

PlotTool - Manual de uso

Juan Sbruzzi

Leandro Yiu

Agosto 2025

Índice

1. Instalación	2
2. Utilización básica	2
2.1. Creación de curva a partir de dataset arbitrario	3
2.2. Sobre escalas y desplazamientos	3
2.3. Opciones de ploteo	4
2.4. Exportación de gráficos	5
2.5. Sobre el formato de los archivos .csv	6
3. Otros tipos de <i>datasets</i>	6
3.1. Manejo de funciones transferencia	6
3.2. Manejo de archivos de simulación de Spice	9

Lista de figuras

1. Ventana de inicio del PlotTool.	2
2. Comparación entre configuraciones de altura en la visualización de plots.	3
3. Interfaz de configuración de opciones de ploteo.	4
4. Ejemplo de utilización de expresiones de \LaTeX en título y labels.	4
5. Opciones adicionales de ploteo.	5
6. Ejemplo de visualización de archivo .pdf en \LaTeX	5
7. Ejemplo de archivo .csv compatible con PlotTool.	6
8. Cuadro de diálogo para creación de <i>dataset</i> a partir de función transferencia.	7
9. Ejemplo de uso de función <i>Show poles and zeros</i> para un <i>dataset</i> creado a partir de una función transferencia.	7
10. Ejemplo de ploteo de datos obtenidos de un <i>dataset</i> creado a partir de una función transferencia.	8
11. Cuadro de diálogo para configurar la respuesta temporal de una función transferencia.	8
12. Ejemplo de uso de función <i>Add time response to TF</i>	9
13. Ejemplo de archivos creados al ejecutar simulación de <i>Spice</i>	9

1. Instalación

1. Clonar [el repositorio](#).
2. Navegar en una terminal al directorio raíz del repositorio.
3. Correr `pip install -r requirements.txt` (alternativamente, pueden instalar las dependencias en un `venv` de acuerdo a lo que se explica en el `readme` del repositorio).

Desde la terminal del directorio raíz, la PlotTool se inicia con `python main.py`.

2. Utilización básica

La PlotTool trabaja con *datasets* y *datalines*.

Un *dataset* puede ser:

- Un archivo `.raw` de simulación de Spice.
- Un archivo `.csv` con datos numéricos organizados en columnas, como los que exporta el osciloscopio.
- Una función transferencia agregada directamente desde la PlotTool.

A partir de cada *dataset* se puede crear una cantidad arbitraria de *datalines*. Por ejemplo, a partir de una transferencia, se puede obtener una línea para graficar la magnitud y otra para la fase; de un archivo con mediciones se puede obtener una línea para cada canal, etc. Todas las fuentes de puntos del *dataset* están disponibles para usar como variables *x* e *y* de cada *dataline*.

El manejo de *datasets* se presenta a la izquierda de la ventana, y el de *datalines* a la derecha.

