Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Ficha de trabajo en clase 1
Introducción a MATLAB
04 de octubre de 2007

A) Escalares, vectores, matrices

En MATLAB, el símbolo "=" se llama operador de asignación. Asigna a una variable cuyo nombre está especificado en la parte izquierda del operador un valor numérico especificado en la parte derecha del operador.

Ejercicio 1: Definir dos variables a y b cuyos valores asignados sean 4 y 5. Calcular $a+b$, $a-b$, $b-a$, ab, a/b, a^b y b^a. Definir una variable x que tome el valor $2a+5b$. Redefinir x como $3(x+6(b-a))$.

En MATLAB, el número iracional π toma la forma de una variable predefinida denotada por `pi`*. El formato numérico por defecto es* `short` *(punto fijo con 4 dígitos decimales). Otros formatos numéricos:* `long` *(punto fijo con 14 dígitos decimales),* `short e`*,* `long e`*,* `short g`*,* `long g`*,* `rational` *(se aproxima un número real por un cociente de dos enteros). Para cambiar el formato numérico, se usa el comando* `format`*, seguido por el nombre del formato numérico al que queremos pasar (por ejemplo,* `format long e`*).*

Ejercicio 2: Calcular π en cada uno de los formatos numéricos antes mencionados. Definir dos variables c y d cuyos valores sean $\sin(\pi/6)$ y $\tan(\pi/6)$ y calcular $c+d$, c^2+d^2, $1-c^2$, $\sqrt{c}$, e, e^d, $|-c-1|$, $\cos(\pi/2)/\sin(\pi)$ y $e/\sin(\pi)$.

Para crear una variable vectorial a partir de una lista de números o expresiones numéricas conocidas, se teclean sus componentes $e_1, \cdots, e_n$ separadas por coma (,) o por espacio dentro de un par de corchetes ([ ]):

$$vect = [e_1\ e_2\ \cdots\ e_n].$$

Para introducir un vector de paso constante h a partir de los extremos a y b se escribe entre corchetes o sin corchetes $a:h:b$

$$vect = [a:h:b] \qquad o \qquad vect = a:h:b.$$

Por $a:b$ se denota el vector de paso constante $h=1$.
Para definir una matriz, se teclean entre corchetes los vectores filas separados por punto y coma (los vectores fila se pueden dar en cualquiera de las formas antes mencionadas, es decir, por sus componentes o como vectores de paso constante):

$$matriz = [fila_1; fila_2; \cdots; fila_m].$$

El símbolo ' se usa para denotar la traspuesta de una matriz. Cuando entre dos matrices $A = (a_{ij})_{i=\overline{1,m},j=\overline{1,n}}$ y $B = (b_{ij})_{i=\overline{1,m},j=\overline{1,n}}$ hay una operación aritmetica precedida por punto (.), el resultado es una matriz $C = (c_{ij})_{i=\overline{1,m},j=\overline{1,n}}$, cuyas componentes c_{ij} se obtienen realizando dicha operación entre los correspondientes componentes a_{ij} y b_{ij}.
Para encontrar los autovalores y los autovectores de una matriz A se usa el comando `eig`*:*

$$[eigenvectors, eigenvalues] = eig(A),$$

donde eigenvectors es una matriz cuyas columnas son los autovectores de A y eigenvalues es una matriz diagonal, los componentes de la diagonal siendo los autovalores de la matriz A.

Ejercicio 3: Definir los vectores $vect_1 = (1,\ 7,\ 60/5-1,\ 7*3-5,\ 4^2-5)$, $vect_2 = [2:0{,}2:3]$, $vect_3 = [2:10]$, $vect_4 = [40:-2:30]$, $vect_5 = 55:-3:36$, $vect_6 = 3:12$, $vect_7 = 1:11$.
Definir $vect_8 = vect_6'$. Calcular $vect_6 + vect_8$, observar y corregir.
Calcular $vect_7+3$, $vect_6.*vect_7$, $vect_6./vect_7$, $(vect_6-vect_7).^3+9$, $vect_6.^2$, $vect_6-4$, $vect_6*vect_6$ y corregir los eventuales errores que aparecen.

Definir la matriz $A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$. Calcular A', $A*A'$ y $A.*A'$. Definir el vector $b = (1,\ 3,\ 5)$.
Calcular $[b;A]$ (*añade una fila dada por el vector* b *a la matriz* A) y $[b'\ A]$ (*añade una columna dada por* b' *a la matriz* A).

Redefinir la matriz $A = \begin{pmatrix} 1 & 4 & -3 \\ 2 & 1 & 5 \\ -2 & 5 & 3 \end{pmatrix}$. Definir B como la inversa de A (*a través del comando inv(A)*) y calcular $A*B$, $A-B$, $A+B$, B', B^3, $B.^3$. Resolver el sistema $A*x = b'$. Encontrar los autovalores y los autovectores de la matriz A.

