Lab Report: Adaptive Simpson's Rule

Introduction

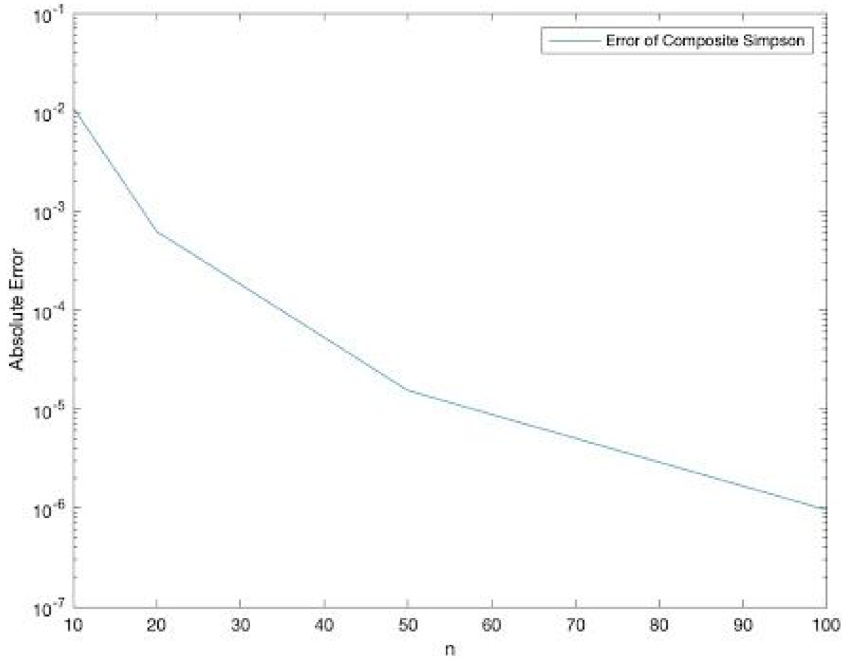
I'll begin by calculating the exact value of the integral using the built-in function of integration in Matlab. Afterwards, I will implement the composite Simpson's algorithm I wrote in the previous lab to evaluate the integral using $x_i = 1 + ih \text{ } where \text{ } i = 0, ..., n$ and $h = 1/n \text{ } for \text{ } n = 10, 20, 50, 100$. Using the results I can then compare them to the actual error and exact integral and plot the numerical quadrature as a function of n. As to perform adaptive numerical quadrature for a general definite integral using the Simpson's rule, I simply follow the algorithms provided from the textbook and test out the algorithm using the accuracy 0.5*10-4 and the maximum number of level N = 20.

Algorithm Method

Part A: To calculate the exact value of the integral, I simply set up the function $f(x) = cos(2x)e^{-x}$ with a = 0 and b = 2*pi and use the built-in function *int* to get the result of 0.1996. To calculate the composite Simpson's rule, I implement the code from Lab 6 and got the result of comp_simp = [0.1886, 0.1990, 0.1996, 0.1996]. The error estimate is error_est_simp = [0.0163, 0.0010, 0.0000, 0.0000] and the actual error is error_comp_simp = [0.0110, 0.0006, 0.0000, 0.0000]. Comparing the two errors, I notice that the exact error is roughly the same so I can conclude that the exact error is justified. The plot of the absolute error is shown below.

Part B: I used the algorithm provided in the textbook to code out the adaptive Simpson's rule. It takes the input a, b, the tolerance ε, and the limit to number of levels and produce the output the approximation of the integral APP. Running the code with the maximum number of levels N = 20 would give me the solution of 0.1996. However, running the code when N = 5 would return "Level Exceeded". This helps conclude that the highest maximum number of levels would be N = 6 in this case to reach the level of accuracy of ε = 0.5*10-4.

Graph



Conclusion

In comparison, the adaptive Simpson's rule uses less quadrature points than the non-adaptive composite Simpson's rule. This is observed from the fact that the composite Simpson's rule reaches the exact solution 0.1996 when n = 50. Meanwhile, the adaptive algorithm reaches the exact integral when the maximum level is higher than N = 6.