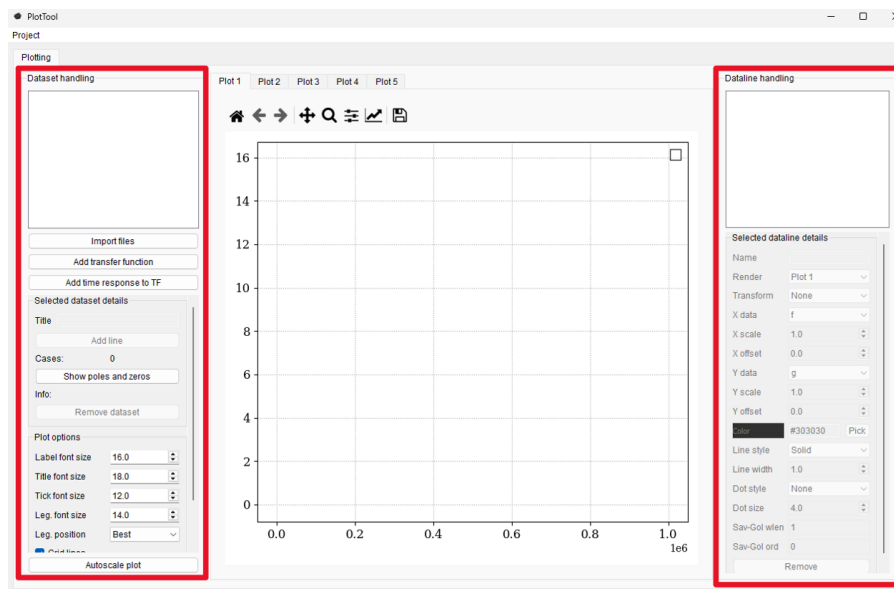
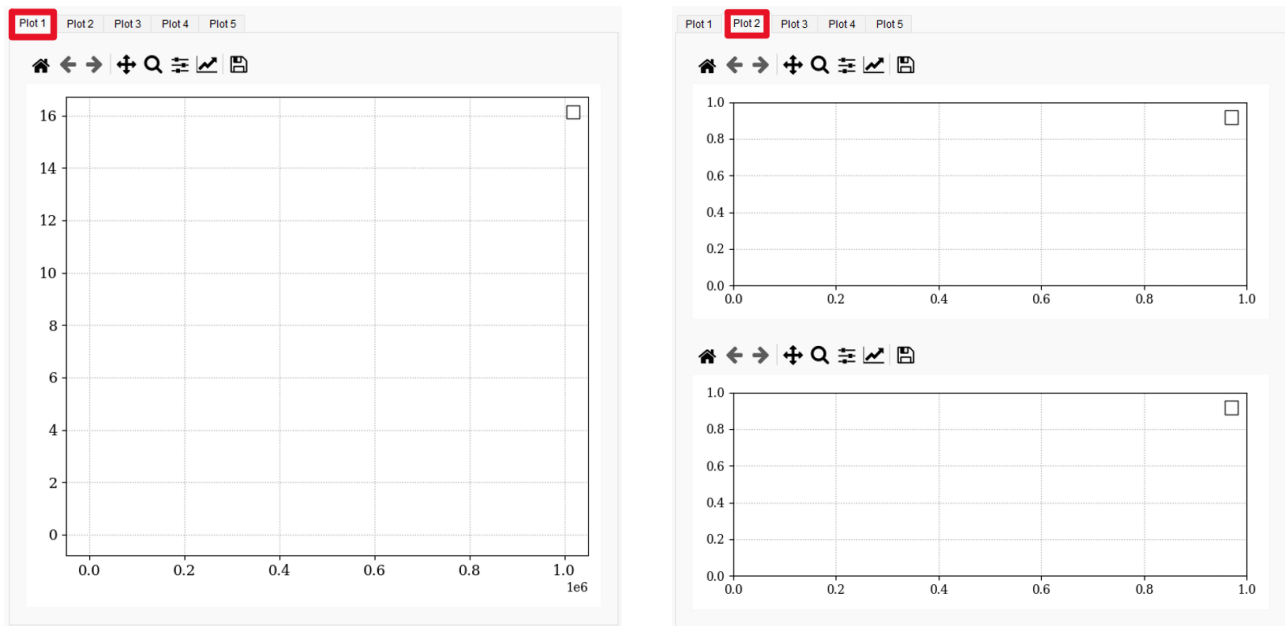


Figura 1: Ventana de inicio del PlotTool.

Cada curva se puede presentar en 1 de 7 paneles de **plot**, que a su vez pueden ser de altura completa como se observa en la *figura 2a* o 50 % en la *figura 2b*, como se demuestra en la *figura 2*. Los plots de 50 % están orientados a la realización de diagramas de módulo y fase.



(a) Plot de altura completa.

(b) Plot de 50 % de altura.

Figura 2: Comparación entre configuraciones de altura en la visualización de plots.

2.1. Creación de curva a partir de dataset arbitrario

Los pasos para representar una curva son:

1. Importar un dataset o más con el botón **Import files** o arrastrando los archivos a la ventana.
2. Seleccionar el dataset del que se quiere obtener la curva.
3. Presionar **Add line**.
4. En el panel de *datalines* se presentará la curva creada y se seleccionará automáticamente.
5. En este punto es conveniente configurar una **Name** apropiado para la curva.
6. Seleccionar el **plot** en el que se mostrará.
7. Definir los datos (columnas, en el caso de un **.csv**) a utilizar como **x** e **y** mediante **X data** e **Y Data**.
8. Aplicar transformaciones, offsets, y escalas para que la representación de los datos sea adecuada.
9. Aplicar transformaciones estéticas de color, tipo de línea y/o vértices, grosor.

