MA3227 Numerical Analysis II

Tutorial 3: Finite differences

Non-uniform grids

The aim of this exercise is to solve the one-dimensional Poisson equation

$$-u''(x) = f(x) \qquad \text{for all } x \in (0,1)$$

with homogeneous Dirichlet boundary conditions

$$u(0) = u(1) = 0$$

using a non-uniform grid

$$0 = x_0 < x_1 < \ldots < x_n < x_{n+1} = 1.$$

We will see that doing so can greatly improve the performance of the finite difference method under certain circumstances.

Let us begin by deriving and implementing the linear system associated with finite differences on non-uniform grids.

1. Convince yourself that

   $$u'\Big(\tfrac{x_{i+1}+x_i}{2}\Big) \approx \tfrac{u(x_{i+1})-u(x_i)}{x_{i+1}-x_i}$$

   and

   $$u''(x_i) \approx \tfrac{2}{x_{i+1}-x_{i-1}}\Big(\tfrac{1}{x_i-x_{i-1}}\,u(x_{i-1}) - \big(\tfrac{1}{x_i-x_{i-1}} + \tfrac{1}{x_{i+1}-x_i}\big)\,u(x_i) + \tfrac{1}{x_{i+1}-x_i}\,u(x_{i+1})\Big)$$

   are reasonable finite-difference approximations to the first and second derivatives on a non-uniform mesh $(x_i)_{i=0}^{n+1}$.

   Hint. There is not much to be done for $u'(x)$, but you will likely require pen and paper and a couple of minutes to derive the formula for $u''(x)$ from the formula for $u'(x)$.

2. Complete the function `laplacian(x)` so it assembles the Laplacian matrix based on the above approximation to $u''(x_i)$. You can test your code by running the provided function `test_laplacian()`.

Next, let us consider an example of a right-hand side $f(x)$ where using non-uniform grids is crucial for achieving optimal performance.

3. Verify that the pair of functions

   $$f(x) = \tfrac{1}{4}\,x^{-3/2}, \qquad u(x) = x^{1/2} - x$$

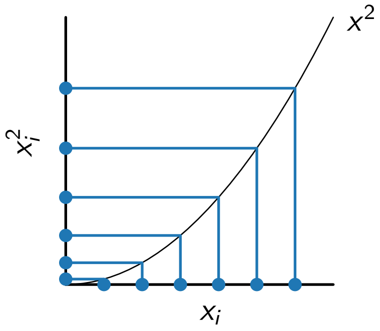
   satisfy the Poisson equation specified above.

4. Run the provided function `plot_solution()` and observe how the finite difference solution (the blue line) looks the "least natural" / "most kinky" near $x \approx 0$. This is because the slope of $u(x)$ blows up for $x \to 0$,

   $$\lim_{x\to 0} u'(x) = \tfrac{1}{2}\,x^{-1/2} + 1 = \infty,$$

   and this unequal variability in $u(x)$ is not reflected in our uniformly spaced grid.

In order to do better, we need to cluster our grid points towards the left end of the interval. A simple way to achieve this is to take uniformly spaced gridpoints $x_i = \frac{i}{n+1}$ and raise them to some power $p > 0$. This is illustrated in the following plot



5. Run `convergence()` and observe how clustering the grid points towards the left improves the convergence of the method. Keep increasing the value of p until the slope of the convergence curve stops improving (this should happen at $p \approx 5$).