1 Problem (Homework 2)

This week's assignment uses the ideas of using Taylor series to achieve factorizations suitable for use in FMM type algorithms.

Let

$$\Phi_{jk} = e^{i\alpha x_k y_j}, \quad k = 1, ..., N, \quad j = 1, ..., M.$$

$$\mathbf{\Phi} = \begin{pmatrix} \Phi_{11} & \Phi_{12} & ... & \Phi_{1N} \\ \Phi_{21} & \Phi_{22} & ... & \Phi_{2N} \\ ... & ... & ... & ... \\ \Phi_{M1} & \Phi_{M2} & ... & \Phi_{MN} \end{pmatrix}, \quad \mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ ... \\ u_N \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ ... \\ v_M \end{pmatrix}, \tag{1}$$

where $i^2 = -1$, $x_1, ..., x_N$, $y_1, ..., y_M, u_1, ..., u_N$, are random real numbers distributed uniformly in $[0, 1]$, and α is some real number. Compute the matrix-vector product

$$\mathbf{v} = \mathbf{\Phi u}, \tag{2}$$

or

$$v_j = \sum_{k=1}^{N} \Phi_{jk} u_k, \quad j = 1, ..., M, \tag{3}$$

with absolute error $\epsilon < 10^{-6}$. The matrix sizes, $N, M > 0$ are given (fixed) positive integers.

1. Write down a factored expression. Estimate the error in truncating the series using residual term evaluation for the Taylor series, and evaluate the truncation number, p,as a function of the required accuracy and N. Provide a formula that can be used for the "fast" ($O(N + M)$) method.
2. Write a program that implements both the straightforward computation based on Eq. (3) and the "fast" method.
3. Plot the absolute maximum error between the straightforward and "Fast" method for $N = 10^3$ and $M = 2N$ and p varying between 1 and 11 for $\alpha = 1$ and $\alpha = 2$. Compare the results with your evaluations of the accuracy.
4. Provide a graph that compares the CPU time required by the straightforward and the "Fast" method for N varying between 10^2 and 10^3 for the straightforward and N varying between 10^2 and 10^4 for the "Fast" method. Take $M = 2N$ and the theoretical value of the truncation number that ensures that the required accuracy is achieved for $\alpha = 1$.
5. Provide a graph of the abs. max. error (between the standard and fast methods) for N varying between 10^2 and 10^3, $M = 2N$ and the truncation numbers used for each N for $\alpha = 1$.

Hints.

1. Note that each source contributes to the error. So the truncation number p,corresponding to a required accuracy ϵ, depends on N . This relationship is an implicit function of p and you can either solve for p (write a Matlab function to do that) or determine it by developing a table of values and interpolating.
2. Use Matlab.
3. The maximum absolute error is defined as

$$error = \max_{i=1,...,N} \left| v_i^{straightforward} - v_i^{fast} \right|. \tag{4}$$

Plot the theoretical error bound on the same graph (use hint 1).

4. You may keep the truncation number constant (using the one evaluated for $N \leqslant 10^4$) or vary it with N according to the theoretical estimate for the error. In this case the function calculated in hint 1 will be helpful.

2 Solution

The kernel function $\Phi(y, x_k) = e^{i\alpha x_k y}$ is regular and so it can be expanded into the Taylor series centered at $x_* = 0.5$

$$\Phi(y, x_k) = \sum_{n=0}^{\infty} \frac{1}{n!} \left.\frac{\partial^n \Phi}{\partial y^n}\right|_{y=x_*} (y - x_*)^n . \tag{5}$$

We have

$$\left.\Phi\right|_{y=x_*} = e^{i\alpha x_k x_*}, \quad \left.\frac{\partial \Phi}{\partial y}\right|_{y=x_*} = i\alpha x_k e^{i\alpha x_k x_*}, ..., \quad \left.\frac{\partial^n \Phi}{\partial y^n}\right|_{y=x_*} = (i\alpha x_k)^n e^{i\alpha x_k x_*}. \tag{6}$$

Hence, the required factorization is

$$\Phi(y, x_k) = e^{i\alpha x_k x_*} \sum_{n=0}^{\infty} \frac{(i\alpha x_k)^n}{n!} (y - x_*)^n . \tag{7}$$

