

COMPARACIÓN RSCHI VS MMF

Miguel Ángel López Di Benedetto

Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina

email: lopezmiguelangel@live.com

Se presenta el desarrollo de un script en R que procesa respuestas al impulso suavizando la misma por un filtro de MMF o por el método de integral inversa de Schroeder. Los resultados muestran que hay una tendencia a que con el primer método se obtengan TR mayores. Es de esperar que, este efecto sea más notable mientras más cortas sean las respuestas al impulso, por la propia respuesta del MMF. En promedio, considerando el resultado en global y por filtros de banda de octava, las diferencias de TR para el EDT es de $0,155 \pm 0,219ms$, para el TR20 es de $0,077 \pm 0,110ms$, y para el TR30 es de $0,040 \pm 0,112ms$ en favor del MMF por sobre la integral inversa.

Keywords: respuesta al impulso, filtro de media móvil, integral inversa de Schroeder, tiempo de reverberación.

1. Introducción

El tiempo de reverberación es el parámetro acústico que nunca se deja de aclarar al caracterizar una sala. Definido como el tiempo que tarda en disminuir 60 dB la energía acústica una vez se cesó el estímulo, conlleva la dificultad de que para ser evaluado la fuente debe ser capaz de generar un L_p 60 dB por sobre el piso de ruido en el espectro de interés.

En conjunto con métodos para poder medir este parámetro, también se desarrollaron otros como TR30, TR20 y EDT. Para complementar, EDT350, EDTt, C50, entre otros, que dan una noción de como es la curva de decaimiento hasta que la energía acústica disminuye 60 dB.

Este trabajo tiene por objetivo el desarrollo de un software para el análisis de RI mediante MMF y la integral inversa de Schroeder [1] para encontrar diferencias en sus resultados. El software desarrollado en R analiza mediciones mediante LSS [2] y da la posibilidad de obtener los resultados por ambos métodos en un archivo .xlsx resumiendo todos las RI analizadas.

2. Desarrollo

Se desarrolla un script en R utilizando Shiny Apps para poder analizar respuestas al impulso y obtener los parámetros RT30, RT20 y EDT.

Para su uso, el programa solo requiere que se le ingrese la carpeta donde se encuentra el filtro inverso y el resto de los archivos .wav se asumen son mediciones LSS. Antes de proceder, se puede elegir el método de cálculo, que puede ser por MMF o por la integral inversa de Schroeder.

2.1 UI

La UI consta de tres tabs o páginas. La primera denominada 'Sine Sweep generation' está disponible para que el usuario genere los archivos y proceder a realizar las mediciones de RI. El input para evitar el ingreso de datos erróneos son tres 'sliders' con datos ingresados por default, y que pueden

ser modificados. Uno para seleccionar el rango de frecuencias de trabajo del LSS, El segundo define la duración del mismo, con un mínimo obligado de 30 segundos. El tercero permite seleccionar la frecuencia de sampleo. Por último debe ingresar el directorio de trabajo y dar click sobre el botón 'Generate sine sweep'. El programa procesará la información ingresada y generara en la carpeta de trabajo el LSS, el filtro inverso y un archivo auxiliar con los datos ingresados.

En la segunda pantalla el usuario debe ingresar en el único cuadro de dialogo que va a encontrar el directorio donde se encuentra el filtro inverso y las mediciones realizadas. Al dar click sobre 'Load directory' se verificará la existencia de estos archivos, y se puede proceder al análisis de los mismos.

En el tercer y último tab, el usuario puede elegir si quiere procesar los archivos por tercio o por banda de octava, si quiere generar los gráficos, si quiere guardar las RI obtenidas (con la opción de solo generar las RI's sin ningún análisis, útil para analizar la RI en Aurora). Si los TR son muy cortos, puede elegir generar gráficos con una escala de tiempo reducida a 2.5 s., sino el predeterminado es de 5 s. Y, por último, elegir si quiere procesar la información por el método MMF o por la integral de Schroeder. En el apéndice se puede observar las figuras A1, A2 y A3 los tres tabs.

Una vez analizadas todas las RI, la salida de este script consta de un archivo .xlsx donde en cada hoja se encuentran los resultados de cada medición disponible en la carpeta de trabajo. Y además se adjuntan las hojas donde se encuentran los promedios y los desvíos estándar de todas las ellas, como se observa en la figura 1.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Global	31.5	63	125	250	500	1000	2000	4000	8000	16000	
2	EDT	1.190	1.295	0.937	1.039	1.263	1.200	1.129	1.210	1.131	1.166	1.178
3	RT20	1.282	1.302	1.131	1.183	1.386	1.215	1.161	1.261	1.187	1.286	1.275
4	RT30	1.295	1.643	1.188	1.298	1.428	1.290	1.181	1.262	1.192	1.294	1.289
	Average	Standard deviation	EW 1.02_04-02.wav	EW 10.02_04-02.wav	EW 11.02_04-02.wav	EW 12.02_04-0 ...	+	:				

Figura 1: Ejemplo del archivo .xlsx generado.

