

## 1. Síntesis basada en muestras

La síntesis basada en muestras permite que a partir de una muestra con cierta duración, se pueda obtener una nota distinta a la original y con distinta duración achicando o alargando la muestra. Para ello es necesario poder modificar tanto la frecuencia fundamental como la duración independientemente.

La duración de la nota se puede cambiar a partir del resampleo. Al resamplear una señal, no solo se modifica la duración, sino que también la frecuencia fundamental de la misma.

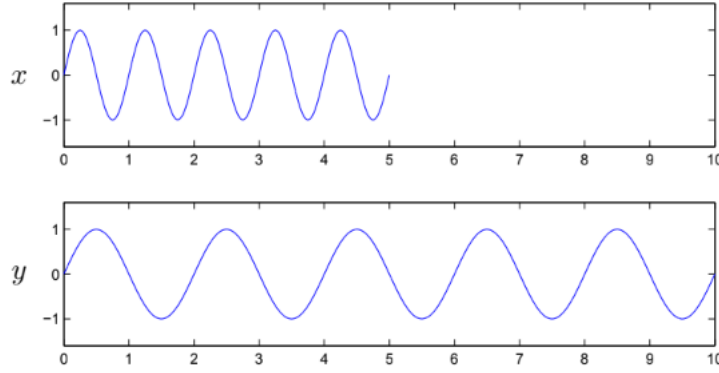


Figura 1:  $x$  está sampleada a una frecuencia de 22050 Hz y tiene una frecuencia de 1Hz. Resampleando la señal con una frecuencia de 44100 Hz se obtiene la señal  $y$  que tiene el doble de duración que  $x$  y la mitad de frecuencia (0.5Hz)

Por lo tanto, una vez cambiada la frecuencia mediante dicha técnica, se necesita un algoritmo que sea capaz de cambiar la duración del audio sin modificar nuevamente su frecuencia.

### 1.1. Escalamiento Temporal (TSM)

Existe una gran variedad de algoritmos que cumplen con la tarea de modificar la duración de un audio. Ya se mencionó el ejemplo del resampleo, a continuación se analizará otro algoritmo denominado Overlap-Add (OLA) que elimina en la mayor escala posible el problema de la alteración en la frecuencia.

#### 1.1.1. Overlap-Add (OLA)

La idea de este algoritmo consiste en cortar la señal de entrada en pequeños segmentos y luego encadenarlos para obtener la longitud deseada.

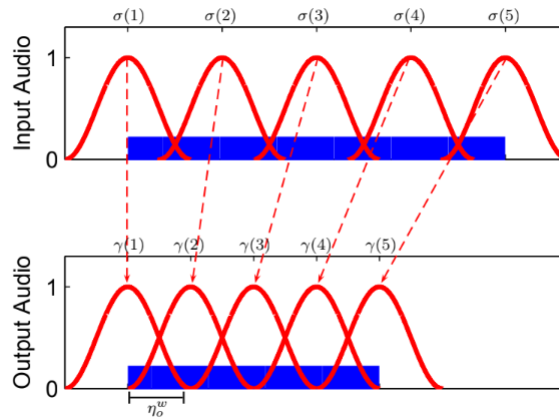


Figura 2: Algoritmo OLA

Los parámetros que utiliza el algoritmo son la muestra original, un factor de solapamiento, una función ventana y una función que 'mapea' los tiempos de la escala original a la nueva escala. Matemáticamente se puede escribir esta síntesis como:

$$y(n) = \frac{\sum_{k=1}^{len(\sigma)} \omega(n - \gamma(k))x(n - \gamma(k) + \sigma(k))}{\sum_{k=1}^{len(\sigma)} \omega(n - \gamma(k))} \quad (1)$$

Donde  $x$  es la señal de muestra,  $\omega$  es la función ventana (en este caso utilizamos la ventana Hann, cuya longitud debe ser mayor que a lo sumo un período de la señal),  $\sigma$  es el vector que contiene las posiciones de las ventanas en la escala de entrada y  $\gamma$  es el vector con las posiciones de las ventanas en la escala de salida. Estos dos están vinculados mediante:

$$\sigma(n) = \tau^{-1}(\gamma(n)) \quad (2)$$

Donde  $\tau$  es la función que convierte los puntos de la escala original a la deseada.