Consideremos m, n dos enteros positivos. A través del comando `eye(n)` *se introduce la matriz identidad de tamaño n. Mediante los comandos* `ones(m,n)`, `zeros(m,n)` *se introducen matrices de m filas y n columnas con todos los elementos 1, respectivamente 0; cuando estos comandos sólo tienen un argumento (*`ones(n)`, `zeros(n)`*), se obtienen matrices cuadráticas de tamaño n con todos los elementos 1 o 0.*
Si x es un vector, el comando `diag(x)` *produce una matriz diagonal con el vector x en la diagonal principal. El comando* `diag(x,k)`*, donde k es un entero, produce una matriz cuadrada diagonal, con el vector x colocado en la k-esima diagonal inferior si k es negativo, en la k-esima diagonal superior si la k es positivo y en la diagonal principal si $k = 0$. Si A es una matriz, el comando diag(A) produce un vector columna que almacena la diagonal principal de dicha matriz.*
El comando `size(A)`*, donde A es una matriz $m \times n$, tiene como salida el vector $[m, n]$. El comando* `length(x)`*, donde x es un vector de n componentes, tiene a n como salida.*
Los comandos `triu(A)` *y* `tril(A)` *generan una matriz triangular superior, respectivamente inferior, a partir de una matriz cuadrada A.*

Ejercicio 4: Calcular $eye(5)$, $ones(2,3)$, $ones(5)$, $zeros(4,2)$, $zeros(4)$, $diag(vect_6)$, $diag(A)$, $diag(diag(A))$, $size(A)$, $length(vector_7)$, $size(vector_6)$, $triu(A)$, $tril(A)$, $size(vector_6')$, $diag(vect_6,-3)$. Definir una matriz $M = diag(ones(5,1)) + diag(ones(3,1),-2) + diag(ones(4,1),1)$.
Usar el comando `spy(M)` para visualizar en una figura los elementos no-triviales de la matriz M.

Si x es un vector de n componentes, el comando $x(j)$, $j = \overline{1,n}$, tiene como salida la j-esima componente del vector. Si A es una matriz $m \times n$, el comando $A(i,j)$, $i = \overline{1,m}$, $j = \overline{1,n}$, tiene como salida el elemento (i,j) de dicha matriz.
Denotemos por `index_1`, `index_2` *dos vectores de enteros positivos en el rango $\overline{1,m}$, respectivamente $\overline{1,n}$. El comando* `A(index_1, index_2)` *tiene como salida la matriz resultada al intersectar las filas de indices contenidos en* `index_1` *con las columnas de indices contenidos en* `index_2`.
`A(:,index_2)` *tiene como salida la matriz resultada al intersectar todas las filas con las columnas de indices contenidos en el vector* `index_2`.
`A(end,index_2)` *tiene como salida el vector resultado al intersectar la última fila con las columnas de indices contenidos en* `index_2`.

Ejercicio 5: Definir la matriz

$$A = \begin{pmatrix} 35 & 1 & 6 & 26 & 19 & 24 \\ 3 & 32 & 7 & 21 & 23 & 25 \\ 31 & 9 & 2 & 22 & 27 & 20 \\ 8 & 28 & 33 & 17 & 10 & 15 \end{pmatrix}$$

y el vector $x = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{pmatrix}$.

Calcular $x(4)$, $x(end)$, $x(2:5)$, $x(1:2:7)$, $x([2,4:end])$, $A(2,3)$, $A(3,1:4)$, $A(3,:)$, $A(:,4)$, $A(end,:)$, $A(end,3:5)$, $A(:,[1,\ 3:6])$, $A(1:3,3:6)$, $A([4,2],[3,1,5])$.

B) Representaciones gráficas.

Para realizar gráficos 2-dimensionales se usa el comando `plot(x,y,option)`*, donde x e y son vectores de la misma longitud y* `option` *es una opción que indica el color y el estilo de la línea.*

Ejercicio 6: Teclear el siguiente conjunto de comandos:

```
clear
z=[-3*pi:0.1:3*pi]
plot(z,sin(z),'r*')
hold on
```
(*este comando mantiene los dos dibujos en la misma figura*)
```
plot(z,cos(z),'bo').
```

Se pueden dar más detalles sobre una figura usando comandos del tipo `title('')` *- añade título al gráfico,* `legend` *- añade una leyenda al gráfico,* `grid` *- añade una cuadricula al gráfico,* `grid off` *- hace desaparecer la cuadricula.*

Para realizar peliculas, se usan los comandos `M=moviein(n)` *- reserva memoria para almacenar n frames,* `movie(M,i,j)` *- hace que visualicemos la pelicula almacenada en* `M` *i veces a la velocidad de j imagenes por segundo.*

Ejercicio 7: Teclear el siguiente conjunto de comandos:

```
clear
z=[-2*pi:0.1:2*pi];
M=moviein(17);
for j=1:17
f=sin(z+j*pi/17);
plot(z,f,'b*')
M(:,j)=getframe;
end
movie(M,10,7).
```

Los siguientes dos programas tienen como objetivos la inicialización al uso del Editor de MATLAB, a la definición de funciones en MATLAB y a las representaciones gráficas 3-dimensionales.

Ejercicio 8: Primero, vamos a definir una función `test3d` en un fichero llamado `test3d.m`:

```
function Z=test3d(X,Y)
Z=3*(1-X).^2.*exp(-X.^2-(1+Y).^2)-10*(X/5-X.^3-Y.^5).*exp(-X.^2-Y.^2)
        -1/3*exp(-(X+1).^2-Y.^2)
```

Observar que el nombre de la función debe coincidir con el nombre del fichero la guardamos.

Vamos a escribir un otro fichero tipo .m que realiza la vizualización 3-d de la función `test3d` antes definida:

```
clear all
u=[0:pi/20:6*pi];
figure(1)
```
(*se abre una ventana gráfica dividida en cuatro sub-ventanas dispuestas en dos lineas y dos columnas*)
```
subplot(2,2,1)
plot3(u,sin(u),cos(u),'b*')
```
(*el comando* `plot3` *facilita la visualización de curvas parametrizadas en tres dimensiones*)
```
x=[-3:0.1:3];
y=[-4:0.1:4];
[X,Y]=meshgrid(x,y);
```
(*el comando* `meshgrid`*, cuyos argumentos son dos vectores x e y de tamaño m y n, produce dos matrices $n \times m$, X, cuyas n filas son copias de x, e Y, cuyas m columnas son copias de y.*)