2.2. Sobre escalas y desplazamientos

- **Cuidado:** los operadores de **scale** y **offset** se aplican en ese orden ($y' = ax + b$).
- El operador **scale** es muy conveniente para corregir la representación del orden de magnitud (pasando el eje de tiempo de s a μs , o un eje de tensión de V a mV).
- El operador **offset** resulta práctico para llevar las mediciones a los puntos de referencia en la escala temporal, o para sincronizar dos mediciones de fuentes distintas (por ejemplo, obtenidas de simulación y la medición correspondiente). Esto último debe aplicarse con particular atención, utilizando una señal de referencia adecuada para la sincronización.

2.3. Opciones de ploteo

Las *labels* de cada curva se toman automáticamente de su *Name*. Las etiquetas de los ejes se pueden configurar mediante la configuración del plot nativa de `matplotlib`, disponible desde el botón indicado en la *figura 3*:

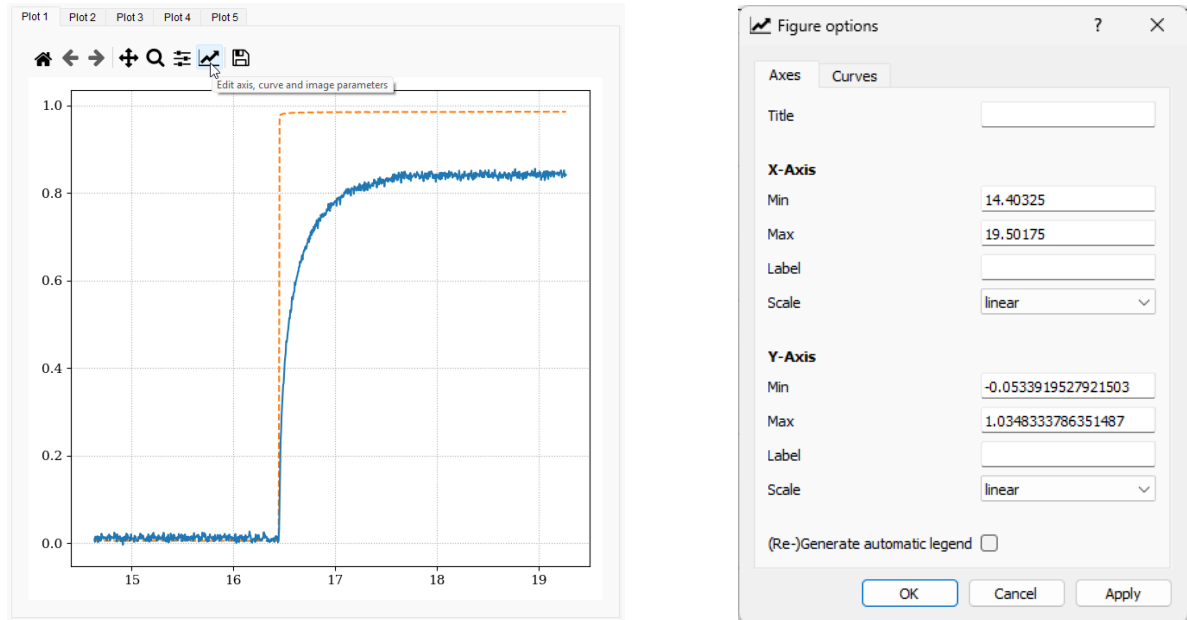


Figura 3: Interfaz de configuración de opciones de ploteo.

Este cuadro de diálogo permite además la utilización de ejes logarítmicos nativos, lo cual es altamente recomendado para gráficos que contienen curvas cuya variable independiente es la frecuencia.

Atención: tanto en el `Name` de los `dataline`, como en los `Label` y `Title` de `Matplotlib` disponibles en este cuadro de diálogo **es posible usar expresiones de L^AT_EX** encerrándolas entre signos `$`.