If we truncate the series leaving only p first terms:

$$\Phi(y, x_k) = e^{i\alpha x_k x_*} \sum_{n=0}^{p-1} \frac{(i\alpha x_k)^n}{n!} (y - x_*)^n + \epsilon_p(x_k), \tag{8}$$

Then the residual is bounded according to the Cauchy's bound:

$$|\epsilon_p(x_k)| \leqslant \frac{|y - x_*|^p}{p!} \sup_{0<y<1} \left| \frac{\partial^n \Phi}{\partial y^n}(y, x_k) \right| \leqslant \frac{1}{2^p p!} \max |i^p \alpha^p x_k^p| \left| e^{i\alpha x_k x_*} \right| \leqslant \frac{\alpha^p}{2^p p!}. \tag{9}$$

Now consider the sum (3). We have

$$\begin{aligned} v_j &= \sum_{k=1}^{N} \Phi_{jk} u_k = \sum_{k=1}^{N} \left(e^{i\alpha x_k x_*} \sum_{n=0}^{p-1} \frac{(i\alpha x_k)^n}{n!} (y_j - x_*)^n + \epsilon_p(x_k) \right) u_k \\ &= \sum_{n=0}^{p-1} (y_j - x_*)^n \frac{(i\alpha)^n}{n!} \sum_{k=1}^{N} x_k^n e^{i\alpha x_k x_*} u_k + err_p, \quad err_p = \sum_{k=1}^{N} \epsilon_p(x_k) u_k. \end{aligned} \tag{10}$$

The last term can be bounded using Eq. (9) and the fact that $|u_k| \leqslant 1$:

$$|err_p| = \left| \sum_{k=1}^{N} \epsilon_p(x_k) u_k \right| \leqslant \sum_{k=1}^{N} |\epsilon_p(x_k)| |u_k| \leqslant \frac{\alpha^p}{2^p p!} \sum_{k=1}^{N} |u_k| = \frac{N\alpha^p}{2^p p!}. \tag{11}$$

This formula is sufficient to accomplish Task 3 of the assignment, as it explicitly expresses the error bound as a function of N, α, and p. To achieve the absolute error $< 10^{-6}$ for $N = 10^4$ and $\alpha = 1$ we need to solve inequality

