



Trabajo Práctico Especial

Análisis y procesamiento del habla

[85.05] Señales y Sistemas
Segundo cuatrimestre de 2023

Alumna	Padrón	Email
Viltes, Denise Oriana	106943	dviltes@fi.uba.ar

Índice

1. Introducción	2
2. Ejercicio 1	2
3. Ejercicio 2	3
4. Ejercicio 3	4
5. Ejercicio 4	5
6. Ejercicio 5	7
7. Ejercicio 6	8
8. Ejercicio 7	10
9. Ejercicio 8	13
10. Ejercicio 9	16
11. Ejercicio 10	19
12. Ejercicio 11	19
13. Ejercicio 12	20
14. Ejercicio 13	25

1. Introducción

Para este proyecto se busca analizar y modificar un segmento de audio mediante las técnicas y herramientas de procesamiento de señales y sistemas aprendidas. La herramienta principal para el procesamiento del audio a utilizar es *MATLAB* y sus paquetes de procesamiento de señales.

2. Ejercicio 1

Antes que nada, se grafica el audio dado, el cual corresponde a la frase "*Alzó la voz para ahuyentar a los perros*". Esto se realiza con la ayuda de la aplicación *Wavesurfer*, en el cual se puede observar la forma de onda e conjunto con el espectrograma del audio (fig. 1).

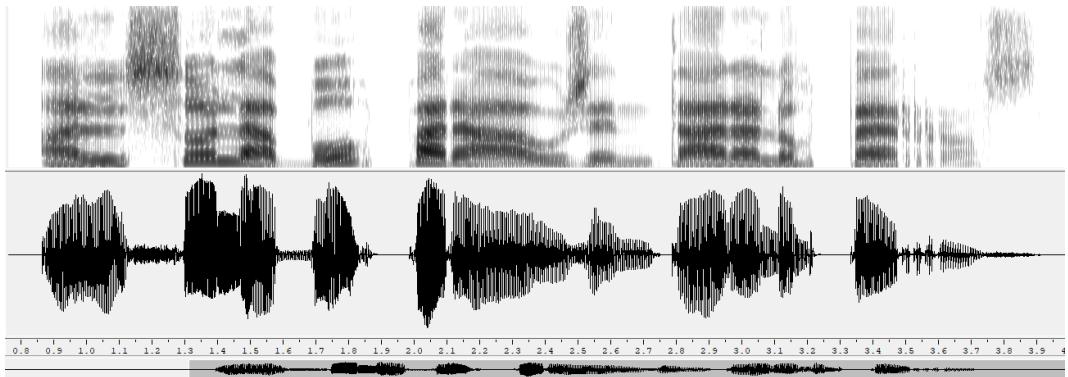


Figura 1: Forma de onda y espectrograma del audio utilizando *Wavesurfer*.

A partir del gráfico del audio y del espectrograma es posible observar en donde comienza cada fonema presente en el audio. Luego, utilizando *MATLAB*, es posible graficar cada segmento anotando su respectivo fonema según el código SAMPA, como se ve en la figura 2.

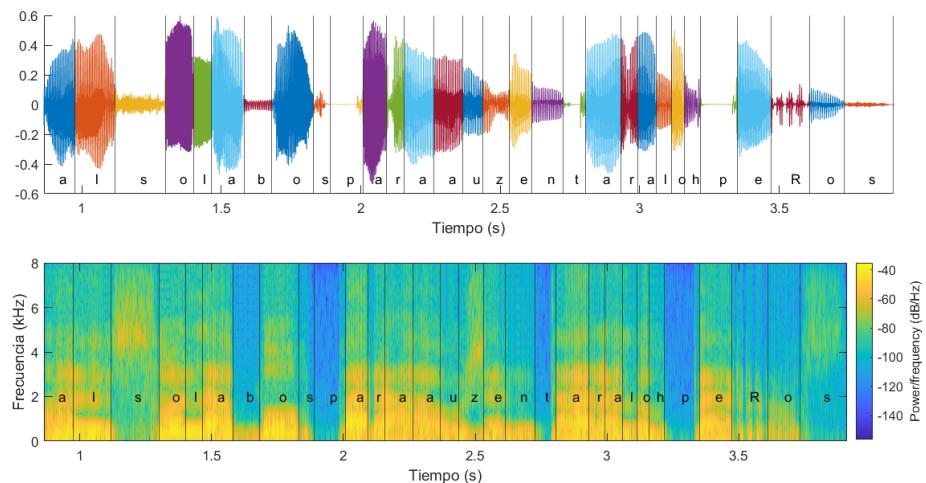


Figura 2: Segmentación del audio

3. Ejercicio 2

En este ítem, se pide elegir un fonema [a] del audio, e identificar sus coeficientes de fourier. Para esto se elige el primer fono [a] del audio, al cual se le aplica la trasformada de fourier, para esto se utiliza la función *fft*, en donde se observan los picos que corresponden a los coeficientes de fourier. Lo mismo se repite para cinco periodos del fono elegido como se ve en la figura 4.

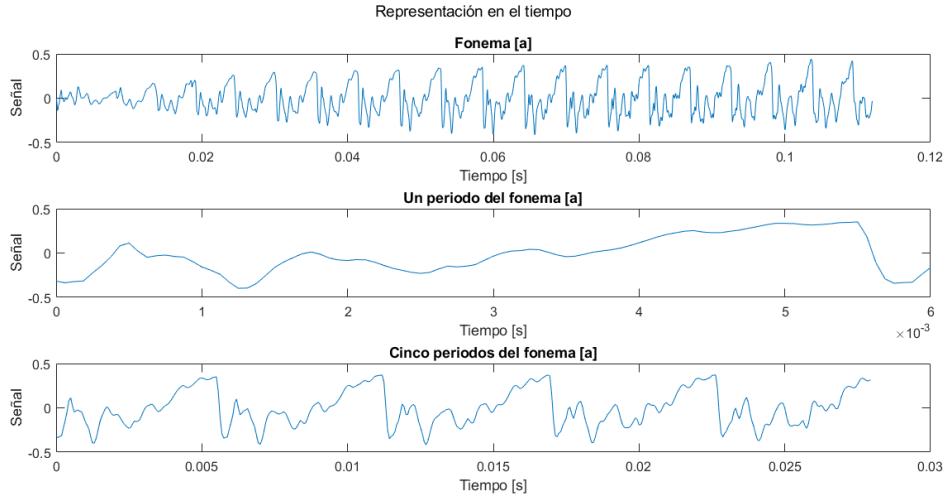


Figura 3: Fonema [a] en el tiempo, y las secciones de 1 y 5 periodos de la misma

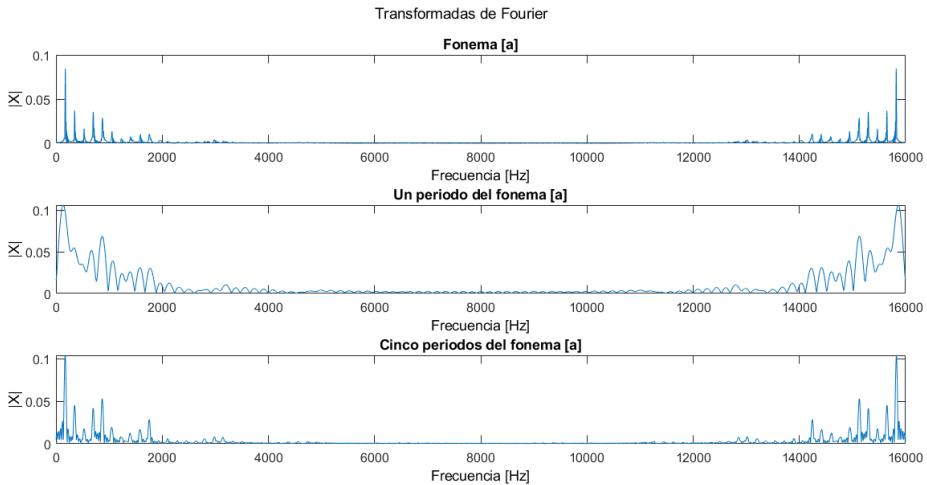


Figura 4: Transformadas de Fourier correspondientes al fonema [a]

Considerando que no hay aliasing, y teniendo en cuenta que la *fft* se trata del calculo de la *dft*, se puede aproximar el valor de los coeficientes como $X[k] = a_k \cdot N$.

4. Ejercicio 3

A partir de los coeficientes encontrados en el ítem anterior, se pide reconstruir la señal. Para ello se realiza la transformada inversa de fourier, en *MATLAB* esto se logra utilizando la función *ifft*. Se realiza la *ifft* con todos los coeficientes, y se observa que se reconstruye la señal exacta, pero al sacar coeficientes se comienza a perder información de la señal. En la siguiente figura (5), se observa la reconstrucción a partir de fft obtenida en el ítem anterior, donde la frecuencia de corte es hasta donde se conservan los valores de la transformada original (fig. 7), luego, son ceros.

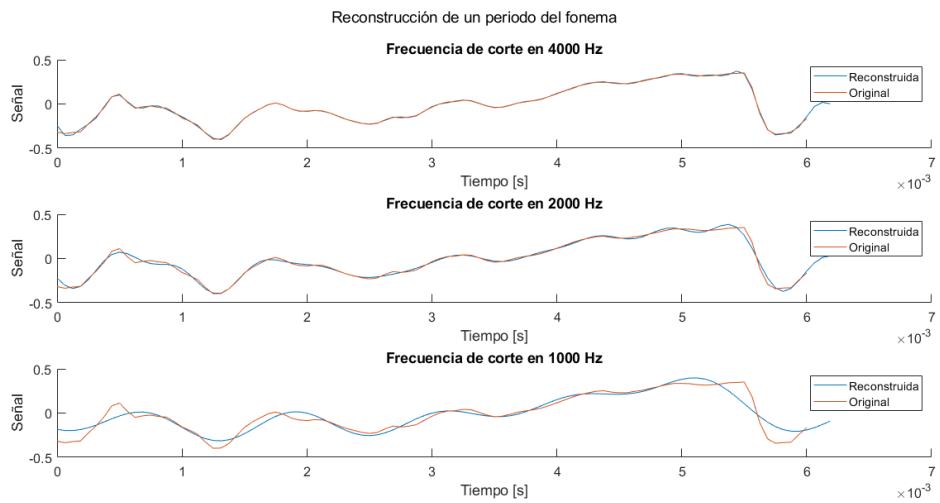


Figura 5: Reconstrucción del primer fonema a partir de distintas frecuencias de corte

De la misma manera, se reconstruyen los cinco períodos segmentados anteriormente, como se ve a continuación,

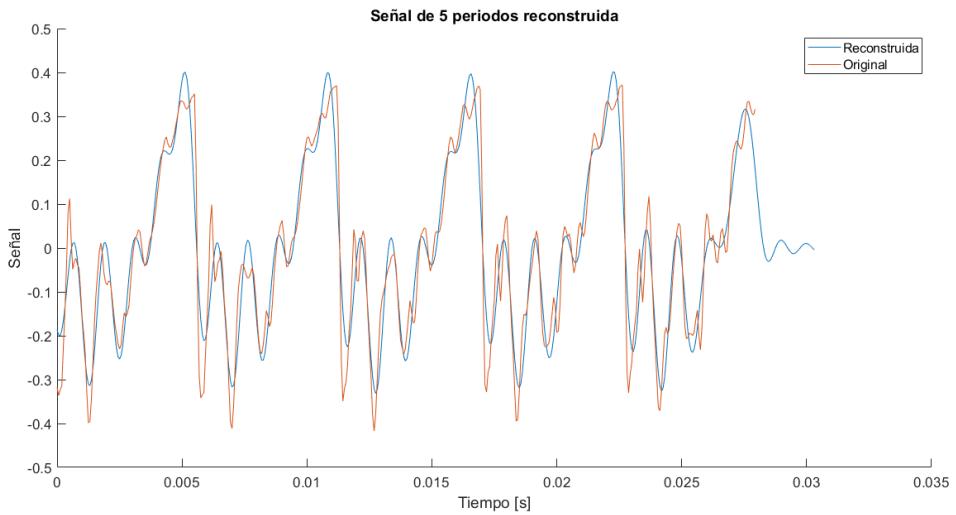


Figura 6: Reconstrucción de los cinco períodos del fono [a]

