

# Laboratorio 1: Eco, Interpolación y Decimación

## Señales y Sistemas

Diego Cardoza Zúñiga  
Benjamin Rodríguez Araya

**Abstract**—Este informe muestra las distintas variaciones que tiene una grabación de un arpegio de guitarra cuando se le aplican algoritmos de Eco, Interpolación y Decimación.

### I. INTRODUCCIÓN

Diariamente escuchamos sonidos, melodías y canciones que no están en su estado inicial, es decir, están procesadas una vez grabadas. Estas señales de audio que transmiten los dispositivos en la actualidad, han sido transformadas por un algoritmo de procesamiento digital que modificó la señal original por una nueva.

En este informe se muestran tres algoritmos de transformación de una señal (Eco, Interpolación y Decimación) en un arpegio de guitarra previamente grabado. El primer problema que se enfrentó fue modificar la señal de este arpegio para agregar un efecto muy común que es el Eco. Esto se logra copiando la misma señal original una cierta cantidad de veces, agregándole un pequeño retardo a las nuevas ondas. El segundo problema se pide expandir la señal, de modo que esta se propague por más tiempo que la original. Para lograr lo anterior, se agregan valores entre cada muestra de manera periódica. El tercer y final problema, es el proceso inverso al segundo, es decir, eliminar muestras de manera periódica en vez de agregar valores entre cada muestra, logrando una compresión de la señal.

### II. ALGORITMOS

Para la solución de cada uno de los problemas propuestos anteriormente, utilizamos el software científico MATLAB, junto con su respectivo lenguaje de programación. Por lo tanto trabajamos en cada caso con:

$$\vec{y} = (y_1, y_2, y_3, \dots, y_n) \forall y_n \in \mathbb{R}$$

$$sr \in [0, \infty^+]$$

Donde  $\vec{y}$  es el vector que contiene todas las muestras de la señal de audio entrante, y  $sr$  es la frecuencia de muestreo o Sample Rate. Debemos considerar también que en cada caso, la cantidad de tiempo en segundos de muestra de señal está dado por:

$$t = sr \cdot \text{length}(\vec{y})$$

Donde  $\text{length}(\vec{y})$  es el largo del vector de muestras.

Escuela de Informática y Telecomunicaciones  
e-mail: diego.cardoza@mail.udp.cl.  
e-mail: benjamin.rodriguez@mail.udp.cl.

#### A. Eco

El efecto conocido como Eco, consiste en sobreponer la señal original junto a una cantidad  $n$  de copias, de la misma señal, donde cada copia está  $t$  segundos retrasada con respecto a la original. Luego, para lograr el delay, se debe añadir la cantidad de ceros necesaria para lograr el retardo  $t$  deseado. Esta cantidad de ceros está dada por:

$$\text{ceros} = sr \cdot t$$

Donde  $sr$  es la frecuencia de muestreo y  $t$  es la cantidad de segundos deseados de delay. Además, logramos que el valor de cada delay se atenúe con el tiempo, utilizando un vector  $\vec{f}$  que tiene la forma.

$$\vec{f}_i = \vec{f}_{i-1} - \frac{1}{n} \mid \vec{f}_0 = 1 \wedge \vec{f}_n = 0$$

Con  $n$  igual al largo del vector original.

Finalmente, por cada valor de los vectores con delay, se multiplican con su respectiva posición en el vector  $\vec{f}$ , y luego se suma al vector original para poder así generar el efecto de Eco.

#### B. Interpolación

Para esta técnica de manipulación de señales muestreadas, debemos agregar valores  $x$  entre cada muestra con el fin de expandir la duración de la señal. Al hacer esto, el largo de la señal final sera:

$$\text{largo}_{\text{final}} = \text{length}(\vec{y}) \cdot n$$

Donde  $n$  corresponde a la cantidad de valores que se va a colocar entre cada muestra. Así, creamos un vector auxiliar con el largo del vector final, el cual llenaremos iterando el vector original y copiando el valor de la muestra cada  $n$  iteraciones del vector final. Según la variación de este método es el valor que tomara cada valor  $x$ .

- $x$  toma el valor de 0.
- $x$  corresponde al valor anterior del vector original.

