Discrete sines, cosines, and complex exponentials

Alejandro Ribeiro

January 18, 2021

Sines, cosines, and complex exponentials play an important role in signal and information processing. The purpose of this lab is to gain some experience and intuition on what these signals look like and how they behave. The signals we consider here are discrete and finite because they are indexed by an integer time index $n = 0, 1, \ldots, N-1$. The constant N is referred to as the length of the signal. Start by considering a separate integer number k to define the *discrete complex exponential* $e_{kN}(n)$ of discrete frequency k and duration N as

$$e_{kN}(n) = \frac{1}{\sqrt{N}}\, e^{j2\pi kn/N} = \frac{1}{\sqrt{N}}\, \exp(j2\pi kn/N). \tag{1}$$

The (regular) complex exponential is defined as $e^{j2\pi kn/N} = \cos(2\pi kn/N) + j\sin(2\pi kn/N)$ so that if we compute the real and imaginary parts of $e_{kN}(n)$ we have that

$$\begin{aligned} \mathrm{Re}\left(e_{kN}(n)\right) &= \frac{1}{\sqrt{N}}\, \cos(2\pi kn/N), \\ \mathrm{Im}\left(e_{kN}(n)\right) &= \frac{1}{\sqrt{N}}\, \sin(2\pi kn/N). \end{aligned} \tag{2}$$

We say that the real part of the complex exponential is a discrete cosine of discrete frequency k and duration N and that the imaginary part is a discrete sine of discrete frequency k and duration N. The discrete frequency k in (2) determines the number of oscillations that we see in the N elements of the signal. A sine, cosine, or complex exponential of discrete frequency k has a total of k complete oscillations in the N samples.

Mathematically speaking, the complex exponential, the sine, and the cosine are all different signals. Intuitively speaking, all of them are oscillations of the same frequency. Since complex exponentials have imaginary parts, they don't exist in the real world. Nevertheless, we work with them instead of sines and cosines because they are easier to handle. Rules of exponential functions are easier than the corresponding rules of trigonometric functions.

1 Signal generation

Let us begin by generating and displaying some complex exponentials and by using the generated signals to explore some of their important properties.

1.1 Generate complex exponentials. Create a Python class to represent a complex exponential of discrete frequency k and signal duration N. The attributes of this class include a vector with N components containing the elements of the signal $e_{kN}(n)$ defined in (1), as well as its real and imaginary parts [cf. (2)]. Plot the real and imaginary components for $N = 32$ and different values of k. Observe that some of these signals don't look much like oscillations. In your report, show the plots for $k = 0$, $k = 2$, $k = 9$, and $k = 16$.

1.2 Equivalent complex exponentials. Use the class in Part 1.1 to generate complex exponentials of the same duration and frequencies, k and l, that are N apart. E.g., make $N = 32$ and plot signals for frequencies $k = 3$ and $l = 3 + 32 = 35$ and $l = 3 - 32 = -29$. You should observe that these signals are identical.

1.3 Conjugate complex exponentials. Use the class in Part 1.1 to generate complex exponentials of the same duration and opposite frequencies k and $-k$. E.g., make $N = 32$ and plot signals for frequencies $k = 3$ and $k = -3$. You should observe that these signals have the same real part and opposite imaginary parts. We say that the signals are conjugates of each other.

1.4 More conjugate complex exponentials. Consider now frequencies k and l in the interval $[0, N-1]$ such that their sum is $k + l = N$. To think about this relationship, order the frequencies from $k = 0$ to $k = N$ and start walking up the chain from $k = 0$, to $k = 1$, to $k = 2$, and so on. Likewise, start walking down the chain from $l = N$, to $l = N - 1$, to $l = N - 2$ and so on. When you have taken the same number of steps in either direction you have that $k + l = N$. Given your observations in parts 1.2 and 1.3 you should expect these signals to be conjugates of each other. Verify your expectation with, e.g., $k = 3$ and $l = 32 - 3 = 29$.

We consider now the energy of complex exponentials and the inner products between complex exponentials of different frequencies. Given two signals x and y of duration N, their inner product is defined as

$$\langle x, y \rangle := \sum_{n=0}^{N-1} x(n) y^*(n). \tag{3}$$

The energy of a signal is defined as the inner product of the signal with itself $\|x\|^2 := \langle x, x \rangle$. We can write the signals x and y as column vectors $x = [x(0), \ldots, x(N-1)]^T$ and $y = [x(0), \ldots, x(N-1)]^T$. Then, by defining the hermitian of a matrix or vector as the complex conjugate and transpose, i.e. $x^H = (x^*)^T$, the inner product is simply written as the product $y^H x$ and the energy as the product $x^H x$. We say that a signal is normal if it has unit energy, i.e., if $\|x\|^2 = 1$. We say that two signals are orthogonal if their inner product is null, i.e., if $\langle x, y \rangle = 0$. Orthogonality looks like an innocent property, but it is nothing like that. It is one of the most important properties that a group of signals can have.