Además, si el usuario lo requiere, se crean los gráficos que contienen las curvas suavizadas de la energía para realizar los cálculos. Como muestra la figura 2:

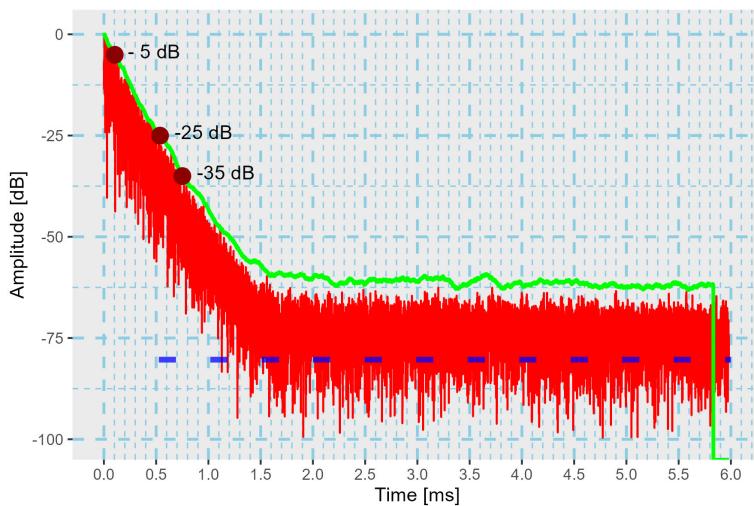


Figura 2: Global.jpg, ejemplo de los gráficos generados.

2.2 Pre procesamiento

Obtención de la RI

Como los archivos ingresados son mediciones de LSS, primero es necesario obtener la RI medida. Para ello se realiza la convolución en frecuencia de cada medición con su filtro inverso:

```

# Carga del filtro inverso.
fInv    <- readWave("filtro inverso.wav")@left
Fs2      <- readWave("filtro inverso.wav")@samp.rate
# La combinación do.call y lapply permiten el procesamiento a través
# de la lista de nombres de los archivos en la carpeta. Este procesamiento
# en paralelo, al contrario del secuencial (bucle for o while) permite el
# procesamiento aprovechando los hilos (threads) disponibles en la
# computadora.
RI <- NULL
RI <- do.call(cbind, lapply(fileNames[, 1], function(filename) {
  aux <- readWave(filename)
  signal <- aux@left
  Fs1 <- aux@samp.rate
  if (Fs2 != Fs1) { signal <- signal::resample(signal, Fs1, Fs2) }
  # Zero padding si es que una señal es corta. Puede suceder que la
  # medición sea más larga que el filtro inverso.
  max_length     <- max(length(fInv), length(signal))
  fInv_padded   <- c(fInv, rep(0, max_length - length(fInv)))
  signal_padded <- c(signal, rep(0, max_length - length(signal)))
  # La función convolve en R calcula la convolución a través de la FFT
  # en el dominio de la frecuencia.
  RI_aux <- convolve(fInv_padded, rev(signal_padded), conj = TRUE)
  # Ubico el máximo de la RI en 10 ms.
  RI_aux <- circshift(RI_aux, -which.max(abs(RI_aux))+Fs*0.01)
  return(RI_aux)
}))
```

RI filtrada por banda

Como se encontraron problemas con los filtros en bajas frecuencias, principalmente la banda de 31.5 Hz y de 63 Hz, en los paquetes 'signal' y 'tuneR', se desarrolló la función para aplicar filtros pasa banda a la señal.

```

filtradoBanda <- function(senal, Fs, fIni, fFin) {
  # Genero el vector de frecuencias con el largo de la señal y Fs.
  freqs <- c (
    seq(0, Fs/2, length.out=length(senal)/2),
    seq(Fs/2, 0, length.out=length(senal)/2)
  )
  total_length <- length(senal)
  # Para evitar la cola de energía que genera la convolución, se agregan
  # 2 segundos a la señal, que luego serán descartados.
  senal    <- c(senal, rep(0, Fs*2))
  filtro   <- rep(0, length(freqs))           # Init 0's en el filtro.
  filtro[freqs >= fIni & freqs <= fFin] <- 1 # Agrego la RI del mismo.
  # Suavizo el filtro para minimizar la cola energética.
  filtro   <- zoo::rollmean(filtro, k = 4, fill=0)
  # Obtengo su respuesta al impulso.
  filtro   <- Re(fft(filtro, inverse=TRUE)/length(filtro))
  # Lmito su respuesta al impulso que asumo es menor a 200 ms (5 Hz).
  filtro   <- filtro[1:(0.2*Fs)]
  # Zero padding para poder ejecutar la convolución.
  filtro   <- c(filtro, rep(0, length(senal)-length(filtro)))
  # Convolución (la realiza en el dominio de la frecuencia).
  output   <- convolve(filtro, rev(senal), conj = TRUE)
  # La salida debe tener el mismo largo que la señal original.
  output   <- output[1:total_length]
```

```
    return (output)
}
```

2.3 Tiempo de Reverberación

En análisis de señales, el filtro de media móvil (MMF) es de las técnicas más sencillas y simples para suavizar una señal. Consiste en ir calculando muestra a muestra el promedio de una cantidad discreta de muestras que pueden ser las muestras siguientes y/o anteriores a la muestra actual. Principalmente se utiliza para quitar ruido de una señal. Su respuesta en frecuencia se puede aproximar a la de un pasa bajos. Es de esperar que no sea de los filtros más precisos que se pueden aplicar a una señal para retirar ruido de la misma, pero compensa por su sencillez y su fácil implementación.

Sin importar el método de cálculo elegido, primero se aplica directamente sobre la RI para suavizarla un MMF. Como la frecuencia de corte de este tipo de filtro se puede aproximar según eq. 1, cada señal será suavizada considerando su frecuencia máxima:

$$F_{\text{cut}} = \frac{Fs}{2 \cdot F_{\text{máx}}} \quad (1)$$

Esto se realiza porque señales de baja frecuencia van a tener un decaimiento muy distinto al que se puede encontrar en una señal con componentes de medias a altas frecuencias.

