Finite Differences and Finite Elements: A Graded Exercise

In the SCCS book, Chapter 23[1] and the solution to its challenges provide algorithms and software for solving boundary value problems for ordinary differential equations.

A major limitation of the software is that it uses a uniform mesh. Often we would like to have more mesh points concentrated in some parts of the domain Ω, so in this homework we develop the algorithms and software to do this.

We'll consider an important special case: a *graded mesh*. In particular, we will restrict ourselves to meshes that have equal numbers of mesh points in each of the 4 intervals $[0, 1/8]$, $[1/8, 1/4]$, $[1/4, 1/2]$, and $[1/2, 1]$. Such meshes are appropriate for problems in which the solution changes more quickly near 0 than near 1. First we develop finite difference formulas for graded meshes.

CHALLENGE 1 *Use Taylor series expansions to determine the values of* a, b, c, d, e *so that the finite difference approximations*

$$u'(x) \approx \frac{au(x+h) - bu(x-h/2)}{h},$$

$$u''(x) \approx \frac{cu(x+h) - du(x) + eu(x-h/2)}{h^2},$$

are as accurate as possible.

Unfortunately, we see that the formulas derived in Challenge 1 are not as accurate as the formulas from the chapter, so we will use the old ones.

CHALLENGE 2 *Write a* MATLAB *function* `finitediff2g.m`*, based on* `finitediff2.m`*, that uses a graded mesh instead of a uniform one. The input value* `M` *should be the number of meshpoints in the interval* $[1/2, 1]$*. Make sure that your function is well documented, clearly indicating the original source and the modifications that you made.*

Now we need to make the same changes to `fe_linear.m`, using a set of basis functions like those in Figure 1.

[1] Chapter 23 is "Finite Differences and Finite Elements: Getting to Know You".

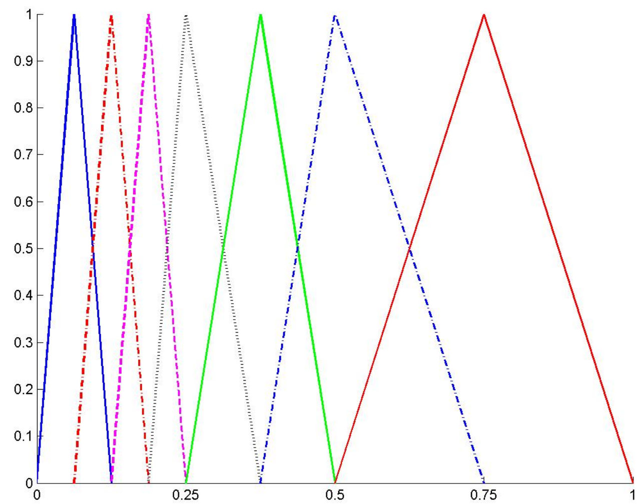

Figure 1: Finite element basis functions when there are 3 points in each subinterval.

CHALLENGE 3 *Write a* MATLAB *function* `fe_linearg.m`*, based on* `fe_linear.m`*, that uses a graded mesh instead of a uniform one. The input value* `M` *should be the number of meshpoints in the interval* $[1/2, 1]$*. Make sure that your function is well documented.*

Now that we have the software, let's see how it works.

CHALLENGE 4 *Test* `finitediff2.m`*,* `fe_linear.m`*,* `fe_quadratic.m`*,* `finitediff2g.m`*, and* `fe_linearg.m` *on the problem with* $a(x) = 1+x^2$*,* $c(x) = 0$ *and true solution defined by* `utrue.m`*. In the graded meshes, use 3, 33, and 333 points per subinterval, and use 9, 129, and 1329 points for the uniform meshes. Make a table containing the maximum absolute error for each approximation at the points* `linspace(0,1,10000)`*. Discuss the accuracy of the methods.*