1.5 Orthonormality. Write a function to compute the inner product $\langle e_{kN}, e_{lN} \rangle$ between all pairs of discrete complex exponentials of length N and frequencies $k, l = 0, 1, \ldots, N-1$. Run and report your result for $N = 16$. You should observe that the complex exponentials have unit energy and are orthogonal to each other. When this happens, we say that the signals form an orthonormal set.

2 Analysis

The numerical experiments of Part 1 demonstrated two properties that discrete complex exponentials have that are very important for subsequent analyses. In this section we study these properties analytically. We first work on the observation that when we consider frequencies k and l that are N apart, the complex exponentials may have formulas that look different but are actually equivalent (part 1.2).

2.1 Equivalent complex exponentials. Consider two complex exponentials $e_{kN}(n)$ and $e_{lN}(n)$ as given by the definition in (1). Prove mathematically that if $|k-l| = N$ the signals are equivalent, i.e., that $e_{kN}(n) = e_{lN}(n)$ for all times n.

2.2 More equivalent complex exponentials. Use the result in part 1 to show that the same is true not only when $|k-l| = N$ but also whenever $k - l \in \dot{N}$, that is, whenever $|k-l|$ is a multiple of N.

The second fundamental property that we want to explore is that whenever we have two complex exponentials that are not equivalent their inner product is null,

$$\langle e_{kN}, e_{lN} \rangle := \sum_{n=0}^{N-1} e_{kN}(n) e_{lN}^{*}(n) = 0 \tag{4}$$

We observed that this was true in Part 1.5 for some particular examples. We will now prove that it is true in general.

2.3 Orthogonality. Consider two complex exponentials $e_{kN}(n)$ and $e_{lN}(n)$ that are *not* equivalent, i.e., for which the difference $|k-l| \notin \dot{N}$ is not a multiple of N. Prove that the signals are orthogonal to each other.

2.4 Orthonormality. Prove that complex exponentials have unit norm $\|e_{kN}(n)\|^2 = \langle e_{kN}(n), e_{kN}(n) \rangle = 1$. The combination of this fact with the orthogonality proven in Part 2.3 means that a set of N consecutive complex exponentials form an orthonormal set. Explain this statement.

The statements that we derived above are for a specific sort of discrete complex exponential. We can write more generic versions if we do not restrict the discrete frequency k to be discrete or if we shift the argument of the complex exponential. When performing these operations it is interesting to ask if the equivalence properties of parts 2.1 and 2.2 and the orthogonality properties of parts 2.3 and 2.4 hold true.

2.5 Phase shifts. Let $\phi \in \mathbb{R}$ be an arbitrary given number that we call a phase shift. We define a shifted complex exponential by subtracting the shift from the exponent in (1)

$$e_{kN}^{\phi}(n) = \frac{1}{\sqrt{N}}\, e^{j(2\pi k \frac{n}{N} - \phi)} = \frac{1}{\sqrt{N}}\, \exp\left[j \left(2\pi k \frac{n}{N} - \phi \right) \right]. \tag{5}$$

The reason why subtracting ϕ from (5) is called a shift, is because the frequency of the oscillation doesn't change. It's just that the oscillation gets shifted to the right. In this problem we consider discrete frequencies $k \neq l$ and a same shift ϕ. Is there a condition to make complex exponentials $e^{\phi}_{kN}(n)$ and $e^{\phi}_{lN}(n)$ of frequencies k and l equivalent? Is there a condition that guarantees that the complex exponentials $e^{\phi}_{kN}(n)$ and $e^{\phi}_{lN}(n)$ are orthogonal?

2.6 Fractional frequencies. Lift the assumption that k in (1) is integer and consider arbitrary frequencies $k, l \in \mathbb{R}$. Is there a condition to make complex exponentials $e_{kN}(n)$ and $e_{lN}(n)$ of frequencies k and l equivalent? Is there a condition that guarantees that the complex exponentials $e_{kN}(n)$ and $e_{lN}(n)$ are orthogonal?