#### C. Decimación

La compresión se lleva a cabo, simplemente eliminando  $n$  cantidad de valores del vector de muestreo. Para ello utilizaremos nuevamente un vector auxiliar que tendrá un largo de:

$$\text{largo}_{\text{final}} = \text{length}(\vec{y}) - n$$

Con  $n$  igual a la cantidad de muestras a eliminar. Tal vector lo llenaremos iterando en el vector original y copiando al vector final solamente cuando el número del iterador es divisible por  $n$ . Así, eliminamos muestras de manera periódica y uniforme.

### III. RESULTADOS

#### A. Eco

Para generar el eco en la señal, se usaron 3 ondas con los siguientes retardos:

$$T = [0.0001, 0.0002, 0.0003]$$

Obteniendo el siguiente gráfico (ver Fig. 1), en el cual se pueden apreciar las distintas señales del arpegio, tales como, la original, las 3 señales de eco, y finalmente la resultante.

#### B. Interpolación

Para el caso de la interpolación, se pedían dos actividades distintas. En la primera actividad se pedía agregar ceros a la señal; En la segunda actividad se pedía copiar el valor de la muestra anterior en una nueva muestra contigua a la anterior.

**Actividad 1:** Se utilizaron dos parámetros de entradas (1 y 10), los cuales representan la cantidad de ceros que se agregarán entre cada muestra.

Lo que se obtuvo de esas entradas fueron los siguientes gráficos (ver Fig. 2 y Fig. 3) para las entrada 1 y 10, respectivamente. En ellos se puede apreciar la señal original y la señal resultante al agregar ceros.

**Actividad 2:** Al igual que la primera actividad, se utilizaron dos parámetros de entradas (1 y 10), los cuales representan la cantidad de copias que se agregarán entre cada muestra.

Lo que se obtuvo de esas entradas fueron los siguientes gráficos (ver Fig. 4 y Fig. 5) para las entrada 1 y 10, respectivamente. En ellos se puede apreciar la señal original y la señal resultante al agregar ceros.

#### C. Decimación

Finalmente, para comprimir la señal se utilizó como parámetro la cantidad de muestras que se deseaban eliminar. El valor que se le dio al parámetro fue de 5000, es decir, se eliminarán 5000 muestras equidistantes una de otra. Dado ese valor, se obtuvo el siguiente gráfico (ver Fig. 6), en el cual se pueden apreciar dos señales, la original y la señal comprimida.

### IV. ANÁLISIS DE RESULTADOS

#### A. Eco

Podemos apreciar sin mayor problema en la Figura. 1 el desfase que tiene cada eco con respecto a la señal original y como al principio se notan el silencio generado por los valores en 0 agregados al inicio de cada vector. Una cosa que debemos tener en cuenta es que, si aplicamos muchos ecos

que se solapen uno sobre otro, la suma de todos estos puede provocar una saturación en la señal final, y por consiguiente una mala calidad en el sonido.

Este es uno de los efectos mas comunes en la música, utilizado mucho en los pedales para instrumentos, al igual que como efecto en las mesas de audio para micrófonos, entre otras aplicaciones.

#### B. Interpolación

Dependiendo de la variación del método que queramos aplicar de la interpolación, el efecto en el gráfico (ver Fig. 2 y Fig. 3), al igual que en audio son muy notorios. Aunque de igual manera se aprecia el cambio con respecto al original, consideramos que la variación de copias de los valores anteriores tiene menos perdida de calidad, obviamente debido a que no perdemos información como copiando los ceros.

Los beneficios de utilizar la interpolación son muy abarcados mayormente en el área musical como moduladores de voz, o de algún instrumento.

#### C. Decimación

Podemos notar en la Figura 6 que al aplicarle la técnica de Decimación, la señal es comprimida en el tiempo, por ende dura menos que la señal original. Es el efecto contrario al de la interpolación. Por lo tanto, puede ser usada para que una señal vaya mas rápido o para reducir su tamaño. Debemos tener en cuenta que en el caso que solo queramos reducir su tamaño, para poder escuchar la señal con la velocidad de reproducción original, debemos aplicar una simple regla de 3 para calcular el nuevo valor de frecuencia de muestreo.

$$sr_{final} = \frac{sr_{original} \cdot length(\vec{f})}{length(\vec{y})}$$

Donde  $\vec{f}$  es el vector de la muestra ya comprimida e  $y$  es el vector original.

