FIRST MIDTERM

Scientific Computing II
Fundación Universitaria Konrad Lorenz
September 05, 2023

- This is an **individual exam**. Therefore, do not provide or receive help from anyone for the completion of this midterm.
- The exam will be available from September 5th, 2023, 6.15pm. From that time on you will have 2 hours and 15 minutes to complete the midterm exam.
- The solution must be turned in only through the Aula virtual. Midterms sent over email will not be graded.
- Full score will only be given to correct and completely justified answers. Miraculous, obtuse and unnecessarily complex solutions will receive partial or null score.
- You can ask any question of this assignment during class or through email: juliano.jimenezc@konradlorenz.edu.co.

Finite Differences Method

The Taylor series expansion of a univariate function f, centered at x_0, is given by

$$f(x_0+h)=\sum_{n=0}^{\infty}\frac{f^{(n)}(x_0)}{n!}h^n=f(x_0)+\frac{f'(x_0)}{1!}h+\frac{f''(x_0)}{2!}h^2+\cdots, \tag{1}$$

so that, at first-order with respect to h, it follows that

$$f(x_0+h)=f(x_0)+f'(x_0)h+O(h^2),$$

from where we get that, if h is sufficiently small, then

$$f'(x_0)\approx\frac{f(x_0+h)-f(x_0)}{h}. \tag{2}$$

If we do a similar procedure for the first-order derivative of f, we get that

$$f''(x_0)\approx\frac{f(x_0+h)-2f(x_0)+f(x_0-h)}{h^2}. \tag{3}$$

Heat Equation

This approach can be extended to solve partial differential equations. For example, consider the unidimensional heat equation, describing the temperature $U(x,t)$ at a point x and time t of a rod of length π, whose ends are at temperature of 0°C:

$$\frac{\partial U}{\partial t}=\frac{\partial^2 U}{\partial x^2}, \tag{4}$$

subject to the following initial and boundary conditions:

$$U(0,t) = U(\pi,t) = 0 \text{ (boundary conditions)}, \tag{5}$$

$$U(x,0) = U_0(t) = \sin x \text{ (initial conditions)}. \tag{6}$$

We partition the space and time using a mesh $x_0 = 0, x_1 = h, ..., x_j = jh, ..., x_J = \pi$ and $t_0 = 0, t_1 = k, ..., t_n = nk, ..., t_J = T$, respectively. The points

$$u(x_j, t_n) = u_j^n$$

will represent the numerical approximation of $U(x_j, t_n)$. Using the finite differences method, the PDE (4) can be approximated as follows:

$$\frac{u_j^{n+1} - u_j^n}{k} = \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{h^2}. \tag{7}$$

1. (0/30) Use equation (7) to numerically integrate the PDE (4) subject to conditions (5). Graph the solution as a 3D surface or as a colormap, where the evolution of the rod's temperature as a function of time can be appreciated. Remember to:

- Modularize and comment your code.
- Optimize your code looking for possible bottlenecks and using the optimization strategies seen in class.
- Profile your code if required.
- Some variables were not explicitly defined, like k, h, T. Your code should receive them as input/arguments.

Monte-Carlo Integration

Monte Carlo integration is a numerical method for approximating definite integrals by employing random sampling. It involves generating random points within a known area encompassing the integration domain, evaluating the function at these points, and then averaging the function values, scaled by the size of the known area, to estimate the integral's value.

For example, we estimated π by sampling the unit square and counting the number of points inside and outside the circle. We could also have estimated the area of the circle using this method. Now we will extend this method to calculate the hypervolume of an n-sphere (note that the circle is a 2-sphere, and the $3D$ sphere is a 3-sphere).

Let's consider a hypercube with sides of length 2 that completely contains the n-dimensional sphere of radius 1. The volume of this hypercube is:

$$V_{cube} = 2^n. \tag{8}$$

Now, we should generate K uniformly distributed random points within this hypercube, where each component of the point is randomly chosen between -1 and 1 in each dimension. After that, we count how many of these random points $\boldsymbol{x}$ are inside the n-dimensional sphere by measuring its distance to the origin. Namely, if

$$\boldsymbol{x}^T\boldsymbol{x} \leq 1,$$

the point is inside the n-dimensional sphere. Moreover, as $K \to \infty$, we expect that

$$\frac{V_{n-sphere}}{V_{cube}} \approx \frac{|\{\text{points inside the n-sphere}\}|}{K}.$$

2. (0/20) Calculate the hypervolume of the n-sphere for $n = 2, 3, \ldots, 25$ using Monte-Carlo Integration. **Do the calculations in parallel for each dimension**, taking the number of processors that you consider adequate (at least 6), and take 10^6 (a million) samples per processor. Make a graph of the dimension vs the hypervolume of the unit n-sphere. Remember to:

- Modularize and comment your code.
- Optimize your code looking for possible bottlenecks and using the optimization strategies seen in class.
- Profile your code if required.