

Actividad 1:

Tema: "Muestreo y Cuantificación en Octave".

Objetivos:

- El alumno entienda el muestreo en Octave
- El alumno entienda la cuantificación en Octave

Nota: El nombre del archivo a enviar debe tener el siguiente formato:

PrimerNombre_ApellidoPaterno_GX.pdf donde X=numero de su grupo

Cada alumno enviará su actividad grupal e indicara que alumno(s) de su grupo no ayudo o colaboro en realización de dicha actividad.

Alumno(s) que no ayudo: _____

Pasos a seguir:

Observar el siguiente video para el entendimiento de muestreo y cuantificación:


Video1:

<https://www.youtube.com/watch?v=9GxcNyGQsuk>

Investigar sobre las funciones de Octave en grupo

Funciones en Octave

Funcion Triplus() –Señal Triangular


Octave Forge

Home
Packages
Developers
Support/Help
Documentation

Function File: $y = \text{tripuls}(t)$

Function File: $y = \text{tripuls}(t, w)$

Function File: $y = \text{tripuls}(t, w, \text{skew})$

Generate a triangular pulse over the interval $[-w/2, w/2]$, sampled at times t . This is useful with the function `pulstran` for generating a series of pulses.

`skew` is a value between -1 and 1, indicating the relative placement of the peak within the width. -1 indicates that the peak should be at $-w/2$, and 1 indicates that the peak should be at $w/2$. The default value is 0.

Example:

```
fs = 11025; # arbitrary sample rate
f0 = 100; # pulse train sample rate
w = 0.3/f0; # pulse width 3/10th the distance between pulses
plot (pulstran (0:1/fs:4/f0, 0:1/f0:4/f0, "tripuls", w));
```

See also: `gauspuls`, `pulstran`, `rectpuls`.

Demonstration 1


The following code

```
fs = 11025; # arbitrary sample rate
f0 = 100; # pulse train sample rate
w = 0.5/f0; # pulse width 1/10th the distance between pulses
x = pulstran (0:1/fs:4/f0, 0:1/f0:4/f0, "tripuls", w);
plot ([0:length(x)-1]*1000/fs, x);
xlabel ("Time (ms)");
ylabel ("Amplitude");
title ("Triangular pulse train of 5 ms pulses at 10 ms intervals");
```

Produces the following figure

Referencia: <https://octave.sourceforge.io/signal/function/tripuls.html>

Funcion Pulstran()


Octave Forge

Home
Packages
Developers
Support/Help
Documentation

Function File: $y = \text{pulstran}(t, d, \text{func}, \dots)$

Function File: $y = \text{pulstran}(t, d, p)$

Function File: $y = \text{pulstran}(t, d, p, Fs)$

Function File: $y = \text{pulstran}(t, d, p, Fs, \text{method})$

Generate the signal $y = \sum(\text{func}(t+d, \dots))$ for each d . If d is a matrix of two columns, the first column is the delay d and the second column is the amplitude a , and $y = \sum(a * \text{func}(t+d))$ for each d, a . Clearly, `func` must be a function which accepts a vector of times. Any extra arguments needed for the function must be tagged on the end.

Example:

```
fs = 11025; # arbitrary sample rate
f0 = 100; # pulse train sample rate
w = 0.001; # pulse width of 1 millisecond
subplot (pulstran (0:1/fs:0.1, 0:1/f0:0.1, "rectpuls", w), fs);
```