Esta técnica es muy útil ya que a veces, dependiendo del tipo de señal que estamos analizando o de la frecuencia de muestreo, cuando elegimos una cantidad de compresión precisa, los cambios son casi imperceptibles para el oído humano y obtenemos muchos beneficios en cuanto al tamaño del archivo.

### V. CONCLUSIÓN

Estas y muchas otras técnicas para manipular señales, en este caso de audio, han sido muy útiles a lo largo de nuestra vida cotidiana.

Además, al realizar el trabajo en MATLAB, nos pudimos dar cuenta de la forma matemática de ver las señales y de comprender qué pasaba cuando se modificaban en los distintos algoritmos.

### VI. BIBLIOGRAFÍA

Link a repositorio de GitHub del laboratorio.

### VII. ANEXO

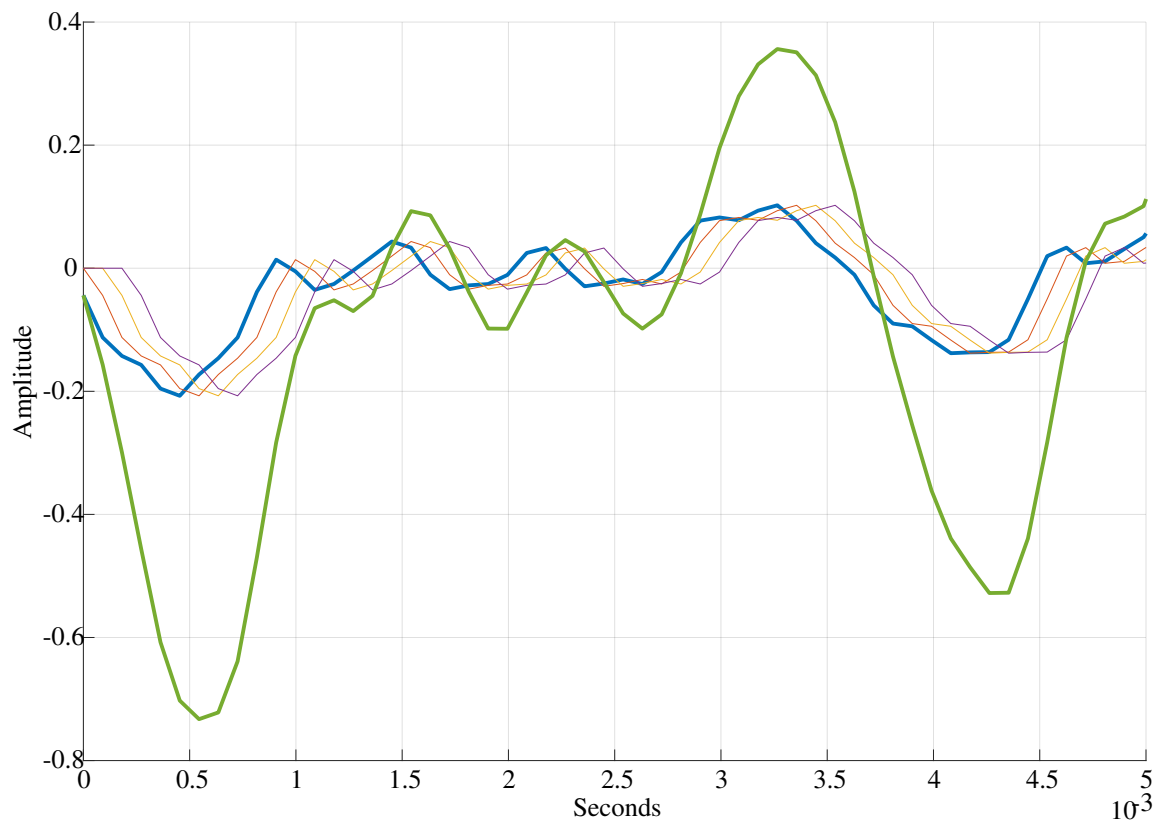


Fig. 1. En esta figura se presentan las distintas señales que están presentes en el eco. De color azul está la señal original; de color verde, la señal completamente procesada; de color rojo, amarillo y morado, las señales de eco individuales.

