Introduction to Numerical Analysis and Applications

Cheng-Chieh Cheng, Ph.D.
Dept of Computer Science and Engineering,
National Sun Yat-sen University, Kaohsiung, Taiwan

Homework #1 (**DUE**: March 19th)

This homework covers the materials from the first two lectures, 1) *Round-off errors & Truncation errors*, and 2) *Seeking Roots*.

1. (5%) In the lecture last week, the forward finite-difference approximation was derived. Now it's your turn to derive the centered finite-difference approximation of the 1st derivative:
   $f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1})}{2h} + O(h^2).$

2. We have a function out of nowhere: $f(x) = 10 \times cos\left(\frac{3\pi}{2}x\right) \times e^{-x} + x^6 - 1$.
   a. (10%) Figure. Plot the above funny function in the range of [–2.5, 2.5], then use graphical method to find all roots (to 2 significant figures). Can you make a prediction of whether there exists other root(s) outside the range of [–2.5, 2.5], based on the curve you plot and/or the function? Why do you make such prediction?
   b. (10%) Figure. Use both bisection and false position methods to find the root in bracket [0, 0.5] (*The stopping criteria:* $|\varepsilon_a| < \varepsilon_s = 10^{-2}$ %). How does each method converge? Also plot the curves of relative error, $\varepsilon_a(\%)$, for each iteration.
   c. (10%) Figure. Now plot the error curves for another root in bracket [0.5, 2].

3. The next question is going to be a thick one, so please bear with me. ☺ Let's start with arrays, vectors, and matrices:
   a. Imagine a unit vector pointing to +z-axis in a 3D space. How would you rotate such vector around the +x-axis by an angle θ? Design a rotation matrix to rotate you 3D vector to do so. (let's name it as $\boldsymbol{R}_{x,\theta}$.)
   b. The second element is a magnitude modulation function, namely $\boldsymbol{D_1}$, such that $S^+ = \boldsymbol{D_1}S^-$. S represents the "signal" we will play around in this set of problems, with superscripts – and + indicating the state of such signal before and after the modulation. The modulation affecting x- and y-components in the same fashion, e.g. $S_x^+ = S_x^- \cdot e^{-t/T_2}$; while the z-component is modulated as $S_z^+ = S_z^- \cdot e^{-t/T_1}$. On top of that, we add another column vector $\boldsymbol{D}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 - e^{-t/T_1} \end{bmatrix}$, after the signal is modulated, i.e. $S^+ = \boldsymbol{D}_1 S^- + \boldsymbol{D}_2$.

c. (20%) The last element is used to "clean up" the x- and y-components, i.e., $S^+ = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} S^-$. Pack all the above elements in one function, such that $S^+ = f(\theta, t, S^-, T_1, T_2)$. This function is a much simplified version of Bloch Equations-based simulator to study the signal behavior in magnetic resonance. You should indicate your signal simulator function, $f(\theta, t, S^-, T_1, T_2)$ in the code.

d. The function defined in part c should be deployed in a loop of 500 iterations, with the initial unit vector pointing to +z, and $t/T_1/T_2 = 250/1500/200$. Answer the following questions by evaluating the vector projected onto the xy-plane, i.e., $S_{xy} = \sqrt{S_x^2 + S_y^2}$. Be sure to output S_{xy} *before* the "crusher" defined in part c, otherwise your results will always be zero.

e. (10%) The goal for this exercise is to find the optimal θ ($0° \leq \theta \leq 90°$) that gives the strongest S_{xy}. Plot your simulation result in a figure (e.g. S_{xy} vs. θ) and use graphical method to make a rough estimate of θ_{opt}.

f. (15%) We know that the 1st derivative is handy when it comes to finding the maxima or minima, i.e. extrema occur at x where $f'(x) = 0$. Now define $g(\theta) = f'(\theta)$ and thus we need to find the root of $g(\theta)$ for this optimization problem. Use centered finite-difference approach to approximate and illustrate $g(\theta)$ in a figure.

g. (15%) Now use the Golden-section search to solve this 1D optimization problem with the stopping criterion of $\varepsilon_s = 10^{-3}$. List your results (for example x_l and x_u) at every iteration.

h. (5%) This optimization of θ can be solved analytically: $\theta_{opt} = \arccos\left(e^{-t/T_1}\right)$. Compare your numerical solution with analytical one. How would you improve your numerical approach?

Historical moment: In the field of magnetic resonance, this optimal angle is best known as the Ernst angle, named after Richard R. Ernst, who won the 1991 Nobel prize in Chemistry. ☺

Advanced discussion: If you're interested in knowing more about Ernst angle and how it could affect the MR image contrast, you are encouraged to visit the following webpages:

http://mriquestions.com/optimal-flip-angle.html

http://mriquestions.com/spoiled-gre-parameters.html