3 Generating and playing musical tones

Up until now we have considered discrete signals as standalone entities. However, discrete signals are most often used as representations of a continuous signal that exists in the palpable—as opposed to virtual—world. To connect discrete signals to the physical world we define the sampling time T_s as the time elapsed between times n and $n+1$. Two ancillary definitions that follow from this one are the definition of the sampling frequency $f_s = 1/T_s$ and the definition of the signal duration $T = NT_s$.

To move from discrete to actual frequencies, say that we are given a discrete cosine of frequency k and duration N with an associated sampling time of T_s seconds. We want to determine the frequency f_0 of that cosine. To do so, recall that a discrete cosine of frequency k has a total of k oscillations in the N samples, which is the same as saying that it has a total of k oscillations in $T = NT_s$ seconds. The period of the cosine is therefore N/k samples, which, as before, is the same as saying that it has a period of $T/k = NT_s/k$ seconds. The frequency of the cosine is the inverse of its period,

$$f_0 = \frac{k}{T} = \frac{k}{NT_s} = \frac{k}{N} f_s. \tag{6}$$

Conversely, if we are given a cosine of frequency f_0 Hertz that we want to observe with a sampling frequency f_s for a total of $T = NT_s = N/f_s$ seconds, it follows that the corresponding discrete cosine has discrete frequency

$$k = N \frac{f_0}{f_s}. \tag{7}$$

In explicit terms, we can use the definition in (2) with the discrete frequency in (7) to write the discrete cosine as

$$x(n) \;=\; \cos\Big[2\pi kn/N\Big] \;=\; \cos\Big[2\pi[(f_0/f_s)N]n/N\Big] \tag{8}$$

Simplifying the signal durations N in (8) and recalling that $T_s = 1/f_s$, the cosine $x(n)$ can be rewritten as

$$x(n) \;=\; \cos\Big[2\pi(f_0/f_s)n\Big] \;=\; \cos\Big[2\pi f_0(nT_s)\Big] \tag{9}$$

The last expression in (9) is intuitive. It's saying that the continuous time cosine $x(t) = \cos(2\pi f_0 t)$ is being sampled every T_s seconds during a time interval of length $T = NT_s$ seconds.

3.1 Discrete cosine generation. Write down a function that takes as input the sampling frequency f_s, the time duration T, and the frequency f_0 and returns the associated discrete cosine $x(n)$ as generated by (9). Your function has to also return the number of samples N. When T is not a multiple of $T_s = 1/f_s$ you can reduce T to the largest multiple of T_s smaller than T.

3.2 Generate an A note. The musical A note corresponds to an oscillation at frequency $f_0 = 440$ Hertz. Use the code of part 3.1 to generate an A note of duration $T = 2$ seconds sampled at a frequency $f_s = 44,100$ Hertz. Play the note in your computer's speakers.

3.3 Generate musical notes. A piano has 88 keys that can generate 88 different musical notes. The frequencies of these 88 different musical notes can be generated according to the formula

$$f_i = 2^{(i-49)/12} * 440 \quad , \; i = 1, \ldots, 88. \tag{10}$$

Modify the code of Part 3.1 so that instead of taking the frequency f_i as an argument receives the piano key number and generates the corresponding musical tone.

3.4 Generate musical notes. To play a song, you just need to play different notes in order. Use the code in Part 3.3 to play a song that has at least as many notes as *Happy Birthday.*

4 Time management

The formulation of the problems in Part 1 is lengthy, but their solutions are straightforward. The goal is to finish them up during the Tuesday lab session. Try to get a head start in solving the problems. You may not succeed, but thinking about them will streamline the Tuesday session. This should require just 1 more hour besides the lab.

The problems in Part 2 will take more time to complete. You should wait until after class on Wednesday to solve them. We will do parts 2.1, 2.2, 2.3, and 2.4 in class. I am asking that you re-solve them yourself to make sure that you understood them. They are very important properties needed to understand Fourier transforms. To solve parts 2.5 and 2.6 you have to work on your own, but the solutions are simple generalizations of earlier parts. You should be able to wrap this up in 3 hours, about 30 minutes for each of the questions.

Part 3 is the one that will take more time because you have to apply your creativity and problem solving skills. If you are familiar with tones, beats, and know how to read music, this should take about 6 hours to complete. If you don't, part of being an engineer is being able to do something you don't know how to do. It'll take you a couple more hours to learn how to read *Happy birthday.*

5 Report presentation

Please remember to label both the x-axis and y-axis of all your figures Remember also to add a legend and/or a title to the plots you're including in your figures (check commands A graph without units and labeled axes makes no sense. The titles help us with grading.

Please include your code along with the lab report.

Lab reports must be named. Write down the names of all members of the group in the first page along with the group number.