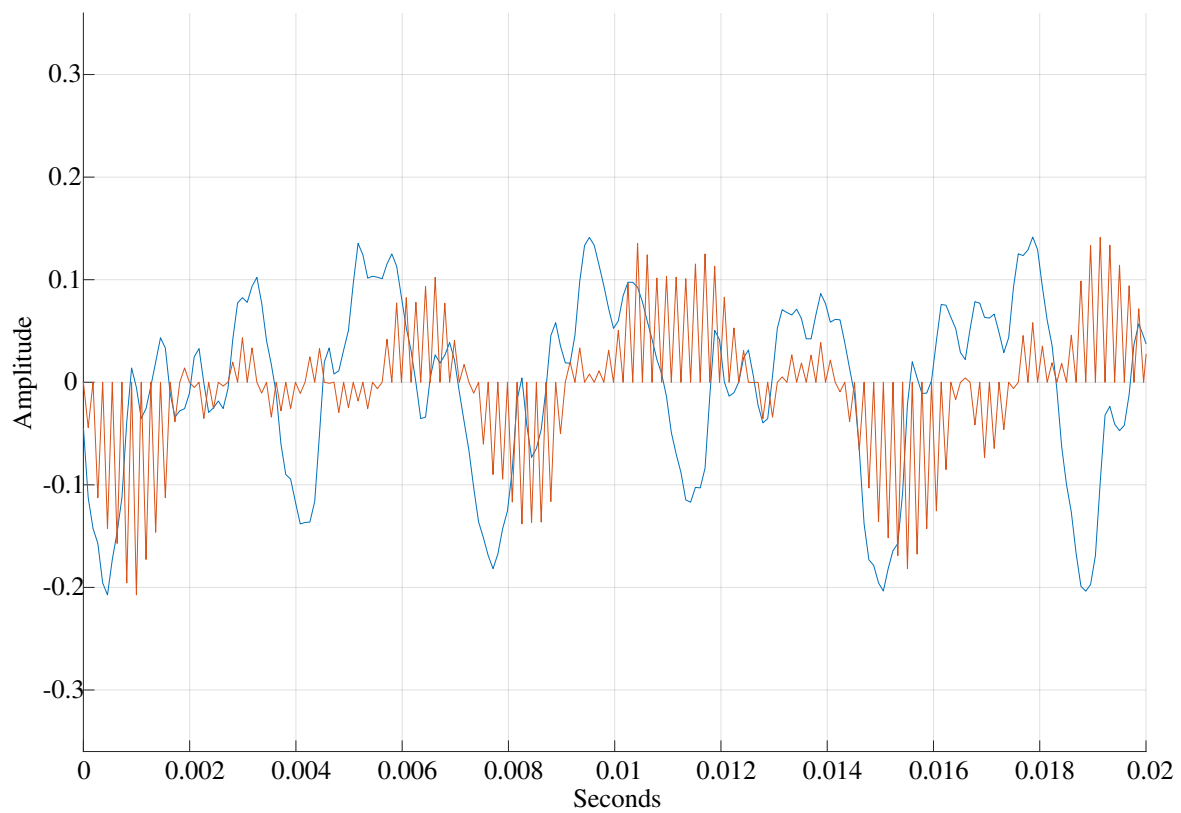


Fig. 2. En esta figura se puede ver en color azul la onda original, y en color rojo la onda procesada después de agregar 1 cero después de cada muestra.