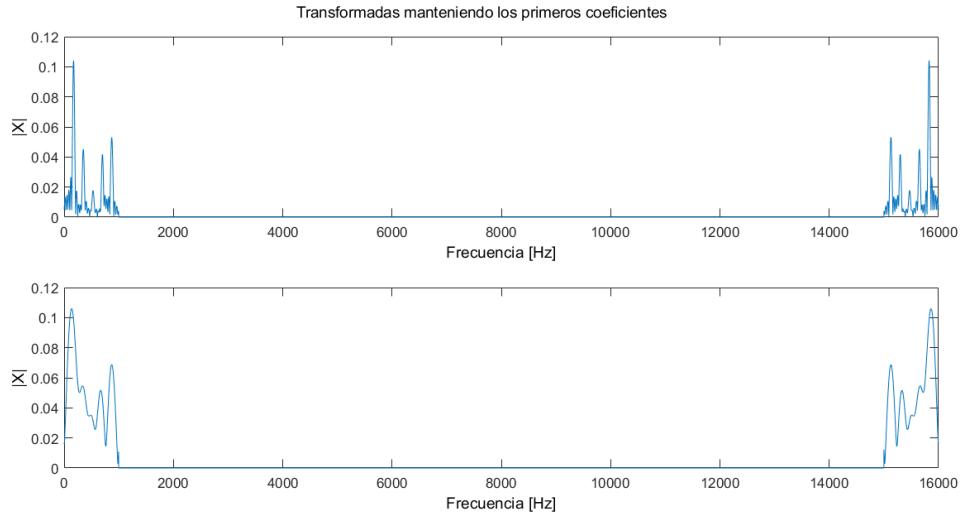


Figura 7: Recorte de la transformada de fourier

Al escuchar los audios de las señales reconstruidas, se encuentra una gran diferencia en ambas señales. La señal de un periodo tiene un sonido más agudo y robotico, incluso es difícil notar que se trata de una vocal "a". En cambio, la de cinco periodos se escucha más grave, pero es posible notar que se trata de una "a". A diferencia de la señal de un periodo, al utilizar varios periodos de la señal original, se guarda más información sobre las variaciones en el tiempo de la señal, ya que aunque varía muy poco la frecuencia de la vocal no es completamente constante. Aunque la mayor variación se ve en la amplitud, cada periodo es distinto al anterior, lo que da como resultado una "a" más real.

Asimismo, se observa que al reconstruir la señal en el tiempo, a pesar de que no se mantiene la forma de onda, si se mantiene el periodo de la señal original.

5. Ejercicio 4

Ahora, se graba un audio con la misma frase, y se lo compara con el audio original, en el tiempo se obtiene la figura 8.

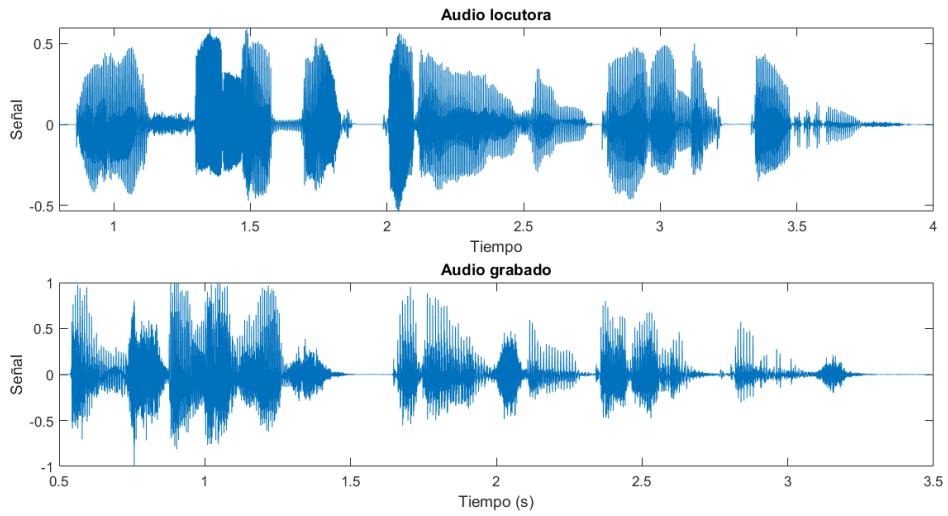


Figura 8: Comparación del audio grabado con el original

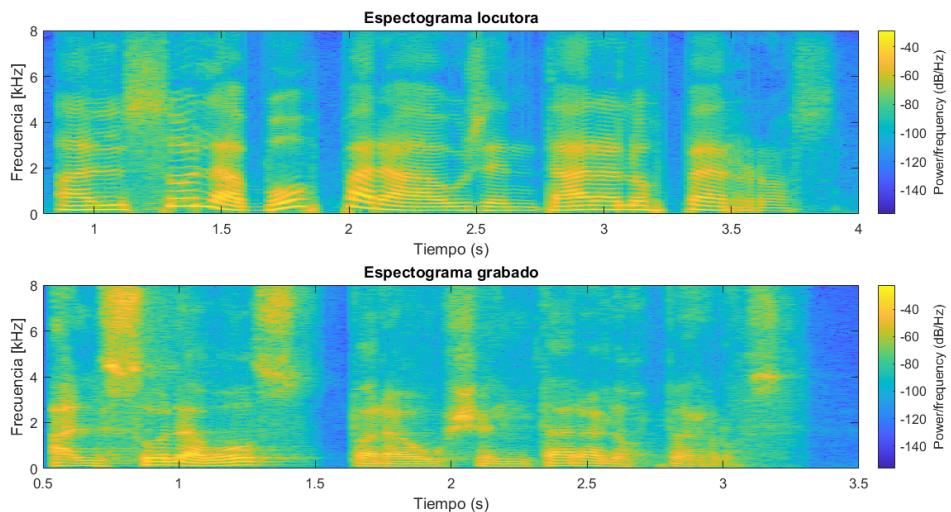


Figura 9: Comparación tiempo-frecuencia del audio grabado y el original, donde se observa la estructura armónica

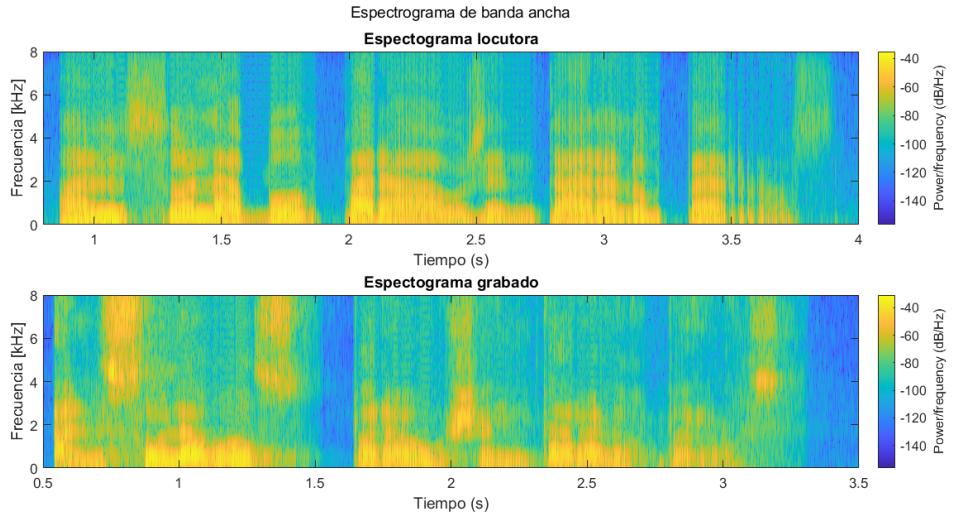


Figura 10: Comparación tiempo-frecuencia del audio grabado y el original, donde se observan los formantes

Si bien en ambos audios se escucha la misma frase, al intentar comparar algunos pocos períodos de un fonema específico, se dificulta notar que se trata del mismo fono. En la grabación, en los sonidos periódicos, se pierde de forma significante la similitud entre cada periodo, ya que no solo cambian en amplitud, si no, que varía la cantidad de picos locales que se observan. En el audio original, los sonidos se ven más limpios y constantes. Además, graficando el espectrograma de banda angosta (fig.9), es posible observar que las frecuencias fundamentales son mucho más bajas, ya que efectivamente se trata de la voz de un hombre. Y con respecto al espectrograma de banda ancha, en este se puede observar que se conserva la disposición de los formantes, claramente cambiando su intensidad, pero es posible distinguir de que fonema se trata en cada tramo.

6. Ejercicio 5

Del audio original se eligen tres vocales, y se grafica el espectrograma de banda angosta. Esto, en MATLAB se logra utilizando la función *spectrogram* obteniendo,

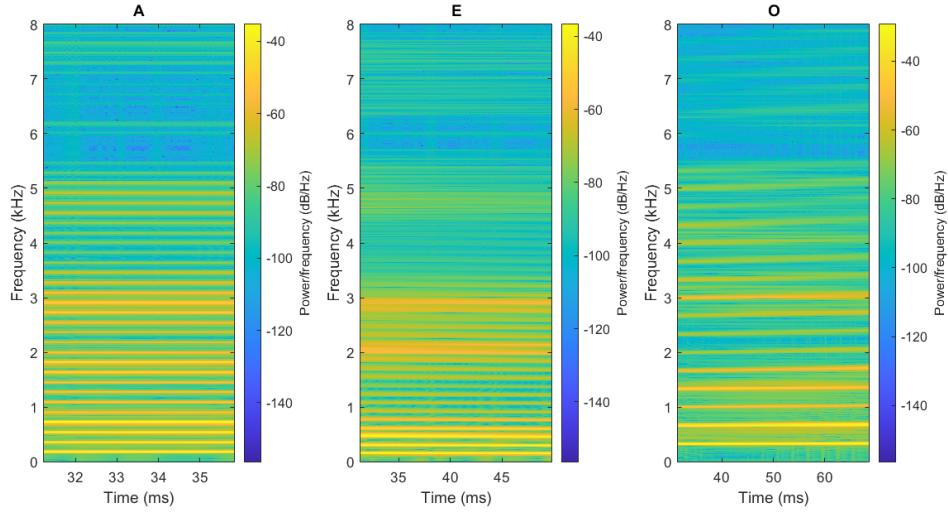


Figura 11: Espectrograma de las vocales a,e,o

Al realizar el espectrograma de banda angosta, es posible observar los formantes de las vocales. Y es justamente la disposición de los formantes la principal diferencia entre las vocales. En la *a*, se ve que los formantes se ubican con una separación casi constante, y la intensidad disminuye de forma gradual a medida que aumenta la frecuencia. En la *o*, se observa algo similar pero con una distancia mayor entre los formante. En cambio con la *e*, si bien la distancia entre formantes es similar a la *a*, se observan sectores, en donde no solo aumenten la intensidad, si no que también disminuye la distancia entre los formantes.