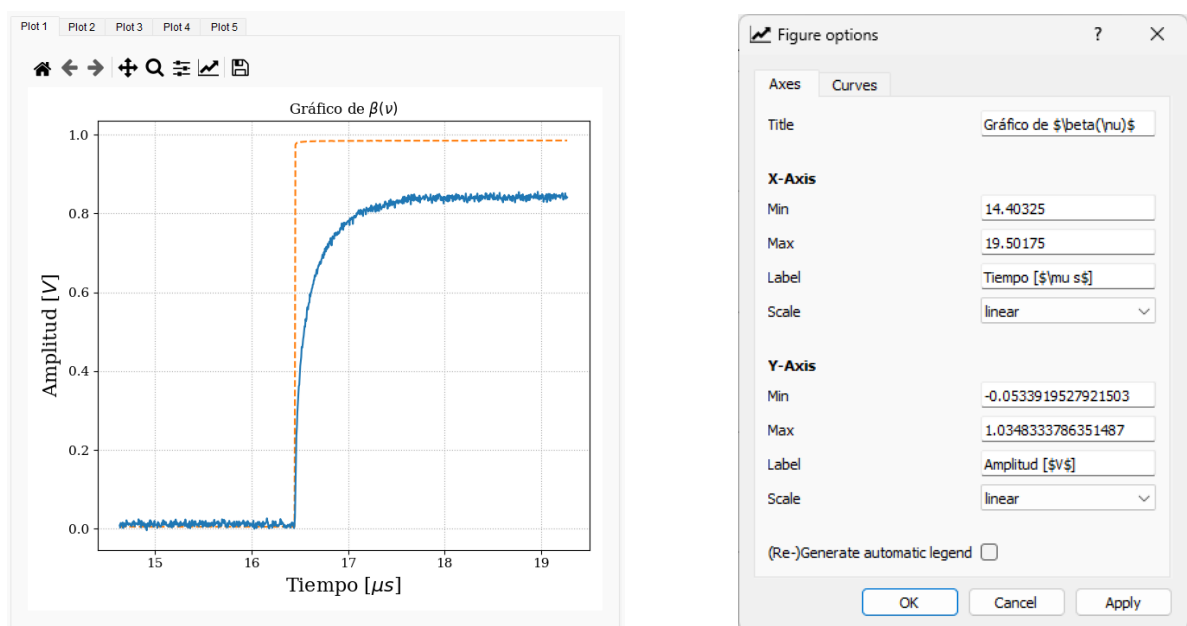


Figura 4: Ejemplo de utilización de expresiones de L^AT_EX en título y labels.

En la esquina inferior izquierda se presentan las opciones adicionales de ploteo, no disponibles

mediante la interfaz gráfica de **Matplotlib**.

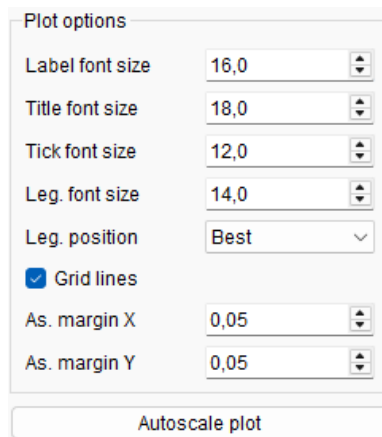


Figura 5: Opciones adicionales de ploteo.

Con ellas es posible modificar los tamaños de fuente de los **labels** (etiquetas de ejes), **title**, **tick** (marcas en los ejes), y **legend** (cuadro que contiene los nombres de las curvas). Además permite modificar la **legend position** a cualquier lugar del gráfico, activar o desactivar las líneas de grilla, y aplicar márgenes porcentuales en ambas dimensiones al tocar el botón **Autoscale plot**.

2.4. Exportación de gráficos

Se recomienda almacenar los gráficos como **.pdf** mediante las herramientas nativas de **Matplotlib**:



El **.pdf** generado se puede utilizar en un **includegraphics** de **L^AT_EX** de manera directa, produciendo un resultado como el siguiente:

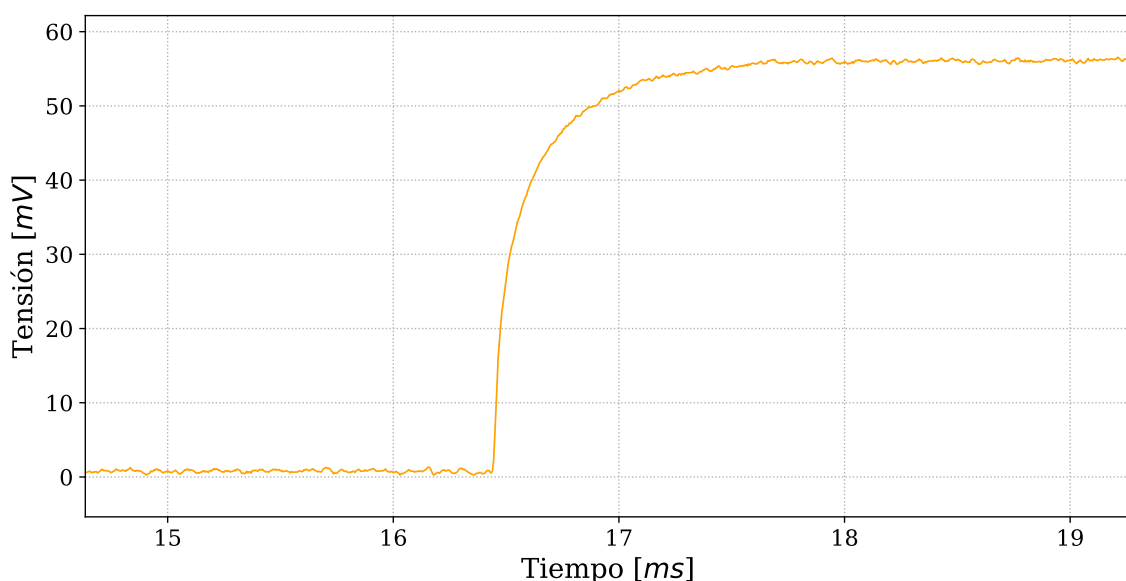


Figura 6: Ejemplo de visualización de archivo **.pdf** en **L^AT_EX**.

2.5. Sobre el formato de los archivos .csv

A la hora de importar archivos .csv a la PlotTool, se debe tener en cuenta que el programa admite un formato específico, como se muestra a continuación:

```
x-axis,1,2
second,Volt,Volt
-9.9600E-03,-174.6231E-03,-160.8040E-03
-9.9500E-03,-201.0050E-03,-177.1357E-03
-9.9400E-03,-160.8040E-03,-167.0854E-03
-9.9300E-03,-160.8040E-03,-128.1407E-03
-9.9200E-03,-146.9849E-03,-160.8040E-03
-9.9100E-03,-120.6030E-03,-136.9347E-03
-9.9000E-03,-120.6030E-03,-160.8040E-03
-9.8900E-03,-120.6030E-03,-120.6030E-03
-9.8800E-03,-80.4020E-03,-94.2211E-03
-9.8700E-03,-64.0704E-03,-120.6030E-03
-9.8600E-03,-46.4824E-03,-120.6030E-03
-9.8500E-03,-40.2010E-03,-120.6030E-03
-9.8400E-03,-66.5829E-03,-94.2211E-03
-9.8300E-03,-40.2010E-03,-80.4020E-03
-9.8200E-03,-40.2010E-03,-80.4020E-03
```

Figura 7: Ejemplo de archivo .csv compatible con PlotTool.

El archivo debe cumplir las siguientes condiciones para garantizar su correcta lectura por el programa:

- El separador de columnas debe ser la **coma** (,).
- Los valores decimales deben utilizar el **punto** (.) como separador decimal.
- Las dos primeras filas contienen, respectivamente, las etiquetas de cada columna y sus unidades.
- Las filas siguientes contienen los datos numéricos, pudiendo expresarse en notación científica (por ejemplo, E-03).

Atención: Cuando opera sobre los datos del .csv mediante otra aplicación, debe revisar que el archivo obtenido esté en el formato correcto para la PlotTool. Se recomienda guardar todos los gráficos previos a la importación de un nuevo archivo, o abrir en una segunda ventana otra PlotTool para probar si el archivo .csv es legible por el programa.

3. Otros tipos de *datasets*

3.1. Manejo de funciones transferencia

Se puede crear un *dataset* a partir de una función transferencia mediante el botón **Add transfer function**. Al presionarlo se presenta el siguiente cuadro de diálogo:

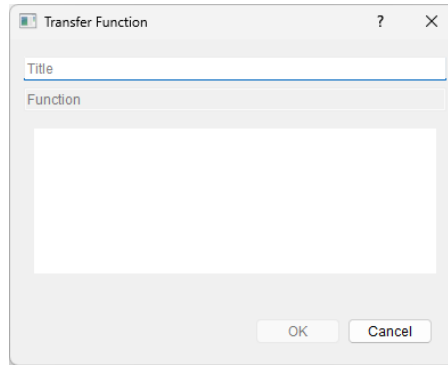


Figura 8: Cuadro de diálogo para creación de *dataset* a partir de función transferencial.

El texto del diálogo se produce poniendo entre símbolos \$ el contenido en minúsculas de la caja de texto. Se pueden utilizar los operadores $^$ o $**$ para la potencia, y se admite el uso de `sqrt(.)` y `pi`.

