Case Study # 1: 1D Transient Heat Diffusion

Background:

The problem of 1D unsteady heat diffusion in a slab of unit length (non-dimensional) with a zero initial temperature (non-dimensional) and both ends maintained at a unit temperature (non-dimensional):

$$\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2} \quad \text{subject to} \quad \begin{cases} T(x, 0^-) = 0 & \text{for } 0 \leq x \leq 1 \\ T(0, t) = T(1, t) = 1 & \text{for } t > 0 \end{cases}$$

has an analytical solution:

$$T^*(x, t) = 1 - \sum_{k=1}^{\infty} \frac{4}{(2k-1)\pi} \sin\left[(2k-1)\pi x\right] \exp\left[-(2k-1)^2\pi^2 t\right]$$

Investigation:

Use the Taylor-series (TSE) method to derive a finite difference approximation to the PDE as demonstrated in class.

Write a Python script to solve this problem numerically using a Forward-Time, Centered-Space (FTCS) explicit scheme and then a FTCS implicit scheme. Use a 21 point mesh ($N = 21$). Compare the Root Mean Square error:

$$\text{RMS} = \frac{1}{N}\sqrt{\sum_{i=1}^{N} \left[T_i^n - T^*(x_i, t_n)\right]^2}$$

obtained for $s(= \Delta t/\Delta x^2) = 1/6$, 0.25, 0.5 and 0.75, at $t = 0.03$, 0.06, and 0.09 (n.d.). Comment your results for each value of s.

A sample Python script for the explicit FCTS case is posted with this assignment; use it as a starting point for your analysis. For the FTCS implicit case, you can first use NumPy's "`numpy.linalg.solve`" operator to solve the resulting system of algebraic equations, but your final solution should use your own implementation of Thomas' algorithm.

NOTE: If you are not familiar with python and do not have it installed on your machine, please refer to the "Lectures" section of the course SmartSite where links to appropriate resources to get you started are provided. The `Spyder` environment is recommended as it provides on install the extensions and libraries needed for numerical analysis and plotting with python (`NumPy`, `SciPy`, `Matplotlib` and `IPython`). Various links to documentation and tutorials are also provided.

Report:

Prepare a report (2-4 pages max in ASME's two-colum article format, templates are available on-line) describing your work and including:

1. Problem description
2. A description of your solution algorithms (can be in pseudo-code, block diagram, or flow chart form)

3. For each value of s
   - A table with the computed Root Mean Square error at $t = 0.03$, 0.06, and 0.09 (n.d.).
   - Plots comparing the predicted temperature profiles to the analytical solution for three values of t between 0 and 0.09.
4. A discussion section
5. APPENDIX: Source code (`.py`, uploaded as a separate file)

The Report (single file, PDF format only) and source code are **due on October 19, 2014 at 5pm** and must be submitted electronically using the class SmartSite.