7. Ejercicio 6

Ahora, se busca crear un propio pulso glótico, para el cual se parte del modelo de Rosenberg. Según este modelo, un periodo del pulso glótico se puede modelar como,

$$P(t) = \begin{cases} \frac{P_0}{2} [1 - \cos(\frac{t}{T_p}\pi)] & \text{si } 0 \leq t \leq T_p \\ P_0 \cos(\frac{t-T_p}{T_n}\frac{\pi}{2}) & \text{si } T_p \leq t \leq T_p + T_n \end{cases} \quad (1)$$

Siendo T_p la duración de la fase de apertura, en este caso es un 40 % de la duración de un periodo, T_n la duración de la fase de cierre, ahora de un 16 % del periodo, y P_0 la amplitud, la cual se pide que tenga valor unitario.

Se genera un periodo de pulso glótico con una frecuencia fundamental $f_0 = 200$ Hz, lo cual significa que se está modelando el pulso glotal perteneciente a una mujer, y luego este periodo se lo repite 10 veces para formar el tren de pulsos pedido (fig. 12).

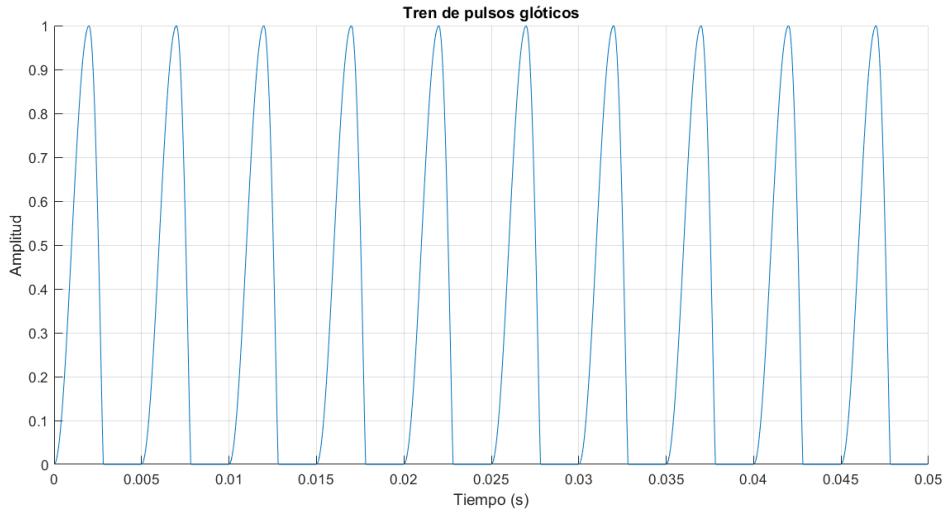


Figura 12: Tren de pulsos glóticos generados.

En la figura 13, se observan las transformadas de fourier de los pulsos glóticos generados. Al realizar la transformada en *MATLAB*, se está realizando la dft, a la cuál se le pasa un segmento finito de muestras, esto significa multiplicar el tren de pulsos en el tiempo por una ventana rectangular. Por otro lado, generar el tren de pulsos glóticos, significa convolucionar en tiempo el pulso glótico junto con un tren de deltas. Por lo tanto, al pasar a frecuencia se observaría el resultado de convolucionar el tren de deltas (la transformada de un tren de deltas, sigue siendo un tren de deltas), con una sinc, correspondiente a la transformada de la ventana, es decir que se observarían sincs ubicadas en cada delta del tren. Luego este tren de sincs, se multiplica por la transformada del pulso glótico, por lo cual quedarían las amplitudes de cada sinc ajustadas a la amplitud de la transformada del pulso glótico.

Lo cual es justamente lo que se observa en la figura 13, en amarillo se grafica la transformada de un solo pulso, mientras que en azul se ve un tren de pulsos que se ajusta a la forma de la transformada del pulso

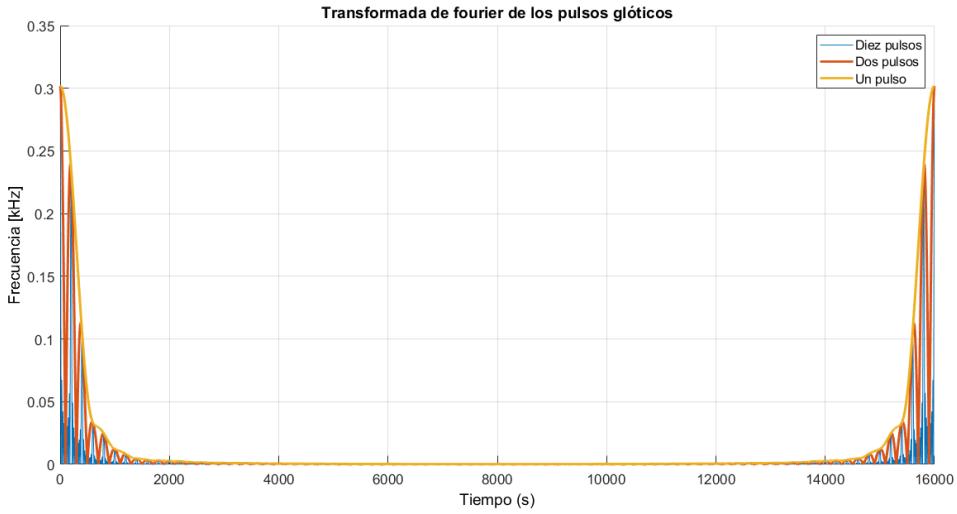


Figura 13: Transformada de fourier de los pulsos glóticos generados.

8. Ejercicio 7

Siendo el objetivo producir el sonido de una vocal, una vez generado el tren de pulsos es necesario moldearlo según el tracto vocal. Es decir, que en el ítem anterior se modeló la forma y frecuencia de la voz, la cual es propia de cada hablante, pero lo que se necesita ahora es modelar el efecto de las distintas configuraciones que toma el tracto vocal para lograr diferentes sonidos. Para esto, se utiliza una aproximación mediante filtros IIR (2), luego para el cálculo de los polos del filtro se utiliza (3), los cuales toman los valores de frecuencia de resonancia y ancho de banda de la tabla 1.

$$H_n(z) = \frac{1}{(1 - p_n z^{-1})(1 - p_n^* z^{-1})} \quad (2)$$

$$p_n = \exp\left(\frac{-2\pi B}{F_s}\right) \exp\left(j\frac{2\pi F_n}{F_s}\right) \quad (3)$$

	F_1	B_1	F_2	B_2	F_3	B_3	F_4	B_4
a	830	110	1400	160	2890	210	3930	230
e	500	80	2000	156	3130	190	4150	220
i	330	70	2765	130	3740	178	4366	200
o	546	97	934	130	2966	185	3930	240
u	382	74	740	150	2760	210	3380	180

Tabla 1: Tabla de valores de frecuencia de resonancia y ancho de banda para el tracto de cada vocal

Para generar cada tracto, se crea un filtro mediante la función `zpk`, en la cual se cargan los polos calculados para cada vocal, y los ocho ceros correspondientes. Utilizando la función `pzmap` para graficar el diagrama de polos y ceros y calculando la transformada de fourier del filtro, se obtienen las siguientes figuras,

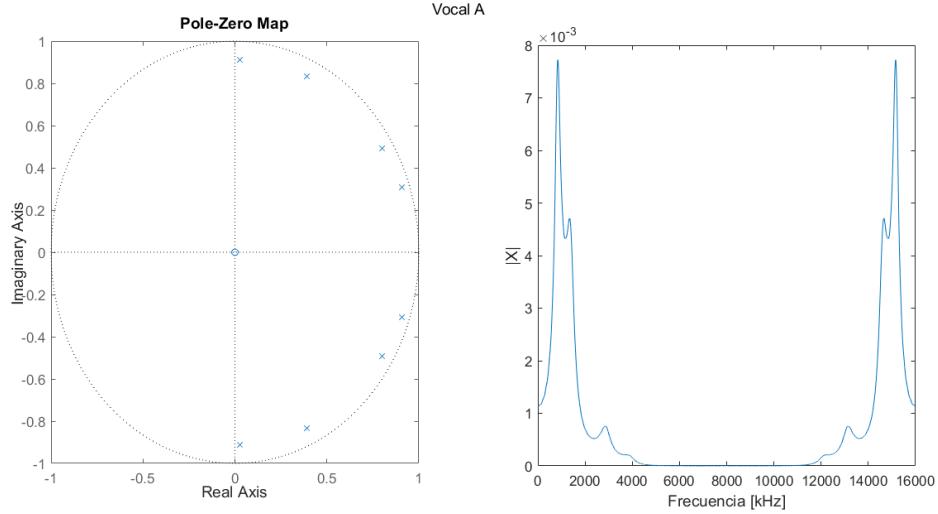


Figura 14: Diagrama de polos y ceros, y respuesta en frecuencia del modelo de tracto vocal para la vocal *a*

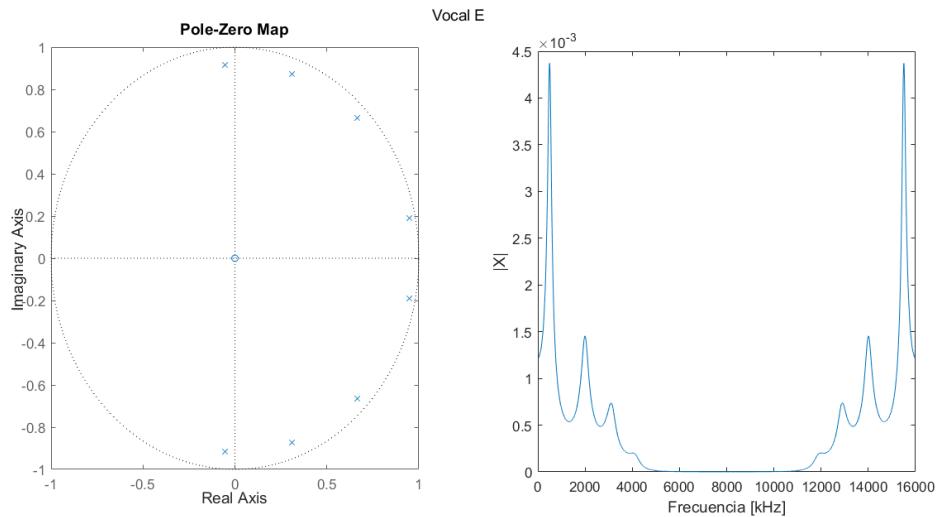


Figura 15: Diagrama de polos y ceros, y respuesta en frecuencia del modelo de tracto vocal para la vocal *e*

