Este capítulo se enfoca a explicar cómo fueron desarrolladas las llamadas de función y el acoplamiento de las mismas a las GUIs creadas. Como complemento se detallan previamente los objetos componentes de cada GUI.

7.1 Descripción de elementos

De acuerdo al desempeño de las interfases gráficas al usuario del **ambiente GUI**, fue primordial en la etapa de diseño determinar la cantidad exacta de elementos que forman parte de cada GUI.

En la tabla 7.1 se aprecian las características de las GUIs del primer capítulo de igual manera que en otras explicaciones me limito a un solo capítulo para ejemplificar el concepto, evitando generar largas listas en el caso que se incluyeran las GUIs de los 10 apartados del ambiente.

Tabla 7.1 Elementos de GUIs primer capítulo.

	# Ejes	Edición	Varía ejes	Fórmula	Botones adicionales	Caja de lista	Enrejado (grid)
GUI1_1	1	1	No implementada	No implementada	1 segundo cálculo	aparición normal	No implementada
GUI1_2	2	2	No implementada	Presente	no tiene	aparición normal	No implementada
GUI1_3	1	2	No implementada	Presente	1 Energía	aparición normal	No implementada
GUI1_4	1	3	eje X	No implementada	1 Potencia	aparición normal	Posible
GUI1_5	2	no tiene	eje X	Presente	no tiene	aparición normal	No implementada
GUI1_6	1	3	eje X	No implementada	no tiene	aparición normal	Posible
GUI1_7	1	1	eje X	Presente	no tiene	aparición normal	No implementada

Entre los segmentos de la tabla se encuentran:

 # Ejes. Es el número de ejes que tiene la GUI disponibles para graficar.

Edición. Indica el número de cuadros de edición útiles en el menú local. Si se trata de variables es posible que se identifiquen con nombre de variable y título (Figura 7.1).

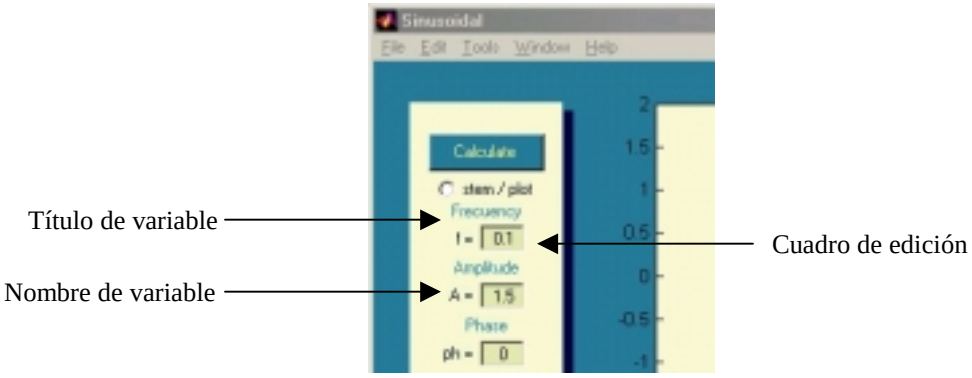

Figura 7.1 Cuadros de edición.

Varía ejes. Muestra si está disponible la opción de variar la magnitud de los ejes. Analizando los casos de todas las GUIs no fue necesario implementar variación del eje Y, únicamente se implemento la del eje X.

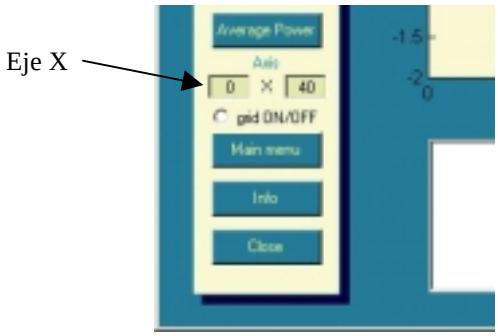

Figura 7.2 Escala del eje X.

Fórmula. Se refiere al marco de fórmula añadido sobre algunos menús locales según se necesitó. Dentro de éste marco se encuentran algunas ecuaciones con su descripción o simplemente las fórmulas básicas involucradas en la GUI.

Botones adicionales. Cuando existieron funciones extras que realizar, se añadieron botones adicionales de función.

Caja de lista. Existen dos modalidades, normal o escondida. En caso que estuviera escondida es debido a que se añadió un tercer eje en la GUI, compartiendo ambos el mismo espacio. Para activarla se agregó un botón de presión (Esconder - Mostrar).