$$\frac{10^4}{2^p p!} < 10^{-6}, \tag{12}$$

or

$$2^p p! > 10^{10}. \tag{13}$$

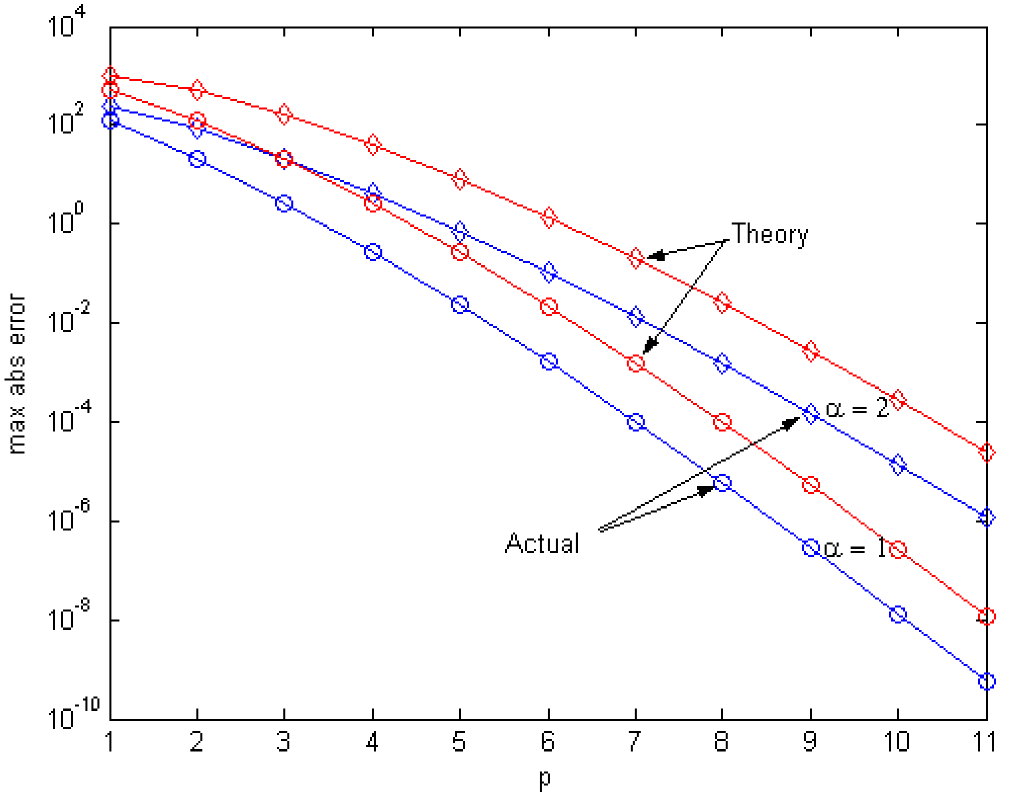

Figure 1:

We can find then that

$$\begin{aligned} 2^{10}10! &< 4\cdot 10^9 < 10^{10}, \\ 2^{11}11! &> 8\cdot 10^{10} > 10^{10}. \end{aligned} \tag{14}$$

Therefore $p = 11$ is the minimal truncation number which provides guarantee that the the max absolute error will be less than 10^{-6} for any point distribution.

To compute the sum using the fast method, according to Eq. (10) we first compute p coefficients

$$c_n = \frac{(i\alpha)^n}{n!}\sum_{k=1}^{N} x_k^n e^{i\alpha x_k x_*} u_k, \quad n = 0, ..., p-1, \tag{15}$$

and then M sums

$$v_j = \sum_{n=0}^{p-1} c_n \left(y_j - x_*\right)^n, \quad j = 1, ..., M. \tag{16}$$

Results of computations required in Tasks 3-5 are presented on the figures below.

Task 3: Fig 1 shows the theoretical error bound (11) (red) and the actual error (blue) for $\alpha = 1$ (the circles) and $a = 2$ (the diamonds) ($p = 1, ..., 11$; $N = 1000$, $M = 2000$). It is seen that the actual error is substantially smaller than the theoretical error bound.

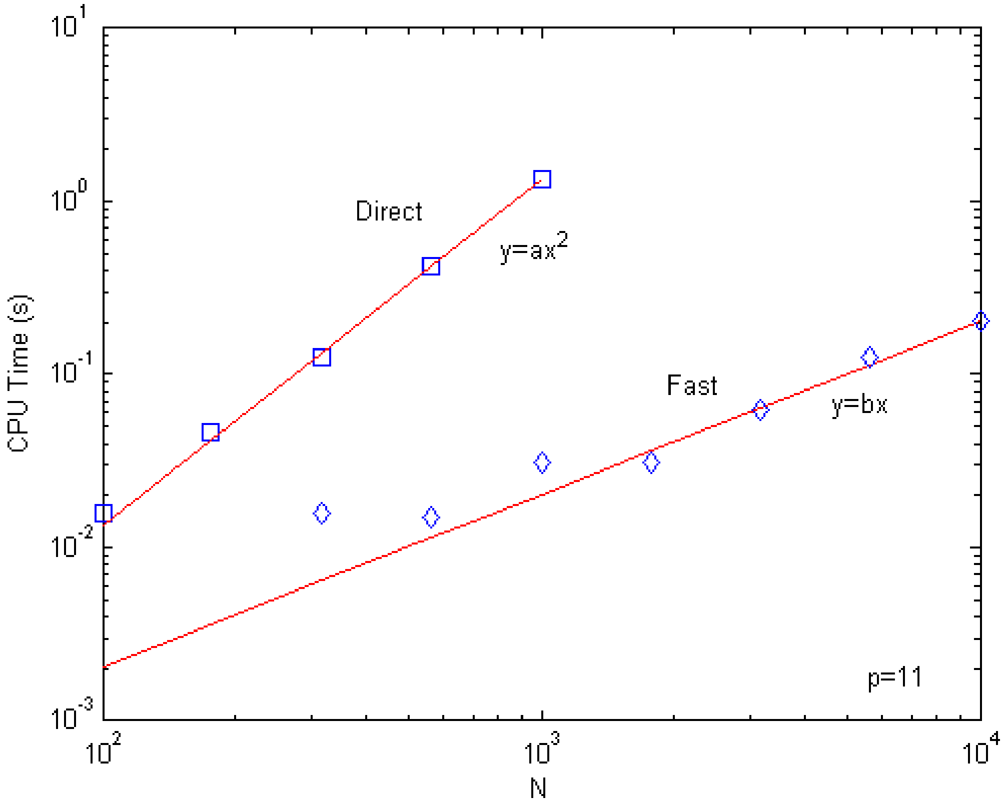
10^1
10^0
10^-1
10^-2
10^-3
CPU Time (s)
Direct
y=ax^2
Fast
y=bx
p=11
10^2
10^3
10^4
N