Si la expresión es correcta, el botón **OK** se habilitará y al presionarlo se agregará un *dataset*, que contiene los campos para seleccionar como variables x o y :

- **f**: frecuencias evaluadas. Notar que se puede pasar de f a ω aplicando la escala $2\pi \approx 6,283$
- **g**: ganancia (en V/V) para cada frecuencia evaluada.
- **ph**: fase para cada frecuencia evaluada.
- **gd**: retardo de grupo para cada frecuencia evaluada.
- **step_time** y **step_res**: tiempos y respuesta al escalón unitario, respectivamente.
- **imp_time** e **imp_res**: tiempos y respuesta al impulso unitario, respectivamente.

Al seleccionar una función transferencial como *dataset*, se habilita el botón de **Show poles and zeros** que permite la visualización del diagrama de polos y ceros en hertz o radianes.

A continuación se presenta un ejemplo de uso:

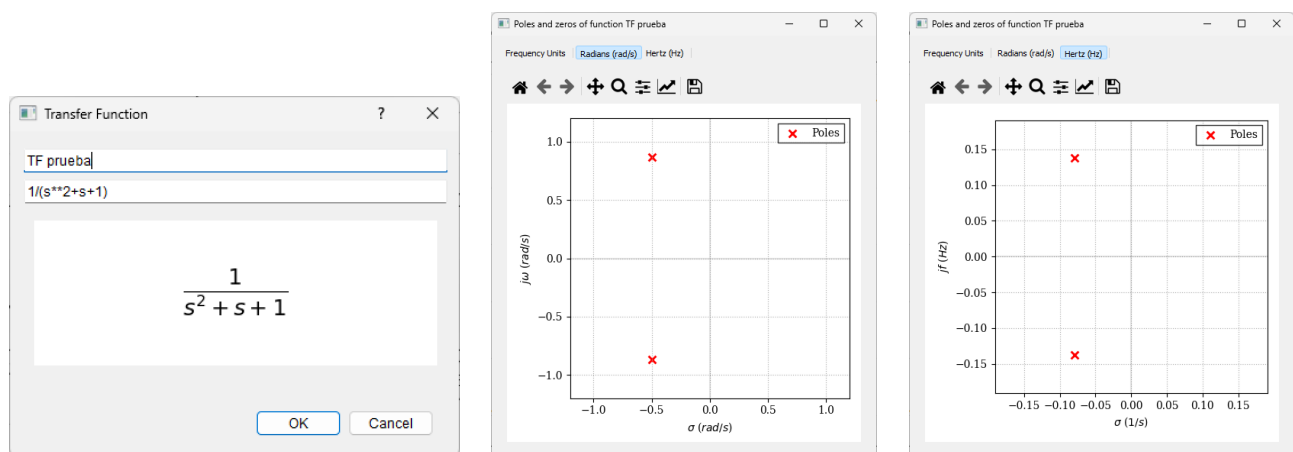


Figura 9: Ejemplo de uso de función *Show poles and zeros* para un *dataset* creado a partir de una función transferencial.