Enrejado. Cuando fue necesario se añadió un botón de selección para activar esta función sobre los ejes (**grid on – grid off**).

7.2 Llamadas de función

Las llamadas de función (**Callbacks**) son el proceso que hace todo el trabajo "detrás de cámaras". En el caso del ambiente GUI existe en cada ventana una caja de lista donde se muestra el código escrito tal cual se tendría que teclear en la ventana de comando para generar una figura en una ventana separada. Es muy importante tener en cuenta que ese código no tiene relación alguna con el desempeño de las llamadas de función, las que generan la graficación en el eje o los ejes dependiendo de la GUI en que se trabaja. Es por ello que se debe tener claro que el contenido de la caja de lista es para que el usuario comprenda los comandos de MATLAB que realizan una función cualquiera (ejemplo, filtrado, graficación, colocación de etiquetas, etc.). Las funciones de llamada de los componentes de las GUIs utilizaron dichos comandos modificados y enriquecidos para realizar las tareas específicas para las que fueron programadas. Con las adaptaciones pertinentes, para que cada elemento hiciera su trabajo independientemente. El usuario al interactuar con la GUI no tiene contacto con los códigos de llamadas de función, ni los manipula (programación), pero tiene acceso a ellos indirectamente al poder realizar la función descrita.

De igual manera que la creación de GUIs se divide en diseño e implementación, la implementación se subdivide en trazado (*guide*) y edición de llamadas de función. Es la diferencia entre cómo se ve la GUI contra lo que hace cuando se interactúa con ella.

Incluso teniendo experiencia en el manejo de MATLAB es difícil poner a trabajar sólida- y eficientemente una llamada de función. En el caso del ambiente GUI cuidando todos los detalles involucrados, los **CALLBACKS** fueron minuciosamente creados, obteniendo un funcionamiento exitoso de todos ellos.

Las llamadas de función con muchos elementos a evaluar son más lentas que las llamadas generadas como una función independiente en memoria. Existe un compromiso entre tiempo de ejecución y espacio en memoria. Las llamadas de función escritas directamente en el editor de llamadas pueden ser tan sencillas o complicadas como se desee pero se debe tener en cuenta el compromiso *velocidad de ejecución*. Al manejar las llamadas como funciones en el disco duro (memoria) se tiene mayor velocidad pues las funciones al ser cargadas y compiladas por el intérprete corren más rápido, el inconveniente es que se genera otro archivo asociado a los mencionados (archivo M y archivo MAT). Entonces para una sola GUI se está hablando de tres archivos que deben permanecer siempre juntos para el correcto funcionamiento.
Originalmente comencé a implementar las llamadas directamente en el editor, pero haciendo pruebas y con la práctica al final decidí convertir todas la llamadas a funciones en disco, pues asociando en un solo archivo los ***CALLBACKS*** aún con la generación de un fichero adicional se produce mayor orden y control sobre las llamadas de función.

En el apéndice A se encuentran las llamadas de función más significativas y ejemplares tal cual serían escritas en el editor, planteado así para que cualquier persona que revise el trabajo pueda analizar por segmentos lo que constituye una función.

En cuanto al menú se refiere, una vez creado el archivo de función como se detalla mas adelante, se escribieron en el editor de llamadas de función (*GUIDE*), los comandos sencillamente como se explica a continuación: **sandinoc seleccion** (en el botón de aparición), **sandinoc eleccion** (en la caja de lista), **sandinoc info** (en el botón de información) y **sandinoc cerrar** (en el botón de cerrar ventana).
La redacción del formato usada para crear un archivo de función, en este ejemplo se refiere a la GUI **MENU**. El archivo de función se guarda en disco con el mismo nombre del archivo M pero con una "c" adicional, con motivo de identificación. Ejemplo: Si el archivo M es **sandino.m** , el archivo de función es **sandinoc.m**. Trabajando ambos en conjunto con el archivo **sandino.mat**.

```
function sandinoc (action)

switch (action)
  case 'seleccion',
    contenido de botón de aparición del MENU [1] (Apéndice A)
  case 'eleccion',
    contenido de caja de lista del MENU [2] (Apéndice A)
  case 'info',
    contenido de botón información de MENU [3] (Apéndice A)
  case 'cerrar',
    contenido de botón cerrar ventana de MENU [4] (Apéndice A)
end
```

La sintaxis del archivo de función del MENU existe tal cual se acaba de presentar. Pero es importante dejar en claro que en lugar del texto presentado en las 4 partes donde dice "contenido de" están insertos los comandos descritos en el apéndice A.