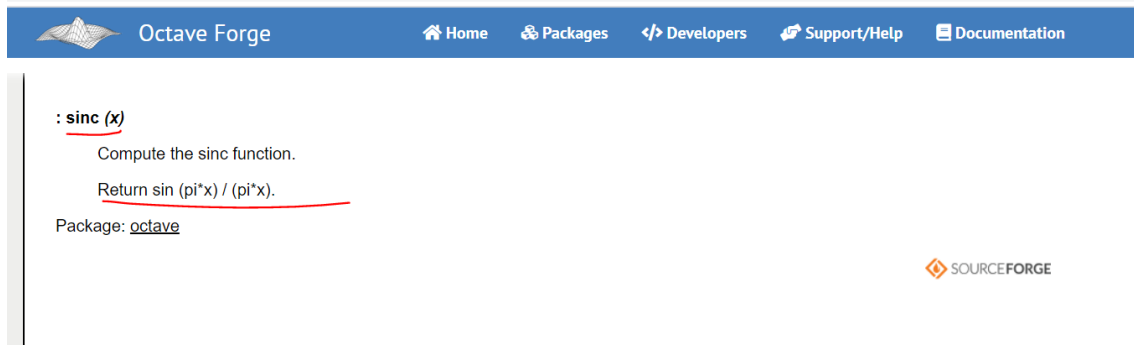
If instead of a function name you supply a pulse shape sampled at frequency F_s (default 1 Hz), an interpolated version of the pulse is added at each delay d . The interpolation stays within the time range of the delayed pulse. The interpolation method defaults to linear, but it can be any interpolation method accepted by the function `interp1`.

Example:

```
fs = 11025; # arbitrary sample rate
f0 = 100; # pulse train sample rate
w = boxcar(10); # pulse width of 1 millisecond at 10 kHz
subplot (pulstran (0:1/fs:0.1, 0:1/f0:0.1, w, 10000), fs);
```

Referencia: <https://octave.sourceforge.io/signal/function/pulstran.html>

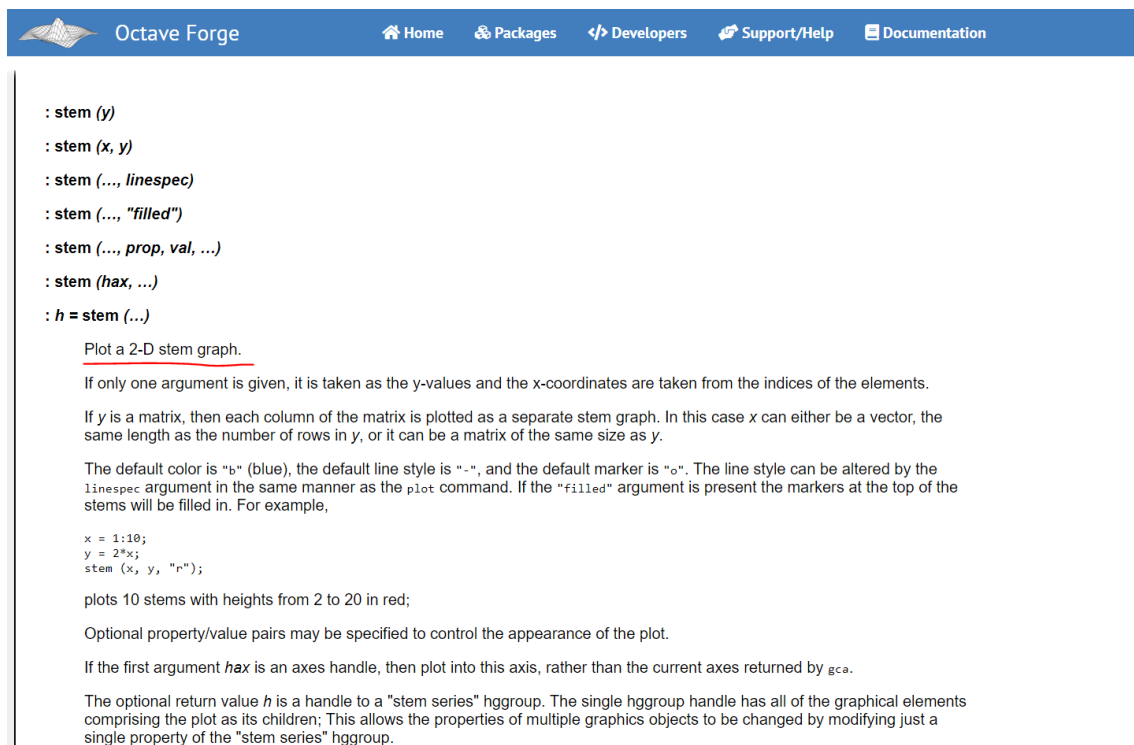
Funcion Sampling (Sinc)



The screenshot shows the Octave Forge website header with navigation links: Home, Packages, Developers, Support/Help, and Documentation. Below the header, the documentation for the `sinc` function is displayed. The function signature is `: sinc (x)`. The description states: "Compute the sinc function. Return $\sin(\pi x) / (\pi x)$." The package is listed as `octave`. The SourceForge logo is visible in the bottom right corner.

