MACM 316 – Computing Assignment 7

- Read the *Guidelines for Assignments* first.
- Submit a one-page PDF report to Canvas and upload you Matlab scripts (as m-files). Do not use any other file formats.
- Keep in mind that Canvas discussions are open forums.
- You must acknowledge any collaborations/assistance from colleagues, TAs, instructors etc.

Numerical integration of an unpleasant function

In this assignment you will be computing the integral

$$I = \int_0^1 \frac{\cos(x^{-1}\log(x))}{x}\,\mathrm{d}x. \tag{1}$$

It is known that $I \approx 0.3$. This is a challenging integral to compute, however, because the integrand $f(x) = x^{-1}\cos(x^{-1}\log(x))$ has infinitely-many oscillations in the interval $[0, 1]$ and is also singular at $x = 0$. You may quickly want to check that any standard numerical quadrature (e.g. composite Trapezoidal, Simpson's, etc) gives you poor approximations to I.

To compute this integral, you will be using a subdivision scheme based on splitting $[0, 1]$ into subintervals defined by the zeros of $f(x)$. Note that $f(x) = 0$ at the points

$$1 > a_1 > a_2 > a_3 > \ldots > 0,$$

where $a_i = \exp(-b_i)$ and b_i is the unique solution to the equation

$$b\exp(b) - (i - 1/2)\pi = 0, \quad 1 < b < \infty. \tag{2}$$

To compute I, you first need to find these roots. Recalling Part 3 of the course, this can be done in Matlab using the *fzero* command:

```
b = fzero(@(x) x*exp(x)-(i-1/2)*pi,0);
a(i) = exp(-b);
```

Given the values $a_1, a_2, \ldots, a_n$, we can now approximate I as follows:

$$I \approx Q_n = \sum_{i=0}^{n} I_i, \qquad I_i = \int_{a_{i+1}}^{a_i} f(x)\,\mathrm{d}x, \qquad a_0 = 1.$$

Write a code that computes Q_n by applying a standard numerical quadrature to each integral I_i. I recommend you use a built-in routine to evaluate each I_i, e.g. Matlab's *integral* or *quad* commands. List your results for $n = 200, 400, 600, \ldots, 2000$. You may find the `format long` command useful. How many digits of I can you accurately compute?

You will hopefully have noticed that Q_n is converging rather slowly to I. Fortunately, there's a way to get a faster converging approximation, known as Aitken's Δ^2 Method (see Burden & Faires, Sec 2.5). Given the sequence $\{Q_n\}_{n=0}^{\infty}$ we define the new sequence $\{\hat{Q}_n\}_{n=0}^{\infty}$ by

$$\hat{Q}_n = Q_n - \frac{(Q_{n+1} - Q_n)^2}{Q_{n+2} - 2Q_{n+1} + Q_n}.$$

Compute this new sequence and use it to get as good an approximation to I as you can (in reasonable computing time). Report your the values for $n = 200, 400, 600, \ldots, 2000$ as before. How many digits of I can you accurately compute using this approach? Make sure to justify the number of digits you give.

Integrals such as (1) can arise in problems in quantum physics. More generally, one might consider an integral of the form

$$I(p) = \int_0^1 \frac{\cos(x^{-p}\log(x))}{x}\,\mathrm{d}x,$$

where $p \geq 1$ is a parameter. The goal is to understand how $I(p)$ varies with p. Modify your code from the previous part of the assignment to compute this integral for a range of values of p. Make sure to adapt the root-finding step (2) suitably to take into account this new integrand. Plot $I(p)$ versus p using suitable axes and discuss their observed relationship.