

Homework 4

due Nov 9, 2022

Exercise 4.1 : Parallelized trapezoidal rule, continued

- a) Implement a second version of the parallel program to solve Exercise 3.1 which uses the collective communication routines `MPI_Bcast` and `MPI_Reduce`.
- b) Do the timing and scaling studies like you did in Exercise 3.2 with your second version of the program.

(8 points)

Exercise 4.2 : Extended trapezoidal rule

As seen in Exercise 3, the trapezoidal rule to integrate a function $f(x)$ by dividing the integration interval $[a, b]$ into N sub-intervals is given by the formula

$$T_N = h \cdot \left[\frac{1}{2}f(x_0) + f(x_1) + \dots + f(x_{N-1}) + \frac{1}{2}f(x_N) \right], \quad h = \frac{b-a}{N}, \quad (1)$$

which has an error that goes as $\mathcal{O}(\frac{1}{N^2})$. An improved estimator for the integral can be obtained by combining the result obtained with the trapezoidal rule on N sub-intervals with that obtained with the same scheme on $2N$ sub-intervals in the following way

$$S_{2N} = \frac{4}{3}T_{2N} - \frac{1}{3}T_N \quad (2)$$

which approximates the integral with an error of $\mathcal{O}(\frac{1}{N^4})$.

please turn the page

Write a program which estimates the integral of Exercise 3.1 to a precision of 10^{-8} , by computing T_{2N} and S_{2N} iteratively until the desired precision is reached. Your program should

- Begin with a number $N = 2 \cdot p$ sub-intervals (i.e. every process should perform only three function evaluations).
- At every new iteration, double the number of sub-intervals $N \rightarrow 2N$ and only perform function evaluations on the new points (i.e. only perform N function evaluations per step).
- Only use collective communication.

(Hint After every iteration use T_{2N} and S_{2N} to estimate the error on the integral)

(12 points)

Exercise Bonus : Make it general!

Improve your solution to Exercise 4.2 in such a way that the function to integrate is not hardwired into the estimator, but is passed to it as an additional parameter (Hint Look up "pointer to a function" on a search engine) (2 points)

Please hand in a *printout* of your working programs together with their input and output and a plot of the time estimates.