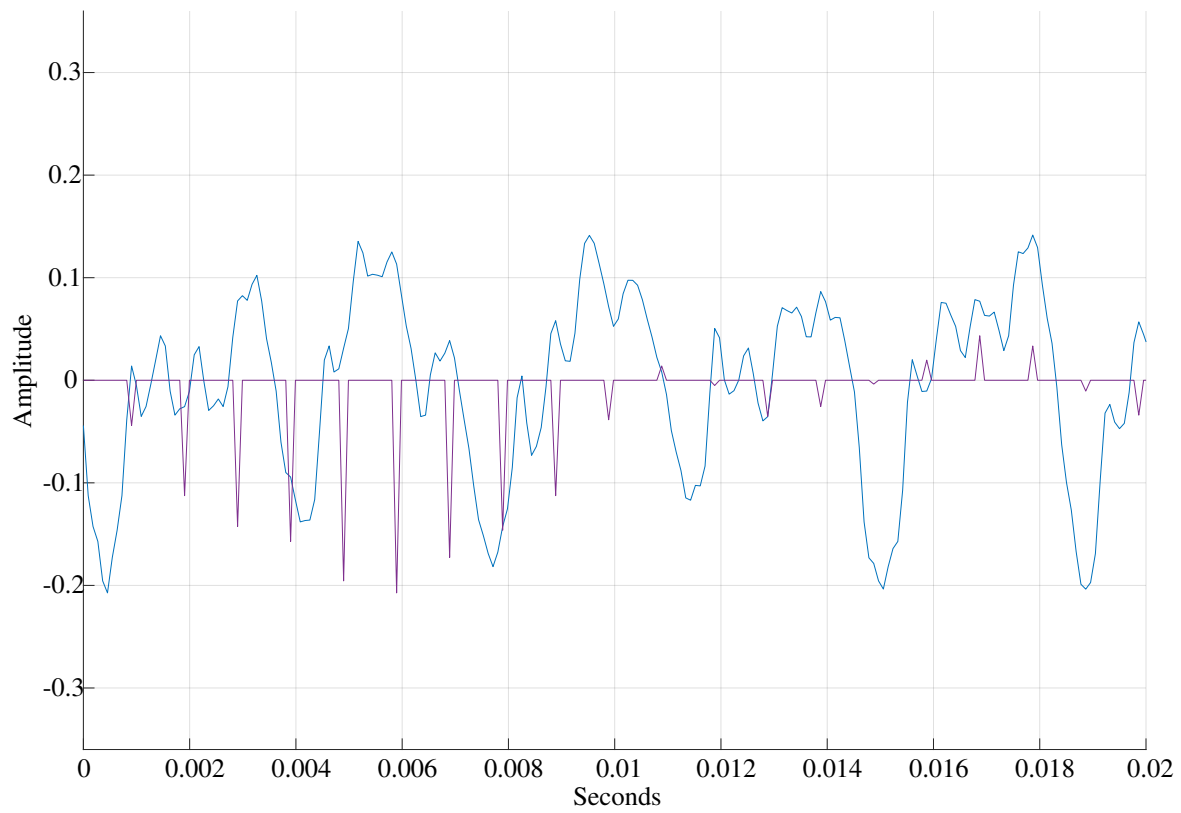


Fig. 3. En esta figura se puede ver en color azul la onda original, y en color morado la onda procesada después de agregar 10 ceros después de cada muestra.