`Z=test3d(X,Y);` (*se aplica la función* `test3d` *a las matrices* X *e* Y, *obteniendose una matriz* $n \times m$, Z.)
`subplot(2,2,2)`
`meshc(x,y,Z)` (*los comandos* `mesh` *o* `meshc` *hace que la superficie sea visualizada como una malla; el segundo comando tiene la ventaja de que se puede visualizar tambien la forma de las curvas de nivel. El comando* `waterfall` *hace visualizar la superficie como una malla unidireccional.*)
`subplot(2,2,3)`
`surfc(Z)`(*los comandos* `surf` *o* `surfc` *hace que visualicemos el gráfico como una superficie continua; el segundo comando tiene la ventaja de que se puede visualizar tambien la forma de las curvas de nivel.*)
`subplot(2,2,4)`
`contour3(x,y,Z,16)` (*los comandos* `contour` *y* `contour3` *ayudan a visualizar las curvas de nivel de la superficie como proyección en el plano* xOy, *respectivamente en tres dimensiones.*)

Referencias

[1] Amos Gilat, Matlab, *Una introducción con ejemplos prácticos*, Editorial Reverté, Barcelona

[2] Javier García de Jalón, José Ignacio Rodríguez, Jesús Vidal, *Aprenda Matlab 7.0 como si estuviera en primero*, Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingenieros Industriales,
http://mat21.etsii.upm.es/ayudainf/aprendainf/Matlab70/matlab70primero.pdf

[3] David Griffiths, *An introduction to Matlab, Version 2.3*,
http://www.maths.dundee.ac.uk/∼ftp/na-reports/MatlabNotes.pdf

[4] Desmond J. Highham, Nicholas Highham, *Matlab guide*,
www.uam.es/personal_pdi/ciencias/fquiros/Numerico2_03_04/mg_final.pdf

[5] Maria Dolores Cárdenas, *Matlab como ayuda al estudiante en Ciencias Matemáticas*,
http://www.uam.es/personal_pdi/ciencias/fquiros

[6] *Getting Started with Matlab 7*,
http://www.mathworks.com/acces/helpdesk/help/pdf_doc/matlab/getstart.pdf

[7] Juan Antonio Infante, José Maria Rey, *Introducción a Matlab*,
http://www.mat.ucm.es/∼jair/matlab/notas.pdf

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Aproximaciones de soluciones de ecuaciones diferenciales ordinarias
11 de octubre de 2007

Ficha de trabajo en clase 2:

Problema 1: Programar el método del punto fijo para aproximar la solución de la ecuación diferencial ordinaria

$$\begin{cases} \dot{x}(t) = \sin(x(t)) & t \in (0,T) \\ x(0) = \pi/2. \end{cases} \tag{1}$$

donde $(0,T)$ es el intervalo donde la existencia local de la solución exacta está garantizada por el Teorema del punto fijo de Banach.

Calcular la solución exacta y dibujar las soluciones exacta y aproximada.
Realizar una tabla para comparar la solución exacta en los nodos y la solución aproximada.
Hallar y dibujar el error cometido.

Problema 2: Programar los métodos de Euler implícito y explícito para el problema (1).

Calcular la solución exacta y dibujar las soluciones exacta y aproximada.
Realizar una tabla para comparar la solución exacta en los nodos y la solución aproximada.
Hallar y dibujar el error cometido.

Problemas para efectuar en casa

Problema 1: Hacer un programa para representar graficamente en la misma figura, pero en sub-ventanas distintas las siguientes funciones:

a. $f(x) = x^2 + 4\sin(2x) - 1$, para $x \in [-3,3]$

b. $f(x,y) = 2^{-\sqrt{x^2+y^2}} \sin(x)\cos(y/2)$, para $x \in [-3,3]$, $y \in [-3,3]$, usando el comando `surf`.

c. $f(x,y) = \sin(\sqrt{x^2+y^2})/\sqrt{x^2+y^2}$, para $x, y \in [-8,8]$, usando el comando `meshc`.

d. la curva parametrizada $(\sqrt{t}\sin(t), \sqrt{t}\cos(t), t/2)$, para $t \in [0, 6\pi]$, usando el comando `plot3`.

Problema 2: Programar el método del punto fijo para aproximar la solución de la ecuación diferencial ordinaria

$$\begin{cases} \dot{x}(t) = x(t) & t \in (0,T) \\ x(0) = 1, \end{cases} \tag{2}$$

$$\begin{cases} \dot{x}(t) = x^3(t) & t \in (0,T) \\ x(0) = 1, \end{cases} \tag{3}$$

donde $(0,T)$ es el intervalo donde la existencia local de la solución exacta está garantizada por el Teorema del punto fijo de Banach para el problema (2) y un subintervalo estrictamente incluido en el intervalo maximal de existencia para el problema (3).

Calcular la solución exacta y dibujar las soluciones exacta y aproximada.
Realizar una tabla para comparar la solución exacta en los nodos y la solución aproximada.
Hallar y dibujar el error cometido.

Problema 3: Programar los métodos de Euler implícito y explícito para los problemas (2) y (3).

Calcular la solución exacta y dibujar las soluciones exacta y aproximada.
Realizar una tabla para comparar la solución exacta en los nodos y la solución aproximada.
Hallar y dibujar el error cometido.

