Problem set 10

[Revised Feb 06, 9:21 PM]

Solving Ordinary Differential Equations

1. Consider the following ODE

$$\frac{dy}{dt} = -2y + \cos(4t), \qquad t \in [0, 8]$$

subject to the initial condition

$$y(0) = 3.$$

The exact solution to this IVP is

$$y(t) = 0.1\cos(4t) + 0.2\sin(4t) + 2.9\exp(-2t).$$

In `euler.py` an empty code implementing the Forward Euler method is provided. Study the function `def ivp_solve_euler(t0, t1, y0, dt)`

- Implement the RHS required for this ODE.
- Run Euler's method like this

```
t, y, ye = ivp_solve_euler(0.0, 8.0, 3.0, dt)
```

The script provided will report the error (difference between the exact and numerical solution) at the final time. * Use timesteps `dt` with values `0.1, 0.05, 0.025, 0.0125, 0.00625` and verify that the error at the final time is $O(h)$ accurate.

2. Consider the same problem in Q1, however this time we you will solve it using backward Euler's method. To do this, make a copy of `euler.py` called `beuler.py`. Rename `ivp_solve_euler()` to `ivp_solve_beuler()`. Change the definition of how the update for `y` is performed such that it implements the backward Euler method.

- Use timesteps `dt` with values `0.1, 0.05, 0.025, 0.0125, 0.00625` and verify that the error at the final time is $O(h)$ accurate.

3. Consider the same problem in Q1, however this time you will solve it using the fourth order Runge Kutta method (RK4). To do this, make a copy of `euler.py` and call it `rk4.py`. Rename `ivp_solve_euler()` to `ivp_solve_rk4()`. Change the definition of how the update for `y` is performed such that it implements RK4.

    - Use timesteps `dt` with values `0.1, 0.05, 0.025, 0.0125, 0.00625` and verify that the error at the final time is $O(h^4)$ accurate.

4. For this question you will examine the stability of Euler's method and backward Euler. You will solve the ODE

$$\frac{dy}{dt} = -20y, \quad t \in [0, 2], \quad y(0) = 3$$

which has the exact solution

$$y(t) = 3\exp(-20t).$$

    - Make a copy of `euler.py` and call it `euler_stiff.py`. Within `euler_stiff.py`:

        - Modify the definition of the exact solution `y_exact` with $y(t) = 3\exp(-20t)$.
        - Modify the RHS update.
        - The conditional stability criterion is $\Delta t \leq 2/20 = \Delta t_{\text{stable}}$. Plot the solution obtained using $\Delta t = 1.1\Delta t_{\text{stable}}$ and $\Delta t = 0.1\Delta t_{\text{stable}}$.

    - Make a copy of `beuler.py` and call it `beuler_stiff.py`. Within `beuler_stiff.py`:

        - Modify the definition of the exact solution `y_exact` with $y(t) = 3\exp(-20t)$.
        - Modify the update of `y`.
        - Plot the solution obtained using $\Delta t = 1.1\Delta t_{\text{stable}}$ and $\Delta t = 0.1\Delta t_{\text{stable}}$, where $\Delta t_{\text{stable}} = 2/20$.