Luego de ser suavizada, se puede manipular la señal y transformarla a energía, pasando a una magnitud en dB. En esta instancia el vector de datos ya no representa la señal en el tiempo, sino su energía, por lo que el suavizado que se realice a partir de este punto solo tiene por finalidad generar una curva que represente su decaimiento.

MMF

Se obtuvieron buenos resultados aplicando un MMF de ventana 150 ms al analizar RI's cuyo RT era del orden de 1 segundo.

También se consideró la utilización de otros filtros, como uno de mediana móvil, y otros con ponderaciones como la exponencial y la triangular, pero su utilización requería desarrollar e implementar dicha herramienta. Lo que implica un costo de horas de trabajo mayor, y los resultados que se pudieron ver en un análisis preliminar eran similares a los obtenidos con el MMF que no requería ser desarrollado.

```
# Cálculos por el método MMF.
Fs <- Fs / resample
if(Fmax>=Fs/8)
{
  Fmax <- Fmax/4 # Sanity check para frecuencias muy bajas.
}
# El primer filtro depende de Fmax. Señal en dominio del tiempo.
sgnl <- as.numeric(
  zoo::rollmean(sgnl, k = round(Fs/(2*Fmax)),
                align="left")
)
# Convierto a energía.
smoothed02 <- 10*log10((sgnl^2) / sum(sgnl^2))
# Aplico downsample a la señal energetica.
smoothed02 <- smoothed02[seq(1, length(smoothed02), resample)]
smoothed02 <- smoothed02 - max(smoothed02)
# Invierto la señal antes de aplicar MMF de 150 ms.
smoothed03 <- rev(smoothed02)
smoothed03 <- zoo::rollmean(smoothed03,
                           k = 0.15*Fs,
```

```

            align = "right", fill = -Inf)
# Normalizo la señal para que 0 dB sea el máximo.
smoothed03 <- smoothed03
smoothed03 <- rev(smoothed03)
smoothed03 <- smoothed03[seq(which.max(smoothed03),
                           length(smoothed03) )]
smoothed03 <- smoothed03 - max(smoothed03)

```

El principal problema de este método es encontrar cual es la ventana temporal (150 ms en este caso) que devuelve una curva que represente correctamente la energía. Este problema se puede observar al ver que el suavizado la pendiente del decaimiento, y siempre termina dando por resultado un TR que es mayor al que representa la curva energética. La figura 3 muestra este efecto:

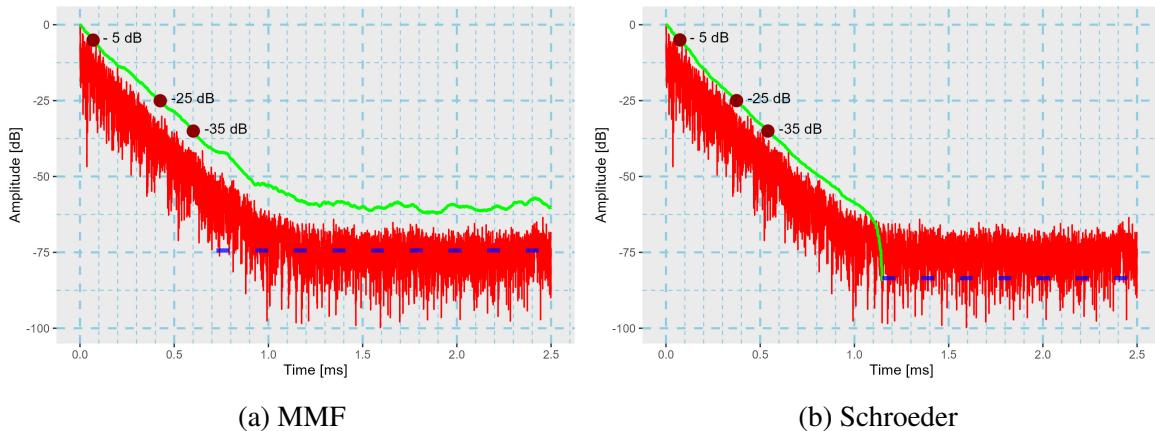


Figura 3: MMF vs Schroeder

Integral inversa de Schroeder

El método propuesto por Schroeder se convirtió en el estándar para obtener la curva del decaimiento energético de una señal impulsiva. Definida por la ec. 2:

$$E_{(t)} = \int_t^{\infty} p_{(t)}^2 dt \quad (2)$$

Es sencillo ver que esta ecuación da por resultado la cantidad de energía que queda por delante del tiempo t . Ec. 3 muestra la igualdad en escala logarítmica:

$$E_{(t)} = 10 \log_{10} \left(\frac{\int_t^{\infty} p_{(t)}^2 dt}{\int_0^{\infty} p_{(t)}^2 dt} \right) \quad (3)$$

La principal ventaja que tiene por sobre el método MMF es que no es necesario hacer ningún tipo de ajuste que contenga un sesgo en los cálculos. No es necesario ajustar el ancho de la ventana del filtro, ni ningún otro parámetro.

La dificultad de la implementación de este método es que es necesario encontrar el piso de ruido para establecer los límites de integración. No encontrar correctamente el piso de ruido hace que la energía que aparece al final de la curva sea muy grande. Esto puede dar por resultado un TR mayor al real, similar al visto en el caso del MMF pero con un efecto mayor. La señal por el método de Schroeder se procesa de la siguiente manera:

```

# Transformada de Hilbert. Señal en el dominio del tiempo.
smoothed01 <- Re(seewave::env((sgnl), Fs, plot = FALSE))
# Downsample.

