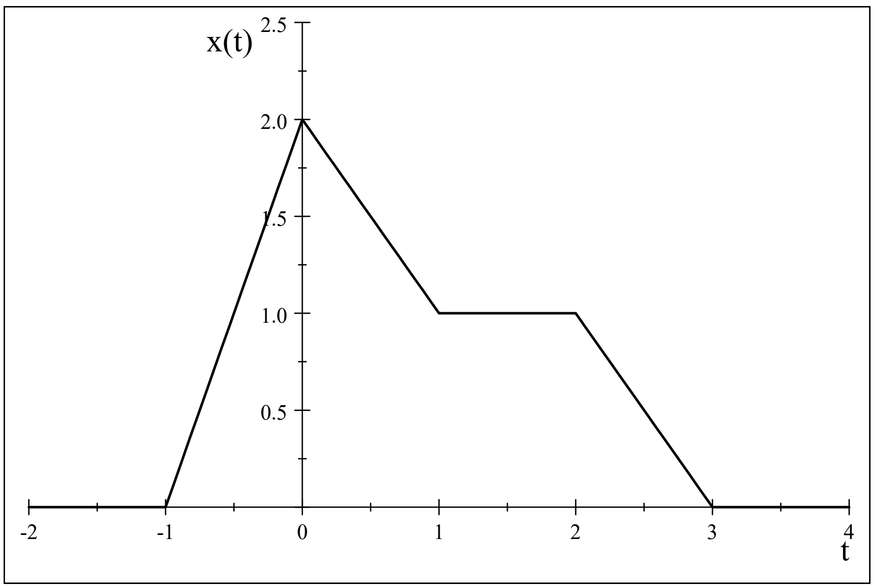
Worksheet 2 - Building Block Signals and Signal Characteristics

Building Block Signals

It is relatively straightforward to understand how a system operates on simple signals, but it is not nearly as intuitive with complicated ones. So one of the more important things that we will want to be able to do is to decompose a complicated signal into smaller, simpler, pieces that we can then analyze separately.

Consider the following signal sketched below:



This can be decomposed in many different ways, all equally valid. In Matlab, run `wav_demo1` and manipulate the GUI so as to reconstruct $x(t)$ as is shown.

Periodicity of Sums of Sinusoids

1. Consider the signal

$$x(t) = \sin(3\pi t) + \sin(5\pi t)$$

   (a) (analytic) Find the fundamental frequency and period of $x(t)$.

   (b) (Matlab) Plot $x(t)$ with Matlab, in such a way that you can see/identify at least one cycle of $x(t)$ to confirm your answer to (a)

(c) Now consider the following signals that are only slightly different than $x(t)$:

$$\begin{aligned} x_1(t) &= 5\sin(3\pi t) + \sin(5\pi t) \\ x_2(t) &= 5\sin(3\pi t) + \sin(5\pi t + \pi/3) \end{aligned}$$

Do you think these new signals will have different fundamental frequencies than $x(t)$? Check via Matlab and see.

2. (Matlab) Run the program `sin_demo2`. This GUI allows you to add up to five sinusoids together, with adjustable frequencies, and actively displays how the fundamental frequency and periodicity changes as you manipulate those sinusoids. Explore various combinations and see if you can find any pattern to be able to predict the fundamental period of the sum of sinusoids.

Power and Energy

Pulse Train

Consider the periodic signal described by

$$\begin{aligned} x(t) &= \begin{cases} A & 0 < t < dT_0 \\ 0 & dT_0 < t < T_0 \end{cases} \\ x(t + kT_0) &= x(t) \text{ for all integers } k \end{aligned}$$

Note that A, d, and T_0 are parameters that are, at this point, allowed to take on any value.

Note: the parameter d only makes sense if we allow it to take on values in the range $0 < d < 1$. It is also referred to as the ***duty cycle****, as it determines for what fraction of one period the pulse is actually "on". This is used in e.g. radar and sonar systems, where there is a short duty cycle, allowing for a short period where a pulse is transmitted, and then a long period of time during which the system listens for a reflection.*

1. (analytic) Sketch this signal.

2. (analytic) Determine the *average power* of this signal in terms of the three parameters, and determine/describe how it depends upon each of the three parameters. What is the average power if $A = 3$, $d = 0.3$, $T_0 = 2.5$?

3. (Matlab) Run the program `tavg_demo`, investigate how things change with the GUI sliders, and confirm your result from above.

Sinusoid

1. (analytic) Determine the *average power* in the generic sinuoid described by

$$x(t) = A \sin(2\pi f_0 t + \theta)$$

How does the average power vary with frequency and phase shift?

Exponential Signal

Consider the signal

$$x(t) = Ae^{-\alpha t}u(t)$$

where $u(t)$ is the unit step function.

1. (analytic) Determine the normalized *energy* of this signal. Does it depend upon α?
2. (analytic) Assume $\alpha > 0$, determine the normalized *power* of this signal.
3. (analytic) Assume $\alpha = 0$, determine both the normalized *energy* and *power* of this signal.

Based on these observations, we can infer that the concept of energy is useful for some signals, but not for others; the same can be said of power. We often classify signals in practice into two categories:

- *Energy signals are those that have finite energy and zero power*
- *Power signals are those that have finite power and infinite energy*

Even And Odd Signals

Consider the signal

$$x(t) = \begin{cases} 0 & t < -3 \\ t+2 & -3 < t < 3 \\ 0 & t > 3 \end{cases}$$

1. (analytic/matlab) Determine the even and odd components of $x(t)$. Plot each of these in Matlab and confirm that they display even/odd symmetry.
2. (analytic/matlab) Determine the even and odd components of $x(t-3)$, and plot them in Matlab to confirm.

You're all done!!