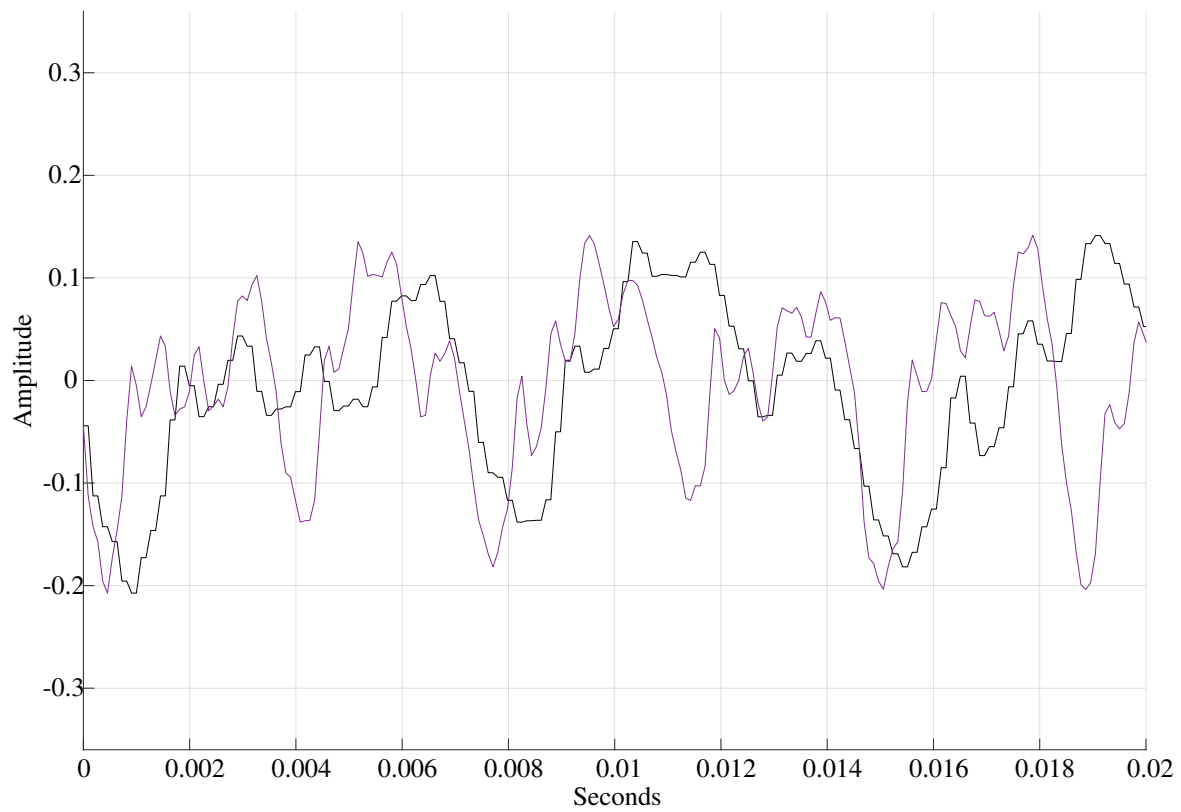


Fig. 4. En esta figura se puede ver en color morado la onda original, y en color negro la onda procesada después de agregar una copia después de cada muestra.

