

Resolución numérica de ecuaciones diferenciales ordinarias.

Trabajo académicamente dirigido



Autores: Miguel Ángel Henche Aguado
José Antonio Zamora Medina

Dirigido por: Ángel Manuel Ramos

Título del programa: **ODESOL** v. 2.0

Fecha: 1 de Febrero de 2007

Índice

1. Propósito del trabajo académicamente dirigido
2. Qué es una ecuación diferencial ordinaria
3. Cómo trabajar con ODESOL v.2.0
 1. Cómo instalar ODESOL v.2.0
 2. Requisitos
 3. Como iniciar ODESOL v.2.0
 4. Trabajando con ODESOL v.2.0
 5. Trabajando con el Módulo de resolución de Problemas de Valor Inicial de ODESOL v.2.0
 6. Trabajando con el Módulo de resolución de Problemas de Contorno Lineales de ODESOL v.2.0
 7. Trabajando con el módulo de resolución de Problemas de Contorno No Lineales de ODESOL v.2.0
4. Novedades con respecto a la versión 1.0

Propósito del trabajo académicamente dirigido

En el presente trabajo hemos hecho una toolbox que permita la resolución de ecuaciones diferenciales ordinarias mediante métodos numéricos problemas de contorno, ampliando la toolbox ya desarrollada el año pasado que resolvía ecuaciones diferenciales ordinarias mediante métodos numéricos para problemas de valor inicial. Además se debe tener en cuenta que no toda ecuación diferencial puede ser resuelta de manera exacta mediante los métodos conocidos, e incluso cuando puede resolverse de esta manera, la solución puede no ser expresable mediante funciones elementales. El paquete informático que aquí presentamos tiene como objetivos:

1. Poder resolver las ecuaciones diferenciales mediante una gran diversidad de métodos.
2. Poder introducir las diferentes ecuaciones que se puedan plantear.
3. Modificar las funciones de forma rápida y fácil.
4. Proporcionar herramientas para la comparación de resultados conseguidos a través de diferentes métodos.
5. Representar de forma gráfica los resultados para su estudio, ya sea de ciclos, tendencias, etc.

Otra aplicación que se ha tenido en cuenta a la hora de realizar esta toolbox es poder ser utilizada en la asignatura ANED que actualmente se imparte como asignatura troncal de segundo ciclo en la licenciatura de matemáticas de la UCM, por ello en el programa se ha tenido en cuenta que los posibles usuarios no sean unos expertos en la materia, por lo que se han incluido una gran variedad de métodos ya implementados para la resolución, así como funciones y una ayuda con todo el potencial del toolbox.

El programa elegido para llevar a cabo el trabajo ha sido Matlab. En la elección de este programa también se ha tenido en cuenta, como hemos mencionado anteriormente, la posible utilización del paquete por parte de los alumnos de la facultad de matemáticas de la UCM. Estos alumnos están familiarizados con el manejo de Matlab en algunas asignaturas desde los primeros cursos de la licenciatura.

Qué es una ecuación diferencial ordinaria

Una ecuación diferencial ordinaria de orden m es una ecuación de la forma:

$F(t, x, x', \dots, x^{(m)}) = 0$, donde F es una función definida en un cierto subconjunto de \mathbb{R}^{m+2} . Se puede suponer que la ecuación esta dada en forma normal, es decir: $x^{(m)} = f(t, x, x', \dots, x^{(m-1)})$ siendo f una función continua en un cierto subconjunto de \mathbb{R}^{m+1} . Una función $x(t)$ se dice que es solución de la ecuación diferencial ordinaria en un intervalo I de \mathbb{R} si tiene derivadas hasta el orden m -ésimo continuas en I y verifica:

$$x^{(m)}(t) = f(t, x(t), x'(t), \dots, x^{(m-1)}(t)), \forall t \in I.$$

Si una función diferencial ordinaria tiene una solución, entonces tiene infinitas soluciones. Alternativamente, bajo condiciones muy generales, queda determinada una solución única si se prefijan unas condiciones iniciales. Entonces se dice que se trata de un *problema del valor inicial*.

Esto en general, bajo hipótesis muy generales, se puede demostrar que el *problema de valor inicial*:

$$\begin{cases} x^{(m)} = f(t, x, x', \dots, x^{(m-1)}) \\ x(t_0) = x_1 \\ x'(t_0) = x_2 \\ \vdots \\ x^{(m-1)}(t_0) = x_m \end{cases}$$

Tiene solución única.

Problema de Contorno

En este caso nos centramos en los problemas de contorno, distinguiendo dos paquetes de resolución, para problemas de contorno lineales y para problemas de contorno no lineales. Al igual que en el problema de valor inicial, en el problema de contorno se puede demostrar que tiene solución única bajo ciertas hipótesis. Formulamos el problema de contorno:

$$\begin{cases} x'' = f(t, x, x') \\ x(t_0) = a \\ x(T) = b \end{cases} \quad \text{ó} \quad \begin{cases} x'' = f(t, x, x') \\ x'(t_0) = a \\ x(T) = b \end{cases} \quad \text{ó} \quad \begin{cases} x'' = f(t, x, x') \\ x(t_0) = a \\ x'(T) = b \end{cases} \quad \text{ó} \quad \begin{cases} x'' = f(t, x, x') \\ x'(t_0) = a \\ x'(T) = b \end{cases}$$

Se dice que una condición es de tipo Dirichlet cuando se fija el valor de la solución en la frontera, $x(t_0) = a$ ó $x(T) = b$.

Se dice que una condición es de tipo Neumann cuando se fija el valor de un flujo sobre la frontera, $x'(t_0) = a$ ó $x'(T) = b$.

En el caso particular de problemas de contorno lineales, reformulamos el problema de la siguiente manera:

$$\begin{cases} x'' = p(t)x'(t) + q(t)x(t) + r(t) \\ x(t_0) = a \\ x(T) = b \end{cases} \quad \text{ó} \quad \begin{cases} x'' = p(t)x'(t) + q(t)x(t) + r(t) \\ x'(t_0) = a \\ x(T) = b \end{cases} \quad \text{ó}$$

$$\begin{cases} x'' = p(t)x'(t) + q(t)x(t) + r(t) \\ x(t_0) = a \\ x'(T) = b \end{cases} \quad \text{ó} \quad \begin{cases} x'' = p(t)x'(t) + q(t)x(t) + r(t) \\ x'(t_0) = a \\ x'(T) = b \end{cases}$$

Los usos que un científico técnico puede darle a las ecuaciones diferenciales son muy diversos tanto por el tipo de ecuaciones como por la profundidad con que éstas se analizan. Una ecuación diferencial es una relación que ha de satisfacer una "función incógnita" junto con un cierto número de sus derivadas.

La teoría de ecuaciones diferenciales ordinarias comenzó a desarrollarse a finales del siglo XVII en estrecha relación con el naciente cálculo diferencial e integral. Las ecuaciones diferenciales ordinarias surgen en la investigación de distintos problemas de la física: en sus aplicaciones a la construcción, el péndulo, la ley de atracción gravitatoria, etc. Hoy por hoy son utilizadas en multitud de ciencias como pueden ser la biología, la economía, etc.

En el paquete informático que aquí presentamos se adjuntan ejemplos de estas ecuaciones diferenciales ordinarias. Por nombrar algunas: funcpéndulo, que describe el movimiento de un péndulo, la función funcdeppresal, que representa la función depredador-presa donde se modeliza la interacción en un medio entre los depredadores y sus presas, piénsese en el conocido caso de los zorros y los conejos.

Cómo trabajar con ODESOL v.2.0

1. Cómo instalarlo

El paquete informático se presenta en un archivo auto instalable que nos permite descomprimir y copiar todos los archivos y carpetas que precisa el programa ODESOL para su correcto funcionamiento. Para instalarlo simplemente hay que ejecutar el archivo de instalación y seguir los pasos que éste indique. Además, utilizando este asistente de instalación nos permitirá, si así lo deseamos, borrar (desinstalar) todos los archivos y carpetas copiados.

2. Requisitos

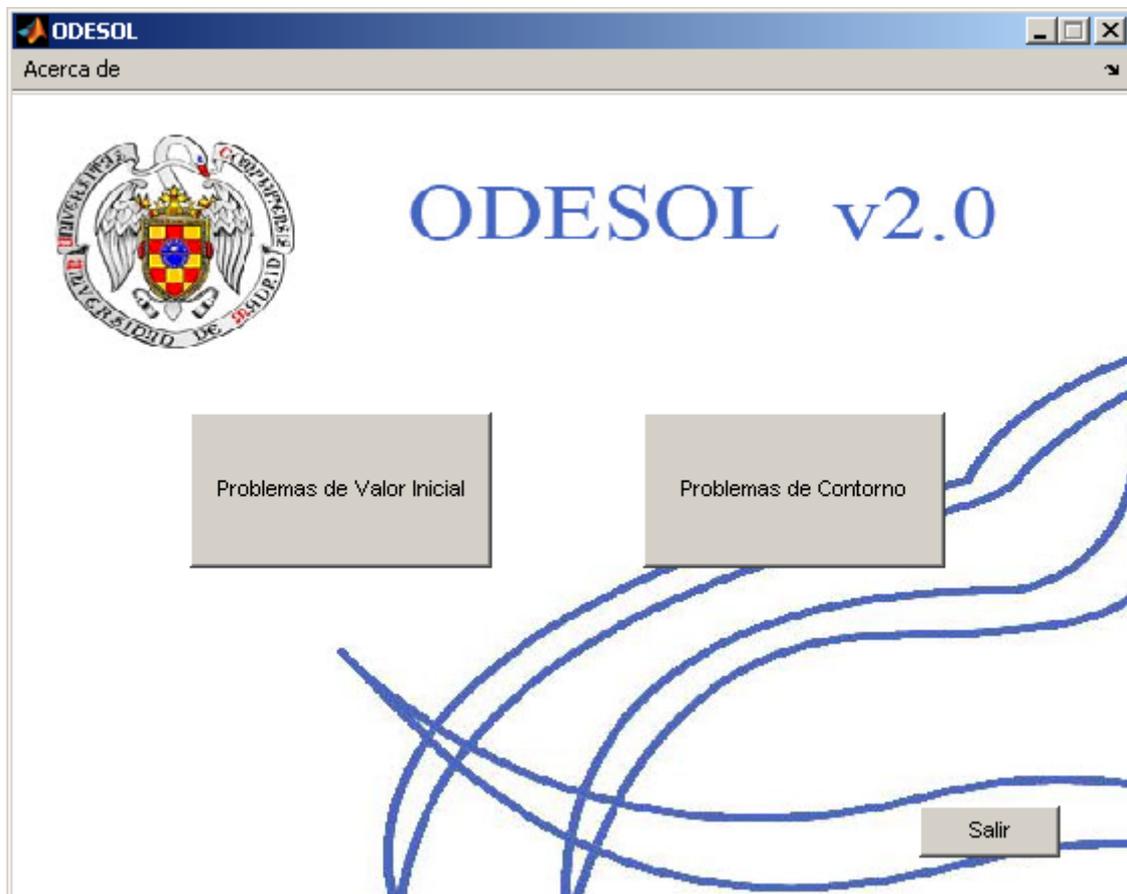
Para la utilización del programa ODESOL es necesario tener instalado en el ordenador de destino el programa MATLAB. ODESOL ha sido creado con MATLAB 7.01 R14 service pack1. Para la correcta utilización de ODESOL se aconseja tener esta versión de MATLAB o posterior, aunque posiblemente pueda funcionar también con versiones anteriores a la mencionada.

3. Cómo inicializar ODESOL

Para inicializar ODESOL abra primero MATLAB. Seleccione como directorio de trabajo aquel en el que haya instalado ODESOL, para ello puede utilizar el comando *cd* seguido del directorio en el que se encuentre ODESOL. Una vez que se encuentre en el directorio adecuado escriba `ODESOL` y presione enter. Le aparecerá la ventana principal del programa.

4. Trabajando con ODESOL

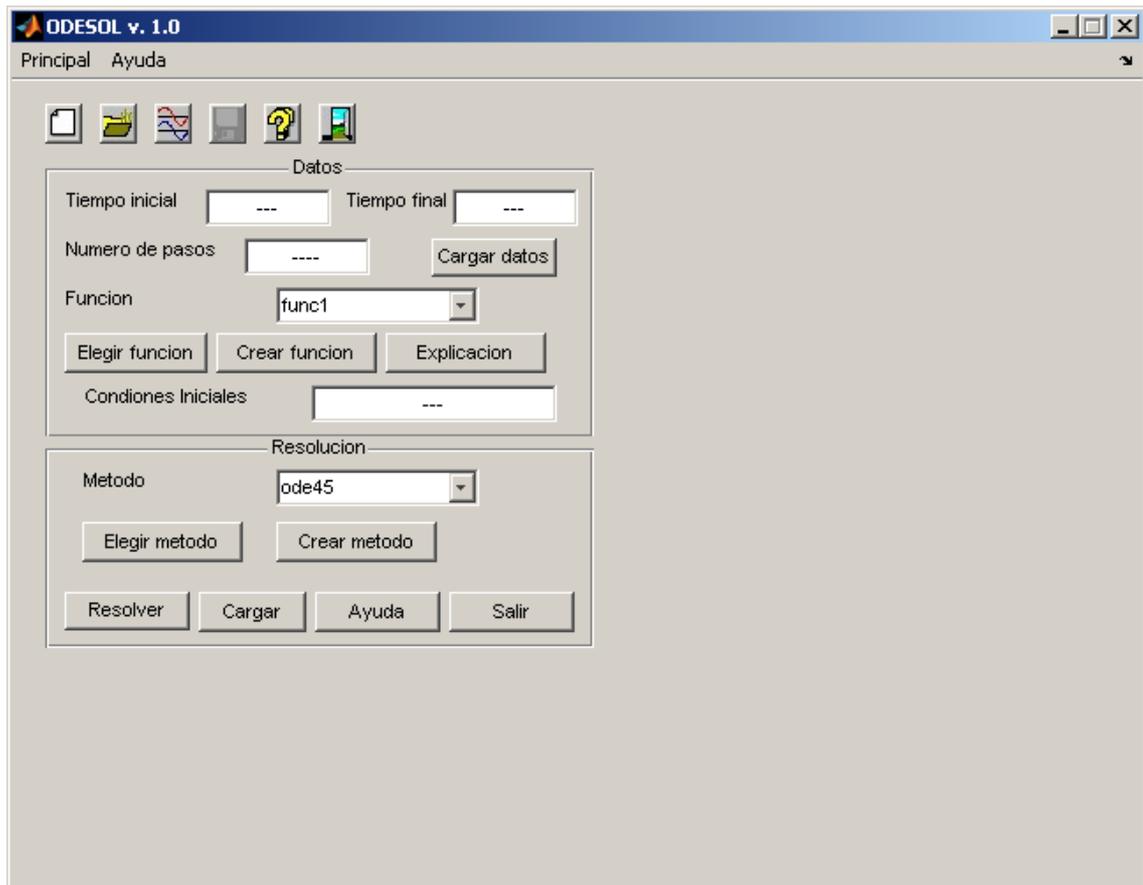
Cuando usted inicia sesión con ODESOL le aparece la ventana principal del programa



En esta pantalla se nos da a elegir el tipo de problema que queremos resolver, distinguiremos en cada caso como debemos introducir los datos.

