EEE424 Homework 4

(Clearly justify all answers.)

(Due 30 April 2023)

In this homework, you will convolve a large array with a finite impulse response using different computational algorithms.

A 120-point ($N_h = 120$) long impulse response is given as $h[n] = 0.96^n$, where $n = 0, 1, \cdots, 119$. Store a long array $x[n]$ whose size is about 120k points (a few seconds of music or speech); state the exact length N_x, and assume that $x[n]$ is an array for $n \in [0, N_x - 1]$.

a) (MATLAB) Convolve $h[n]$ and $x[n]$ using an algorithm which directly implements the convolution, $y_1[n] = \sum_{k=0}^{N_h-1} h[k]x[n-k]$, for $n = 0, 1, \cdots, N_x+N_h-1$. Do not use MATLAB commands for convolution. Store the result in an array. Find the number of real multiplications to find the result; this will be the measure of your algorithm complexity.

b) (MATLAB) Split the array into segments whose length is N_s points; concatenate $x[n]$ with zeros to increase N_x to an integer multiple of N_s. Write an algoritm which uses direct computation of 2048-pt DFT to implement an overlap-and-add procedure to compute $y_2[n] = h[n] * x[n]$. Do not use MATLAB commands for DFT; write your own code based on the definition of DFT. Use the largest possible N_s. Store the result in an array. Again, find the number of needed real multiplications as the measure of complexity of your algorithm.

c) (MATLAB) Clearly understand which algorithm is used to implement the MATLAB commands related to FFT computation; clearly indicate the algorithm, by giving references if needed, so the reader of this report clearly knows the used algorithm. Repeat item (b) by replacing the direct DFT computation by FFT; call the output $y_3[n]$.

d) (MATLAB) Compare the three outputs $y_1[n]$, $y_2[n]$, $y_3[n]$, by computing a difference measure $\epsilon_{i,j} = \frac{1}{(N_x+N_h-1)} \sum_{n=0}^{N_x+N_h-1} (y_i[n] - y_j[n])^2$, for $i, j = 1, 2, 3$, $i \neq j$ (i.e., you will do it three times). What is the source of error? Also compute $e_{i,j} = \max |y_i[n] - y_j[n]|$ in the range $n = 0, (N_x + N_h - 1)$ for $i, j = 1, 2, 3$, $i \neq j$. ((Note: Your error must be very small; if there is a large error, your algorithms may be wrong.)

e) (Both analytic part and MATLAB) Now you will investigate the computational error for each output $y_1[n]$, $y_2[n]$, $y_3[n]$. Replace the input used in items (a), (b) and (c) by $x[n] = \cos(0.05\pi n)$, $n = 0, 1, \cdots, N_x - 1$. Repeat items (a), (b) and (c) for this input. Analytically find the steady state output for this filter when it is passed through an LTI system whose impulse response is the given $h[n]$; call the analytical output $y_4[n]$. Delete the transient parts of your stored arrays; therefore, your output arrays will become shorter; call the new size N_1. For the steady state parts of the outputs, compare the computational error for each $y_i[n]$, by computing $\epsilon_{i,j} = \frac{1}{N_1} \sum_n (y_4[n] - y_i[n])^2$, $i = 1, 2, 3$, where the summation is over those n that represent the steady state. Also compute $e_{i,j} = \max |y_4[n] - y_j[n]|$

for the same range of n. (Note: Your error must be very small; if there is a large error, either your algorithms, or your analytical answer, may be wrong.) What is the source of error? Compare the error in three convolution algorithms you have implemented.

f) Make a plot of about size 512 (to fit your page) of a segment of your input and one of the corresponding outputs; chose the segment not close the starting or ending range of n (i.e., do not chose $n < 1000 \leq$ or $n > 110000$.)