Entonces, como muestra la ecuación, se toma segmentos del original definidos por el tamaño y la amplitud de la ventana utilizada y luego se solapan estos segmentos de acuerdo lo indica la función  $\tau$ , que es la que estira o acorta la escala de acuerdo con la duración que se desea para la salida.

La calidad del sonido que se obtiene a partir de este algoritmo es más aceptable para muestras que no tienen naturaleza armónica como es por ejemplo una señal de voz. Pero cuando se trata de estiramiento temporal para una señal con contenido armónico, como por ejemplo una nota musical, se introducen en la salida ciertos defectos como modulación y saltos de fase producidos por el solapamiento de las ventanas.

A continuación se muestran los resultados obtenidos para una señal con contenido armónico, donde se utilizó como señal de entrada una senoidal pura, y para una señal no armónica, utilizando como muestra una nota de voz.

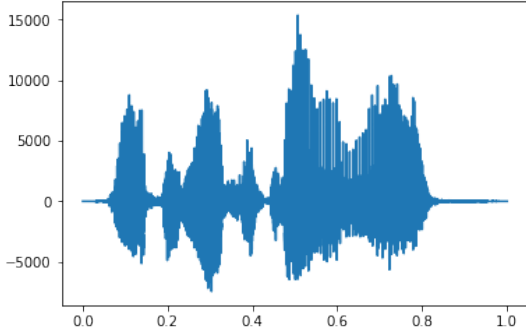


Figura 3: Señal de voz entrada

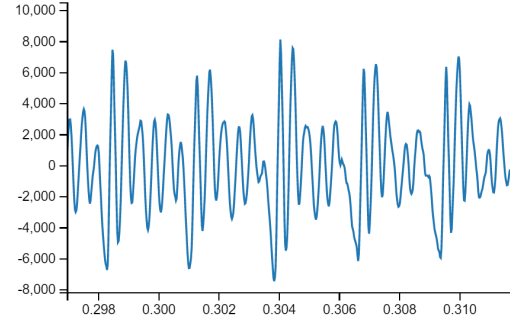


Figura 4: Señal de entrada con zoom

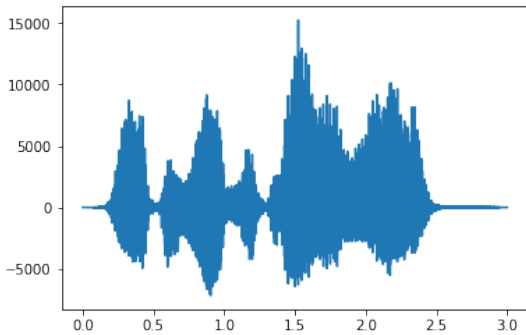


Figura 5: Señal a la salida con el triple de duración que la entrada

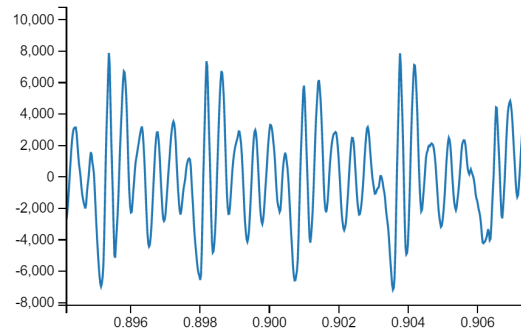


Figura 6: Señal de salida con zoom

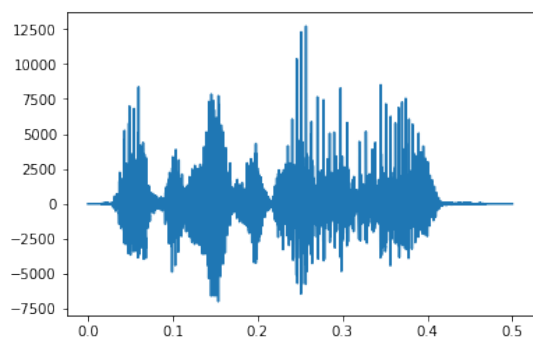


Figura 7: Señal a la salida con la mitad de duración que la entrada

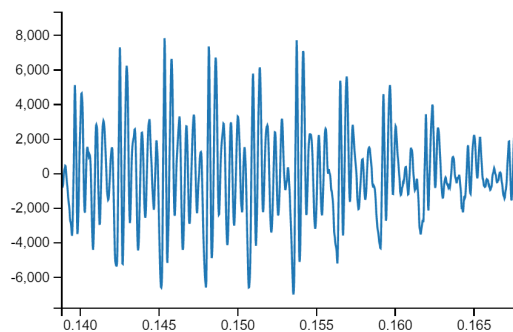


Figura 8: Señal de salida con zoom

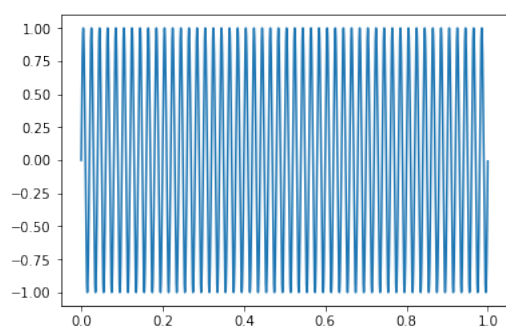