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Método de diferencias finitas para problemas elípticos 1-d
30 de octubre de 2007

Ficha de trabajo en clase 3:
Problema 1: Se considera la ecuación de Poisson con condiciones de tipo Dirichlet

$$\begin{cases} -u''(x) = \sin(4\pi x), \quad 0 < x < 1, \\ u(0) = 0, \quad u(1) = 0. \end{cases} \tag{1}$$

Se pide:

1. Hallar la solución exacta.
2. Aproximar la solución usando el método de diferencias finitas.
3. Realizar una tabla comparando la solución exacta en los nodos y la solución aproximada, con $N = 10, 20, 40, 80$.
4. Para cada valor de N, hallar el error cometido en norma infinito.
5. Dibujar las soluciones exacta y aproximada para todos los valores de N mencionados.
6. Dibujar el error cometido en función de N.

Problema 2: Se considera la ecuación de Poisson con condiciones de contorno tipo Dirichlet

$$\begin{cases} -u''(x) + c(x)u(x) = f(x), \quad a < x < b, \\ u(a) = \alpha, \quad u(b) = \beta. \end{cases} \tag{2}$$

En las aplicaciones, c y f son funciones continuas, con $c \geqslant 0$. Se pide:

1. Aproximar la solución usando el método de diferencias finitas.
2. Aplicar el algoritmo anterior al caso particular:
$$a = 0, \quad b = 2, \quad u(0) = 10, \quad u(2) = \frac{10e^2}{21}, \quad c(x) = \frac{2}{(x+0{,}1)^2} \text{ y } f(x) = \frac{2e^x}{(x+0{,}1)^2} - \frac{e^x}{x+0{,}1}.$$
3. En este caso particular, la solución explícita es $u(x) = \frac{e^x}{x+0{,}1}$. Realizar una tabla comparando la solución explícita en los nodos y la solución aproximada, con $N = 10, 20, 40, 80$ puntos interiores.
4. Para cada valor de N, hallar el error cometido en norma infinito.
5. Dibujar las soluciones exacta y aproximada para los valores de N antes mencionados.
6. Dibujar el error cometido en función de N.

Problema 3: Se considera el problema espectral asociado al laplaciano $1 - d$ con condiciones de contorno de tipo Dirichlet homogeneas:

$$\begin{aligned} -\varphi_{xx} = \lambda\varphi \qquad x \in (0,1) \\ \varphi(0) = \varphi(1) = 0. \end{aligned} \tag{3}$$

Se pide:

1. Aproximar el problema espectral (3) mediate el método de diferencias finitas.
2. Dibujar los autovalores aproximados y los primeros N autovalores del problema continuo y observar la posición de las dos curvas de autovalores.
3. Dibujar los autovectores aproximados (en una pelicula).

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Método de diferencias finitas para problemas elípticos 1-d con condiciones de contorno mixtas
6 de noviembre de 2007

Ficha de trabajo en clase 4:
Se considera el problema de contorno con condiciones de tipo Dirichlet:

$$\begin{cases} -u''(x) + p(x)u'(x) + q(x)u(x) = f(x), \quad a < x < b, \\ u(a) = \alpha, \quad u(b) = \beta. \end{cases} \tag{1}$$

Supondremos que p, q y f son funciones continuas. Si $q \geqslant 0$, usando la alternativa de Fredholm y el principio de máximo, la solución del problema (1) es única.
Problema 1: Construir un esquema en diferencias finitas del mayor orden posible para el problema (1). Implementar en Matlab este esquema, considerando los siguientes casos particulares:

a) $p = 0$, $q = 1$, $f(x) = 2\cos(x)$, $u(0) = 2$, $u(1) = e + \cos(1)$

b) $p(x) = \sin(x)$, $q = 1$, $f(x) = \cos^2(x)$, $u(0) = u(\pi/2) = 0$

c) $p(x) = \sin(x)$, $q = 1$, $f(x) = \cos^2(x)$, $u(0) = 1$, $u(\pi/2) = 2$

Se considera el siguiente problema con condiciones de contorno mixtas:

$$\begin{cases} -u''(x) + p(x)u'(x) + q(x)u(x) = f(x), & a < x < b, \\ \alpha_1 u(a) + \alpha_2 u'(a) = \alpha, \quad \beta_1 u(b) + \beta_2 u'(b) = \beta. \end{cases} \tag{2}$$

Estudiemos el ejemplo $\alpha_1 = \beta_2 = 1$ y $\alpha_2 = \beta_1 = 0$, que corresponde a una condición de Dirichlet a la izquierda y condición de Neumann a la derecha.
A partir de este caso particular, construir un esquema en diferencias finitas para aproximar la solución del problema (2). Usar tanto diferencias centradas, como progresivas y regresivas para aproximar $u'(b)$ y concluir que tipo de aproximación tiene ventajas desde el punto de vista del orden de convergencia.
En el caso de las diferencias centradas o progresivas para la derivada en b, observar que se introduce un nodo ficticio y una incognita más. Para tener el mismo número de incognitas y ecuaciones, hay que introducir una ecuación más, que se obtiene pidiendo que la ecuación discretizada se verifique en el $N+1$-ésimo nodo (en el modelo continuo, esto se corresponde a observar que podemos construir un problema con condiciones de Dirichlet en el intervalo $(a, 2b-a)$ a partir de los datos f, p, q prolongados por paridad en el intervalo $(b, 2b-a)$, cuya solución, por ser única, coincide con la solución del problema puesto en (a, b)). Las mismas ideas se pueden emplear para tratar condiciones de contorno más generales (con α_i, β_i, $i = \overline{1,2}$, números reales cualquieros).
Problema 2: Se considera el caso particular:

$$a = -1/2, \quad b = 3/2, \quad u'(-1/2) = u'(3/2) = 0, \quad p = 0, \quad q = 1 \text{ y } f(x) = (\pi^2 + 1)\sin(\pi x) + 2.$$

Se pide:

1. Aproximar la solución usando el método de diferencias finitas. Para aproximar las condiciones de contorno utilizar

   a) diferencias centradas, introduciendo los nodos ficticios x_{-1} y x_{N+2} fuera del intervalo.

   b) diferencia progresiva en el extremo izquierdo y diferencia regresiva en el derecho (sin nodos ficticios)

2. Realizar una tabla que contenga la solución exacta y las soluciones aproximadas por los métodos a) y b) para $N = 10, 20, 40, 80$ nodos interiores.