```

```

smoothed01 <- sgnl[seq(1, length(sgnl), resample) ]
Fs           <- Fs/resample
smoothed02 <- 10*log10((smoothed01^2)/ sum(smoothed01^2))
# Normalizado a 0 dB.
smoothed02 <- smoothed02 - max(smoothed02)
# findNoiseFloor se encarga de encontrar la muestra en la que la
# señal llega al piso de ruido, establece los límites de integración.
new_min <- findNoiseFloor(smoothed02, Fs)
# Calculo de la integral de Schroeder
smoothed03 <- smoothed01[1:(new_min)]
smoothed03 <- rev(10*log10(cumsum(rev(smoothed03^2)/sum(smoothed03^2))))
smoothed03 <- smoothed03[seq(which.max(smoothed03), length(smoothed03))]
smoothed03 <- smoothed03 - max(smoothed03)

```

Smoothed03 es en ambos casos la señal suavizada que se puede observar en verde en la figura 3. En el anexo se encuentran las comparaciones para una misma RIR todos los gráficos obtenidos por banda de octava en las figuras desde A4 a A14.

3. Resultados

Se evaluaron 100 mediciones de LSS de 3 segundos de duración. Todas las mediciones se realizaron bajo las mismas condiciones y el RT30 promedio obtenido es de 0.7 s. Se puede observar en la tabla 1 los resultados de cada método, y las diferencias globales y por banda de octava:

Tabla 1: Media

MMF	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.68	0.92	0.82	0.99	0.87	0.76	0.60	0.63	0.63	0.64	0.67
RT20	0.76	0.92	0.97	1.03	0.90	0.79	0.61	0.70	0.73	0.72	0.73
RT30	0.74	0.82	0.85	0.92	0.85	0.76	0.65	0.72	0.72	0.74	0.74
Sch.	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.49	0.66	0.66	0.60	0.66	0.62	0.64	0.52	0.57	0.58	0.50
RT20	0.67	0.88	0.88	0.90	0.89	0.66	0.63	0.68	0.62	0.55	0.64
RT30	0.70	0.84	0.84	0.88	0.85	0.75	0.68	0.66	0.61	0.57	0.68
Dif.	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.18	0.26	0.17	0.40	0.22	0.13	-0.04	0.11	0.06	0.06	0.17
RT20	0.09	0.04	0.09	0.13	0.01	0.13	-0.02	0.02	0.11	0.17	0.09
RT30	0.04	-0.02	0.00	0.04	0.00	0.02	-0.02	0.06	0.12	0.17	0.05

Valores positivos en la diferencia indican que resultados del método MMF es mayor al método de Schroeder. Se puede observar que hay una clara tendencia a que el de MMF de por resultados TR mayores al método de Schroeder. Un efecto que era esperado, ya que a los cálculos del propio TR, se está agregando la respuesta del filtro. Además, dicha diferencia está en concordancia con la duración de la ventana del filtro MMF elegido, que es de 0.15 s. Una curiosidad que no se encuentra explicación es que la relación en global RT20/RT30 es similar a la relación EDT/RT20.

Al analizar los desvíos de ambos métodos se puede observar que es menor en el método de Schroeder. Nuevamente, valores positivos indican que el método MMF tiene un desvío estándar mayor, como indica la tabla 2:

Tabla 2: Desvío estándar

MMF	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.018	0.015	0.016	0.028	0.015	0.025	0.015	0.024	0.016	0.015	0.016
RT20	0.015	0.011	0.014	0.013	0.012	0.018	0.012	0.018	0.018	0.016	0.027
RT30	0.008	0.010	0.012	0.011	0.011	0.011	0.010	0.011	0.011	0.009	0.012
Sch.	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.010	0.005	0.005	0.011	0.003	0.002	0.001	0.012	0.010	0.011	0.009
RT20	0.007	0.006	0.006	0.007	0.009	0.006	0.004	0.009	0.007	0.009	0.009
RT30	0.007	0.003	0.003	0.008	0.004	0.003	0.006	0.003	0.007	0.009	0.008
Dif.	Glob.	31.5	63	125	250	500	1 k	2 k	4 k	8 k	16 k
EDT	0.008	0.010	0.011	0.017	0.012	0.023	0.014	0.012	0.006	0.004	0.007
RT20	0.008	0.005	0.008	0.006	0.003	0.012	0.008	0.009	0.011	0.007	0.018
RT30	0.001	0.007	0.009	0.003	0.007	0.008	0.004	0.008	0.004	0.000	0.004

4. Conclusiones

Se pudieron desarrollar ambos métodos propuestos en R. El diseño de los filtros pasa banda propuestos da buenos resultados, aunque deben ser evaluados con mayor profundidad ya que están sobredimensionados al ser obtenidos a partir de una respuesta en frecuencia de tipo brickwall. Otros métodos de diseño de filtros puede también ser tomado en cuenta, como por ejemplo a partir de sus polos y ceros.

Ambos métodos de cálculo no tienen diferencias notables, comparándolos con el JND [3] de los parámetros de TR, pero aún así los tiempos obtenidos tienden a ser mayores para el método MMF en la mayoría de los casos y esta diferencia esta ligada al largo de la ventana utilizada. La ventaja de este método es su fácil implementación, y su principal desventaja es la dificultad de ajustar la ventana de tiempo de dicho filtro, en particular para bajas frecuencias. La implementación de otro tipo de ventana y filtro para bajas frecuencias también puede ser objeto de estudio.