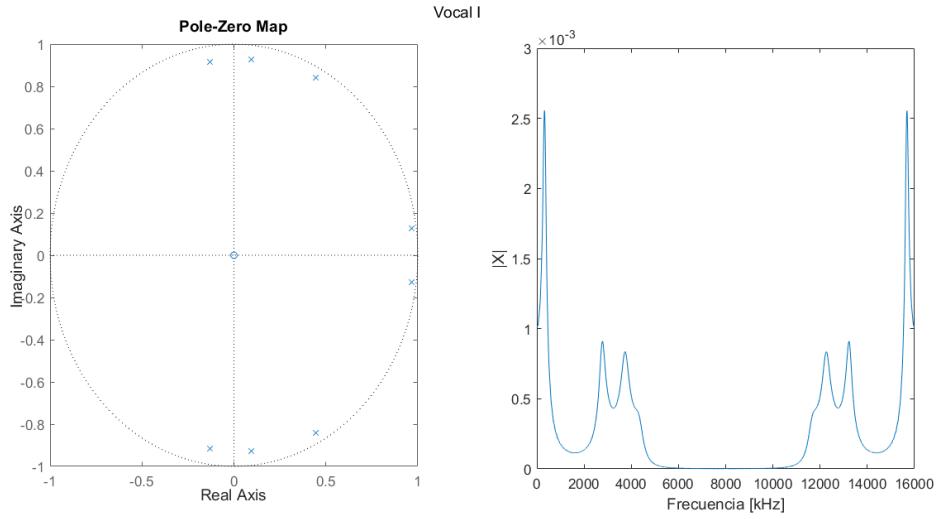


Figura 16: Diagrama de polos y ceros, y respuesta en frecuencia del modelo de tracto vocal para la vocal *i*

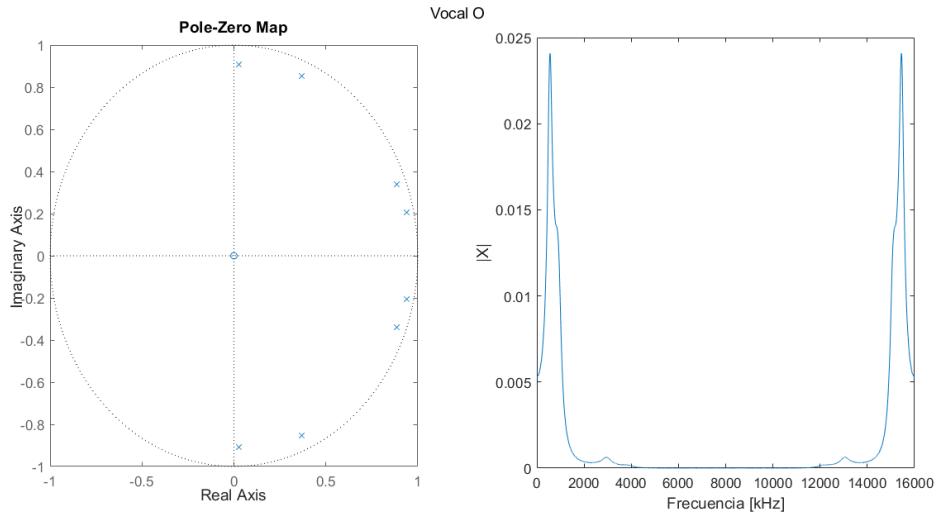


Figura 17: Diagrama de polos y ceros, y respuesta en frecuencia del modelo de tracto vocal para la vocal *o*

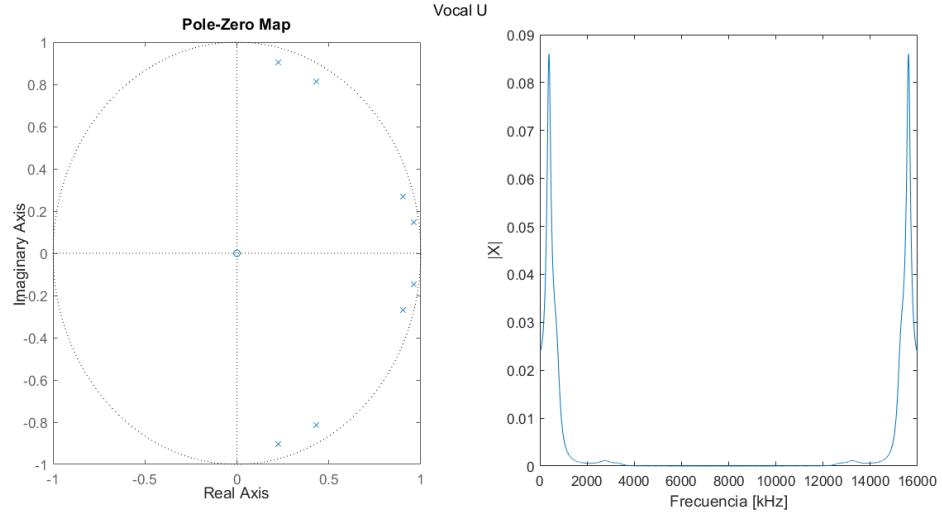


Figura 18: Diagrama de polos y ceros, y respuesta en frecuencia del modelo de tracto vocal para la vocal u

9. Ejercicio 8

Finalmente, utilizando los resultados de las secciones 7 y 8, es posible sintetizar el sonido de una vocal. El objetivo es emular sonido de una vocal a partir de estos resultados, es decir encontrar la respuesta temporal del pulso glótico al pasar por el filtro diseñado.

Para esto, una función útil es *lsim*, la cual justamente devuelva la respuesta en tiempo de un sistema para una entrada determinada. Por lo tanto, se convoluciona mediante dicha función, un tren de pulsos glóticos de 200 periodos, para ver un 1 s de la señal, y el filtro del tracto vocal de cada vocal.

A continuación se exponen los resultados de la sintetización de las vocales, junto con su transformada de fourier y el espectrograma de banda angosta.

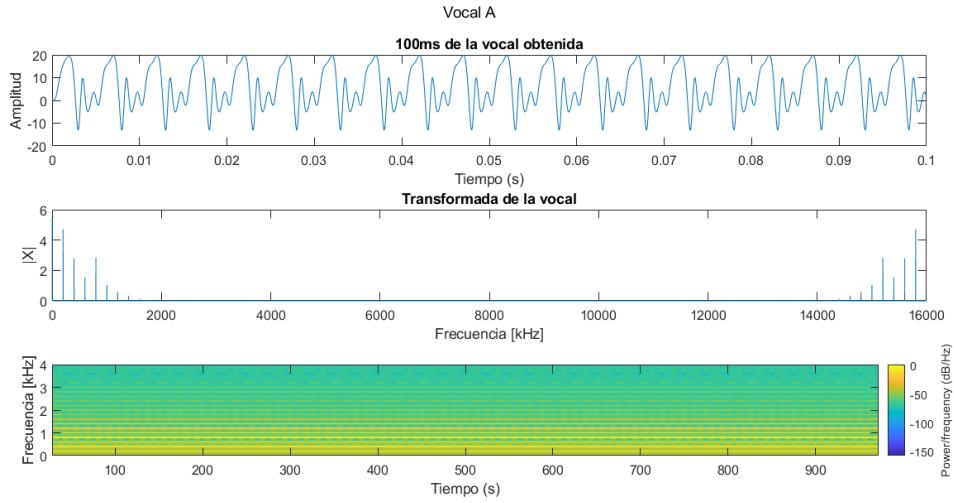


Figura 19: Resultados de sintetizar la vocal *a* junto con los análisis en frecuencia y tiempo-frecuencia.

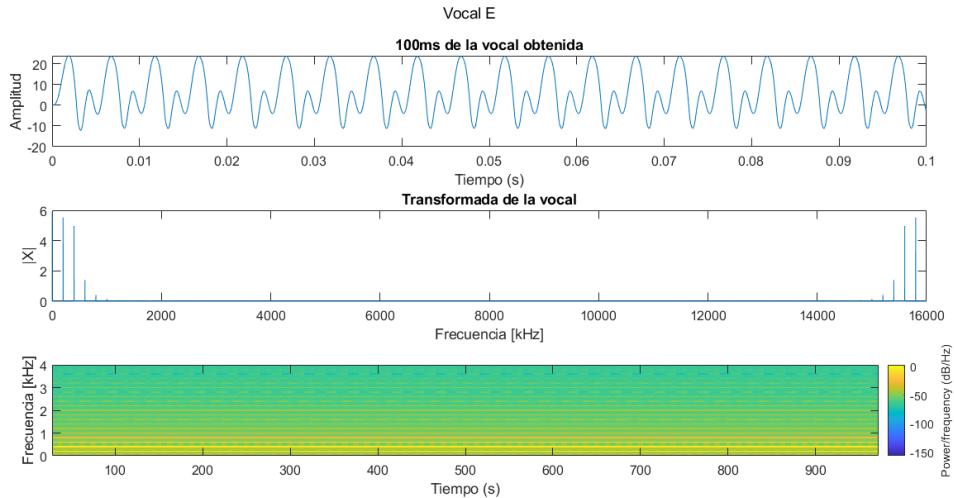


Figura 20: Resultados de sintetizar la vocal *e* junto con los análisis en frecuencia y tiempo-frecuencia.

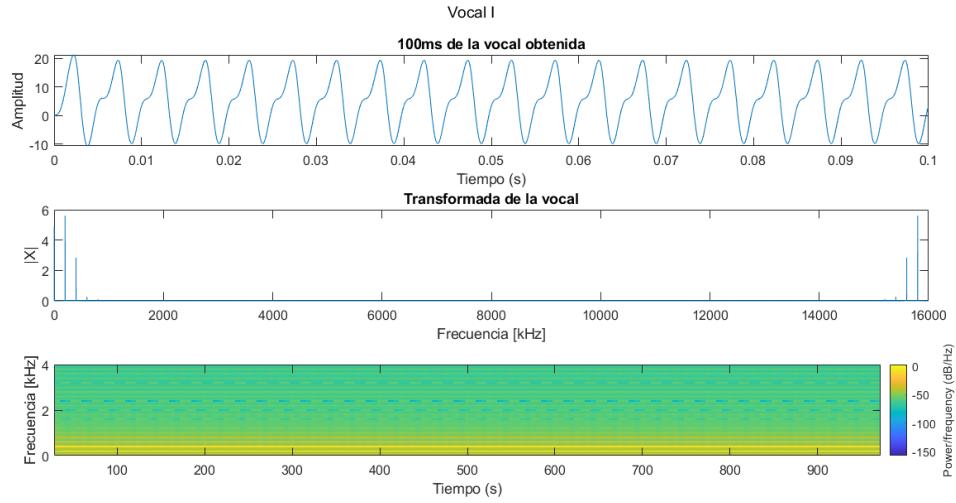


Figura 21: Resultados de sintetizar la vocal *i* junto con los análisis en frecuencia y tiempo-frecuencia.

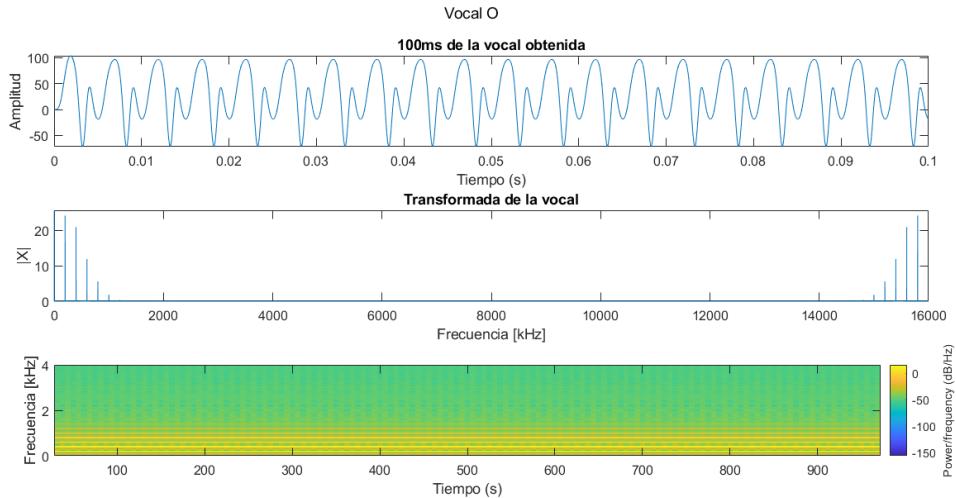


Figura 22: Resultados de sintetizar la vocal *o* junto con los análisis en frecuencia y tiempo-frecuencia.