3. Dibujar las soluciones aproximadas correspondientes a los métodos a) y b) y la solución exacta dada por $u(x) = 2 + \sin(\pi x)$, para todos los valores anteriores de N.

4. Calcular los errores cometidos en los métodos a) y b) y dibujarlos en función de N.

5. Fijemos $N = 100$ nodos interiores. Realizar una tabla comparativa y dibujar las soluciones aproximadas cuando se toma q constante, de valores comprendidos entre 10^{-4} y 10^4.

Problema 3: Aproximar mediante diferencias finitas la solución del problema (2) con $p = 0$, $q = 1$, $f(x) = 2\cos(x)$ y con las condiciones de contorno

a) $u'(0) = 1$, $u'(1) = e - \sin(1)$

b) $2u'(0) - u(0) = 0$, $(e + \cos(1))u'(1) - (e - \sin(1))u(1) = 0$.

Comparar las aproximaciones obtenidas con la solución exacta, $u(x) = e^x + \cos(x)$.

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Relación de dispersión. Velocidad de grupo y velocidad de fase
8 de noviembre de 2007

Ficha de trabajo en clase 5:
Problema 1: Hallar las relaciones de dispersión para las ecuaciones de transporte, calor, ondas y Schrodinger continuas.
Problema 2: Hallar las relaciones de dispersión para las semi-discretizaciones espaciales de estas ecuaciones. Dibujarlas frente a las continuas en el intervalo espacial $[-\pi/h, \pi/h]$.
Problema 3: Hallar las relaciones de dispersión para las aproximaciones completamente discretas abajo enumeradas. Dibujar las relaciones de dispersión non-disipativas (con parte imaginaria cero) frente a las continuas, para $\lambda = 0{,}5$ y para $\lambda = 2$ y marcar las diferencias.
Problema 4: Hallar las velociades de grupo y de fase para todas las ecuaciones continuas antes mencionadas y las correspondientes velocidades de fase y de grupo para los esquemas semi-discretos y completamente discretos (los no-disipativos). Dibujar las velocidades de fase y de grupo semi-discretas y completamente discretas frente a las continuas.

Aproximaciones completamente discretas
En lo que sigue, h es el paso espacial, k es el paso temporal, u_j^n es una aproximación de $u(jh, nk)$.
Esquemas numéricos para la ecuación del transporte (en todos estos esquemas, $\lambda = k/h$ es una constante)

- **Euler explícito:** $u_j^{n+1} = u_j^n + \frac{1}{2}\lambda(u_{j+1}^n - u_{j-1}^n)$
- **Euler implícito:** $u_j^{n+1} = u_j^n + \frac{1}{2}\lambda(u_{j+1}^{n+1} - u_{j-1}^{n+1})$
- **Leap frog:** $u_j^{n+1} = u_j^{n-1} + \lambda(u_{j+1}^n - u_{j-1}^n)$
- **Crank-Nicolson:** $u_j^{n+1} = u_j^n + \frac{1}{4}\lambda(u_{j+1}^n - u_{j-1}^n) + \frac{1}{4}\lambda(u_{j+1}^{n+1} - u_{j-1}^{n+1})$
- **Lax-Friedrich:** $u_j^{n+1} = \frac{1}{2}(u_{j+1}^n + u_{j-1}^n) + \frac{1}{2}\lambda(u_{j+1}^n - u_{j-1}^n)$
- **Upwind:** $u_j^{n+1} = u_j^n + \lambda(u_{j+1}^n - u_j^n)$
- **Lax-Wendroff:** $u_j^{n+1} = u_j^n + \frac{1}{2}\lambda(u_{j+1}^n - u_{j-1}^n) + \frac{1}{2}\lambda^2(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$

Esquemas numéricos para la ecuación del calor (en todos estos esquemas, $\lambda = k/h^2$ es una constante)

- **Euler explícito:** $u_j^{n+1} = u_j^n + \lambda(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$
- **Euler implícito:** $u_j^{n+1} = u_j^n + \lambda(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1})$
- **Leap frog:** $u_j^{n+1} = u_j^{n-1} + 2\lambda(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$
- **Crank-Nicolson:** $u_j^{n+1} = u_j^n + \frac{1}{2}\lambda(u_{j+1}^n - 2u_j^n + u_{j-1}^n) + \frac{1}{2}\lambda(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1})$

Esquemas numéricos para la ecuación de ondas (en todos estos esquemas, $\lambda = k/h$ es una constante)

- **Leap frog:** $u_j^{n+1} - 2u_j^n + u_j^{n-1} = \lambda^2(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Discretizaciones por diferencias finitas de los problemas de evolución 1-d
13 de noviembre de 2007

Ficha de trabajo en clase 6:
Problema 1: Ecuación de transporte
Se considera la ecuación de transporte

$$\left\{\begin{array}{cc} u_t - u_x = 0 & x \in (0,1), \quad t > 0 \\ u(x,0) = \psi(x) & x \in (0,1) \\ u(0,t) = u(1,t) = 0 & t > 0. \end{array}\right. \tag{1}$$

Vamos a estudiarla con dos datos iniciales,

$$\psi(x) = \sin(\pi x) \quad y \quad \psi(x) = \mathbb{I}_{[1/3,2/3]} = \left\{\begin{array}{ll} 1 & x \in [1/3,2/3] \\ 0 & x \notin [1/3,2/3] \end{array}\right.$$

Se pide:

1. Discretizar totalmente el problema usando un método explícito (por ejemplo: Euler explícito), un método implícito (por ejemplo: Euler implícito) para la derivada en tiempo y el método de diferencias finitas centradas y progresivas para la derivada en espacio. Estudiar la consistencia y estabilidad.
2. Realizar una película que represente las soluciones aproximadss obtenidas mediante los métodos numéricos empleados.

