton iteration steps per time step for the implicit schemes.

[4] Yes, you are doing a lot of time steps here. See this as a simulation of an ecological systems over a millennium time scale.