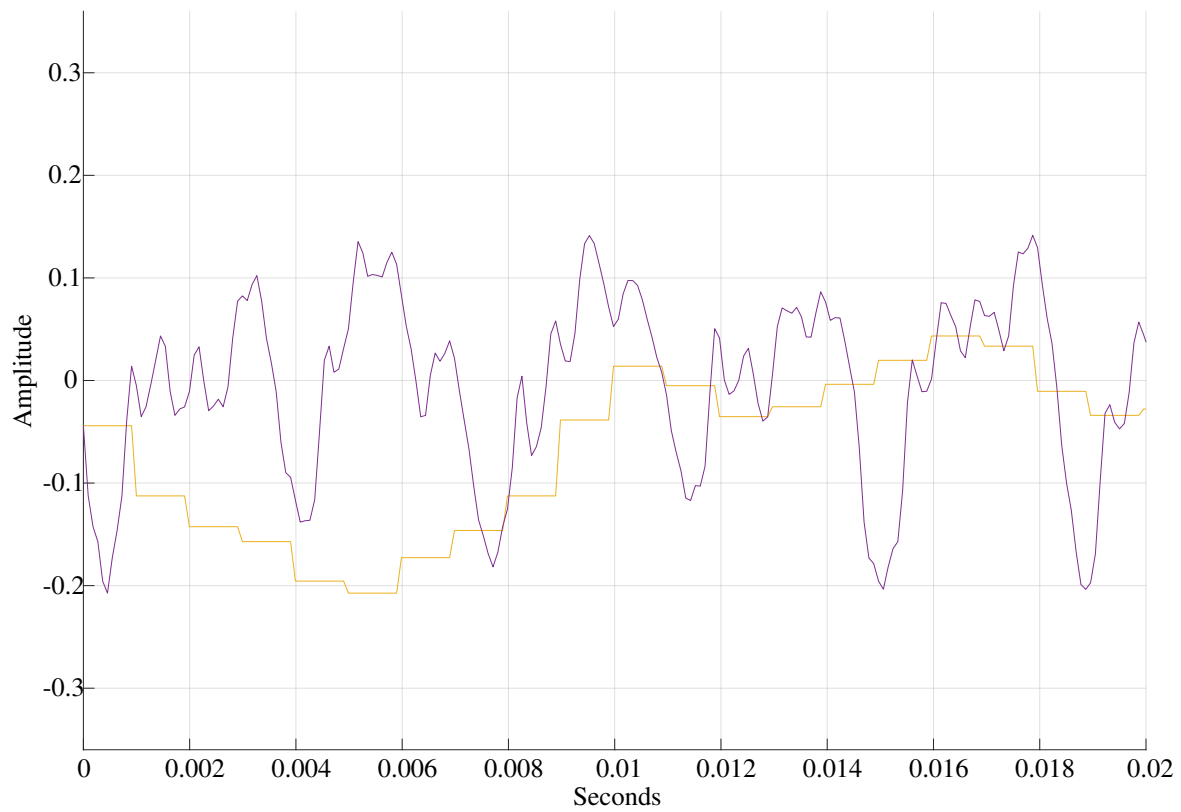


Fig. 5. En esta figura se puede ver en color morado la onda original, y en color amarillo la onda procesada después de agregar 10 copias después de cada muestra.

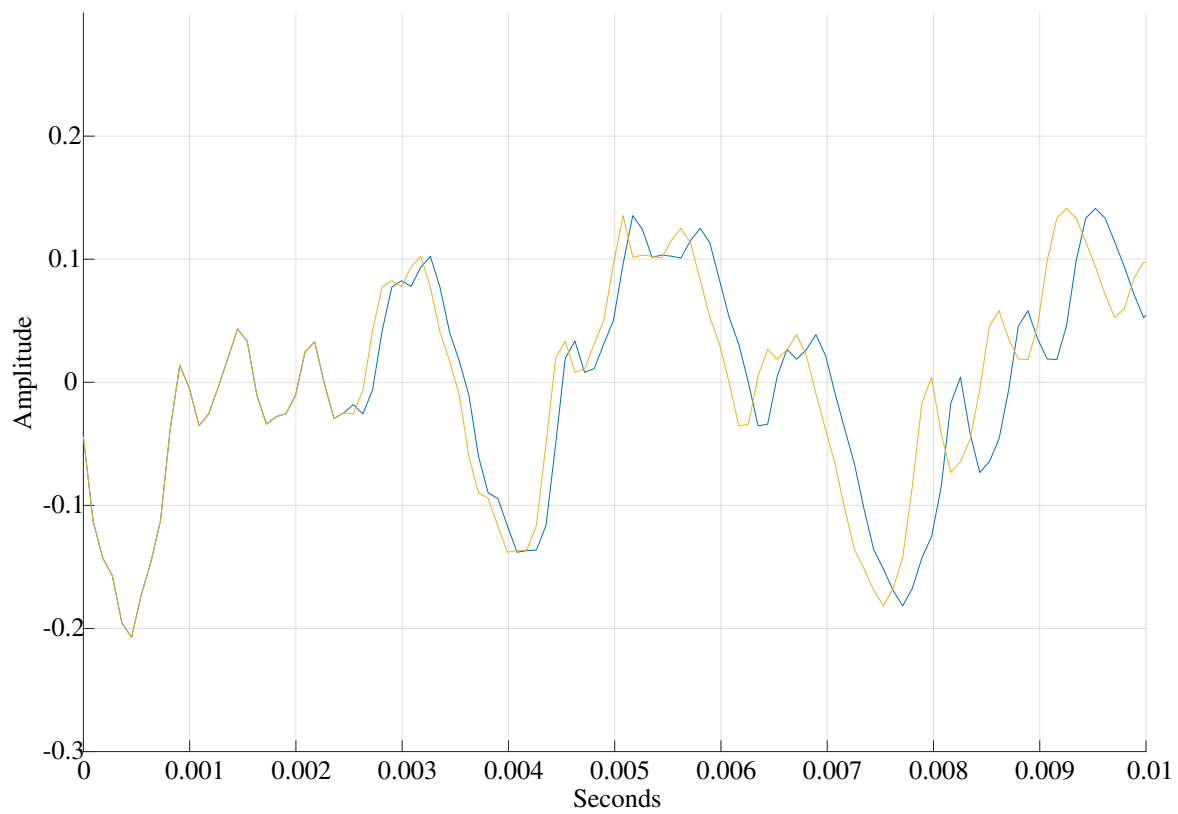


Fig. 6. En esta figura se puede ver en color celeste la onda original, y en color amarillo la onda procesada después de agregar comprimir 5000 muestras de la señal original.