Problema 2: Ecuación de convección-difusión estacionaria
Se considera el problema:

$$\left\{\begin{array}{l} -\epsilon u_{xx} + u_x = 0 \quad x \in [0,1] \\ u(0) = 0, \quad u(1) = 1. \end{array}\right. \tag{2}$$

- Encuentra la solución exacta.
- Estudia la aproximación del problema mediante el método de diferencia finitas centradas, progresivas o regresivas, en el método numérico programado, y de la solución numérica obtenida.
- Dibujar las soluciones exacta y aproximada para cada uno de los métodos para $N = 20, 40, 80, 100$.
- Hacer el estudio anterior para diferentes valores de ϵ, (por ejemplo: $\epsilon = 0{,}1, 0{,}01, 0{,}001$).

Problema 3: Ecuación de convección-difusión
Se considera el siguiente problema de convección-difusión:

$$\left\{\begin{array}{l} u_t = \epsilon u_{xx} + u_x \quad (0,1) \times (0,T) \\ u(0,t) = 0, u(1,t) = 1, \\ u(x,0) = sin(\pi x). \end{array}\right. \tag{3}$$

- Utilizar esquemas implícitos y explícitos para la resolución en tiempo, diferencias finitas centradas y progresivas para la derivada espacial de orden uno.
- Dibujar la solución aproximada para cada uno de los métodos para N=20,40,80,100.

- Rehacer el estudio anterior para distintos valores de ϵ, (por ejemplo: $\epsilon = 0{,}1, 0{,}01, 0{,}001$).

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Diferencias finitas para problemas elipticos 2-d en un cuadrado
22 de noviembre de 2007

Ficha de trabajo en clase 7: Consideramos el siguiente problema elíptico con condiciones de contorno de Dirichlet en un cuadrado:

$$\begin{cases} -u_{xx} - u_{yy} = f, & (x,y) \in (a,b) \times (c,d) \\ u(x,0) = \psi_1(x), & x \in [a,b] \\ u(x,1) = \psi_2(x), & x \in [a,b] \\ u(0,y) = \phi_1(y), & y \in [c,d] \\ u(1,y) = \phi_2(y), & y \in [c,d]. \end{cases} \tag{1}$$

Se pide:

a) Tomando los datos $f, \psi_1, \psi_2, \phi_1, \phi_2$ continuos, aproximar la solución del problema (1) mediante el método de diferencias finitas $2-d$.

   Para esto, hay que dividir el intervalo (a,b) en $M+1$ subintervalos de longitud $h = (b-a)/(M+1)$ y el intervalo (c,d) en $N+1$ subintervalos de longitud $\widetilde{h} = (d-c)/(N+1)$ y considerar $u_{j,k}$ como una aproximación de $u(jh, k\widetilde{h})$, para $j = \overline{0, M+1}$, $k = \overline{0, N+1}$.

   Teneis que reducir el problema continuo a un sistema de la forma $L\overline{u} = F$, donde L es la matriz del laplaciano discreto, $\overline{u}$ es el vector de incognitas de $M \times N$ componentes y F es una discretización de la función f a la cual hay que sumar las contribuciones de las condiciones de contorno.

b) Tomar $a = c = 0$, $b = d = 1$, $h = \widetilde{h} = 1/(N+1)$, donde $N = 10, 20, 30, 40$ (el número de nodos interiores). En el caso particular $f = 0$, $\psi_1 = \phi_1 = 0$, $\phi_2(y) = \sin(\pi y/10)/\sin(\pi/10)$, $\psi_2(x) = \sinh(\pi x/10)/\sinh(\pi/10)$ dibujar la solución aproximada y la solución exacta,

$$u(x,y) = \frac{\sinh(\pi x/10)\sin(\pi y/10)}{\sinh(\pi/10)\sin(\pi/10)}.$$

c) Usando el comando `spy`, para cada valor de N, representar gráficamente la estructura de la matriz laplaciana L.

d) Para cada valor de N, hallar el error cometido en norma l^∞ y dibujar el vector $log(error)$ frente al vector $log(N+1)$. Son consistentes los resultados con lo que cabría esperar teóricamente?

Pasos que hay que seguir para resolver el item a):

- Ordenar el vector de incógnitas:

$$\overline{u} = (u_{1,1}, \cdots, u_{M,1}, u_{1,2}, \cdots, u_{M,2}, \cdots, u_{1,N}, \cdots, u_{M,N})'.$$

- Construir el problema discreto

$$\begin{cases} -\frac{u_{j+1,k} - 2u_{jk} + u_{j-1,k}}{h^2} - \frac{u_{j,k+1} - 2u_{j,k} + u_{j,k-1}}{\widetilde{h}^2} = f_{j,k} = f(jh, k\widetilde{h}), & j = \overline{1, M}, k = \overline{1, N} \\ u_{0,k} = \phi_1(k\widetilde{h}), & k = \overline{1, N} \\ u_{M+1,k} = \phi_2(k\widetilde{h}), & k = \overline{1, N} \\ u_{j,0} = \psi_1(jh), & j = \overline{1, M} \\ u_{j,N+1} = \psi_2(jh), & j = \overline{1, M}. \end{cases} \tag{2}$$