En el anexo se adjuntan figuras de los resultados para 31.5 Hz (fig. A4), 63 Hz (fig. A5) y 125 Hz (fig. A6) donde se aprecia la dificultad de ambos métodos para aproximar a la curva de energía obtenida, especialmente en 31.5 Hz.

Por el otro lado, el método de Schroeder es consistente en sus resultados, esto se ve en el menor desvío estándar obtenido. La desventaja de este método puede recaer en la detección del piso de ruido. Un algoritmo que no es capaz de detectar la muestra en la señal alcanza el piso de ruido no va a dar buenos resultados, mientras que en el caso de MMF no era necesario ese desarrollo.

Referencias

1. M. R. Schroeder. New method of measuring reverberation time. *J Acoust Soc Am* 1, 37(3):409–412, 2005.
2. A. Farina. Advancements in impulse response measurements by sine sweeps. *J Acoust Soc Am* 1, 37(3):409–412, 2005.
3. M. G. Blevins, A. T. Buck, Z. E. Peng, and L. M. Wang. Quantifying the just noticeable difference of reverberation time with band-limited noise centered around 1000 hz using a transformed up-down adaptive method. *Proceedings of the International Symposium on Room Acoustics*, 2013.

A. Apéndice

UI

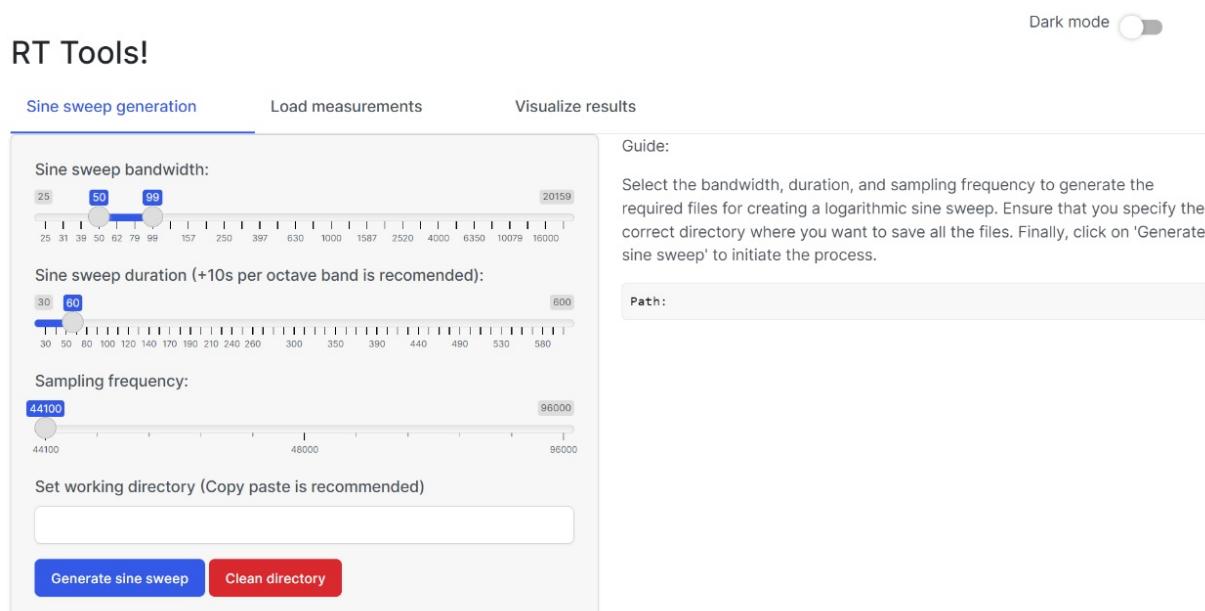


Figura A1: Generación de LSS

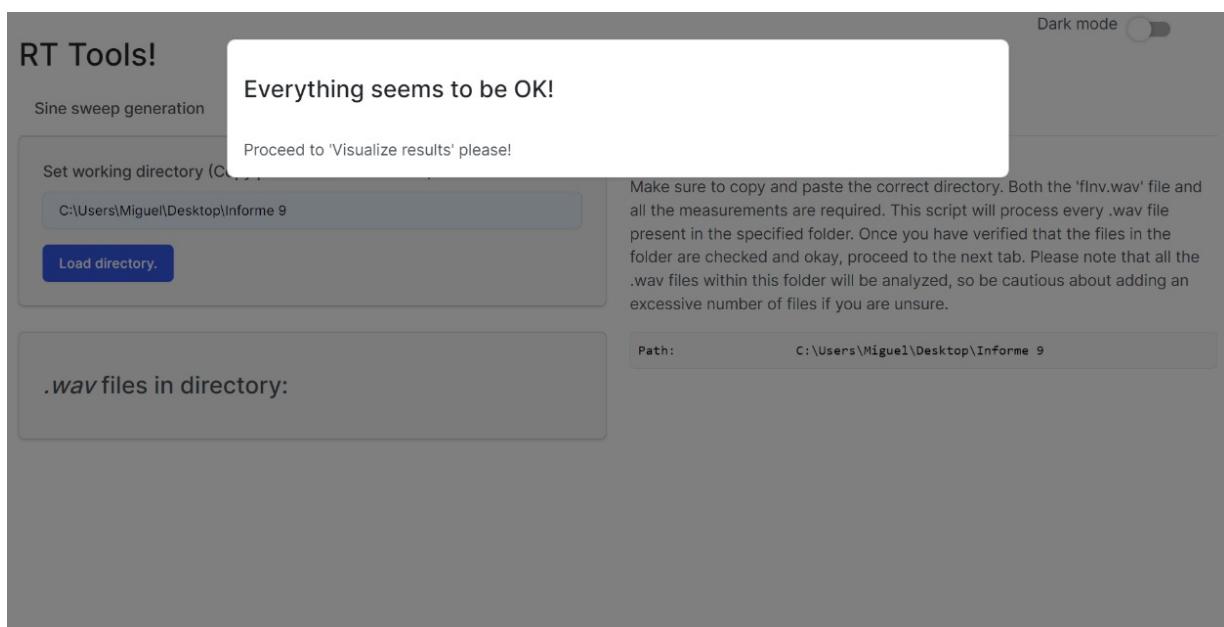


Figura A2: Carga de datos

Dark mode **RT Tools!**

Sine sweep generation Load measurements **Visualize results**

