COMPUTATIONAL METHODS IN MECHANICS: Task 1

Vesa-Ville Hurskainen, 19 Jan 2018

Introduction

This is a report on the first assignment of the course *Computational Methods in Mechanics*, concerning the trapezoidal integration rule. A MATLAB script with an implementation of the rule (`integral_trapezoid.m`) was provided and six tasks involving the script were given. The tasks are as follows:

1. To test that the script works correctly.
2. To use GitHub.
3. To compare the script with built-in MATLAB procedure.
4. To test the debugger.
5. To speed up the script.
6. To use the script to compute a 2D integral.

Methods

In order to complete Task 5, an optimized script (`integral_trapezoid_optimized.m`) was written, employing vectorization to speed up computation. In addition, two test scripts (`test_integral_1D.m` and `test_integral_2D.m`) were written in order to test and compare the various different types of integration functions in one- and two-dimensional cases. There, the MATLAB profiler was used to investigate computation times and the MATLAB debugger to find errors. A function was chosen to test the integration procedures in each case. The functions, along with their analytical definite integrals, are as follows:

$$\int_0^1 \sin(x)\ dx = 1 - \cos(1) \tag{1}$$

$$\int_0^1 \int_0^1 \sin(x) + \cos(y)\ dx\ dy = 1 + \sin(1) - \cos(1) \tag{2}$$

Results

All employed scripts can be found in a GitHub repository. Numerical results are presented in Table 1.

$\int_0^1 \sin(x)dx$				$\int_0^1 \int_0^1 \sin(x) + \cos(y)dx\ dy$		
Integration	Steps	Result	Avg. time	Integration	Steps	Result
Analytical		0.4596977		Analytical		1.3011687
MATLAB `trapz`	10^6	0.4596977	35 ms	MATLAB `integral2`	1000 / D	1.3011687
Original script	10^6	0.4596977	1100 ms	Original script	(defaults)	1.3011687
Optimized script	10^6	0.4596977	20 ms			

Table 1: Comparison of numerical results yielded by test scripts.

Analysis

Tasks 2 and 4 were completed during coding and the completion of Tasks 1 and 3 can be verified by comparing the results presented in Table 1. The script appears to work correctly, since it produces results that coincide with the analytical results in the one-dimensional case. However, the script is unoptimized and therefore significantly slower than than the MATLAB function `trapz`. On the other hand, the optimized script is even faster than the built-in script when using vectorized functions, which completes Task 5. Finally, as Table 1 shows, in the two-dimensional case the same result was gained from both the script and the built-in MATLAB function `integral2`, which completes Task 6. Therefore, all tasks are complete.