5. Trabajando con el módulo de resolución de Problemas de valor inicial de ODESOL

Cuando usted inicia el módulo de resolución de Problemas de valor inicial de ODESOL le aparece la siguiente ventana.



Como se puede ver en la imagen aparecen dos paneles: el primero es el panel en el que introduciremos los datos que definen la ecuación diferencial ordinarias, estos son: el tiempo inicial y final que nos da el intervalo de definición de la ecuación diferencial ordinaria. Las condiciones iniciales de la ecuación diferencial ordinaria ($x(t_0) = x_0$). El número de pasos representa el número de nodos equiespaciados en los que se va a calcular la ecuación diferencial ordinaria. Usted dispone de un menú desplegable en el que puede elegir entre cualquiera de las funciones de las que tiene guardadas en la carpeta "funciones". Estas funciones tienen que tener un formato específico. Supongamos que tenemos la ecuación diferencial ordinaria: $x''(t) = -16x(t) + 4\text{sen}(2t)$

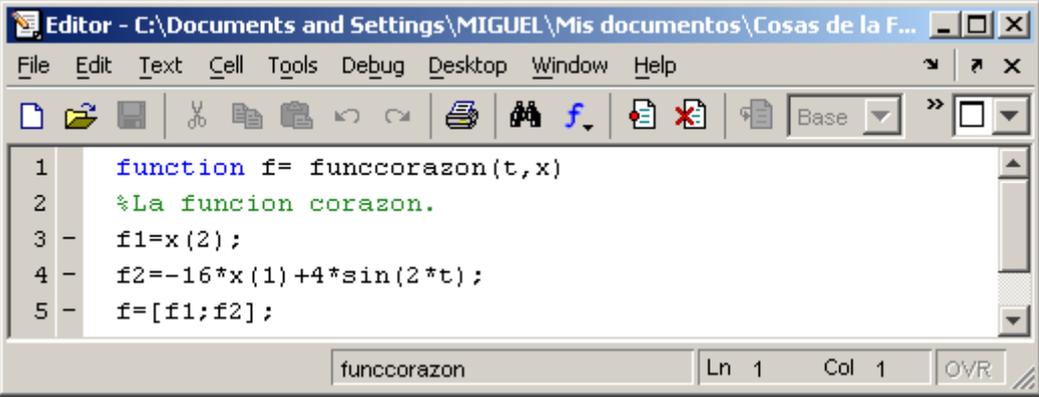
con condiciones iniciales:

$$\begin{cases} x(0) = 0; \\ x'(0) = 2; \end{cases}$$

entonces tendremos que utilizar el sistema de ecuaciones diferenciales equivalente. Para ello se debe hacer la siguiente transformación:

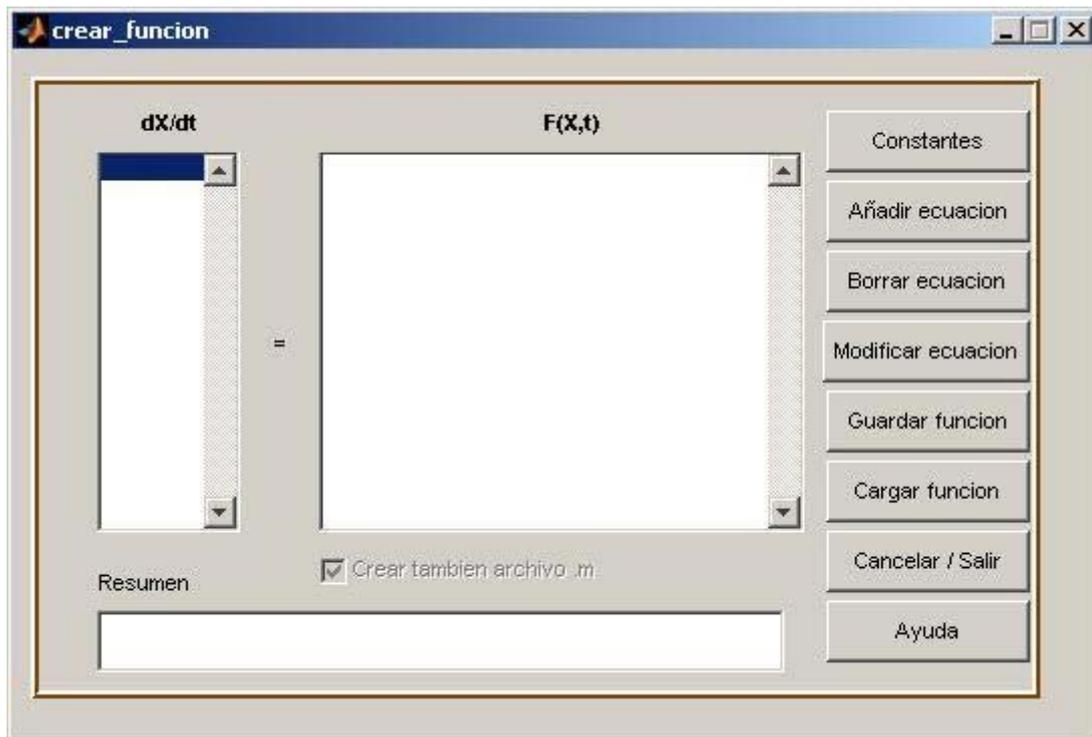
$$\begin{cases} x_1(t) = x(t) \\ x_2(t) = x'(t) \end{cases} \rightarrow \begin{cases} x_1'(t) = x_2(t) \\ x_2'(t) = -16x_1(t) + 4\text{sen}(2t) \end{cases}$$

Ahora deberá escribir esta función en un archivo del tipo función (con extensión m) de MATLAB de la siguiente forma:

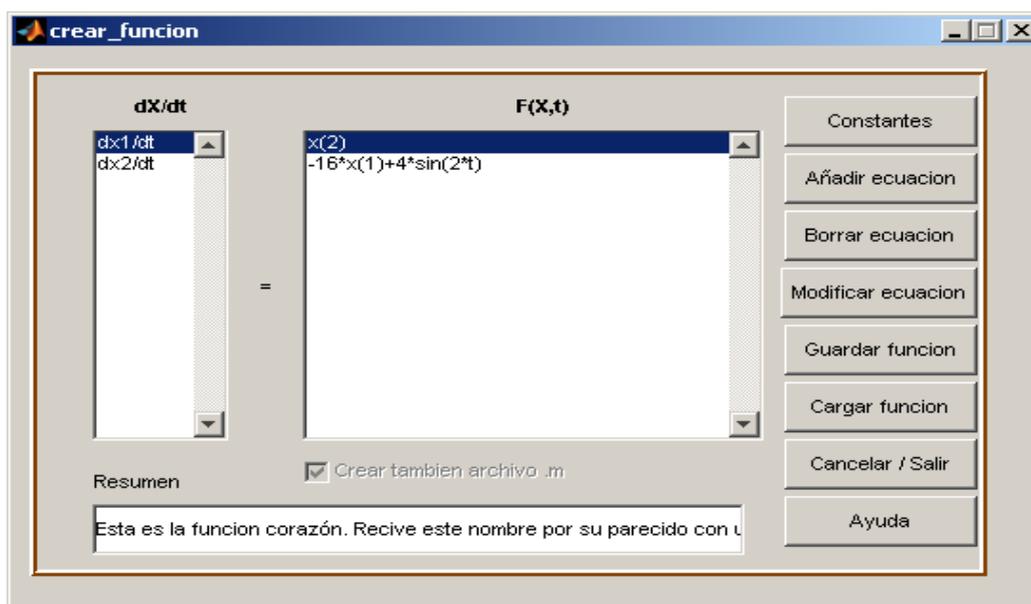


```
Editor - C:\Documents and Settings\MIGUEL\Mis documentos\Cosas de la F...
File Edit Text Cell Tools Debug Desktop Window Help
Base
1 function f= funcorazon(t,x)
2 %La funcion corazon.
3 f1=x(2);
4 f2=-16*x(1)+4*sin(2*t);
5 f=[f1;f2];
funcorazon Ln 1 Col 1 OVR
```

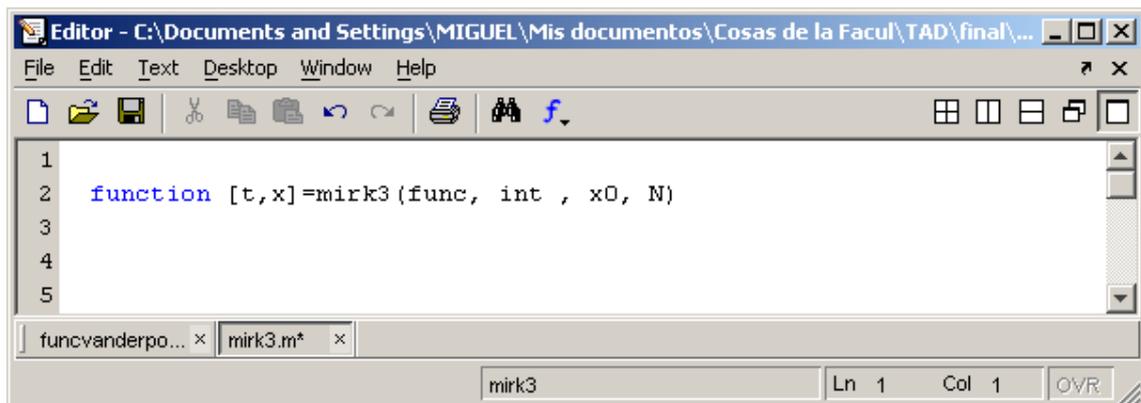
Esta función sería reconocida por ODESOL . Obsérvese el orden de las variables de entrada t y x así como la salida de la función, que debe ser por columnas, por eso en el vector salida de la función f se utiliza ;. Para más información sobre cómo crear funciones con MATLAB mire la ayuda de MATLAB. De todos modos ODESOL le proporciona un editor de ecuaciones para poder introducir éstas de una forma cómoda. Si quisiésemos introducir esta misma función utilizando ODESOL sólo tendremos que presionar el botón "crear función" que se encuentra en el panel datos. Si usted presiona este botón le aparecerá:



Con este módulo puede ir introduciendo una a una cada ecuación del sistema equivalente. Para ello sólo tendrá que pulsar el botón "añadir ecuación" e introducir la ecuación. Además ODESOL le permite introducir constantes en las ecuaciones, para ello utilice una variable alfanumérica que no comience con un número. Más tarde presione el botón "constantes" y asígnele a cada variable que haya creado el valor que quiera utilizando el símbolo igual. Vea la siguiente imagen:

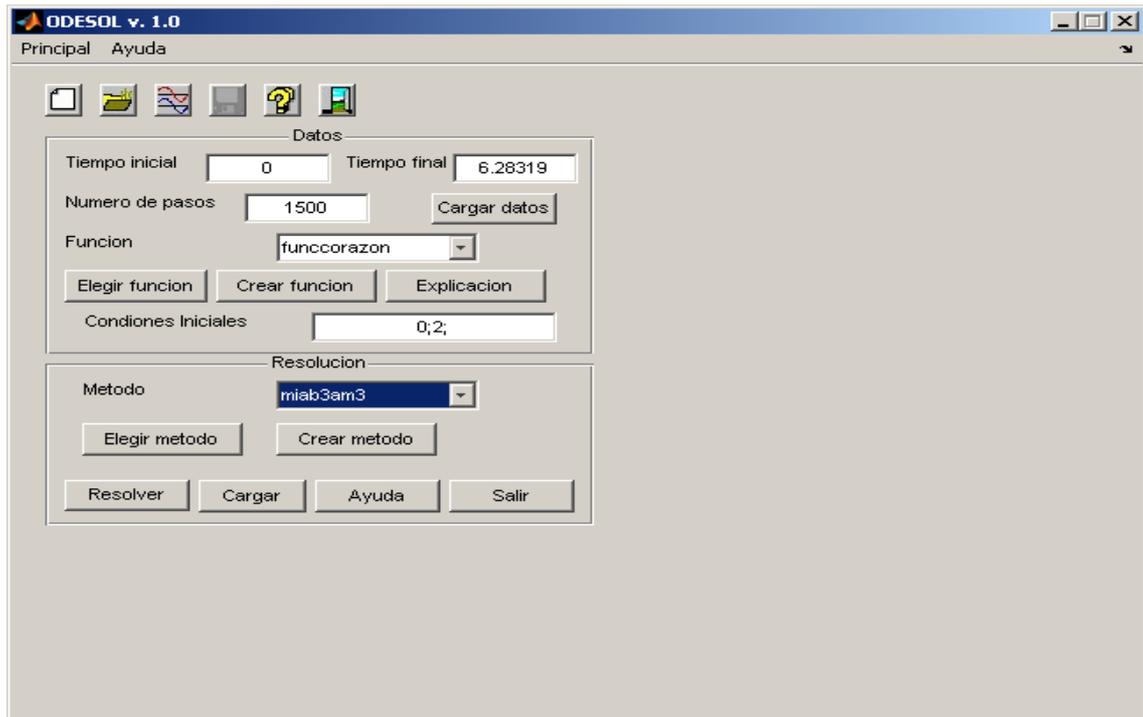


Ahora sólo tiene que presionar el botón "guardar" y llamar a la función como desee. Ahora está en condiciones de resolver la ecuación diferencial. Para resolverla usted puede elegir uno de los métodos adjuntos al programa ODESOL o bien elegir cualquier método creado por usted. Si se decanta por este último caso tenga en cuenta que el método tiene que ser una función de MATLAB guardada en un archivo m que tenga como datos de entrada a la función las variables: *func*, del tipo char con el nombre de la función, *int*, que se trata de un vector numérico de dimensión uno por dos y representa el intervalo de definición de la función, *x0*, que también es un vector numérico, esta vez de dimensión m por uno, que representa las condiciones iniciales de la ecuación diferencial ordinaria, por último la variable *N*, que representa el número de nodos equiespaciados en los que se calculará la solución. La salida de la función debe de ser un vector columna N por uno con los tiempos en los que se ha calculado la solución, y otra variable *x* de dimensión N por m con las soluciones de la ecuación diferencial ordinaria. Mírese la siguiente figura:



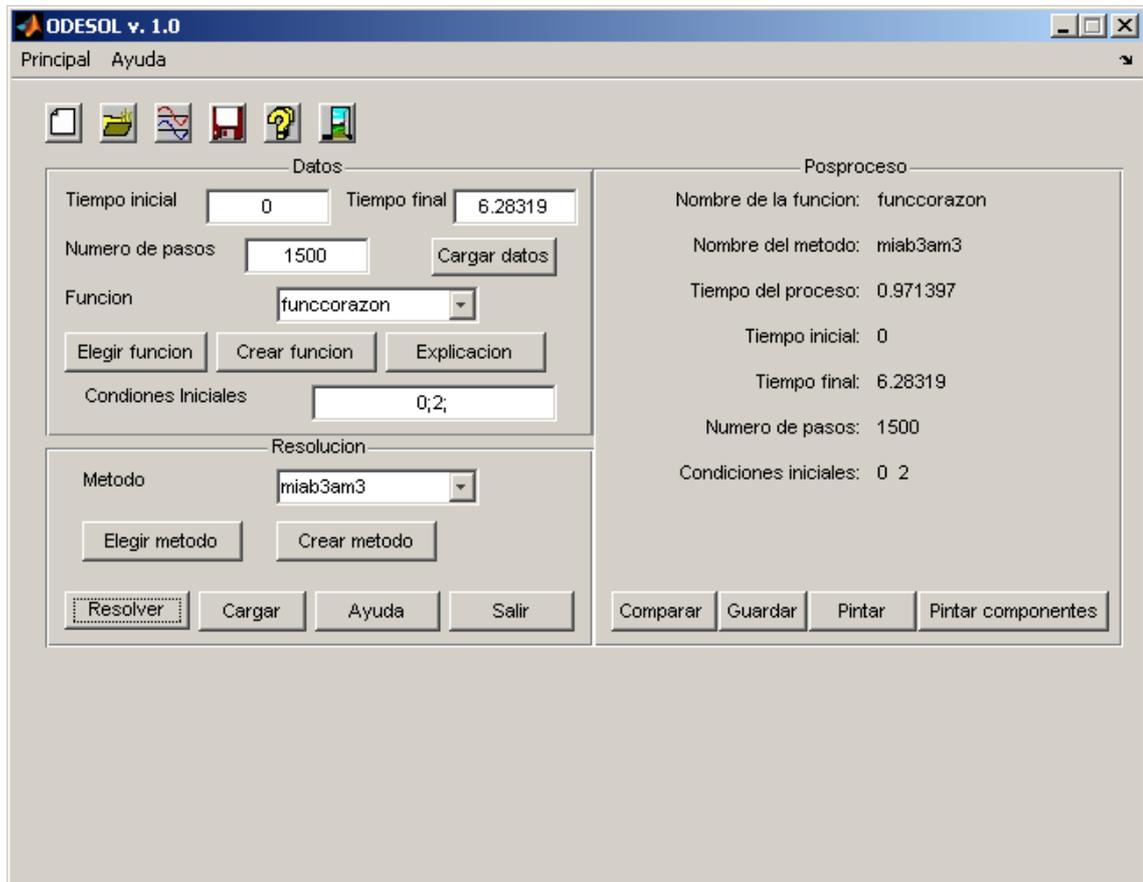
```
1  
2 function [t,x]=mirk3(func, int, x0, N)  
3  
4  
5
```

Para resolver una ecuación diferencial ordinaria es necesario pasar al programa todos los datos mencionados. Debería parecerse a lo siguiente:



Ahora sencillamente presione el botón "resolver" para resolver la ecuación diferencial ordinaria.

Una vez resuelta le aparecerá:

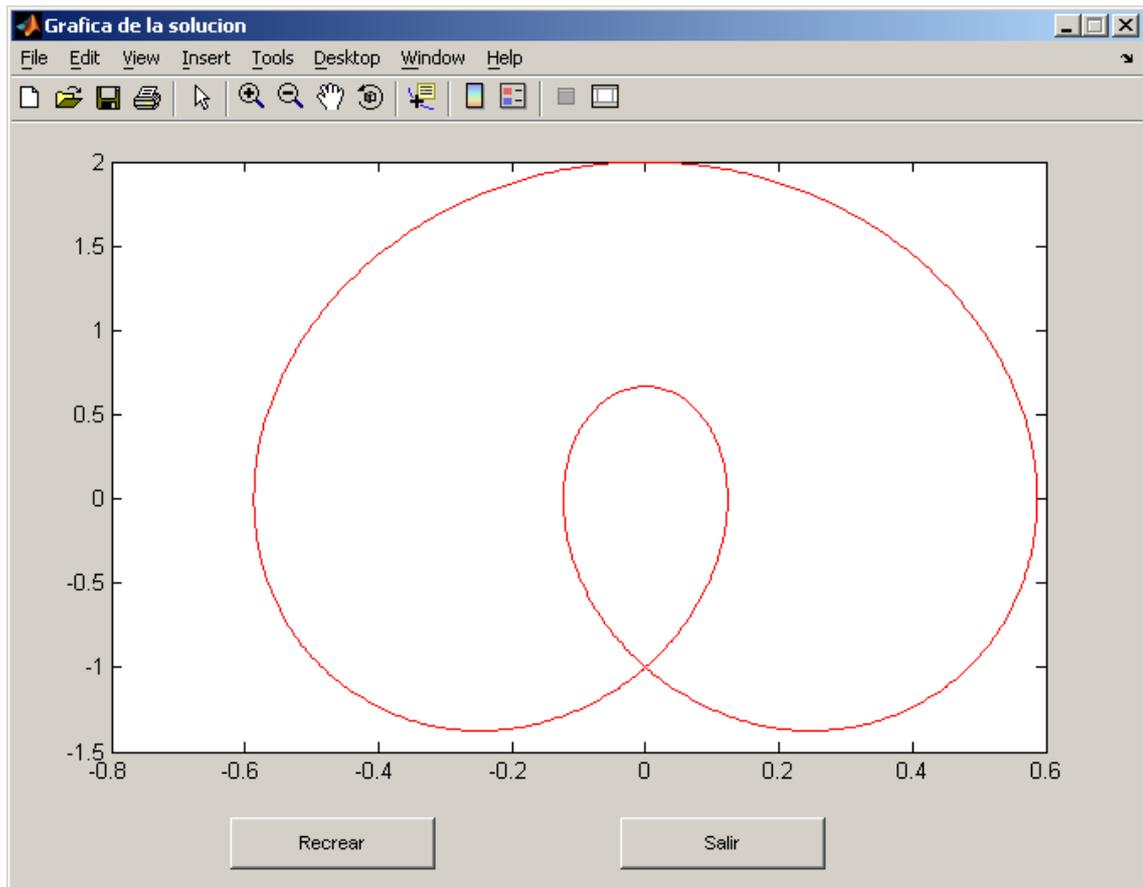


Observe como ahora dispone de un nuevo panel llamado "posproceso" con todos los datos del problema que pretendía resolver. En él se incluyen el tiempo en el que se ha llevado a cabo el proceso así como con qué método ha sido resuelto. Usted puede guardar todos los datos del problema sencillamente presionando el botón "guardar". En el archivo guardado, además de los datos que usted ve en pantalla, en el panel posproceso, se guardará el tiempo y la solución halladas. Usted podrá recuperar estos datos en sesiones posteriores con el programa ODESOL, además de poder exportar los resultados a otras aplicaciones. El archivo guardado contiene una variable llamada Sys del tipo struct en formato binario con los siguientes campos:

- Sys.x la solución del problema.
- Sys.t vector de tiempos donde se ha calculado la solución.
- Tanto x como t son vectores columna de la misma longitud.
- Sys.tiempo el tiempo que se ha tardado en calcular la solución.
- Sys.metodo el nombre del método con que se ha calculado la solución.
- Sys.funcion el nombre de la función que ha resuelto.
- Sys.N el número de pasos que se han dado para calcular la solución.
- Sys.t_ini tiempo inicial del problema.
- Sys.t_fin tiempo final del problema.
- El tiempo inicial y final dan el intervalo de definición del problema.
- Sys.x0 condiciones iniciales del problema.

Usted puede cargar estos datos en el programa en otra sesión, o bien exportarlos a otra aplicación, por ejemplo en el propio Matlab. Si está trabajando en la ventana command window de MATLAB puede cargar estos datos desde el menú File y eligiendo Open o bien Import Data. También puede hacerlo usando el comando load de Matlab.

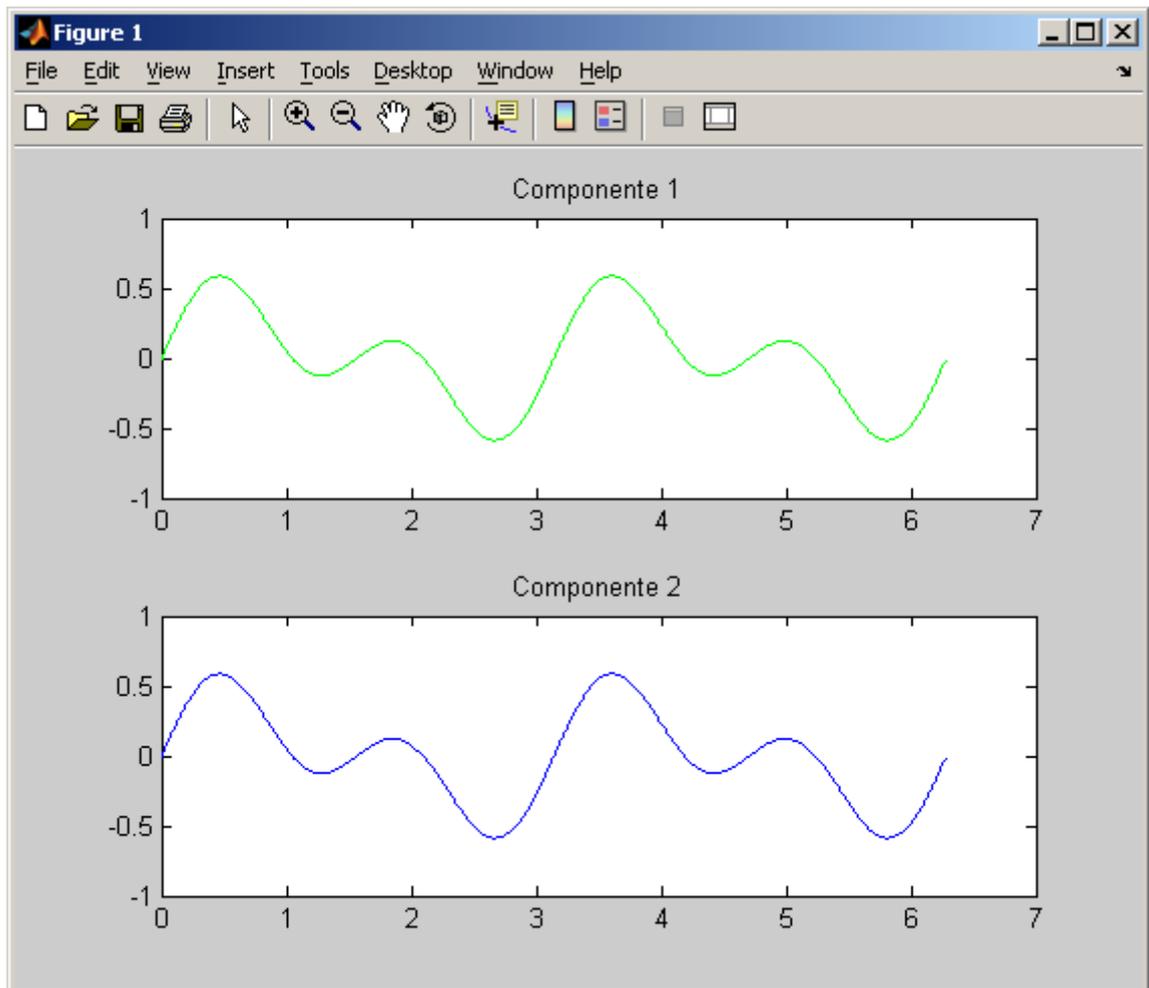
Usted puede pintar la solución del problema. Para ello sencillamente pulse el botón "pintar", le aparecerá una nueva ventana:



En esta ventana usted podrá guardar la gráfica de la solución, podrá verla de distintas posiciones, ampliar zonas, y un largo etc. Además se ha incluido el botón "recrear" que le permite ver la solución en movimiento.