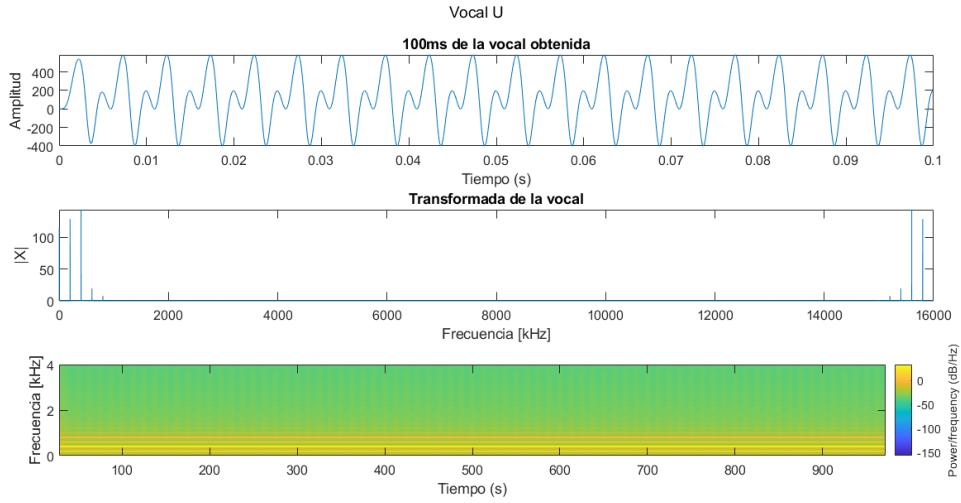
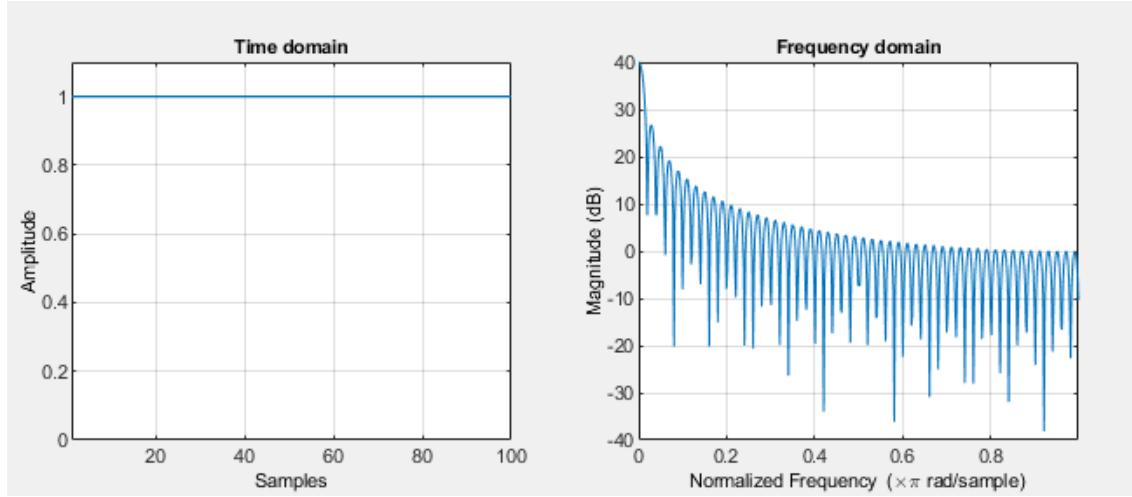
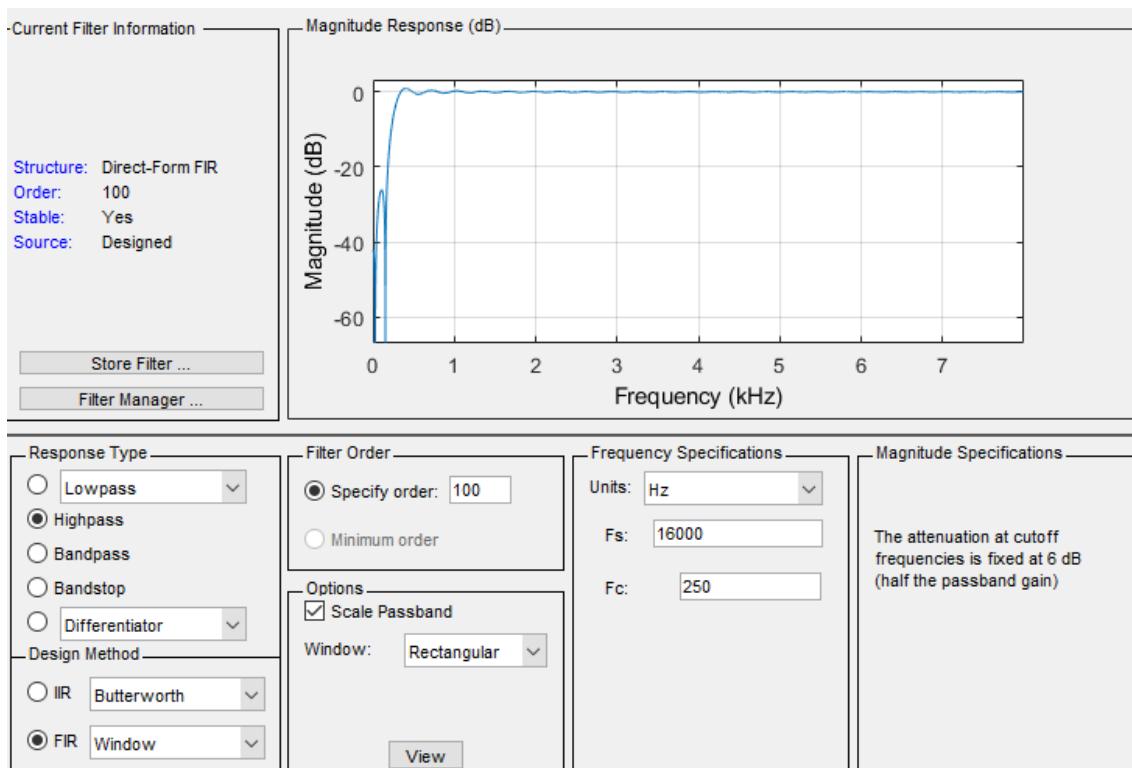


Figura 23: Resultados de sintetizar la vocal *u* junto con los análisis en frecuencia y tiempo-frecuencia.

10. Ejercicio 9

Para este ejercicio se pide filtrar las vocales obtenidas, con un filtro tal que elimine la frecuencia fundamental, es decir $f_0 = 200$ Hz.

En cuanto al diseño del filtro, se utiliza la aplicación de MATLAB, "Filter Designer"(25). Ya que no es necesario un filtro complejo, y se busca que sea estable, se elige un filtro FIR, se elige una ventana rectangular (32), tal que atenúe la frecuencia f_0 pero no modifique en lo posible el resto de las frecuencias siguientes. Por esta misma razón se elige un filtro pasa altos, que deje pasar frecuencias mayores a f_0 . Otro parámetro que se modifica es el orden del filtro, al aumentar el orden, se aumentan la cantidad de coeficientes, y se logra que se produzca un aumento abrupto de la ganancia luego de pasada la frecuencia de corte, logrando que no se atenúen los armónicos siguientes al fundamental.

Figura 24: Ventana obtenida en utilizando *fdatool*.Figura 25: Diseño de la ventana en *fdatool*.

Se aplica el filtro diseñado a cada vocal utilizando la función *filter* (fig.26), y se observa que genera una significante atenuación de la amplitud de la señal, y si bien se observa un desplazamiento de la señal debido a la respuesta transitoria del filtro, este no modifica la frecuencia de la señal original. Por esta razón, no hay grandes cambios perceptibles en el audio de las vocales filtradas, más allá de una pequeña disminución en la intensidad.

Es interesante observar, que al atenuar el armónico, la señal cada vez se asemeja más a una

senoidal, exceptuando la vocal *a*. Esto ocurre porque los formantes de la vocal *a*, se encuentran alejados de la frecuencia fundamental, por lo que al atenuar la frecuencia fundamental, no se pierde información de los formantes de la vocal. En el resto de las vocales, esto último no ocurre, ya que las frecuencias de los formante son similares a la frecuencia eliminada, por lo que son atenuadas, o incluso eliminadas.

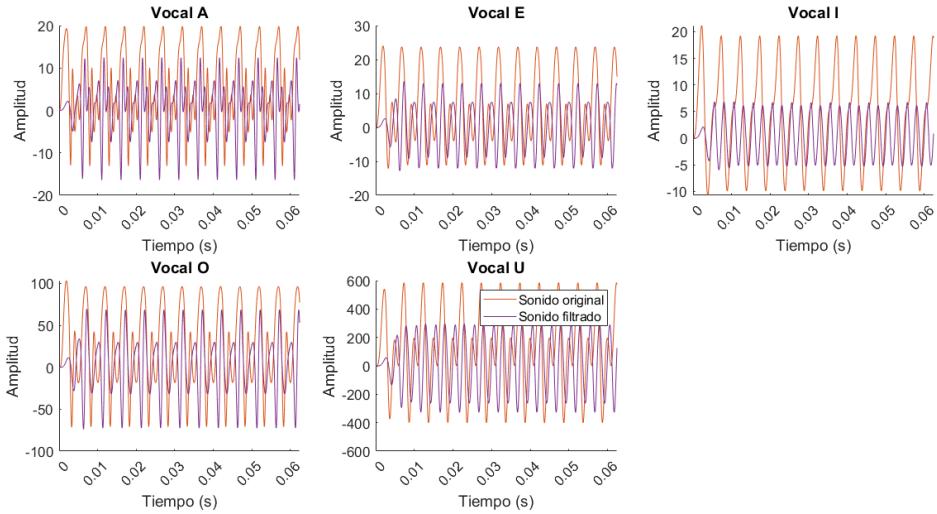


Figura 26: Filtrado de las vocales sintetizadas

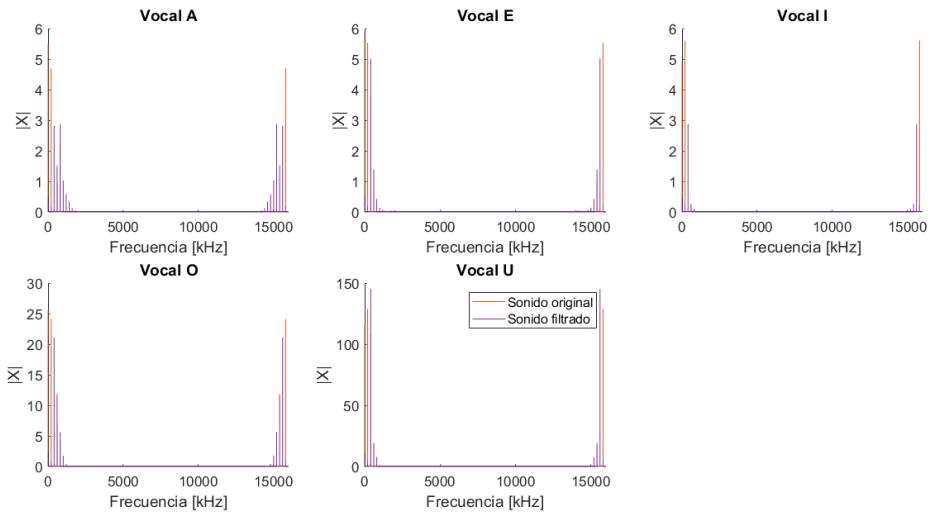


Figura 27: Análisis en frecuencia de las vocales sintetizadas y filtradas

En la figura 27 es posible observar como al filtrar la señal, se elimina el primer pico de la transformada, correspondiente a la frecuencia fundamental, pero no afecta al resto de las frecuencias mayores.