- Construir la matriz L. Para esto, consideramos las matrices auxiliares

$$A = \begin{pmatrix} \frac{2}{h^2}+\frac{2}{\widetilde{h}^2} & -\frac{1}{h^2} & 0 & \cdots & 0 & 0 \\ -\frac{1}{h^2} & \frac{2}{h^2}+\frac{2}{\widetilde{h}^2} & -\frac{1}{h^2} & \cdots & 0 & 0 \\ 0 & -\frac{1}{h^2} & \frac{2}{h^2}+\frac{2}{\widetilde{h}^2} & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \ddots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & \frac{2}{h^2}+\frac{2}{\widetilde{h}^2} & -\frac{1}{h^2} \\ 0 & 0 & 0 & \cdots & -\frac{1}{h^2} & \frac{2}{h^2}+\frac{2}{\widetilde{h}^2} \end{pmatrix} \in \mathcal{M}_M \text{ and } I_{\widetilde{h}} = \frac{1}{\widetilde{h}^2} I_M,$$

donde $I_M \in \mathcal{M}_M$ es la matriz identidad. Entonces

$$L = \begin{pmatrix} A & -I_{\widetilde{h}} & 0 & \cdots & 0 & 0 \\ -I_{\widetilde{h}} & A & -I_{\widetilde{h}} & \cdots & 0 & 0 \\ 0 & -I_{\widetilde{h}} & A & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \ddots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & A & -I_{\widetilde{h}} \\ 0 & 0 & 0 & \cdots & -I_{\widetilde{h}} & A \end{pmatrix} \in \mathcal{M}_{MN}.$$

Es decir, la matriz L es una matriz tridiagonal por bloques, dividida en N^2 bloques $M \times M$.

- Construir el vector F, donde

$$F_{(k-1)M+j} = \begin{cases} F_{1,1} = f_{1,1} + \frac{\phi_1(\widetilde{h})}{h^2} + \frac{\psi_1(h)}{\widetilde{h}^2}, & \text{si } j,k=1 \\ F_{j,1} = f_{j,1} + \frac{\psi_1(jh)}{\widetilde{h}^2}, & \text{si } k=1, j=\overline{2,M-1} \\ F_{M,1} = f_{M,1} + \frac{\phi_2(\widetilde{h})}{h^2} + \frac{\psi_1(Mh)}{\widetilde{h}^2}, & \text{si } j=M, k=1 \\ F_{1,k} = f_{1,k} + \frac{\phi_1(k\widetilde{h})}{h^2}, & \text{si } j=1, k=\overline{2,N-1} \\ F_{M,k} = f_{M,k} + \frac{\phi_2(k\widetilde{h})}{h^2}, & \text{si } j=M, k=\overline{2,N-1} \\ F_{1,N} = f_{1,N} + \frac{\phi_1(N\widetilde{h})}{h^2} + \frac{\psi_2(h)}{\widetilde{h}^2}, & \text{si } j=1, k=N \\ F_{j,N} = f_{j,N} + \frac{\psi_2(jh)}{\widetilde{h}^2}, & \text{si } j=\overline{2,M-1}, k=N \\ F_{M,N} = f_{M,N} + \frac{\phi_2(N\widetilde{h})}{h^2} + \frac{\psi_2(Mh)}{\widetilde{h}^2}, & \text{si } j=M, k=N \\ F_{j,k} = f_{j,k}, & \text{si } j=\overline{1,M}, k=\overline{1,N}, \text{ en el resto de los casos.} \end{cases}$$

Por tanto, $F = (F_{1,1}, \cdots, F_{M,1}, F_{1,2}, \cdots, F_{M,2}, \cdots, F_{1,N}, \cdots, F_{M,N})'$.

- Hallar la solución: $\overline{u} = L^{-1}F$.

- Reorganizar el vector $\overline{u}$ en una matriz $U = (U_{k,j}) = (u_{j+M(k-1)})$, con $k = \overline{1,N}$, $j = \overline{1,M}$. Hay que bordar esta matriz con los vectores correspondientes a las condiciones de contorno.

- Dibujar la solución: $\texttt{mesh}(x, y, U)$, donde $x = [a : h : b]$ e $y = [c : \widetilde{h} : d]$.

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

Problemas espectrales
10 de diciembre de 2007

Ficha de trabajo en clase 8:

Problema 1: Encontrar mediante ondas planas la forma de las soluciones del problema espectral continuo $1-d$:

$$\begin{cases} \phi \in H_0^1(0,1), \lambda \in \mathbf{R}_+ \\ -\phi_{xx} = \lambda\phi, \quad x \in (0,1) \end{cases} \tag{1}$$

y del problema espectral aproximado por diferencias finitas $1-d$:

$$\begin{cases} (\phi_j^h)_{j\in\overline{0,N+1}}, \quad \lambda^h \in \mathbf{R}_+ \\ -\frac{\phi_{j+1}^h - 2\phi_j^h + \phi_{j-1}^h}{h^2} = \lambda^h \phi_j^h, \quad j = \overline{1,N} \\ \phi_0^h = \phi_{N+1}^h = 0, \end{cases} \tag{2}$$

donde $h = 1/(N+1)$.

Dibujar en la misma ventana las dos curvas de autovalores para los problemas (1) y (2) y hacer una pelicula que recorra todos los autovectores. Encontrar las semejanzas y las diferencias entre las soluciones de los problemas espectrales (1) y (2).

Aproximar la solución del problema

$$\begin{cases} -u_{xx} = f, \quad x \in (0,1), \quad f \in L^2(0,1), \\ u(0) = u(1) = 0 \end{cases} \tag{3}$$

usando series de Fourier. Tomar $f(x) = e^x$, encontrar la solucion exacta del problema (3) y compararla con la obtenida mediante series de Fourier.