Use third Octave Band Filters
 Generate plots
 Save RI's
 Just save RIs, no analysys
 Short RI. If checked plot will be 2.5 seconds long.
 Calculate usin MMF instead of RSchl

Proceed with calculations.

.wav files loaded :

Guide:

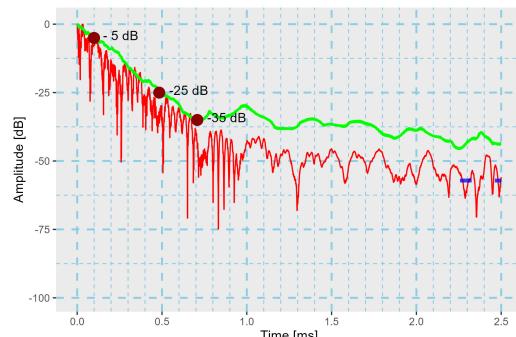
If desired, select the third-octave band filters. Please note that the computation times will be tripled when this option is selected, so it is not recommended to analyze an excessive number of files with this option enabled.

Select the 'Generate plots' option to create graphs by the selected bands. These graphs can be found in the folders created within the selected directory. The created folders have the same names as the provided files.

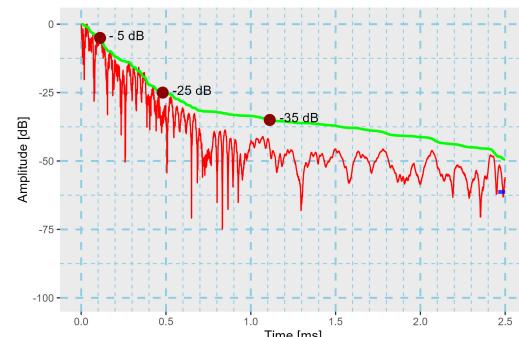
ATTENTION: Plots are 5 seconds long. If Short RI is checked the plots will be 2.5 seconds long. This option might crash the program if RI is too long.

The file 'Output.xlsx' will be generated with all the results. If only one measurement is found, Output will have a single sheet. If two measurements are found, the IACC Early parameter will also be generated. If two or more measurements are found, the average sheet and standard deviation sheet will be generated as well.

Figura A3: Procesamiento de datos

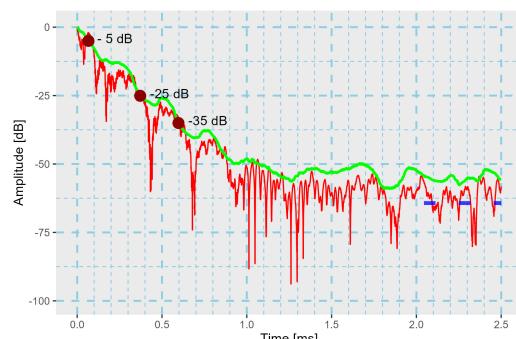
Ejemplos figuras por banda de octava para una misma medición