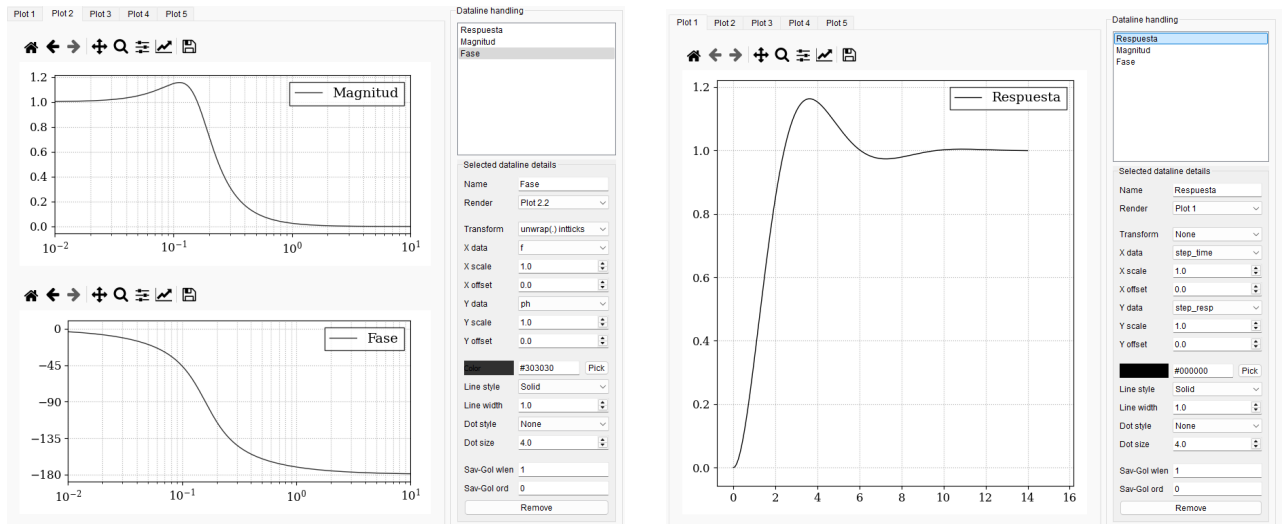


Figura 10: Ejemplo de ploteo de datos obtenidos de un *dataset* creado a partir de una función transferencia.

Atención: para el caso en el que se quiera mostrar la fase en función de la frecuencia (seleccionando *f* como *X data* y *ph* como *Y data*), se recomienda aplicar la transformación `unwrap(.)` (o su variante con `intticks`) ya que mostrará los ticks de fase en divisores enteros de 90.

3.1.1. Obtención de respuestas a señales arbitrarias

Cuando el *dataset* seleccionado es una función transferencia, es posible utilizar la funcionalidad de *Add time response to TF* que permite agregar al *dataset* los campos correspondientes a la respuesta temporal obtenida al aplicar una señal definida por el usuario:

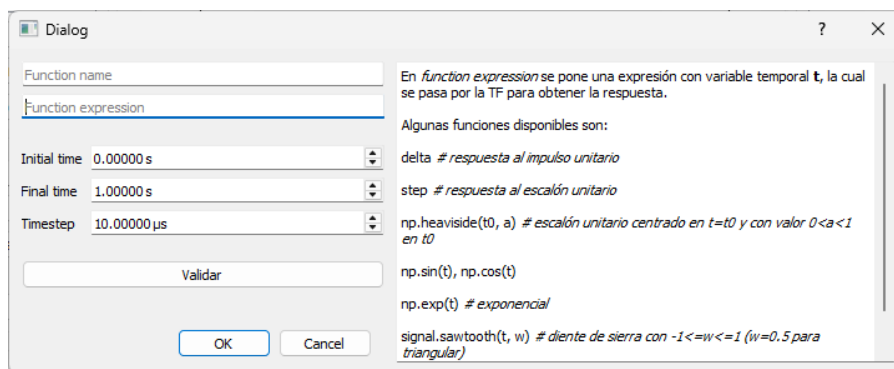


Figura 11: Cuadro de diálogo para configurar la respuesta temporal de una función transferencia.

Se aplicará al sistema la función evaluada en los intervalos temporales especificados. Esto se explicita agregando al *dataline* las siguientes curvas, donde *[nombre]* es el nombre de la función elegido en el cuadro de diálogo:

- *[nombre]_time*: los puntos de prueba temporales, generados de acuerdo a los parámetros seleccionados. Debe ser el *X data* de las líneas utilizadas.
- *[nombre]_inp*: la función de entrada evaluada en los puntos de prueba generados.
- *[nombre]_res*: la función obtenida al pasar la entrada especificada por el sistema seleccionado, evaluada en los puntos de prueba generados.

Atención: no tenemos control sobre las condiciones iniciales del sistema al aplicarle la función. En general, las condiciones iniciales serán nulas como se aprecia en el gráfico de ejemplo.

A continuación se presenta un ejemplo de uso:

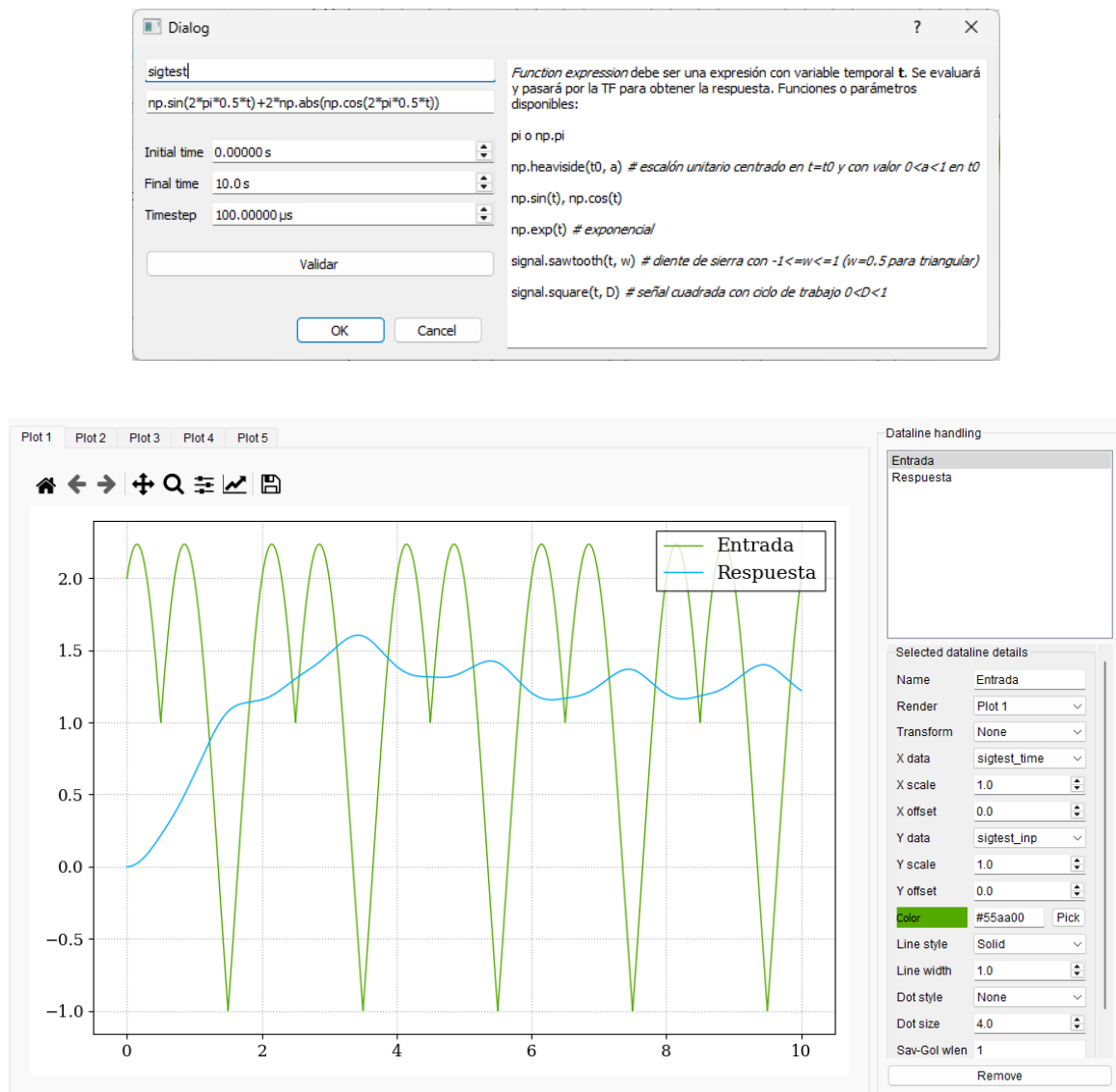


Figura 12: Ejemplo de uso de función *Add time response to TF*.

3.2. Manejo de archivos de simulación de Spice

Como se mencionó previamente, se puede crear un *dataset* a partir de un archivo de simulación de *Spice*. Una vez ejecutada la simulación presionando *Run*, *LTspice* crea 3 archivos en la misma carpeta donde se encuentra el archivo *.asc*. Entre estas 3, se debe importar el archivo *.raw* que no tiene *.op* en el nombre. A continuación se muestra un ejemplo de los archivos generados por simulación:

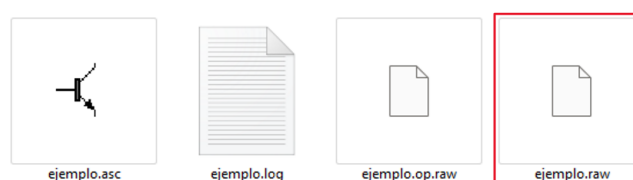


Figura 13: Ejemplo de archivos creados al ejecutar simulación de *Spice*.

En este caso, el archivo encuadrado en rojo es el que se debe utilizar en la PlotTool. Una vez importado el archivo, el manejo del *dataset* es igual al de los otros tipos de archivos. La PlotTool permitirá seleccionar como variables de x o y cada tensión y corriente relevantes.

Atención: Para poder identificar los nodos medidos en la PlotTool, se recomienda utilizar *Label Net* para cada nodo relevante en el esquemático.