11. Ejercicio 10

La transformada cepstrum resulta ser una herramienta útil para discernir entre el tracto vocal y el pulso glótico de un sonido ya sintetizado. Esta transformada se define como,

$$c[n] = \mathcal{F}^{-1}\{\log(|\mathcal{F}\{x[n]\}|)\} \quad (4)$$

Al aplicar esta transformada al sonido, es posible aislar el tracto vocal, el cual al tratarse de una señal no periódica que se desarrolla lento en el tiempo, se va a encontrar al principio del gráfico de la transformada cepstrum, y luego sumado a este la transformada del pulso glótico, la cual resulta, otro tren de pulsos. Por lo tanto al seccionar la parte correspondiente al tracto vocal y anti-transformarlo, es posible observar el tracto vocal nuevamente. Esto se puede observar en la siguiente figura

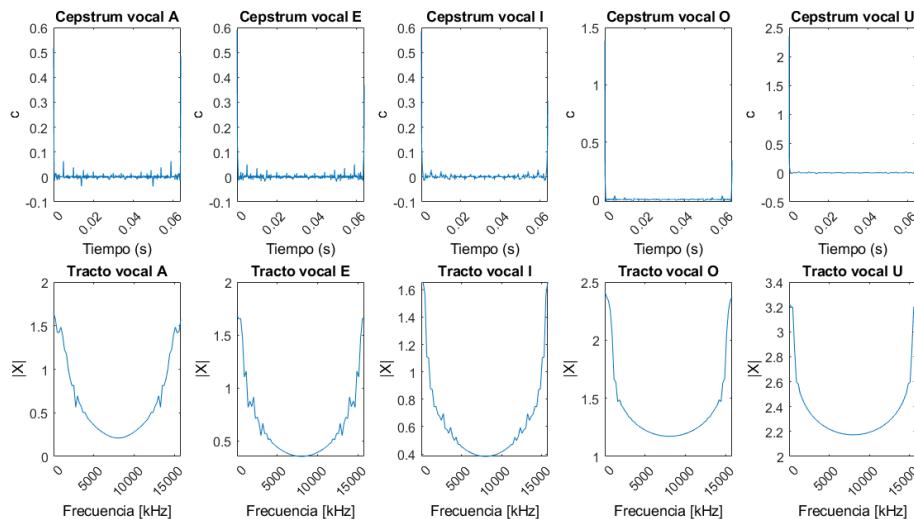


Figura 28: Transformadas cepstrum, y reconstrucción del tracto vocal

La frecuencia fundamental es posible hallarla sabiendo que se corresponde al primer pico luego del tracto vocal. Para la voz humana, este pico debe buscarse en el rango de tiempo $\frac{1}{50}$ s y $\frac{1}{500}$ s. Para el caso de estas vocales, ya que se construyeron de forma periódica, la frecuencia fundamental es constante y corresponde a los 200 Hz utilizados en el diseño de los pulsos.

12. Ejercicio 11

En el caso del audio dado en el enunciado, el contorno de frecuencia fundamental ya no se tratará de una constante, de hecho va a variar en el tiempo. Por lo tanto para hallar este contorno, se aplica la transformada cepstrum en pequeñas secciones del audio, a lo largo de todo el audio. Al graficar este contorno (fig.29), es posible observar como varía la frecuencia para los distintos fonemas. En el caso de los periódicos, se trata de una variación suave, pero en los fonemas

que se componen de silencios y ruidos, este análisis pierde sentido, ya que estos no tienen una frecuencia, por lo que se observan valores dispersos y aleatorios.

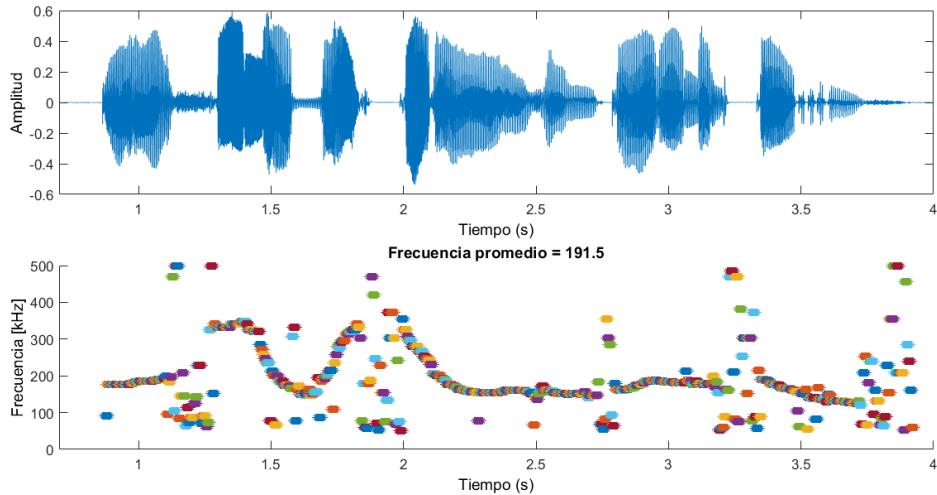


Figura 29: Contorno de frecuencias para el audio completo utilizando la transformada cepstrum

13. Ejercicio 12

En esta ultima parte, se centra en la modificación de la prosodia, para esto se pide utilizar el método **PSOLA**. Para aplicar este método se siguen los siguientes pasos,

- Se buscan los picos máximos de la señal y la distancia pico a pico.

Para este resultado útil la transformada cepstrum, ya que se puede encontrar fácilmente la frecuencia fundamental del segmento de audio. Luego en base a esta, se procede a buscar las posiciones de los picos máximos.

- En segundo lugar, se toman segmentos de señal centrados en cada pico, buscando que estos se solapen.

Para lograr una mayor suavidad en la señal final, se le aplica una ventana con los bordes atenuados, para evitar información duplicada, y evitar distorsionar la relación entre las amplitudes de los picos.

- Luego, se suman estos segmentos (fig. 31), ya sea aumentando o disminuyendo el periodo de la nueva señal. (fig. 30)

La distancia entre cada pico de la señal nueva dependerá de la distancia de los picos originales y cuanto se quiera aumentar o disminuir la señal nueva. Esto se puede expresar como $T_{nuevo} = \frac{T_{original}}{1+k}$, siendo el k el valor de cuanto se quiere variar la señal, y se toma $k > 0$ para aumentar la frecuencia, y $k < 0$ para disminuir la frecuencia (cabe aclarar que esto se aplica periodo a periodo, ya que puede no ser constante la distancia entre picos).

d. Por último, es necesario escalar la nueva señal, para que se mantenga la duración de la original.

Se busca realizar este escalamiento, para que la señal mantenga la misma velocidad que la original.

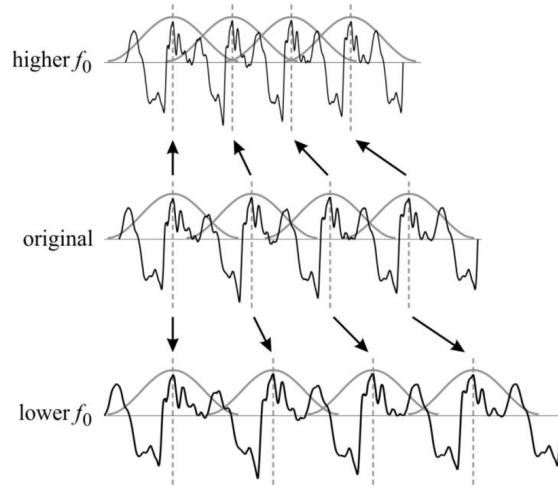


Figura 30: Cambio del largo de la distancia entre picos al cambiar la frecuencia

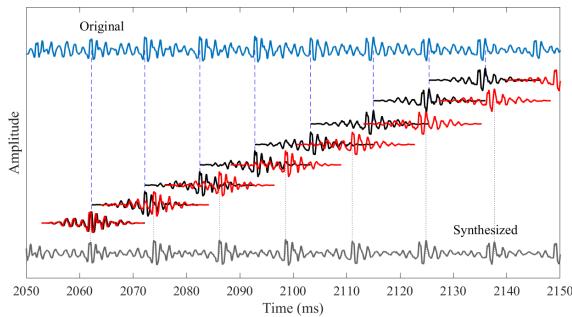


Figura 31: Técnica *overlap add* para unir los segmentos

La ventana elegida (fig. 32) para se trata de una rectangular con los bordes atenuados con una ventana *hamming*, de esta forma se obtiene la mayor información posible cerca del pico principal, y las zonas superpuestas quedan atenuadas. Variando el tamaño de esta, es posible obtener más o menos información de los períodos contiguos, el mejor resultado se obtuvo al utilizar una ventana un 100 % más grande que la distancia entre picos. Con este valor fue posible mantener la amplitud original al sumar los segmentos.

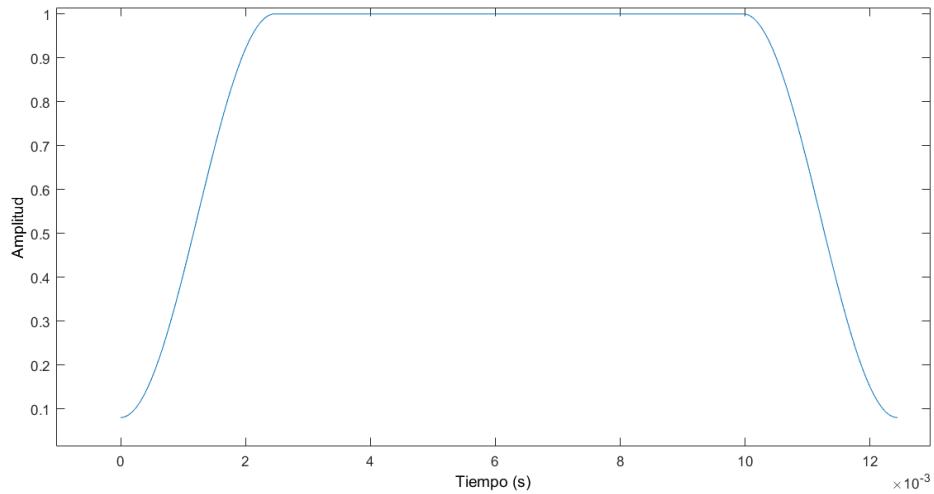


Figura 32: Modelo de ventana elegido para segmentar el audio

A continuación se observan los resultados para la variación de un 10 % de la frecuencia fundamental para cada vocal. Si bien se observan cambios en la forma de onda de algunas vocales, es posible identificar de qué fonema se trata luego del cambio de frecuencia. Esto se debe a limitaciones en la implementación del algoritmo.