Referencia: <https://octave.sourceforge.io/octave/function/sinc.html>

Funcion Stem() –Plot Discrete Sequence



The screenshot shows the Octave Forge website header with navigation links: Home, Packages, Developers, Support/Help, and Documentation. Below the header, the documentation for the `stem` function is displayed. The function signature is `: stem (y)`. Other signatures include `: stem (x, y)`, `: stem (... , linespec)`, `: stem (... , "filled")`, `: stem (... , prop, val, ...)`, `: stem (hax, ...)`, and `: h = stem (...)`. The description states: "Plot a 2-D stem graph. If only one argument is given, it is taken as the y-values and the x-coordinates are taken from the indices of the elements. If y is a matrix, then each column of the matrix is plotted as a separate stem graph. In this case x can either be a vector, the same length as the number of rows in y, or it can be a matrix of the same size as y. The default color is 'b' (blue), the default line style is '--', and the default marker is 'o'. The line style can be altered by the `linespec` argument in the same manner as the `plot` command. If the 'filled' argument is present the markers at the top of the stems will be filled in. For example, `x = 1:10; y = 2*x; stem (x, y, "r");` plots 10 stems with heights from 2 to 20 in red; Optional property/value pairs may be specified to control the appearance of the plot. If the first argument `hax` is an axes handle, then plot into this axis, rather than the current axes returned by `gca`. The optional return value `h` is a handle to a 'stem series' hgroup. The single hgroup handle has all of the graphical elements comprising the plot as its children; This allows the properties of multiple graphics objects to be changed by modifying just a single property of the 'stem series' hgroup.

Referencia: <https://octave.sourceforge.io/octave/function/stem.html>

Preguntas:

1-Defina que es el Muestreo y la Cuantificación. En que aplicaciones se usan

2-Defina que es el teorema de Nyquist y jitter

3- Dada una señal sinusoidal $y(t)$ de frecuencia 2Hz , de amplitud de G Voltios y de frecuencia de Muestreo $f_s=40\text{Hz}$. El intervalo del tiempo es desde 0 a 2.

G=Numero de su Grupo

Graficar la señal $y(t)$ continuo en el tiempo

Comandos

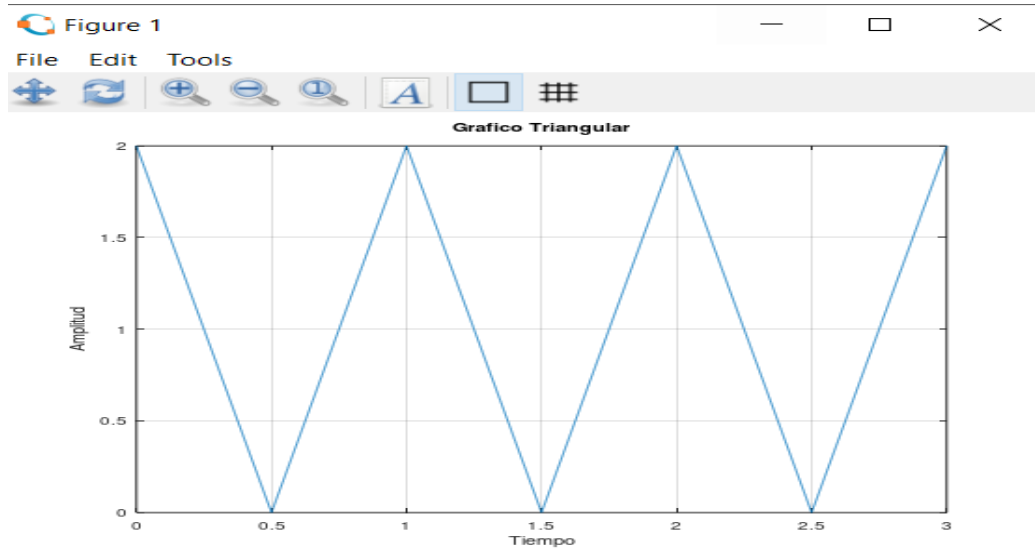
Pantallazo de graficas

Graficar la cuantificación de la señal $y(t)$ discreta en el tiempo

Comandos

Pantallazo de graficas

4-Dada la señal triangular $y(t)$ donde la frecuencia de Muestreo $f_s=20\text{Hz}$. La amplitud de esta señal es de G Voltios. $G=\text{Numero de su Grupo}$