(a) MMF 31.5 Hz

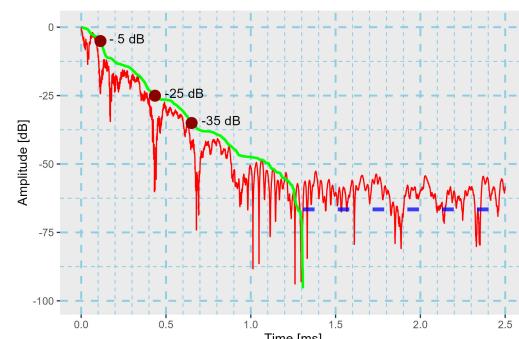


(b) SCH 31.5 Hz

Figura A4: Comparación 31.5 Hz

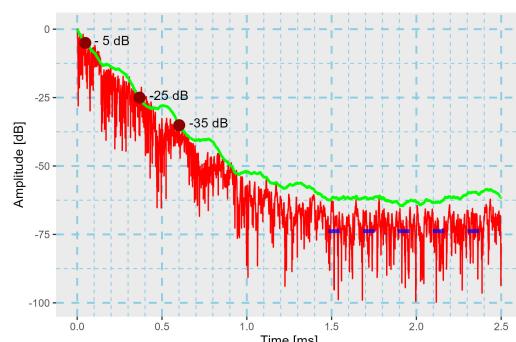


(a) MMF 63 Hz

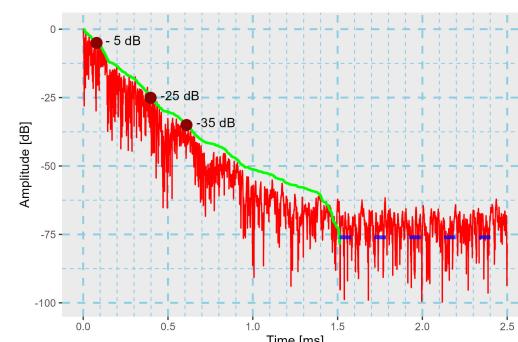


(b) SCH 63 Hz

Figura A5: Comparación 63 Hz

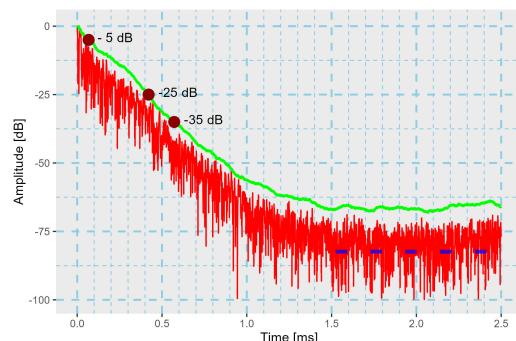


(a) MMF 125 Hz

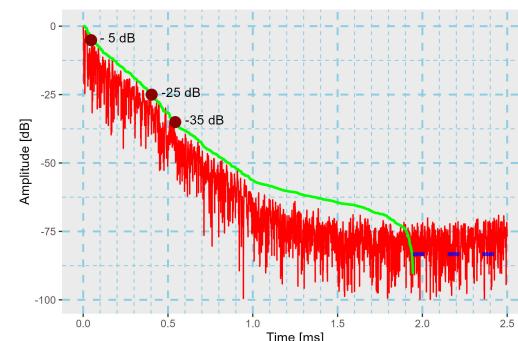


(b) SCH 125 Hz

Figura A6: Comparación 125 Hz

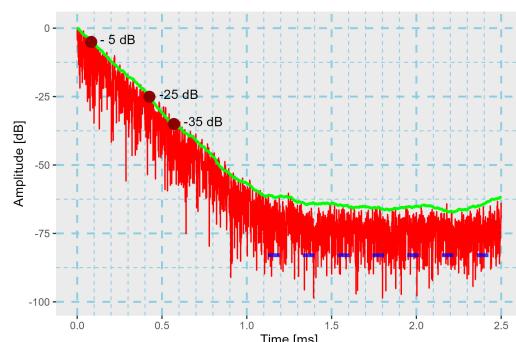


(a) MMF 250 Hz

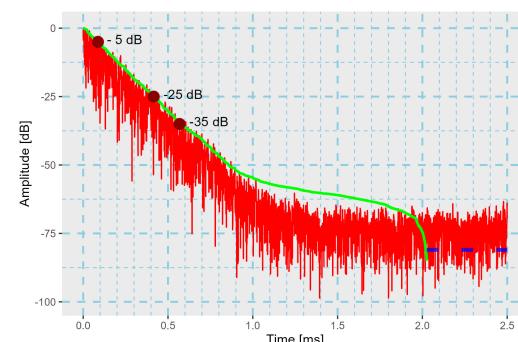


(b) SCH 250 Hz

Figura A7: Comparación 250 Hz

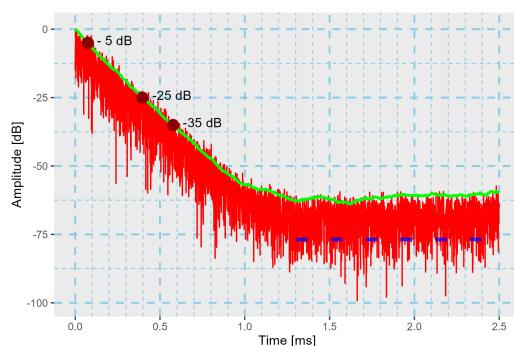


(a) MMF 500 Hz

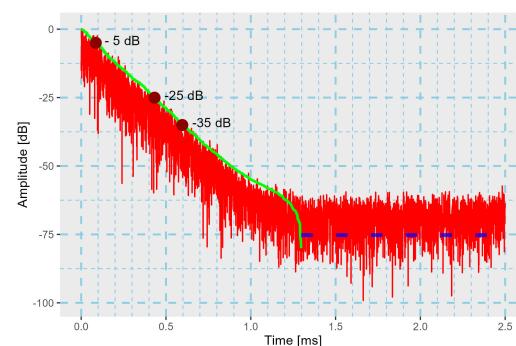


(b) SCH 500 Hz

Figura A8: Comparación 500 Hz

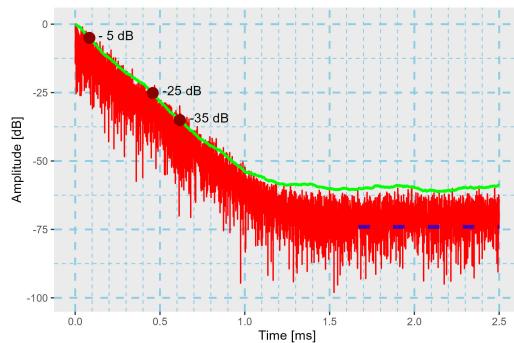


(a) MMF 1000 Hz