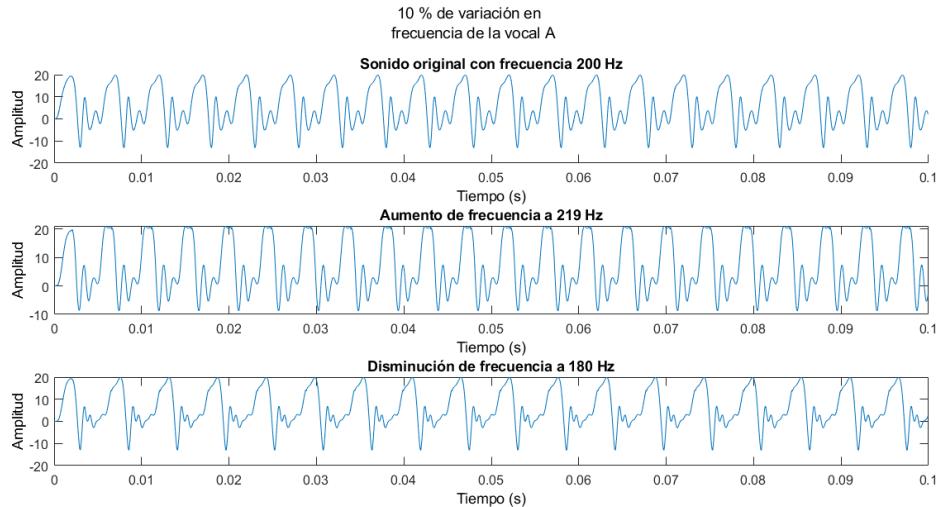


Figura 33: Resultados de utilizar el método PSOLA para la vocal A.

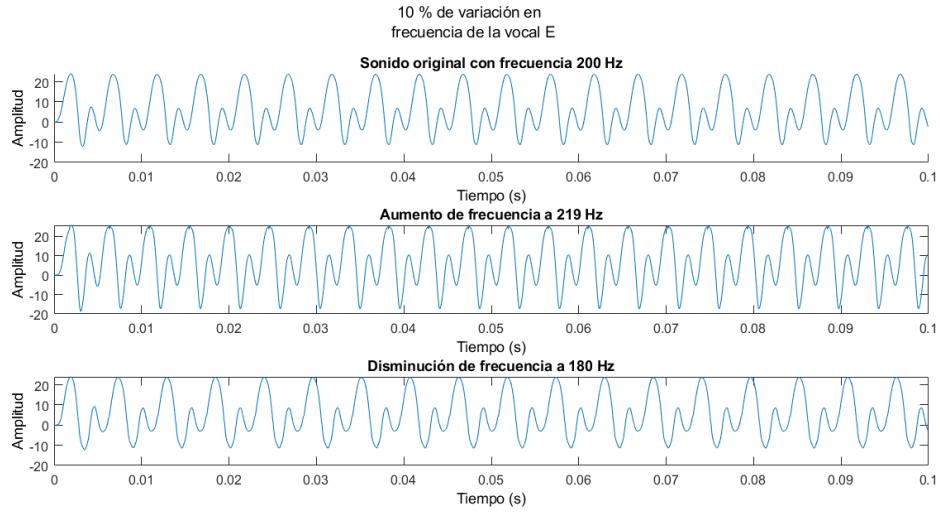


Figura 34: Resultados de utilizar el método **PSOLA** para la vocal E .

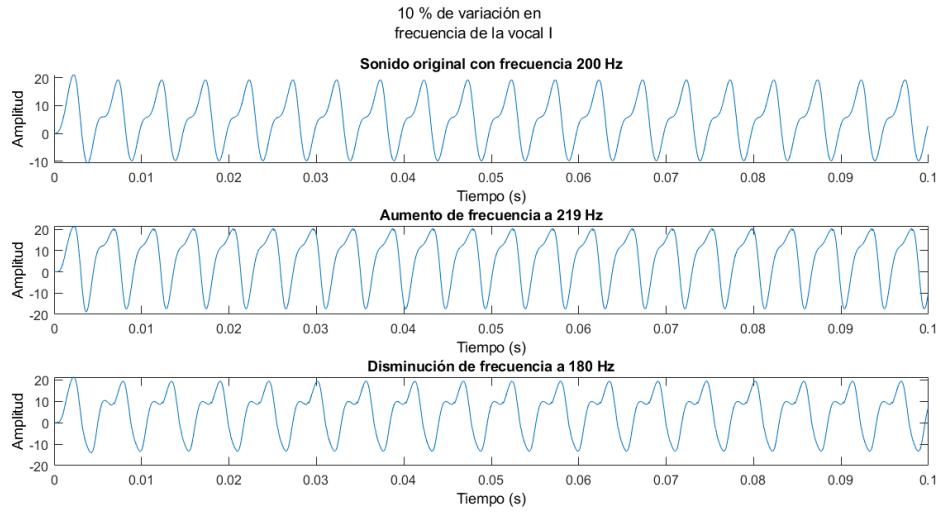


Figura 35: Resultados de utilizar el método **PSOLA** para la vocal I .

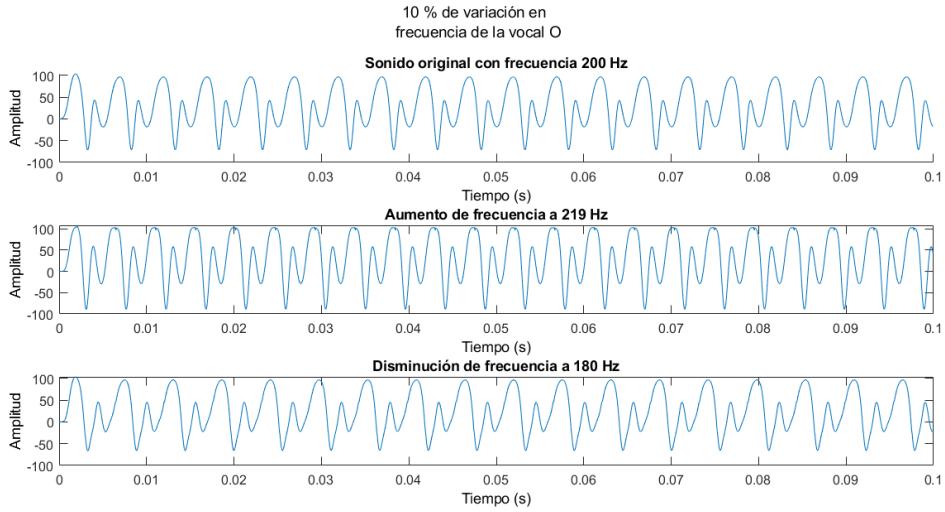


Figura 36: Resultados de utilizar el método **PSOLA** para la vocal O .

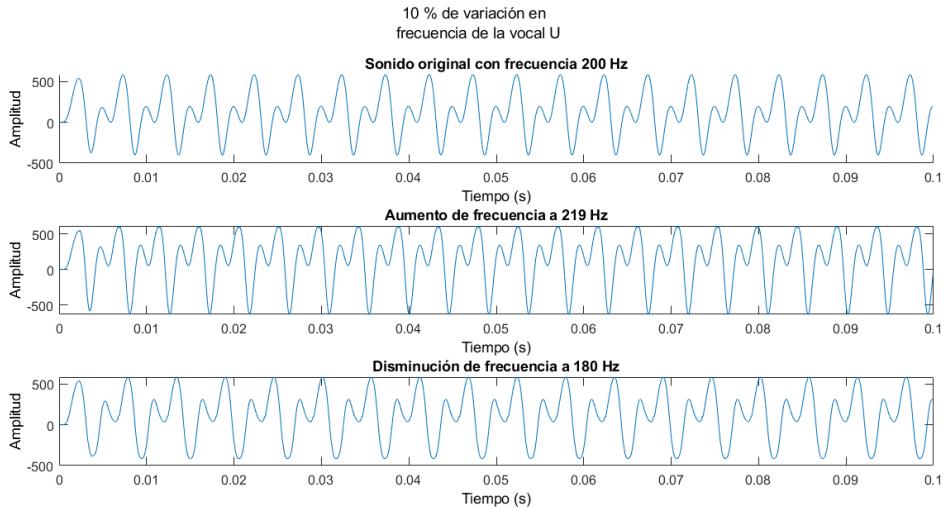


Figura 37: Resultados de utilizar el método **PSOLA** para la vocal U .

Luego, se ve como varia para la vocal *a* con un 20 % y 30 % de variación de frecuencia. Para las vocales restantes, se observa un efecto similar.

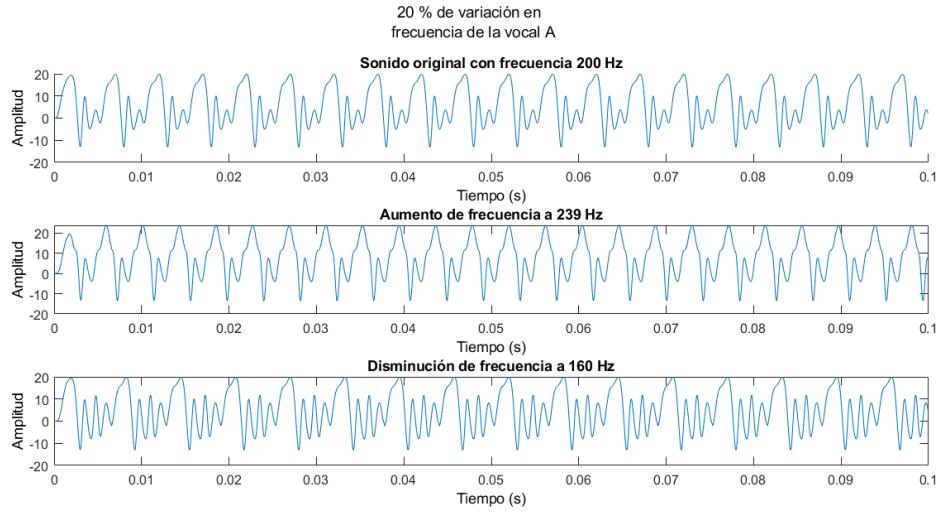


Figura 38: Resultados de utilizar el método **PSOLA** para la vocal A.

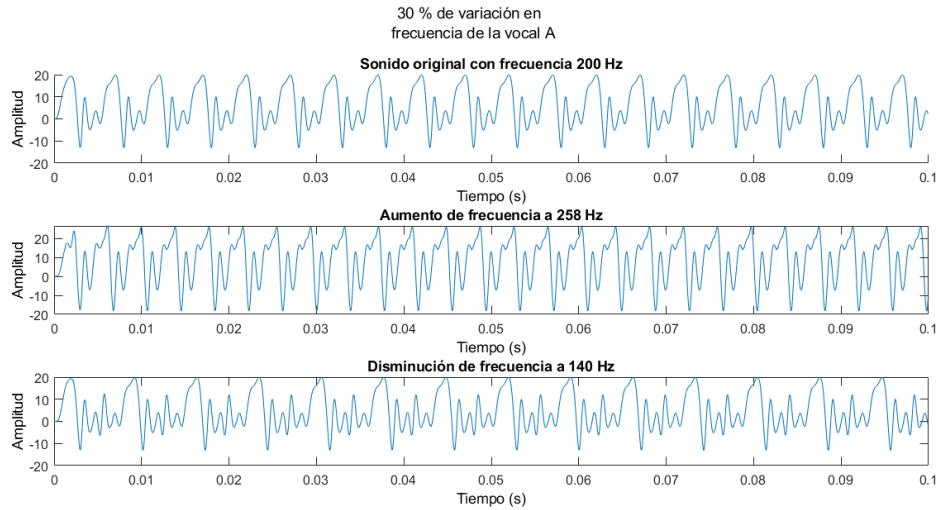


Figura 39: Resultados de utilizar el método **PSOLA** para la vocal A.