Nota importante: por cuestiones obvias sólo se podrán representar funciones de una, dos o tres dimensiones.

Además de pintar la solución usted podrá pintar cada una de las componentes por separado utilizando el botón "pintar componentes". Si presiona este botón le aparecerá una nueva ventana en la que podrá elegir qué componentes pintar y en qué color:

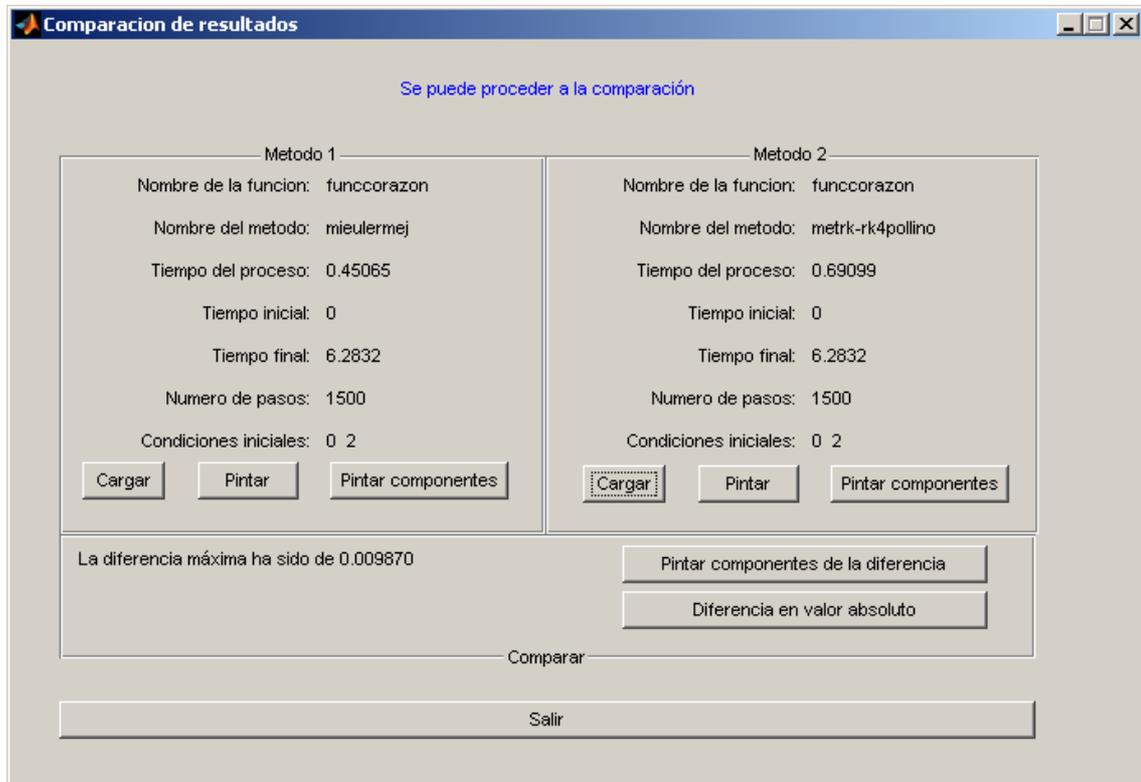


Al igual que antes usted podrá guardar estas gráficas para incorporarlas en sus trabajos.

Por último el panel posproceso le brinda la oportunidad de comparar la solución hallada con la solución exacta de la ecuación diferencial ordinaria, por supuesto si se conoce. Para ello presione el botón "comparar", el programa le pedirá el fichero en el que se encuentra guardada la función de la solución exacta. Le aparecerá un nueva ventana con todas la opciones creadas para comparar las soluciones. Más abajo se explica como se puede hacer la comparación de los resultados, comparar un resultado con la solución exacta se hace de la misma manera.

Si usted quiere comparar soluciones resueltas con distintos métodos, guarde cada una de las soluciones en archivos distintos y presione el botón : . Le aparecerá una nueva ventana, con los botones "cargar" cargue los ficheros en los que

ha guardado las diferentes soluciones. Para proceder a la comparación es necesario que se haya resuelto la misma ecuación diferencial ordinaria en las mismas condiciones, es decir, mismas condiciones iniciales, mismo intervalo de definición y mismo número de pasos. Entonces podrá proceder a la comparación. Le aparecerá algo parecido a lo siguiente:

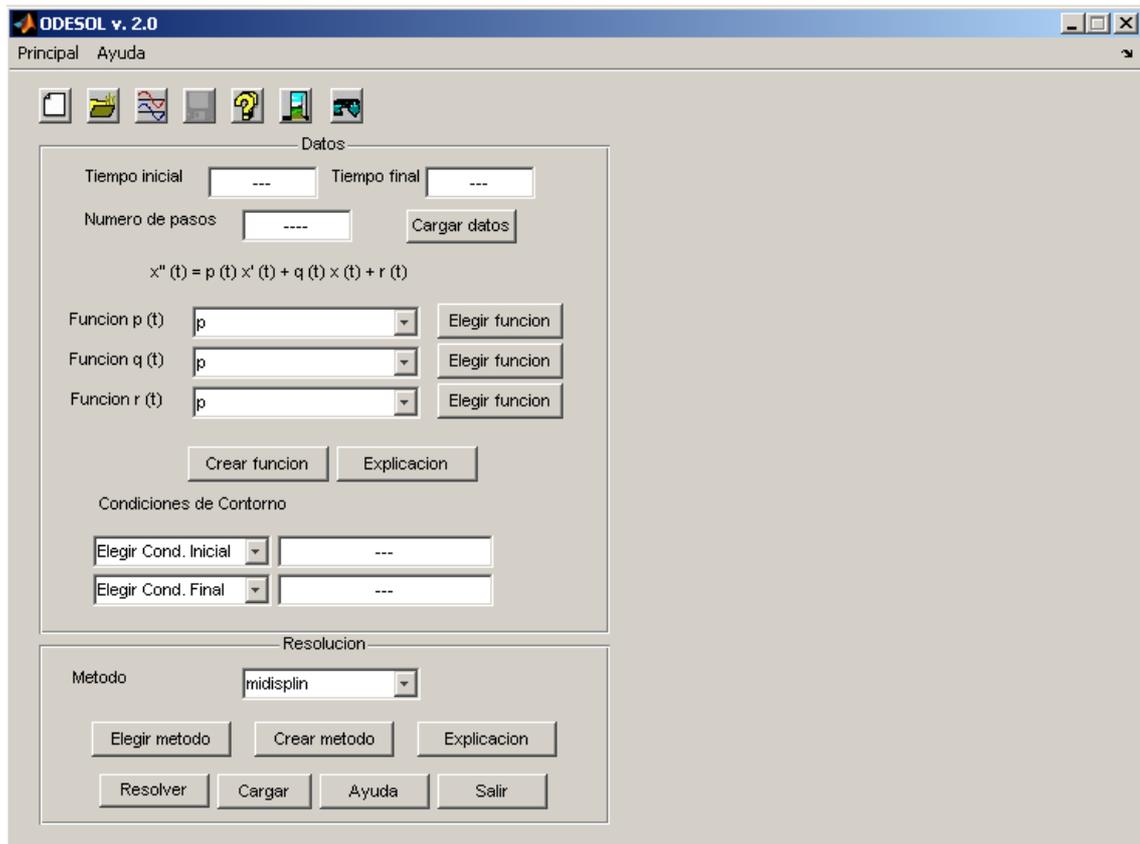


Le informará de cuál ha sido la diferencia máxima y le permitirá pintar las gráficas de la diferencia de las diferentes componentes incluido la diferencia en valor absoluto, para ello presione los botones "pintar componentes de la diferencia" o "diferencia en valor absoluto" respectivamente.

Para más información del programa vea las ayudas que se adjuntan con el paquete informático.

6. Trabajando con el módulo de resolución de Problemas de Contorno Lineal de ODESOL

Cuando usted inicia el módulo de resolución de Problemas de contorno lineales de ODESOL le aparece la siguiente ventana.



Como se puede ver en la imagen aparecen dos paneles: el primero es el panel en el que introduciremos los datos que definen la ecuación diferencial ordinaria, estos son: el tiempo inicial y final que nos da el intervalo de definición de la ecuación diferencial ordinaria. Las condiciones iniciales y las condiciones finales de la ecuación diferencial ordinaria ($x(t_0) = x_0$ y $x(T) = x_T$). El tipo de las condiciones iniciales y las condiciones finales, Dirichlet o Newmann. El número de pasos representa el número de nodos equiespaciados en los que se va a calcular la ecuación diferencial ordinaria. Usted dispone de 3 menús desplegables en los que puede elegir entre cualquiera de las funciones de las que tiene guardadas en la carpeta "funciones" para elegir las funciones $p(t)$, $q(t)$ y $r(t)$. Estas funciones

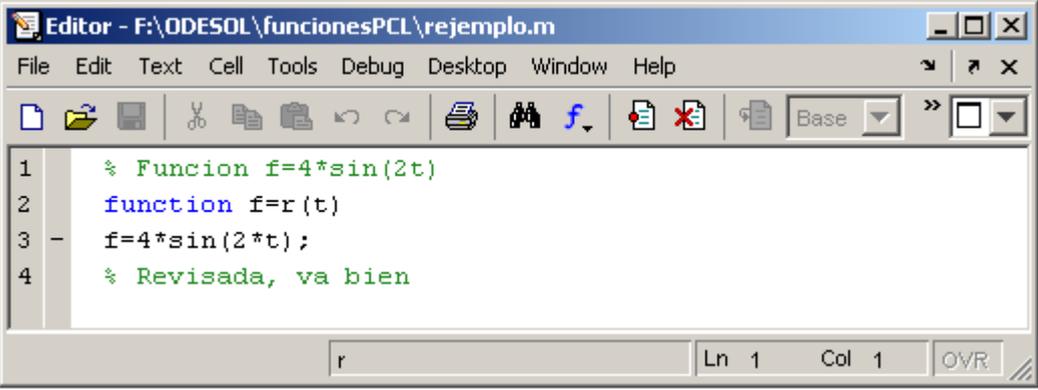
tienen que tener un formato específico. Supongamos que tenemos la ecuación diferencial ordinaria: $x''(t) = -16x(t) + 4\text{sen}(2t)$.

con condiciones iniciales:

$$\begin{cases} x(0) = 0; \\ x(T) = 0; \end{cases}$$

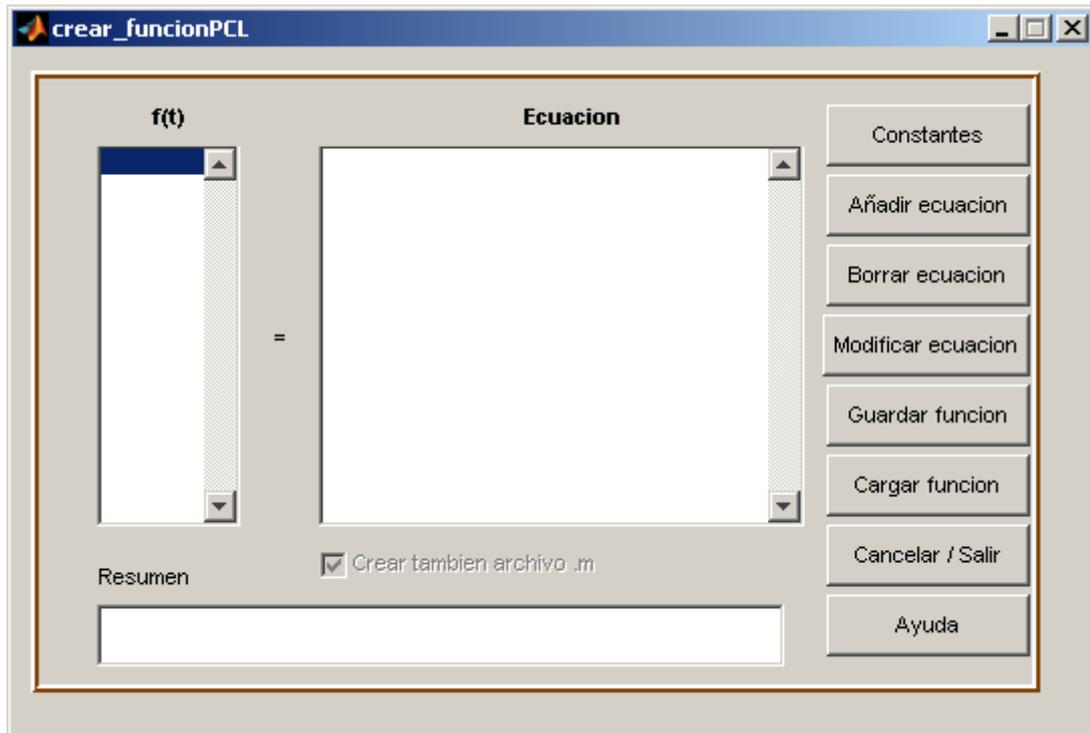
Dónde en nuestro caso particular $p(t) = 0$, $q(t) = -16$ y $r(t) = 4\text{sen}(2t)$.