Figura 9: Señal senoidal de entrada (50Hz)

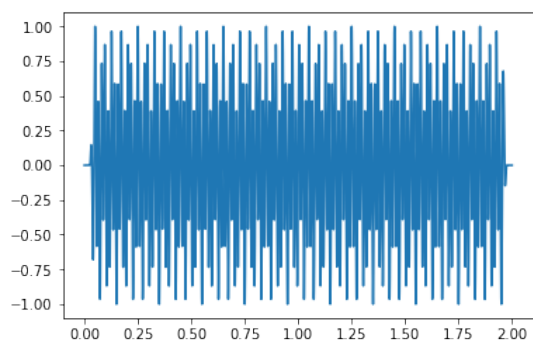


Figura 10: Señal a la salida con el doble de duración que la entrada

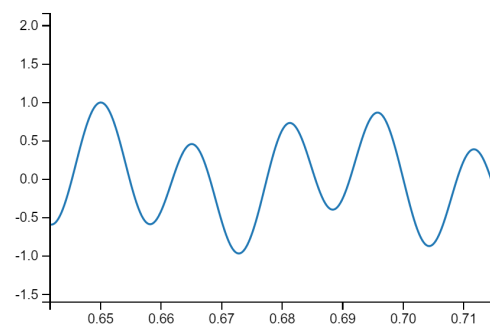


Figura 11: Señal de salida con zoom

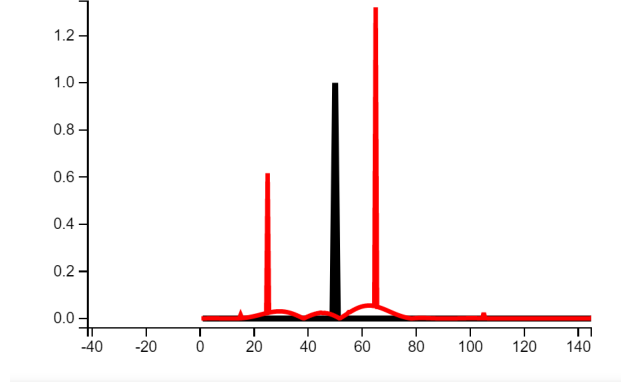


Figura 12: Espectro de las señales de entrada y salida. La negra es la entrada y la roja es la salida

Como se mencionó anteriormente, el algoritmo es mas fiel a la forma de la señal cuando esta no es de naturaleza armónica, pero introduce distorsiones cuando si lo es. En las fotos se puede observar la modulación presente en la salida.

Existen otros algoritmos más complejos que tienen como objetivo evitar este mal comportamiento de OLA en las señales con contenido armónico, uno de ellos es WSOLA.

Su funcionamiento es muy similar al de OLA, pero para que los segmentos sean lo más parecidos posible en la zona que se solapan y el resultado sea más satisfactorio, permite que los estos se puedan desplazar una cierta tolerancia temporal respecto de la posición original. Para encontrar este desplazamiento se busca la correlación máxima entre dos fragmentos adyacentes.

Matemáticamente se describe:

$$y(n) = \frac{\sum_{k=1}^{len(\sigma)} \omega(n - \gamma(k))x(n - \gamma(k) + \sigma(k) + \Delta_k)}{\sum_{k=1}^{len(\sigma)} \omega(n - \gamma(k))} \quad (3)$$

Es muy similar al OLA, salvo por los desplazamientos  $\Delta_k$

## 1.2. Referencias

Time-scale Modification Algorithms for music audio signals by Jonathan Driedger