Problema 2: Encontrar mediante ondas planas la forma de las soluciones del problema espectral $2-d$:

$$\begin{cases} \phi \in H_0^1((0,1)\times(0,1)), \lambda \in \mathbf{R}_+ \\ -(\phi_{xx} + \phi_{yy}) = \lambda\phi, \quad (x,y) \in (0,1)\times(0,1) \end{cases} \tag{4}$$

y del problema espectral aproximado por diferencias finitas $2-d$:

$$\begin{cases} (\phi_{j,k}^h)_{j,k\in\overline{0,N+1}}, \quad \lambda^h \in \mathbf{R}_+ \\ -\frac{\phi_{j+1,k}^h + \phi_{j-1,k}^h - 4\phi_{j,k}^h + \phi_{j,k+1}^h + \phi_{j,k-1}^h}{h^2} = \lambda^h \phi_{j,k}^h, \quad j,k = \overline{1,N} \\ \phi_{0,k}^h = \phi_{N+1,k}^h = \phi_{j,0} = \phi_{j,N+1} = 0, \quad j,k = \overline{0,N+1} \end{cases} \tag{5}$$

donde $h = 1/(N+1)$.

Dibujar en la misma ventana las superficies que contienen los autovalores de los problemas (4) y (5) y hacer una pelicula que recorra todos los autovectores. Discutir las semejanzas y las diferencias entre las soluciones de los problemas espectrales (4) y (5).

Problema 3: Se considera la ecuación del calor estacionaria en el espacio determinado por la corona esférica $1 < r < 2$:

$$\begin{cases} \frac{1}{r^2}\frac{d}{dr}(r^2\frac{dT}{dr}) = 0, \quad 1 < r < 2 \\ T(1) = 2, \quad T(2) = 100. \end{cases} \tag{6}$$

Aproximar la solución del problema (6) mediante diferencias finitas y dibujarla frente a la corona esférica.

Sucesiones de comandos para dibujar frente a la corona esférica:

a)$r = [1 : h : 2];$
$\theta = [0 : h_\theta : 2 * \pi];$
for $\quad j = 1 : \mathrm{length}(\theta)$
plot3$(r * \cos((j)), r * \sin((j)), T,'\text{ opcion color}')$
hold on
end.

b)$r = [1 : h : 2];$
$\theta = [0 : 2 * \pi * h : 2 * \pi];$
$X = r' * \cos(\theta);$ - matriz (un vector columna por un vector fila de la misma dimensión)
$Y = r' * \sin(\theta);$
for $\quad j = 1 : length(r)$
$Z(j, :) = T(j) * ones(1, length(r));$
end
contourf(X, Y, Z)
colorbar$('\text{horiz}')$.

El comando contourf(X, Y, Z)**:** X, Y e Z son matrices de la misma dimensión. Se representa en los puntos (X_{ij}, Y_{ij}) la altura Z_{ij} en un cierto color conforme a una escala de colores, que se puede visualizar mediante el comando **colorbar**. El en el dibujo aparecen las curvas de nivel de la superficie dada por la matriz Z.

Docencia 2007-2008
Máster en Matemáticas y Aplicaciones
Métodos numéricos
Prácticas con MATLAB

El método de elementos finitos $1-d$ para problemas elípticos
13 de diciembre de 2007

Ficha de trabajo en clase 9:

Problema 1: Calcular las matrices de masa M_h y de rigidez R_h correspondientes al método de los elementos finitos $1-d$ P_1.

Problema 2: Sea $V_h = \{v \in H_0^1(0,1) | v|_{[x_j,x_{j+1}]} \in P_1\}$. Encontrar mediante ondas planas las soluciones del problema espectral

$$\begin{cases} \lambda^h \in \mathbf{R}_+, \phi^h \in V_h \\ \int_a^b \phi_x^h v_x \, dx = \lambda^h \int_a^b \phi^h v \, dx, \quad \forall v \in V_h. \end{cases} \tag{1}$$

Dibujar en el mismo dibujo la curva de autovalores correspondiente al método de los elementos finitos y la curva de los primeros N autovalores continuos. Encontrar diferencias con respeto a las soluciones del problema espectral aproximado por diferencias finitas.

Problema 3: Se considera la ecuación de Poisson en un segmento con condiciones de tipo Dirichlet

$$\begin{cases} -u''(x) + c(x)u(x) = f(x), \quad a < x < b, \\ u(a) = \alpha, \quad u(b) = \beta. \end{cases} \tag{2}$$

En este ejercicio vamos a aproximar la solución usando el método de elementos finitos. Aplicaremos el programa al caso particular:

$$a = 0, \quad b = 2, \quad u(0) = 10, \quad u(2) = \frac{10e^2}{21}, \quad c(x) = 0 \text{ y } f(x) = \frac{2e^x(x-0{,}9)}{(x+0{,}1)^3} - \frac{e^x}{x+0{,}1}.$$

Antes de resolver numéricamente el problema (2), hay que reducirlo a uno equivalente en el que las condiciones de contorno sean homogéneas. Para ello se introduce una función $g(x)$ y con las siguientes propiedades:

$$g(a) = \alpha, \quad g(b) = \beta.$$

De esta forma, podemos escribir

$$u = v + g, \quad v \in H_0^1(a,b)$$

y el problema queda reducido a buscar $v \in H_0^1(a,b)$ tal que

$$\begin{cases} -v''(x) + c(x)v(x) = \hat{f}(x), \quad a < x < b, \\ v(a) = 0, \quad v(b) = 0, \end{cases} \tag{3}$$

donde $\hat{f} = f - (-g'' + cg)$