Ahora deberá escribir estas funciones en archivos de tipo función (con extensión m) de MATLAB de la siguiente forma:

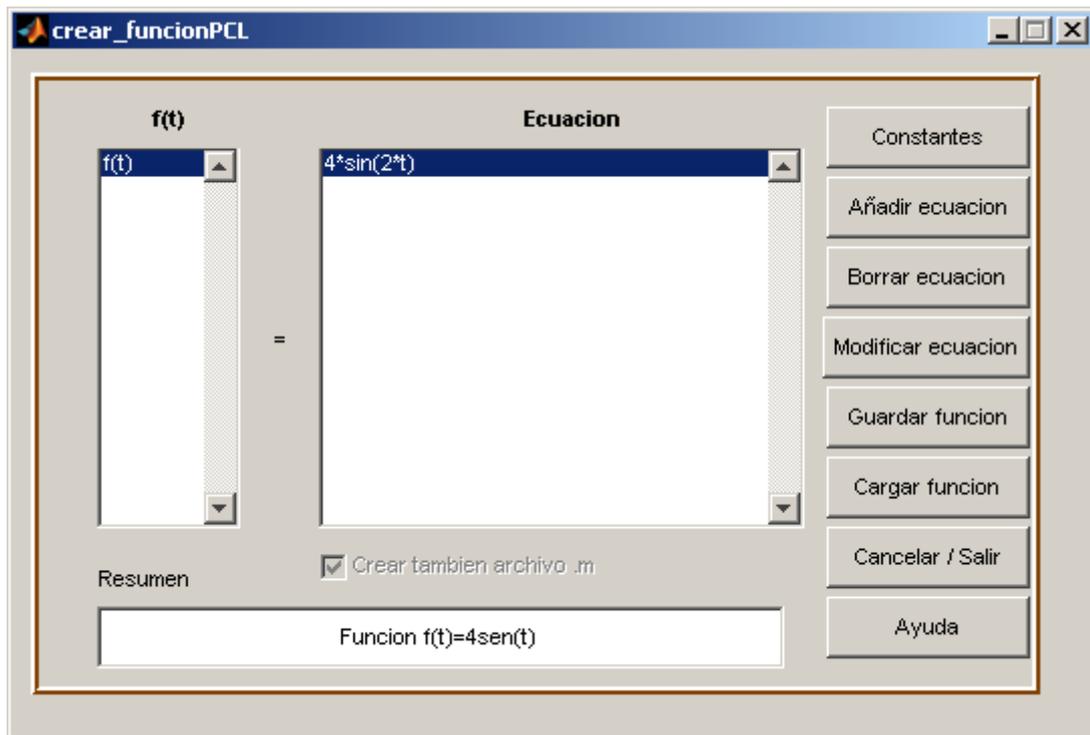


```
Editor - F:\ODESOL\funcionesPCL\rejemplo.m
File Edit Text Cell Tools Debug Desktop Window Help
[Icons] Base >>
1   % Funcion f=4*sin(2t)
2   function f=r(t)
3   -   f=4*sin(2*t);
4   % Revisada, va bien
Ln 1 Col 1 OVR
```

Esta funciones serían reconocidas por ODESOL . Obsérvese que tanto la variable de entrada t, así como la salida de la función, debe ser por columnas. Para más información sobre cómo crear funciones con MATLAB mire la ayuda de MATLAB. De todos modos ODESOL le proporciona un editor de funciones para poder introducir éstas de una forma cómoda. Si quisiésemos introducir esta misma función utilizando ODESOL sólo tendremos que presionar el botón "crear función" que se encuentra en el panel datos. Si usted presiona este botón le aparecerá:



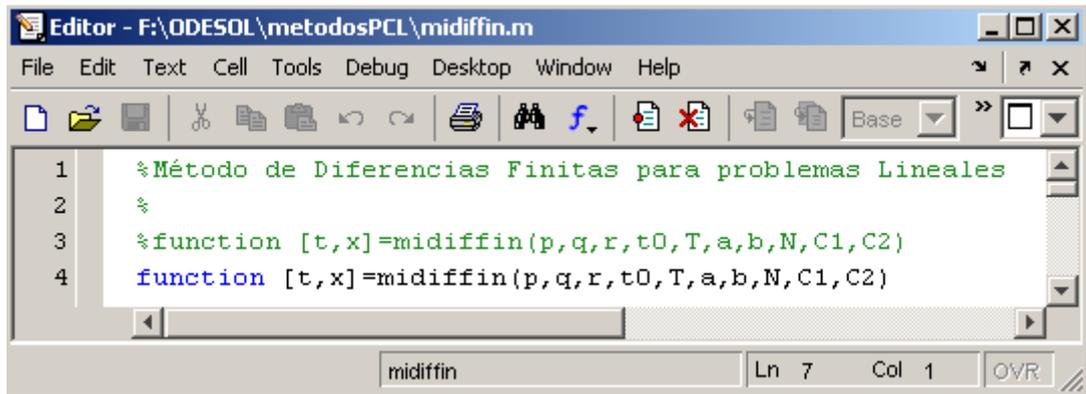
Con este módulo puede ir introducir la ecuación de la función. Para ello sólo tendrá que pulsar el botón "añadir ecuación" e introducir la ecuación. Además ODESOL le permite introducir constantes en las ecuaciones, para ello utilice una variable alfanumérica que no comience con un número. Más tarde presione el botón "constantes" y asígnele a cada variable que haya creado el valor que quiera utilizando el símbolo igual. Vea la siguiente imagen:



Ahora sólo tiene que presiona el botón "guardar" y llamar a la función como desee.

Ahora está en condiciones de resolver la ecuación diferencial. Para resolverla usted puede elegir uno de los métodos adjuntos al programa ODESOL o bien elegir cualquier método creado por usted. Si se decanta por este último caso tenga en cuenta que el método tiene que ser una función de MATLAB guardada en un archivo m que tenga como datos de entrada a la función las variables: *funcion_p*, del tipo char con el nombre de la función p, *funcion_q*, del tipo char con el nombre de la función q, *funcion_r*, del tipo char con el nombre de la función r, x_0 y x^F que son escalares, que representan la condición inicial y la condición final de la ecuación diferencial ordinaria, *int*, que se trata de un vector numérico de dimensión uno por dos y representa el intervalo de definición de la función, la variable *N*, que representa el número de nodos equiespaciados en los que se calculará la solución, y por ultimo los valores del tipo de condición de contorno inicial (0 si es de tipo Dirichlety 1 si es de tipo Newmann) y de tipo de condición final (0 si es de tipo Dirichlety 1 si es de tipo Newmann). Para ver los parámetros anteriormente comentados y su orden, puede seleccionar un método para la resolución y pulsar el botón de explicación en el panel de Resolución. El programa le mostrará una nueva

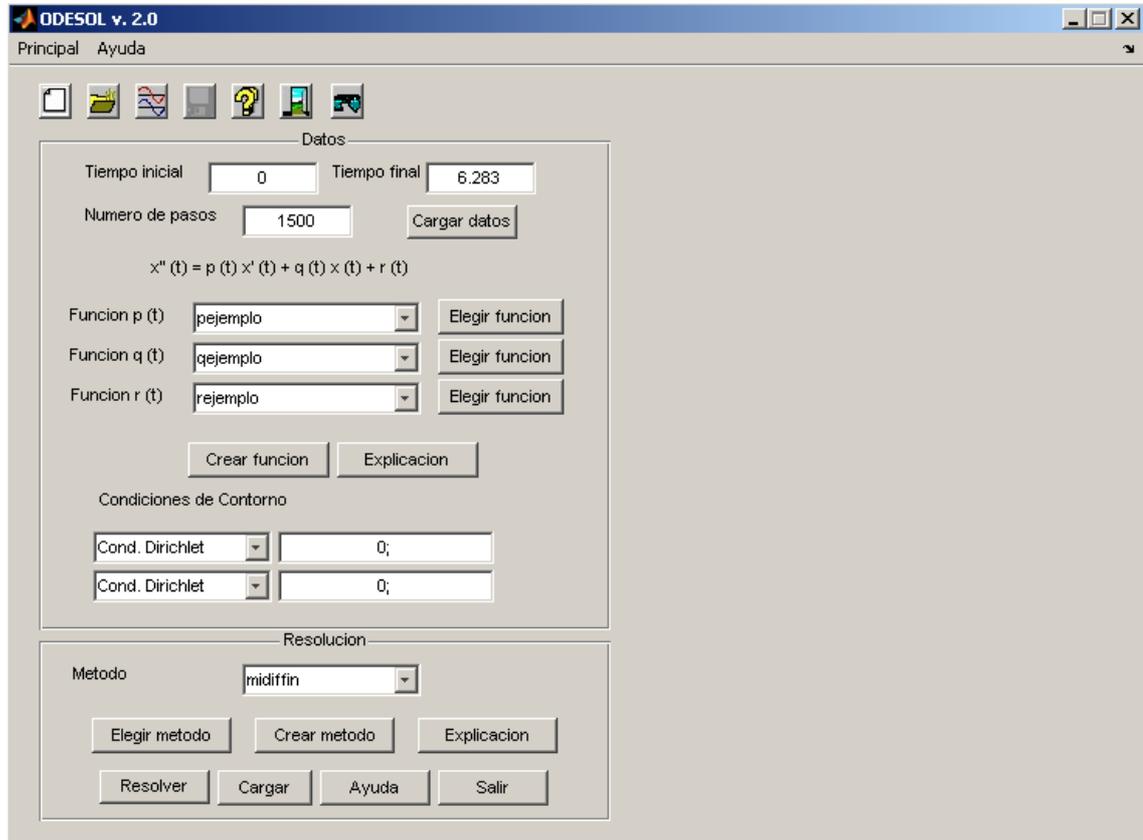
ventana con una breve descripción del método elegido y la definición de la función y sus parámetros de entrada.



```
Editor - F:\ODESQL\metodosPCL\midiffin.m
File Edit Text Cell Tools Debug Desktop Window Help
[Icons] Base >>
1 %Método de Diferencias Finitas para problemas Lineales
2 %
3 %function [t,x]=midiffin(p,q,r,t0,T,a,b,N,C1,C2)
4 function [t,x]=midiffin(p,q,r,t0,T,a,b,N,C1,C2)
midiffin Ln 7 Col 1 OVR
```

La salida de la función debe de ser un vector columna N por m con los tiempos en los que se ha calculado la solución, y otra variable x de dimensión N por m con las soluciones de la ecuación diferencial ordinaria.

Para resolver una ecuación diferencial ordinaria es necesario pasar al programa todos los datos mencionados. Debería parecerse a lo siguiente:



ODESQL v. 2.0
Principal Ayuda

Datos

Tiempo inicial Tiempo final

Numero de pasos

$x''(t) = p(t)x'(t) + q(t)x(t) + r(t)$

Funcion p (t)

Funcion q (t)

Funcion r (t)

Condiciones de Contorno

Cond. Dirichlet

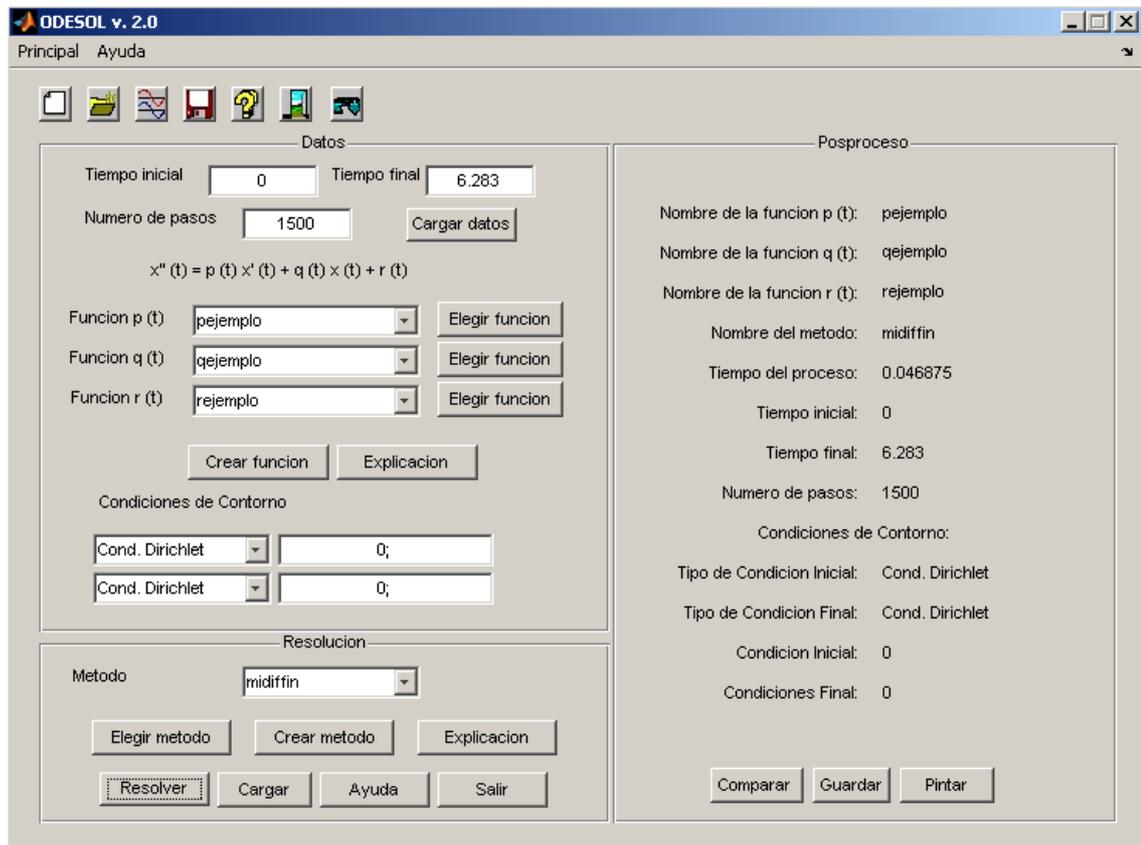
Cond. Dirichlet

Resolucion

Metodo

Ahora sencillamente presione el botón "resolver" para resolver la ecuación diferencial ordinaria.