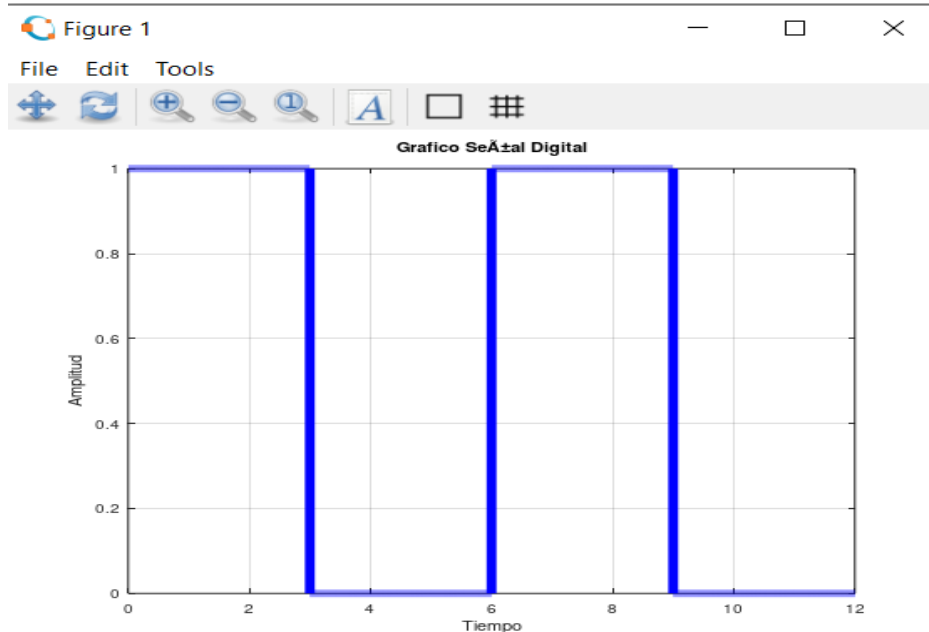


Graficar la cuantificación de la señal $y(t)$ discreta en el tiempo

Comandos

Pantallazo de graficas

5- Graficar la señal periódica digital $y(t)$) donde la frecuencia de Muestreo $f_s=20\text{Hz}$