Figure 2:

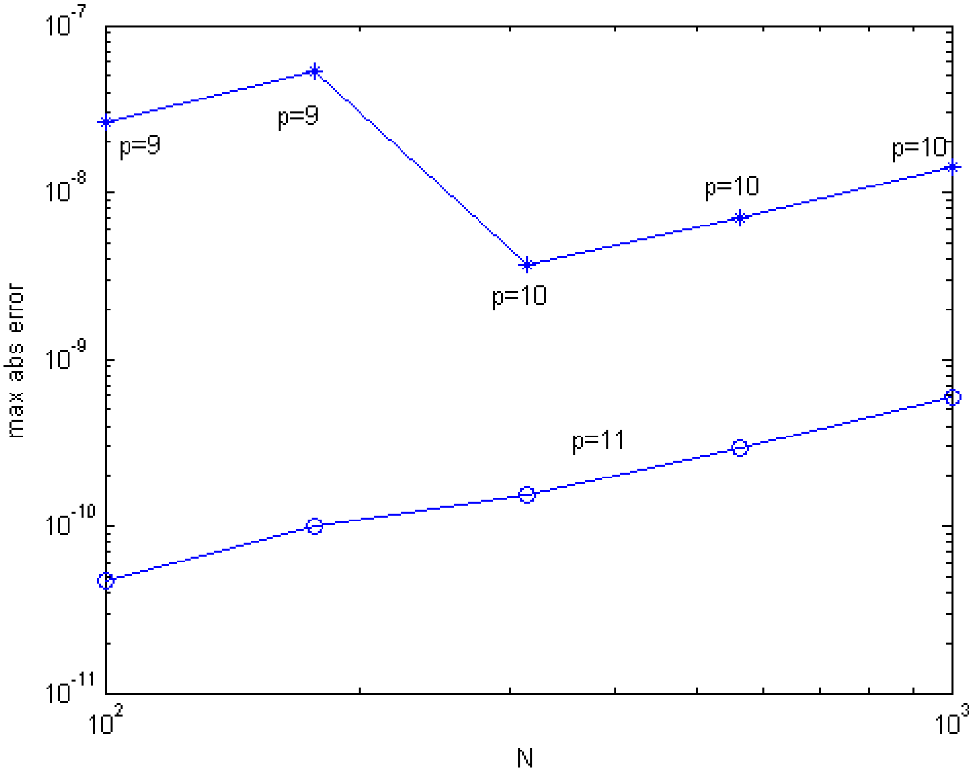

Figure 3:

Task 4: Fig 2 shows the CPU time in seconds required for the matrix-vector multiplication for the direct method (the squares) and the fast (the diamonds) method. For the fast method the truncation number is fixed, $p = 11$, which guarantees max abs errors below 10^{-6} for $\alpha = 1$ ($M = 2N$). The red lines show linear and quadratic dependences in the log-log coordinates.

Task 5: Fig 3 shows the actual maximum absolute error as a function of N. For the line marked by stars the truncation number p was varying with N according to the theoretical error bound (11) ($\alpha = 1$). Here the truncation number is shown near each point. The line marked by the cicles corresponds to computations with constant $p = 11$, which guarantees the error below 10^{-6} for any $N \leqslant 10000$.