Una vez resuelta le aparecerá:



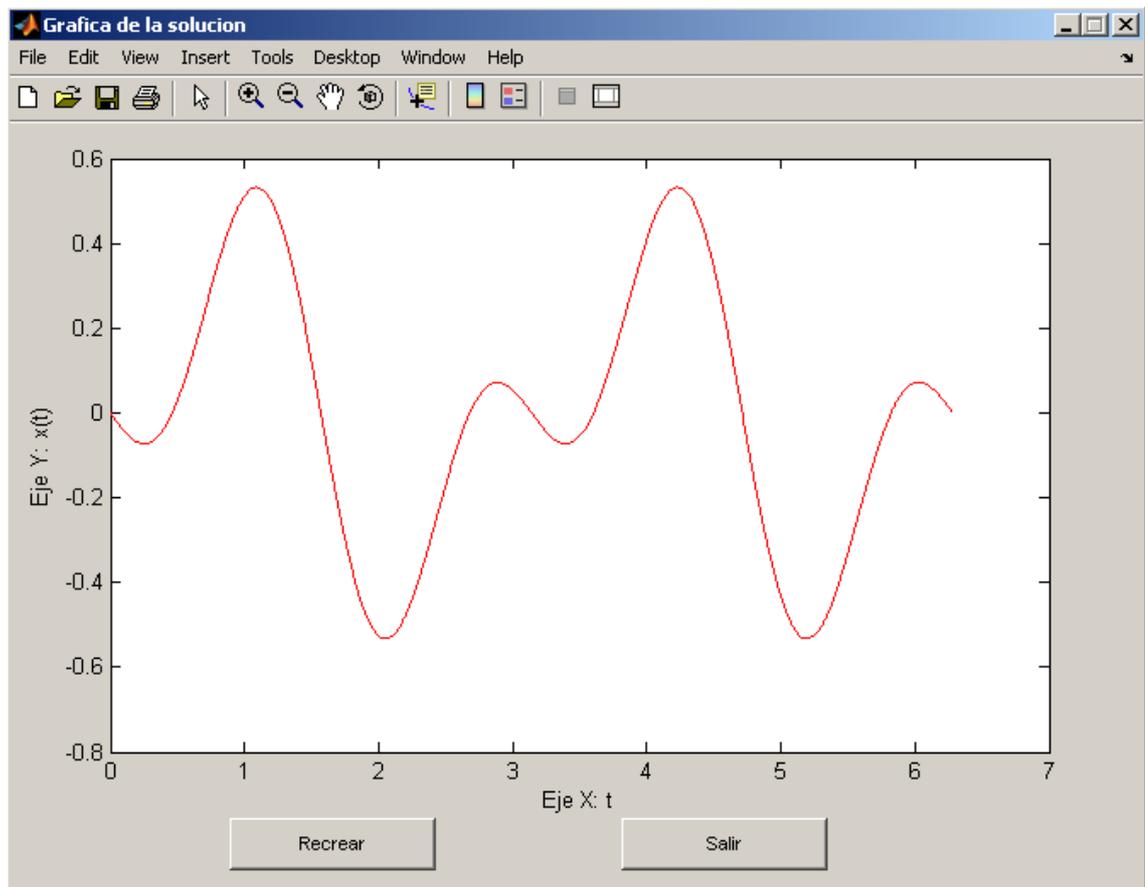
Observe como ahora dispone de un nuevo panel llamado "posproceso" con todos los datos del problema que pretendía resolver. En él se incluyen el tiempo en el que se ha llevado a cabo el proceso así como con qué método ha sido resuelto. Usted puede guardar todos los datos del problema sencillamente presionando el botón "guardar". En el archivo guardado, además de los datos que usted ve en pantalla, en el panel posproceso, se guardará el tiempo y la solución halladas. Usted podrá recuperar estos datos en sesiones posteriores con el programa ODESOL, además de poder exportar los resultados a otras aplicaciones. El archivo guardado contiene una variable llamada Sys del tipo struct en formato binario con los siguientes campos:

- Sys.x la solución del problema.
- Sys.t vector de tiempos donde se ha calculado la solución.
- Tanto x como t son vectores columna de la misma longitud.

- Sys.tiempo el tiempo que se ha tardado en calcular la solución.
- Sys.metodo el nombre del método con que se ha calculado la solución.
- Sys.funcion el nombre de la función p usada para resolver.
- Sys.funcion_q el nombre de la función q usada para resolver.
- Sys.funcion_r el nombre de la función r usada para resolver.
- Sys.N el número de pasos que se han dado para calcular la solución.
- Sys.t_ini tiempo inicial del problema.
- Sys.t_fin tiempo final del problema.
- El tiempo inicial y final dan el intervalo de definición del problema.
- Sys.x0 condición inicial del problema.
- Sys.xF condición final del problema.
- Sys.contorno tipo de condición inicial del problema.
- Sys.contorno_Fin tipo condición final del problema.

Usted puede cargar estos datos en el programa en otra sesión, o bien exportarlos a otra aplicación, por ejemplo en el propio Matlab. Si está trabajando en la ventana command window de MATLAB puede cargar estos datos desde el menú File y eligiendo Open o bien Import Data. También puede hacerlo usando el comando load de Matlab.

Usted puede pintar la solución del problema. Para ello sencillamente pulse el botón "pintar" , le aparecerá una nueva ventana:



En esta ventana usted podrá guardar la gráfica de la solución, podrá verla de distintas posiciones, ampliar zonas, y un largo etc. Además se ha incluido el botón "recrear" que le permite ver la solución en movimiento.

Nota importante: por cuestiones obvias sólo se podrán representar funciones de una, dos o tres dimensiones.

Por último el panel posproceso le brinda la oportunidad de comparar la solución hallada con la solución exacta de la ecuación diferencial ordinaria, por supuesto si se conoce. Para ello presione el botón "comparar", el programa le pedirá el fichero en el que se encuentra guardada la función de la solución exacta. Le aparecerá un nueva ventana con todas la opciones creadas para comparar las soluciones. Más abajo se explica como se puede hacer la comparación de los resultados, comparar un resultado con la solución exacta se hace de la misma manera.

Si usted quiere comparar soluciones resueltas con distintos métodos, guarde cada una de las soluciones en archivos distintos y presione el botón : . Le

aparecerá una nueva ventana, con los botones "cargar" cargue los ficheros en los que ha guardado las diferentes soluciones. Para proceder a la comparación es necesario que se haya resuelto la misma ecuación diferencial ordinaria en las mismas condiciones, es decir, mismas condiciones iniciales, mismo intervalo de definición y mismo número de pasos. Entonces podrá proceder a la comparación. Le aparecerá algo parecido a lo siguiente:

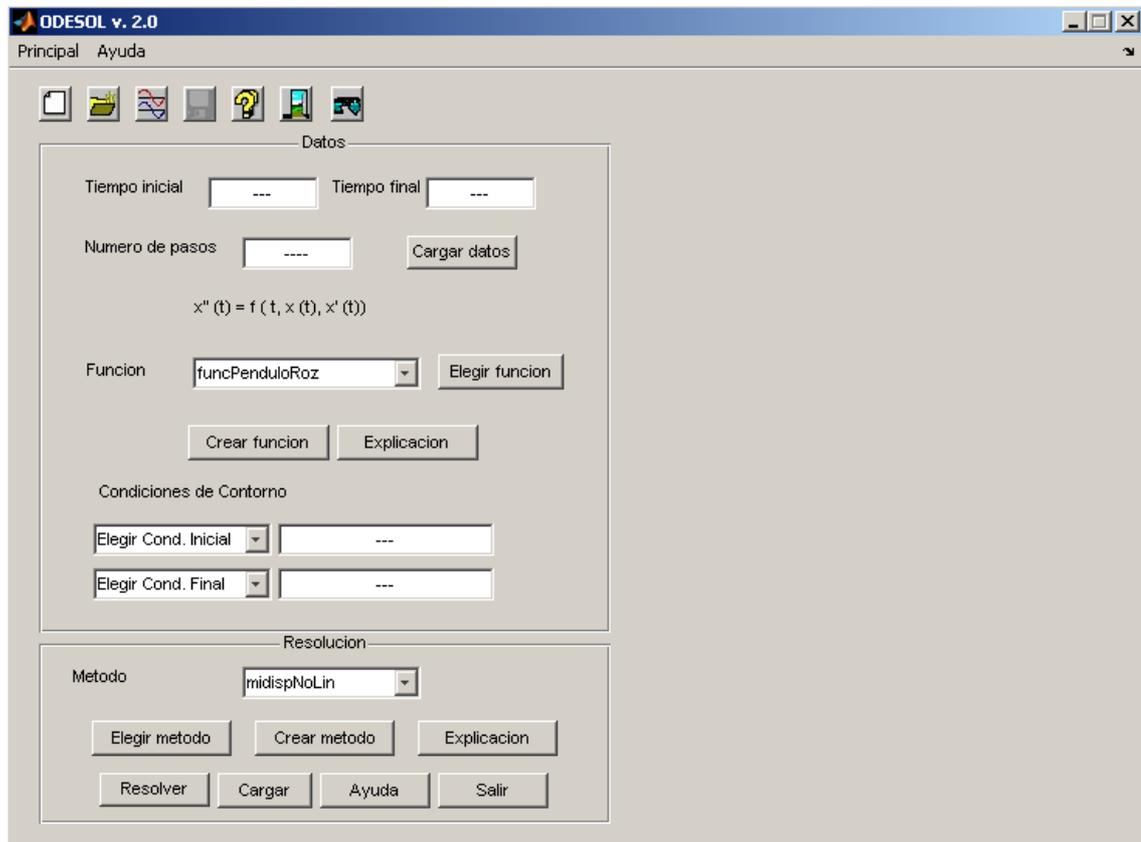


Le informará de cuál ha sido la diferencia máxima y le permitirá pintar las gráficas de la diferencia de las diferentes componentes incluido la diferencia en valor absoluto, para ello presione los botones "pintar componentes de la diferencia" o "diferencia en valor absoluto" respectivamente.

Para más información del programa vea las ayudas que se adjuntan con el paquete informático.

7. Trabajando con el módulo de resolución de Problemas de Contorno No Lineal de ODESOL

Cuando usted inicia el módulo de resolución de Problemas de Contorno No Lineales de ODESOL le aparece la siguiente ventana.

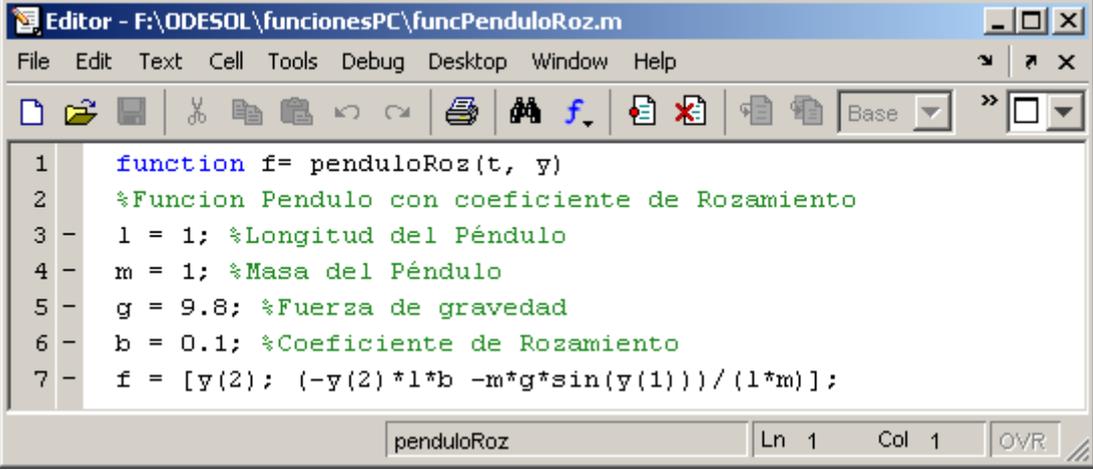


Como se puede ver en la imagen aparecen dos paneles: el primero es el panel en el que introduciremos los datos que definen la ecuación diferencial ordinarias, estos son: el tiempo inicial y final que nos da el intervalo de definición de la ecuación diferencial ordinaria. Las condiciones iniciales y las condiciones finales de la ecuación diferencial ordinaria ($x(t_0) = x_0$ y $x(T) = x_T$). El tipo de las condiciones iniciales y las condiciones finales, Dirichlet o Newmann. El número de pasos representa el número de nodos equiespaciados en los que se va a calcular la ecuación diferencial ordinaria. Usted dispone de un menú desplegable en el que puede elegir entre cualquiera de las funciones de las que tiene guardadas en la carpeta "funcionesPC" para elegir las función a resolver. Estas funciones tienen que tener un formato específico. Supongamos que tenemos la ecuación diferencial

ordinaria: $x''(t) = \frac{-x'(t) \cdot l \cdot b - m \cdot g \cdot \text{sen}(x(t))}{l \cdot m}$ (Ecuación del Péndulo, donde l es la longitud del Péndulo, m la masa del Péndulo, g la fuerza de gravedad y b el coeficiente de rozamiento) con condiciones iniciales:

$$\begin{cases} x(0) = 1; \\ x(25) = 0; \end{cases}$$

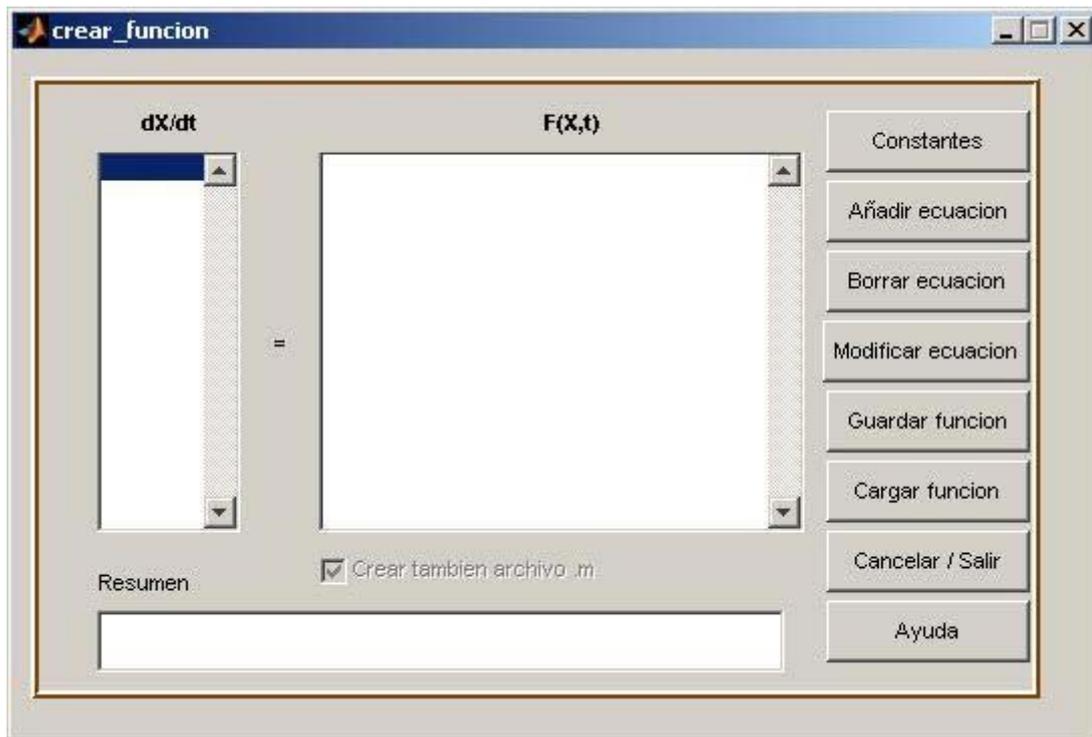
Ahora deberá escribir la función en archivos de tipo función (con extensión m) de MATLAB de la siguiente forma:



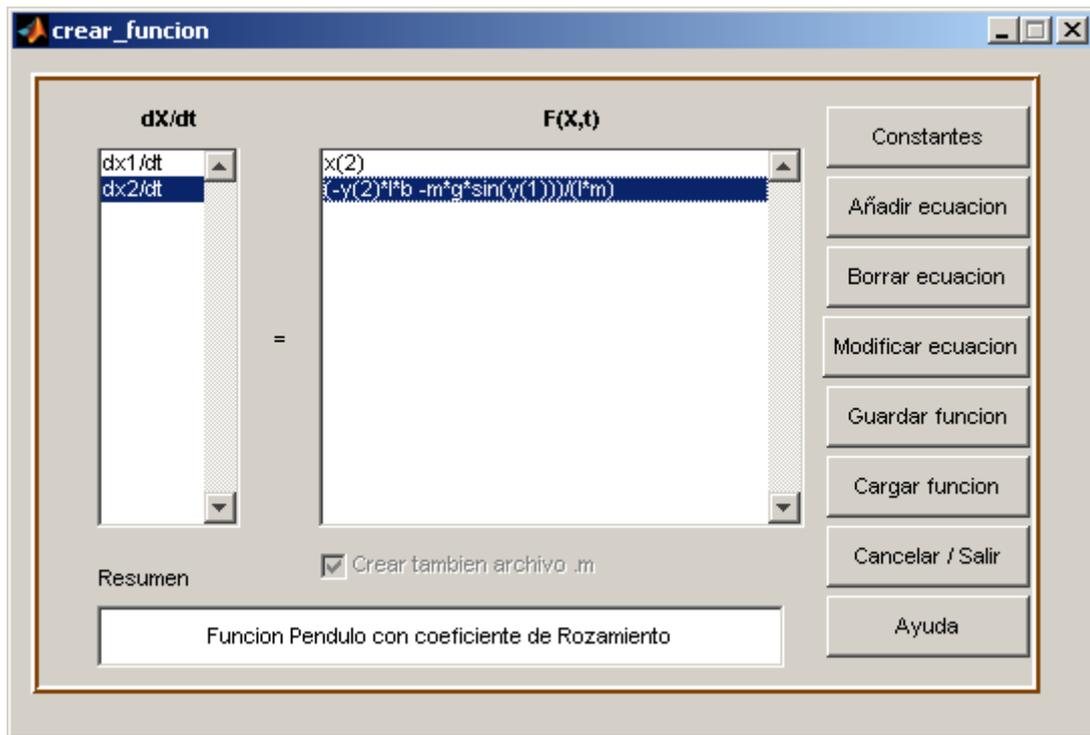
```

Editor - F:\ODESOL\funcionesPC\funcPenduloRoz.m
File Edit Text Cell Tools Debug Desktop Window Help
function f= penduloRoz(t, y)
%Funcion Pendulo con coeficiente de Rozamiento
l = 1; %Longitud del Péndulo
m = 1; %Masa del Péndulo
g = 9.8; %Fuerza de gravedad
b = 0.1; %Coeficiente de Rozamiento
f = [y(2); (-y(2) * l * b - m * g * sin(y(1))) / (l * m)];
  
```

Esta función sería reconocida por ODESOL . Obsérvese el orden de la variable de entrada t, así como la salida de la función, debe ser por columnas, por eso en el vector salida de la función f se utiliza ;. Para más información sobre cómo crear funciones con MATLAB mire la ayuda de MATLAB. De todos modos ODESOL le proporciona un editor de ecuaciones para poder introducir éstas de una forma cómoda. Si quisiésemos introducir esta misma función utilizando ODESOL sólo tendremos que presionar el botón "crear función" que se encuentra en el panel datos. Si usted presiona este botón le aparecerá:



Con este módulo puede ir introduciendo una a una cada ecuación del sistema equivalente. Para ello sólo tendrá que pulsar el botón "añadir ecuación" e introducir la ecuación. Además ODESOL le permite introducir constantes en las ecuaciones, para ello utilice una variable alfanumérica que no comience con un número. Más tarde presione el botón "constantes" y asígnele a cada variable que haya creado el valor que quiera utilizando el símbolo igual. Vea la siguiente imagen:



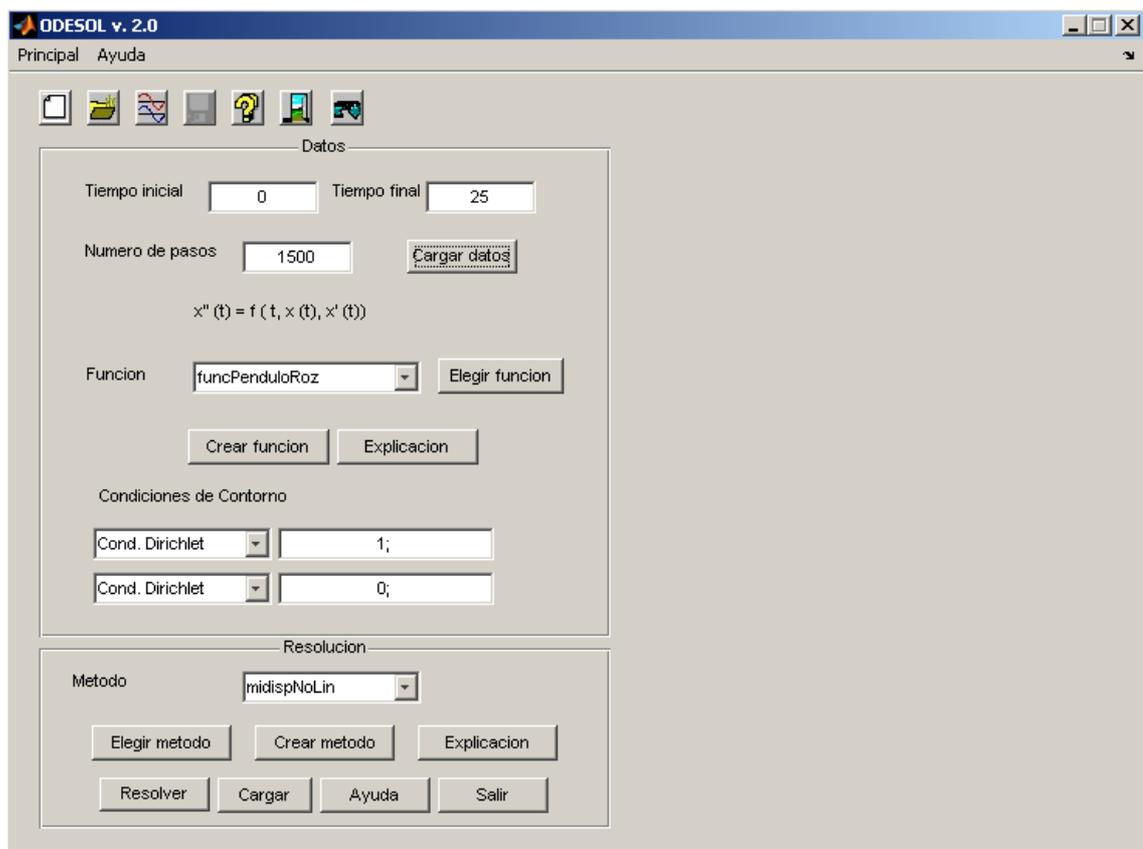
Ahora sólo tiene que presionar el botón "guardar" y llamar a la función como desee.

Ahora está en condiciones de resolver la ecuación diferencial. Para resolverla usted puede elegir uno de los métodos adjuntos al programa ODESOL o bien elegir cualquier método creado por usted. Si se decanta por este último caso tenga en cuenta que el método tiene que ser una función de MATLAB guardada en un archivo m que tenga como datos de entrada a la función las variables: *func*, del tipo char con el nombre de la función, *x0* y *xF* que son escalares, que representan la condición inicial y la condición final de la ecuación diferencial ordinaria, *int*, que se trata de un vector numérico de dimensión uno por dos y representa el intervalo de definición de la función, la variable *N*, que representa el número de nodos equiespaciados en los que se calculará la solución, y por último los valores del tipo de condición de contorno inicial (0 si es de tipo Dirichlety 1 si es de tipo Newmann) y de tipo de condición final (0 si es de tipo Dirichlety 1 si es de tipo Newmann). Para ver los parámetros anteriormente comentados y su orden, puede seleccionar un método para la resolución y pulsar el botón de explicación en el panel de Resolución. El programa le mostrará una nueva ventana con una breve descripción del método elegido y la definición de la función y sus parámetros de entrada.

```
Editor - F:\ODESOL\metodosPC\midispNoLin.m*
File Edit Text Cell Tools Debug Desktop Window Help
1 %Método del Disparo para problemas No Lineales
2 %
3 %function [tdisplin,xdisplin]=midisplin(f, interval, a,b,N,C1,C2)
4 function [tdisplin,xdisplin]=midispNoLin(f, interval, a,b,N,C1,C2)
5
midispNoLin Ln 7 Col 1 OVR
```

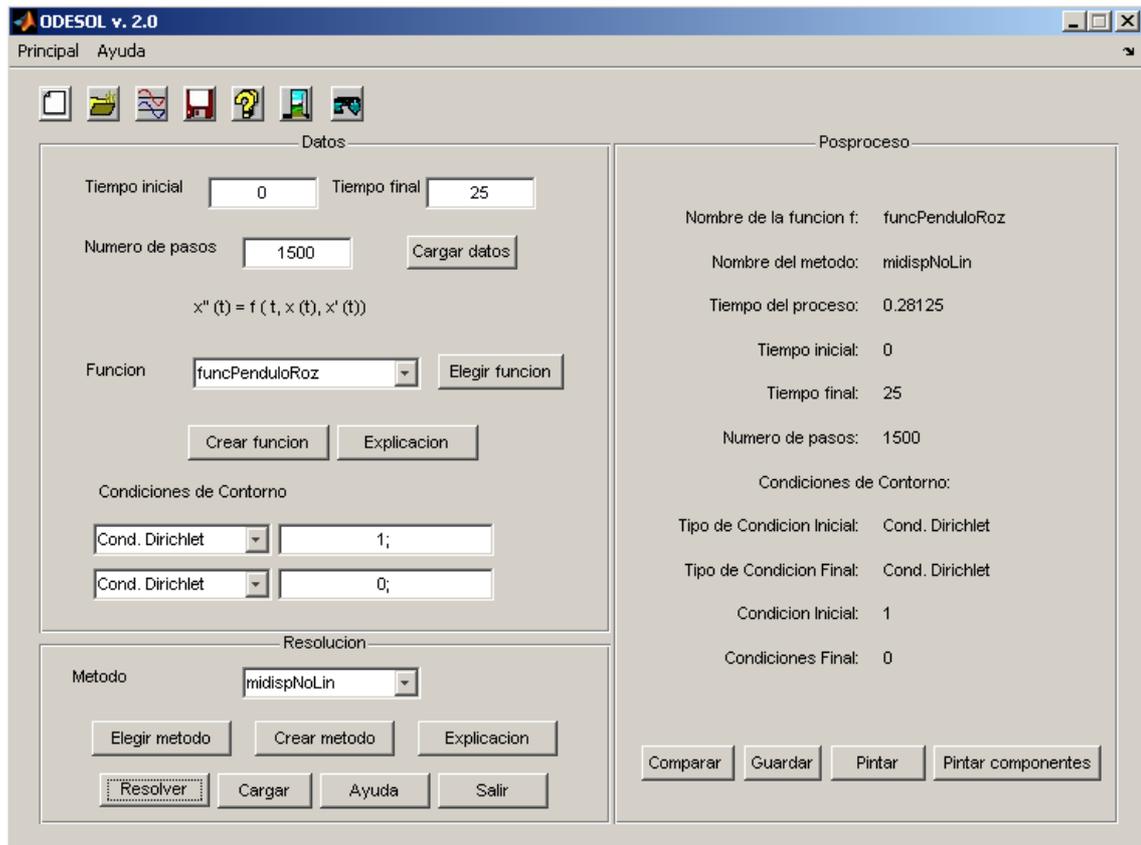
La salida de la función debe de ser un vector columna N por m con los tiempos en los que se ha calculado la solución, y otra variable x de dimensión N por m con las soluciones de la ecuación diferencial ordinaria.