Graficar la cuantificación de la señal $y(t)$ discreta en el tiempo

Comandos

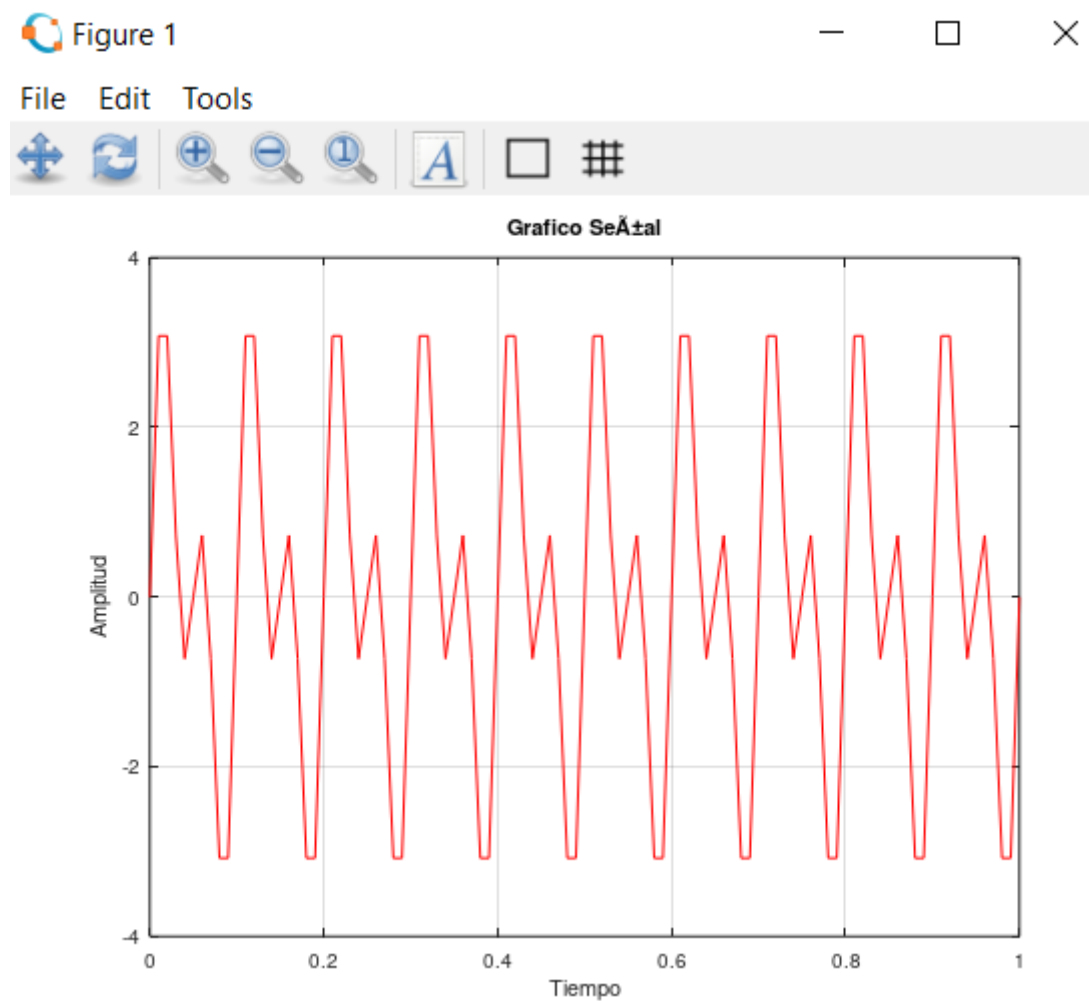
Pantallazo de graficas

6-Se tiene la señal $y(t)$ que es la sumatoria de 2 señales sinusoidales que son las siguientes:

$X(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t)$; donde la $A=G$ y la $f=10\text{Hz}$ $G=\text{Numero de su Grupo}$

$Z(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t)$; donde la $A=G$ y la $f=20\text{Hz}$ $G=\text{Numero de su Grupo}$

Dando el siguiente grafico



Además, la frecuencia de Muestreo $f_s=100\text{Hz}$

Graficar la cuantificación de la señal $y(t)$ discreta en el tiempo

Comandos

Pantallazo de graficas

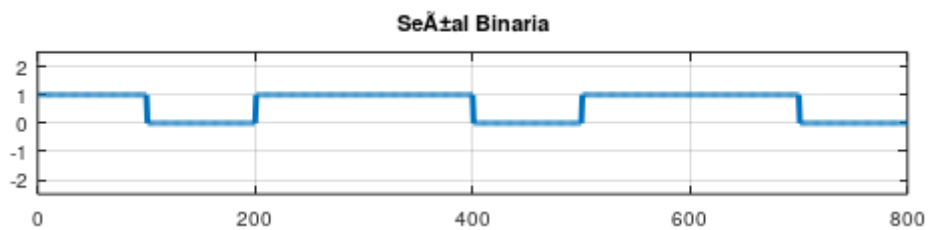
7-Graficar la señal sampling para un intervalo del tiempo de $-2 \cdot G$ a $2 \cdot G$ segundos donde:

G =Numero de su Grupo

Comandos

Pantallazo de graficas

8-Se tiene la siguiente señal Banda Base "10110110"



Si el ancho del pulso de cada bit es 0.2 milisegundos.

Calcular la Velocidad binaria y el Ancho de banda de esta señal

Comandos