14. Ejercicio 13

Por último, se pide implementar el método **PSOLA**, para todo el audio. Para el cuál se utiliza el algoritmo implementado en el ejercicio anterior, el cuál se aplica para cada segmento obtenido en la sección 2.

En este caso, como cada periodo varía significativamente respecto al resto, se agranda la ventana a una longitud total de 2 períodos, de esta forma cada segmento tiene más información de la evolución de cada periodo en el tiempo. Luego para evitar que se superponga esta información nueva, también se agranda la longitud de muestras atenuadas a los costados de la ventana.

Finalmente se obtiene, Por último, se pide implementar el método **PSOLA**, para todo el audio.

Para el cuál se utiliza el algoritmo implementado en el ejercicio anterior, el cuál se aplica para cada segmento obtenido en la sección 2.

En este caso, como cada periodo varía significativamente respecto al resto, se agranda la ventana a una longitud total de 2 períodos, de esta forma cada segmento tiene más información de la evolución de cada periodo en el tiempo. Luego para evitar que se superponga esta información nueva, también se agranda la longitud de muestras atenuadas a los costados de la ventana.

Finalmente, se realiza la variación de la frecuencia para el audio completo.

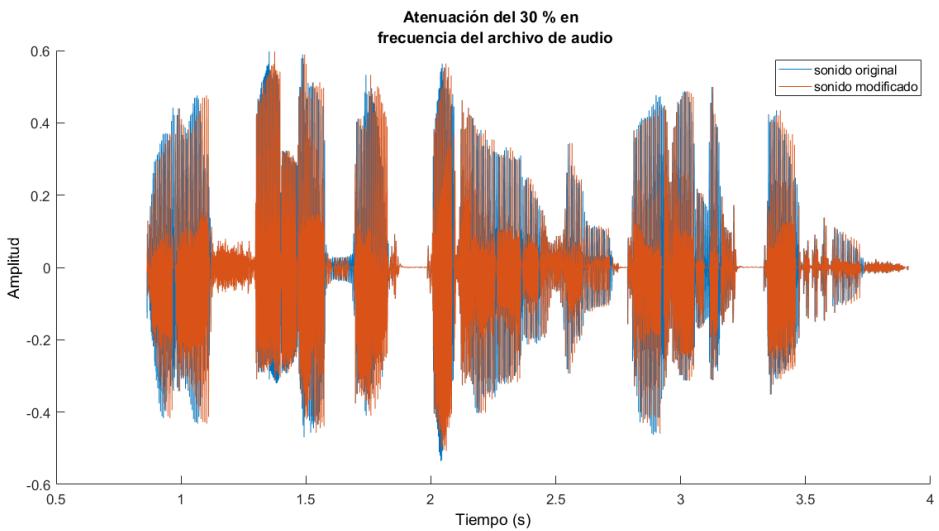


Figura 40: Superposición del audio original, y el audio atenuado en frecuencia un 30 %.

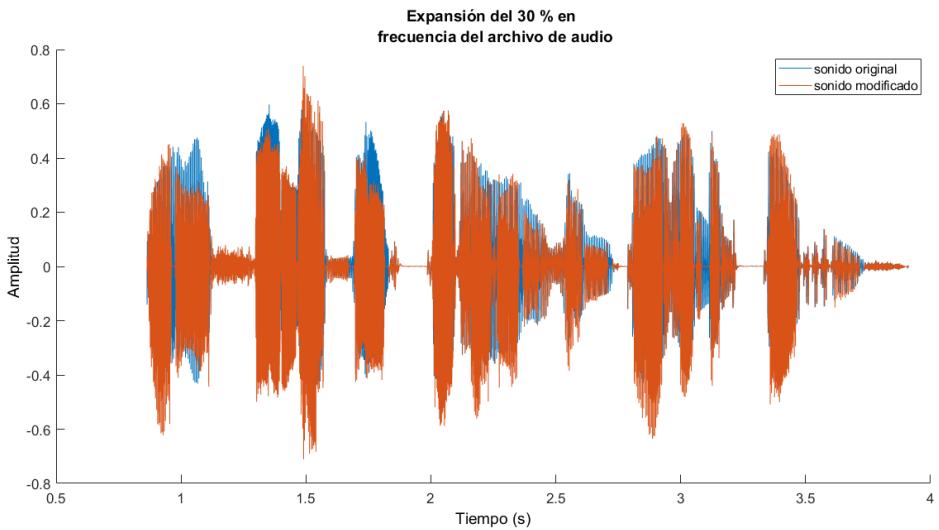


Figura 41: Superposición del audio original, y el audio expandido en frecuencia un 30 %.

Al escuchar el audio, el sonido ahora se escucha más robotico, y es posible notar la presencia de los ceros que se observan en las figuras 40 y 41 que se agregan al final de cada fonema. Pero se nota claramente el cambio de frecuencia en ambos casos y es posible identificar que se trata de la

misma frase. Además, como se observa, efectivamente el nuevo audio tiene la misma duración y velocidad que el original.

A continuación se observan los resultados del algoritmo para los otros dos casos.

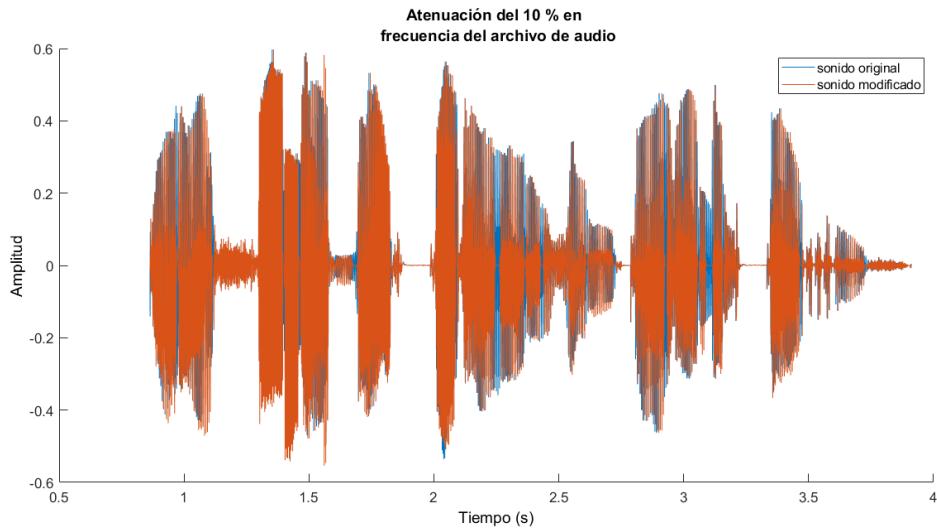


Figura 42: Audio atenuado un 10 %

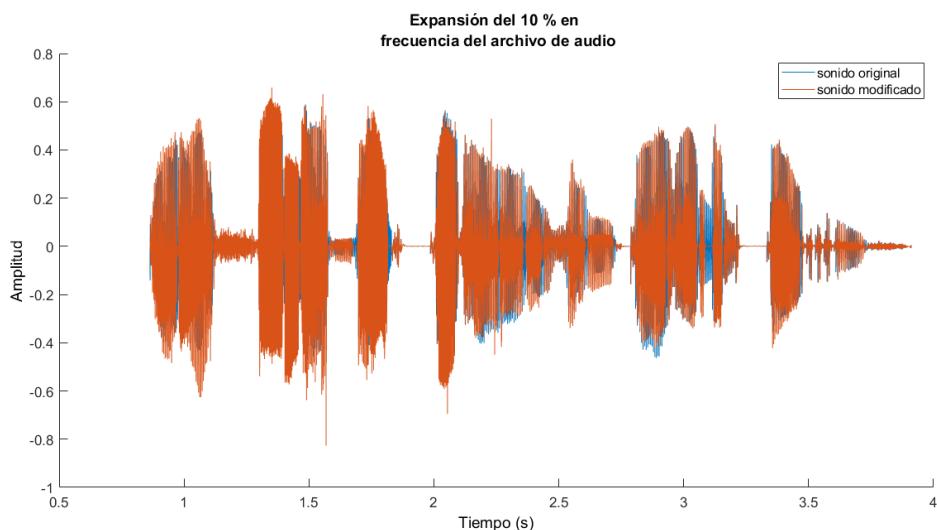


Figura 43: Audio expandido un 10 %

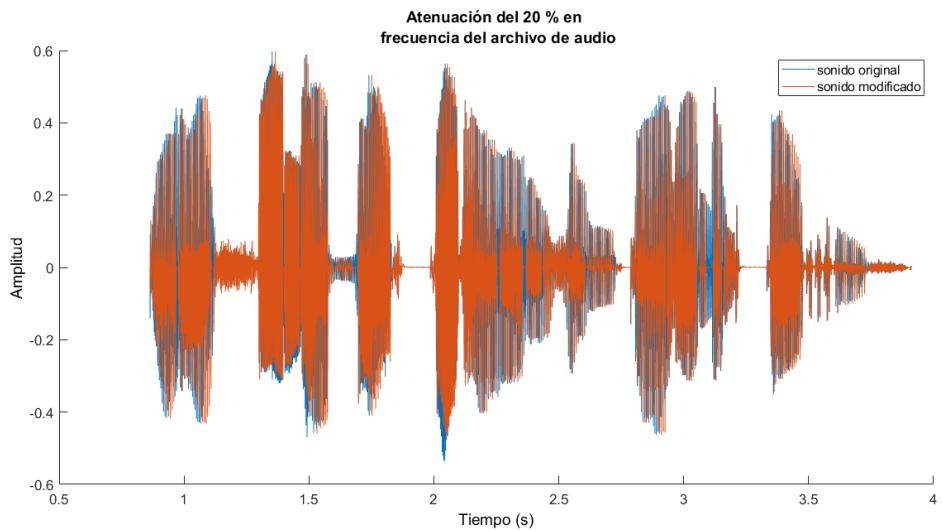


Figura 44: Audio atenuado un 20 %

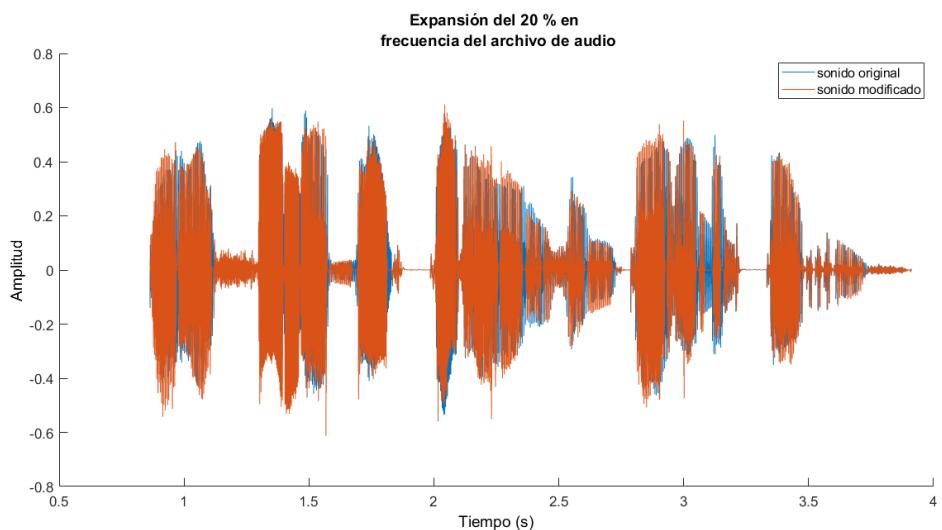


Figura 45: Audio expandido un 20 %