Para resolver una ecuación diferencial ordinaria es necesario pasar al programa todos los datos mencionados. Debería parecerse a lo siguiente:



Ahora sencillamente presione el botón "resolver" para resolver la ecuación diferencial ordinaria.

Una vez resuelta le aparecerá:



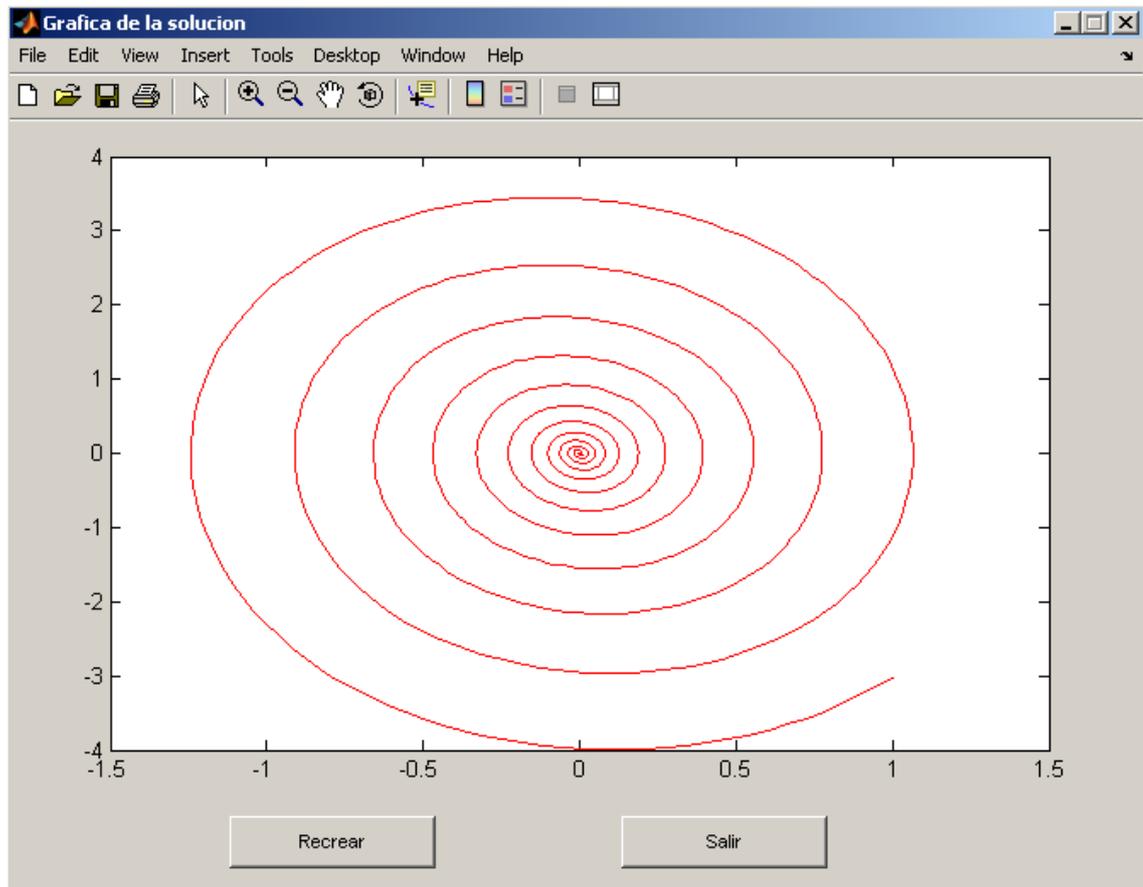
Observe como ahora dispone de un nuevo panel llamado "posproceso" con todos los datos del problema que pretendía resolver. En él se incluyen el tiempo en el que se ha llevado a cabo el proceso así como con qué método ha sido resuelto. Usted puede guardar todos los datos del problema sencillamente presionando el botón "guardar". En el archivo guardado, además de los datos que usted ve en pantalla, en el panel posproceso, se guardará el tiempo y la solución halladas. Usted podrá recuperar estos datos en sesiones posteriores con el programa ODESOL, además de poder exportar los resultados a otras aplicaciones. El archivo guardado contiene una variable llamada Sys del tipo struct en formato binario con los siguientes campos:

- Sys.x la solución del problema.
- Sys.t vector de tiempos donde se ha calculado la solución.
- Tanto x como t son vectores columna de la misma longitud.
- Sys.tiempo el tiempo que se ha tardado en calcular la solución.
- Sys.metodo el nombre del método con que se ha calculado la solución.
- Sys.funcion el nombre de la función usada para resolver.

- Sys.N el número de pasos que se han dado para calcular la solución.
- Sys.t_ini tiempo inicial del problema.
- Sys.t_fin tiempo final del problema.
- El tiempo inicial y final dan el intervalo de definición del problema.
- Sys.x0 condición inicial del problema.
- Sys.xF condición final del problema.
- Sys.contorno tipo de condición inicial del problema.
- Sys.contorno_Fin tipo condición final del problema.

Usted puede cargar estos datos en el programa en otra sesión, o bien exportarlos a otra aplicación, por ejemplo en el propio Matlab. Si está trabajando en la ventana command window de MATLAB puede cargar estos datos desde el menú File y eligiendo Open o bien Import Data. También puede hacerlo usando el comando load de Matlab.

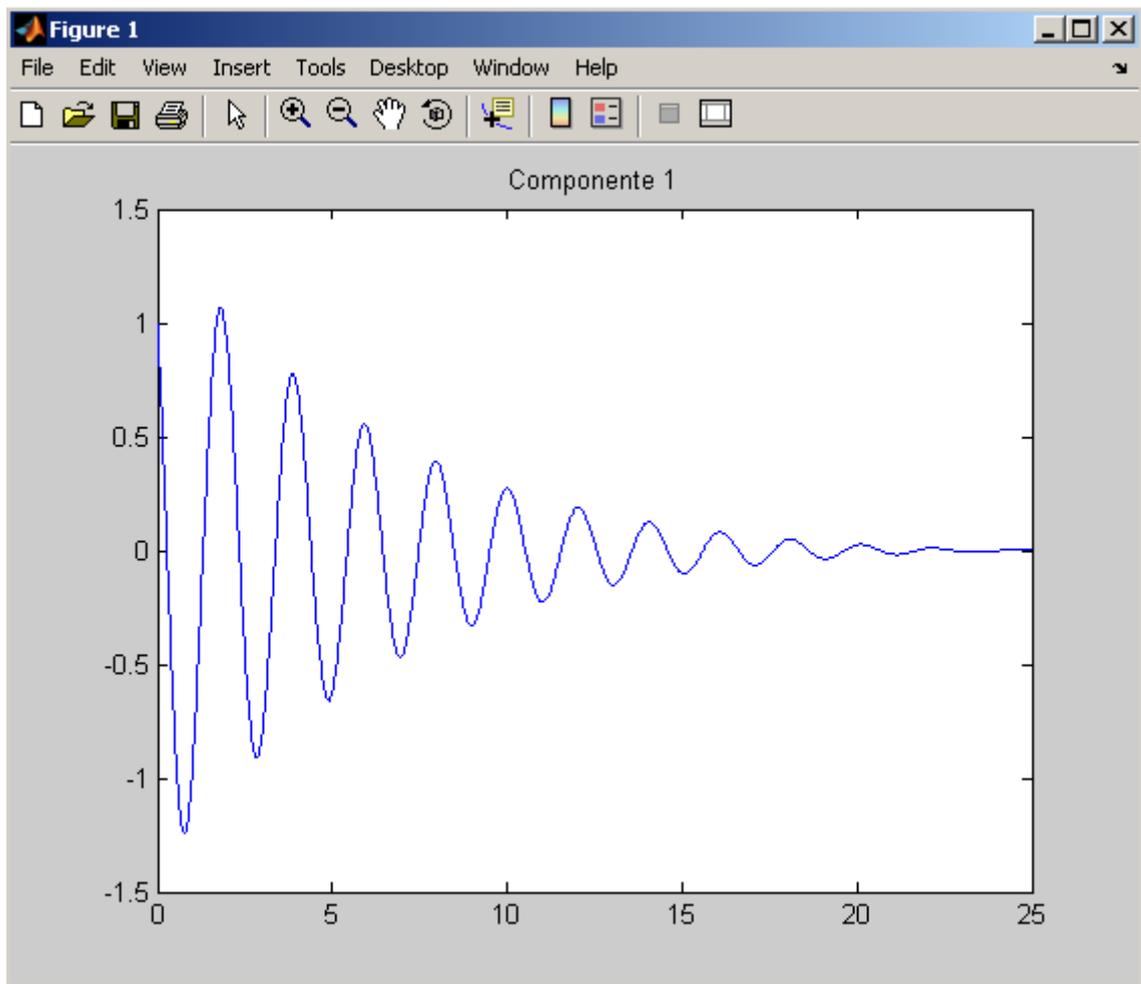
Usted puede pintar la solución del problema. Para ello sencillamente pulse el botón "pintar" , le aparecerá una nueva ventana:



En esta ventana usted podrá guardar la gráfica de la solución, podrá verla de distintas posiciones, ampliar zonas, y un largo etc. Además se ha incluido el botón "recrear" que le permite ver la solución en movimiento.

Nota importante: por cuestiones obvias sólo se podrán representar funciones de una, dos o tres dimensiones.

Además de pintar la solución usted podrá pintar cada una de las componentes por separado utilizando el botón "pintar componentes". Si presiona este botón le aparecerá una nueva ventana en la que podrá elegir qué componentes pintar y en qué color:



Al igual que antes usted podrá guardar estas gráficas para incorporarlas en sus trabajos.

Por último el panel posproceso le brinda la oportunidad de comparar la solución hallada con la solución exacta de la ecuación diferencial ordinaria, por supuesto si se conoce. Para ello presione el botón "comparar", el programa le pedirá el fichero en el que se encuentra guardada la función de la solución exacta. Le aparecerá un nueva ventana con todas la opciones creadas para comparar las soluciones. Más abajo se explica como se puede hacer la comparación de los resultados, comparar un resultado con la solución exacta se hace de la misma manera.

Si usted quiere comparar soluciones resueltas con distintos métodos, guarde cada una de las soluciones en archivos distintos y presione el botón : . Le aparecerá una nueva ventana, con los botones "cargar" cargue los ficheros en los que

ha guardado las diferentes soluciones. Para proceder a la comparación es necesario que se haya resuelto la misma ecuación diferencial ordinaria en las mismas condiciones, es decir, mismas condiciones iniciales, mismo intervalo de definición y mismo número de pasos. Entonces podrá proceder a la comparación. Le aparecerá algo parecido a lo siguiente:

Comparacion de resultados

No hay suficientes datos para comparar

Metodo 1

Nombre de la funcion: -----
Nombre del metodo: -----
Tiempo del proceso: -----
Tiempo inicial: -----
Tiempo final: -----
Numero de pasos: -----
Condiciones de Contorno:
Tipo de Condicion Inicial: -----
Tipo de Condicion Final: -----
Condiciones iniciales: -----
Condiciones Final: -----

Cargar Pintar Pintar componentes

Metodo 2

Nombre de la funcion: -----
Nombre del metodo: -----
Tiempo del proceso: -----
Tiempo inicial: -----
Tiempo final: -----
Numero de pasos: -----
Condiciones de Contorno:
Tipo de Condicion Inicial: -----
Tipo de Condicion Final: -----
Condiciones iniciales: -----
Condiciones Final: -----

Cargar Pintar Pintar componentes

Salir

Le informará de cuál ha sido la diferencia máxima y le permitirá pintar las gráficas de la diferencia de las diferentes componentes incluido la diferencia en valor absoluto, para ello presione los botones "pintar componentes de la diferencia" o " diferencia en valor absoluto" respectivamente.

Para más información del programa vea las ayudas que se adjuntan con el paquete informático.

Novedades con respecto a ODESOL v 1.0

En esta nueva versión se han incluido las siguientes mejoras:

- Nuevo modulo para la resolución de Problemas de Contorno Lineales.
- Nuevo módulo para la resolución de Problemas de Contorno No Lineales.
- Nueva versión estudiante, con métodos y funciones en archivos con código protegido.
- Posibilidad de introducir métodos con código protegido (archivos .p)
- Posibilidad de introducir funciones con código protegido.
- Icono de acceso directo a ODESOL v 2.0.
- Icono de acceso directo a Ayuda.

En todos los nuevos módulos de ODESOL v 2.0 se ha mantenido siempre el mismo formato y modo de uso que en la versión 1.0.