Methods of Mathematical Physics I: Numerical Analysis 2
IUCAA-NCRA Graduate School
August - September 2012

22 August 2012
To be returned in the class on 31 August 2012

- The deadline for the submission of the solutions of this assignment will be strictly enforced. No credits will be given if the assignment is not returned in time.
- You are free to discuss the solutions with friends, seniors and consult any books. However, you are expected to write the numerical program by yourself. You should understand and be clear about every step in the program. Marks may be reduced if you have not understood what you have written even though the answer and the plots are correct.
- Let me know if you find anything to be unclear or if you think that something is wrong in any of the questions.

Stiff Differential Equations:

Consider the following set of equations:

$$\dot{u}(t) = 998u(t) + 1998v(t); \quad \dot{v}(t) = -999u(t) - 1999v(t)$$

with boundary conditions

$$u(0) = 1; \quad v(0) = 0.$$

You can show that the solution of the above system will be given by

$$u(t) = 2\mathrm{e}^{-t} - \mathrm{e}^{-1000t}; \quad v(t) = -\mathrm{e}^{-t} + \mathrm{e}^{-1000t}.$$

1. Solve the above system of equations from $t = 0$ to $t = 2$ using the simple Euler's method:

   $$u(t + \Delta t) = u(t) + \Delta t\ \dot{u}(t); \quad v(t + \Delta t) = v(t) + \Delta t\ \dot{v}(t).$$

   Take $\Delta t = 0.1$ and try to plot $u(t)$ and $v(t)$.

   [10 marks]

2. Now repeat the same problem with $\Delta t = 0.001$. Compare with the exact analytical solution in the same plot.

   [5 marks]

You might have realized that the system is extremely unstable if you take the time-step to be larger than $\sim 10^{-3}$. This happens because of the presence of the e^{-1000t} term in the solution. Though this term is completely negligible in determining the values of u and v as soon as one is away from the origin, its presence forces us to choose a step size as small as 10^{-3} to make the system stable. In realistic physical problems, one may require unacceptably high computing power to solve such equations with small step sizes. This difficulty cannot be addressed by simply using a more advanced scheme like leap-frog or Runge-Kutta algorithms.

A set of differential equations where there are two or more very different scales of the independent variable on which the dependent variables are changing are called **stiff equations**. A somewhat better method of solving them would be to use **implicit differencing scheme** where the derivatives are evaluated at the new time t, i.e.,

$$u(t + \Delta t) = u(t) + \Delta t\ \dot{u}(t + \Delta t); \quad v(t + \Delta t) = v(t) + \Delta t\ \dot{v}(t + \Delta t).$$

Use the expressions for $\dot{u}$ and $\dot{v}$ given in the problem and write $u(t + \Delta t),\ v(t + \Delta t)$ in terms of $u(t),\ v(t)$. Since the system of differential equations is linear, you can do this analytically.

3. Now, using the new scheme, solve the problem from $t = 0$ to $t = 2$ taking $\Delta t = 0.1$. Plot $u(t)$ and $v(t)$ and compare with the exact analytical solution.

   [15 marks]

Root Finding:

If the system of equations in the previous problem was not linear, then it would not have been so straightforward to write $u(t+\Delta t),\ v(t+\Delta t)$ in terms of $u(t),\ v(t)$ using the implicit differencing scheme. In such cases, one has to take help of root finding algorithms to solve the equations.

A root finding problem is defined as finding the roots of the equation $f(x) = 0$. The simplest method for doing this is called the **bisection method**. The idea of this method is based on the fact that a function will change sign when it passes through zero. So start with an interval (a_0, b_0) so that $f(a_0)f(b_0) < 0$, i.e., we know that the root lies between these two limits. Then evaluate the function at the intervals midpoint and examine its sign, i.e., check the sign of $f((a_0 + b_0)/2)$. Use the midpoint to replace whichever limit has the same sign. So the bounds containing the root will decrease by a factor of two. You can keep on repeating this procedure and making the interval smaller and smaller. The algorithm will stop when you have achieved the required tolerance limit ϵ, i.e., $|f(x_k)| < \epsilon$.

4. Suppose we want to find the value of $x = x_0$ where the function

$$u(x) = \frac{x^3}{\mathrm{e}^x - 1}; \quad x > 0$$

has a maximum. Differentiate $u(x)$, use the condition $u'(x_0) = 0$ and write down the algebraic equation for the location of the maximum x_0. Then write a code to solve the equation and find the numerical value of x_0. You can take the tolerance ϵ to be 10^{-3}.

[20 marks]