CIS 515

Fundamentals of Linear Algebra and Optimization
Jean Gallier

Project 2A: Compression of audio signals using Haar wavelets

The purpose of this project is to investigate properties and applications of Haar wavelets. In particular, methods for compressing audio signals are investigated.

During the lectures, we explained how a vector vector $u = (u_1, \ldots, u_m)$ corresponds to a piecewise linear function over the interval $[0, 1)$. The function $\mathrm{plf}(u)$ is defined such that

$$\mathrm{plf}(u)(x) = u_i, \qquad \frac{i-1}{m} \leq x < \frac{i}{m},\ 1 \leq i \leq m.$$

In words, the function $\mathrm{plf}(u)$ has the value u_1 on the interval $[0, 1/m)$, the value u_2 on $[1/m, 2/m)$, etc., and the value u_m on the interval $[(m-1)/m, 1)$.

(Part 1) (**5 points**) Write a `Matlab` program called `drawplf` that takes as input a vector u and plots the corresponding function $\mathrm{plf}(u)$. Test your program on several inputs including

$$u = \begin{bmatrix} 0 & 2 & 4 & 6 & 6 & 4 & 2 & 1 & -1 & -2 & -4 & -6 & -6 & -4 & -2 & 0 \end{bmatrix}$$

and the vector w obtained by concatenating u with itself 8 times, obtained by

$$u2 = [u\ \ u]; \quad u3 = [u2\ \ u2]; \quad w = u4 = [u3\ \ u3].$$

Note that u contains 16 integers, and the last one is 0, not -20 (spaces are important). These are constructed in the output script and saved in the output folder. For an example of how it should look like, on the vector $\begin{bmatrix} 2 & 5 & -1 & 2 & -3 & 4 \end{bmatrix}$ - it should look like Figure 1, though the exact colors, line thickness, corners, etc. are left to your own discression - you have free reign of those.

(Part 2) (**25 points**) Write two `Matlab` functions `haar` and `haar_inv` implementing the method for computing the Haar transform of a vector and the reconstruction of a vector from its Haar coefficients, as described in the notes.

Test your programs on many inputs, including

$$u = \begin{bmatrix} 0 & 2 & 4 & 6 & 6 & 4 & 2 & 1 & -1 & -2 & -4 & -6 & -6 & -4 & -2 & 0 \end{bmatrix}$$

and the string w from Part 1. What do you observe?

(Part 3) (**25 points**) Write a `Matlab` function `haar_step` that performs only k rounds of averaging and differencing on some input vector u; the function `haar_step` takes as input u and k.

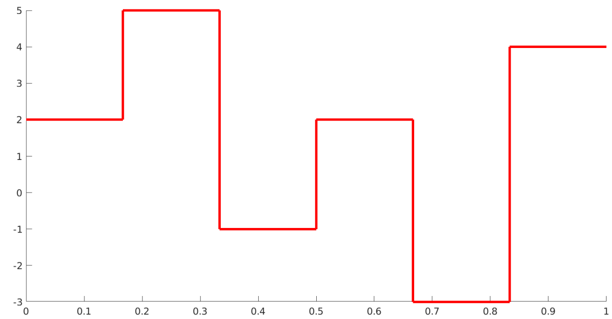

Figure 1: Example plf plot

The output script will test your program on the vector w obtained by concatenating u with itself 8 eight times, along with other vectors. If you plot it, you will notice something remarkable happens starting with $k = 4$. In the report, explain the behavior that you observe for $k = 4, 5, 6, 7$.

(Part 4) (**25 points**) Write a function `haar_inv_step` inverting the function `haar_step`. The function `haar_inv_step` takes as input a vector v and the number of rounds k.

To check that this function is correct, first apply `haar_step` and then `haar_inv_step` for the same number of steps k. You should get back the orginal vector. Note that while this is done in the output script and gives a warning but you should check yourself beforehand.

(Part 5) (**20 points**) This step is all done in the report. Load the audio file `handel` using `load handel`. This file is saved in the variable `y`. Keep the first 65536 elements of this vector by doing `handel = y(1:65536);`. To play and hear the music, do `sound(handel)`. Run `haar_step` on the vector `handel` for $k = 1$. Then play the result. What happens. Can you explain it? Do this again for $k = 2, 3$. What do you observe? Try to extrapolate any patterns.

Run `haar` on the vector `handel` to get the Haar transform c. Set the detail coefficients to zero by doing `c1 = c; c1(32768:end) = 0;`. Then apply `haar_inv` to $c1$ to get `handel1`. Play `handel1`. What difference to you observe compared to playing `handel`? Try plotting the various sound vectors help your observations. Experiment with other compressions of c and write them up in the report.