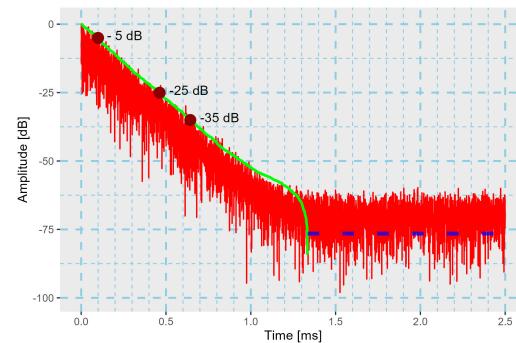


(b) SCh 1000 Hz

Figura A9: Comparación 1000 Hz

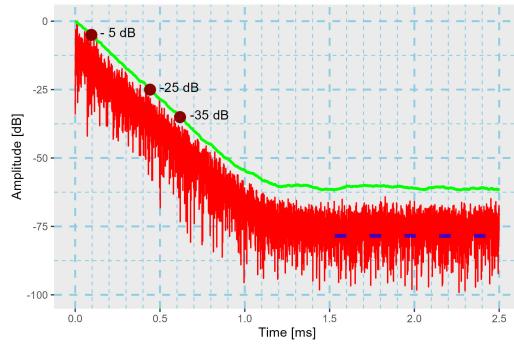


(a) MMF 2000 Hz

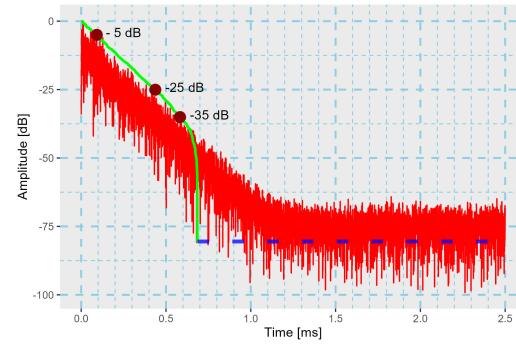


(b) SCh 2000 Hz

Figura A10: Comparación 2000 Hz

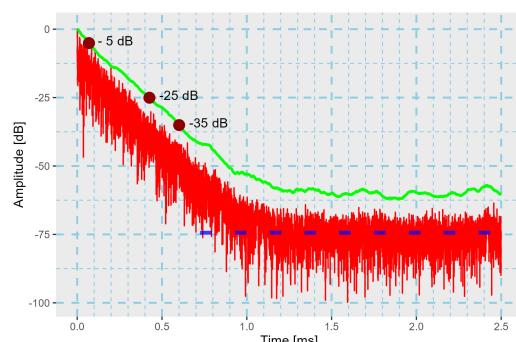


(a) MMF 4000 Hz

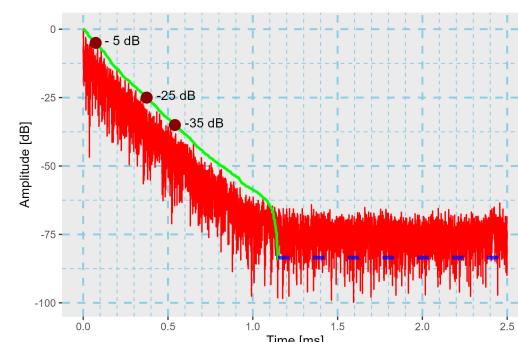


(b) SCh 4000 Hz

Figura A11: Comparación 4000 Hz

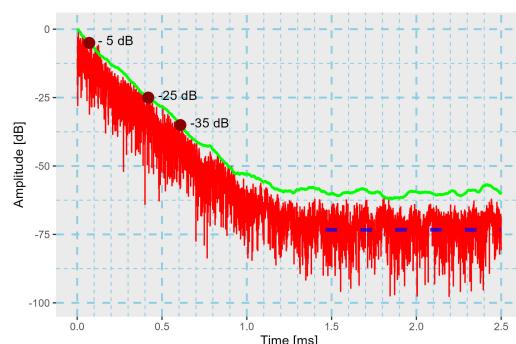


(a) MMF 8000 Hz

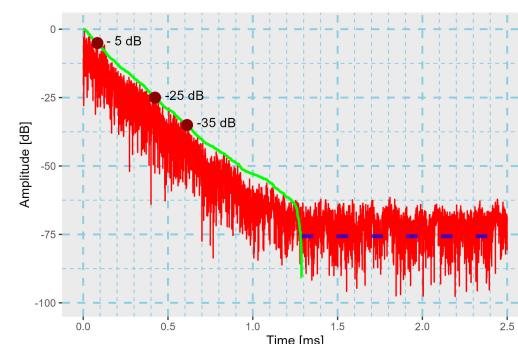


(b) SCh 8000 Hz

Figura A12: Comparación 8000 Hz

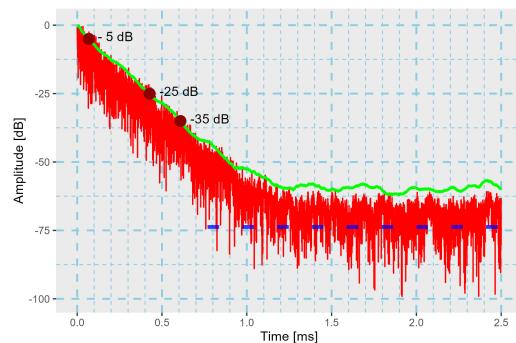


(a) MMF 16000 Hz

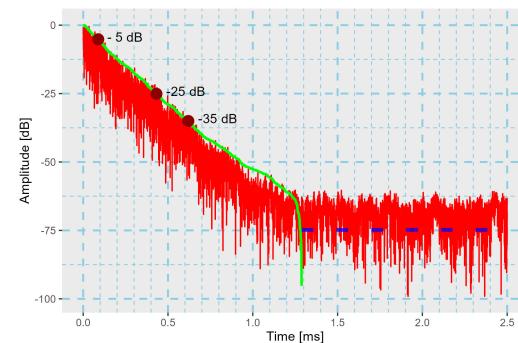


(b) SCh 16000 Hz

Figura A13: Comparación 16000 Hz



(a) MMF Global Hz



(b) SCh Global Hz

Figura A14: Comparación Global