

**ME´TODOS MATEMA´TICOS APLICADOS A LA INGENIER´IA**

**EJERCICIOS Y PROBLEMAS RESUELTOS**

*M´etodos Matem´aticos aplicados a la Ing. de Materiales M´etodos Num´ericos en el M´aster en Ing. Industrial*

A. I. Mun˜oz Montalvo, A. Nolla de Celis, E. Schiavi

Septiembre 2022

### @2022. Autores: A. I. Mun˜oz Montalvo, A. Nolla, E. Schiavi.

### Algunos derechos reservados.

### Este documento se distribuye bajo la licencia internacional Creative Commons Attribution-ShareAlike 4.0 International License.

### Disponible en: <http://creativecommons.org/licenses/by-sa/4.0/>

### Publicado en: https://burjcdigital.urjc.es

### <http://hdl.handle.net/10115/20115>

2

**´Indice general**

1. [Introducci´on](#_bookmark0) 5
2. [Resoluci´on de ecuaciones no lineales](#_bookmark1) 7
   1. [Problemas resueltos](#_bookmark2) 7
   2. [Problemas propuestos](#_bookmark5) 18
   3. [Aplicaciones](#_bookmark17) 22
3. [Problemas de Valor Inicial](#_bookmark19) 25
   1. [Problemas resueltos](#_bookmark20) 25
   2. [Problemas propuestos](#_bookmark26) 38
   3. [Aplicaciones](#_bookmark44) 45
4. [Problemas de Contorno](#_bookmark51) 49
   1. [Problemas resueltos](#_bookmark52) 49
      1. [Problemas de Transporte estacionarios en dominios uni-](#_bookmark53) [dimensionales](#_bookmark53) 49
      2. [Problemas de Transporte estacionarios en dominios bidi-](#_bookmark56) [mensionales](#_bookmark56) 55
      3. [Problemas evolutivos en una dimensi´on espacial](#_bookmark61) 63
   2. [Problemas propuestos](#_bookmark65) 80
      1. [Problemas de Transporte estacionarios en dominios uni-](#_bookmark66) [dimensionales](#_bookmark66) 80
      2. [Problemas de Transporte estacionarios en dominios bidi-](#_bookmark73) [mensionales](#_bookmark73) 82
      3. [Problemas evolutivos en una dimensi´on espacial](#_bookmark80) 85
   3. [Aplicaciones](#_bookmark89) 91
5. [Soluciones a los problemas propuestos](#_bookmark92) 93
   1. [Resoluci´on de ecuaciones no lineales](#_bookmark93) 93
   2. [Problemas de Valor Inicial](#_bookmark94) 95
   3. [Problemas de Contorno](#_bookmark104) 102

3

4 *´INDICE GENERAL*

|  |  |  |  |
| --- | --- | --- | --- |
| [**6.**](#_bookmark113) | [**Ap´endice A. Comandos de Octave/Matlab**](#_bookmark113) | | **113** |
|  | [6.1. Funciones](#_bookmark114) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 113 |
|  | [6.1.1. Definici´on](#_bookmark115) . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 113 |
|  | [6.1.2. Gr´aficas](#_bookmark116) . . . . . . . . . . . . . . . . . . . . . . . . . . . . | | 114 |
|  | [6.1.3. Presentaci´on de gr´aficas](#_bookmark117) . . . . . . . . . . . . . . . . . . . | | 115 |
|  | [6.2. Comandos y funciones u´tiles](#_bookmark118) . . . . . . . . . . . . . . . . . . . . . | | 116 |
|  | [6.3. Ecuaciones no lineales](#_bookmark119) . . . . . . . . . . . . . . . . . . . . . . . .  [6.4. Sistemas de Ecuaciones no lineales.](#_bookmark120)  [M´etodo de Newton-Raphson.](#_bookmark120) . . . . . . . . . . . . . . . . . . . . | | 120  121 |
|  | [6.5. Problemas de Valor Inicial](#_bookmark121) . . . . . . . . . . . . . . . . . . . . . . | | 122 |
|  | [6.6. Problemas de Contorno](#_bookmark122) . . . . . . . . . . . . . . . . . . . . . . . | | 123 |
|  | [6.6.1. Problemas de transporte estacionario 1-dimensionales](#_bookmark123) . . | | 123 |
|  | [6.6.2. Problemas de difusi´on evolutiva 1-dimensionales](#_bookmark124) . . . . . | | 124 |
|  | [6.6.3. Problemas de transporte estacionario 2-dimensionales](#_bookmark125) . . | | 125 |
| [**7. Ap´endice B. F´ormulas en diferencias finitas**](#_bookmark126) | |  | **127** |
| [7.1. F´ormulas en diferencias finitas 1-dimensionales](#_bookmark127) | | . . . . . . . . . . | 127 |
| [7.2. F´ormulas en diferencias finitas bidimensionales](#_bookmark128) | | . . . . . . . . . . | 128 |

#### [Bibliograf´ıa](#_bookmark128) 129

**Cap´ıtulo 1**

**Introduccio´n**

Esta colecci´on de problemas constituye la segunda parte del libro *“M´etodos Ma- tem´aticos para los grados en ingenier´ıa. Primera parte: Teor´ıa”*, de E. Schiavi,

A. I. Mun˜oz y C. Conde [[2].](#_bookmark130) Contiene numerosos problemas y ejercicios, tanto resueltos como propuestos, sobre los temas desarrollados en la primera parte.

Para los problemas marcados con *(Octave)* es necesario el uso del software ma- tem´atico libre Octave para resolverlos, aunque todas las funciones y comandos usados en este libro funcionan tambi´en para MATLAB.

Los c´odigos de las funciones usadas en este libro se pueden encontrar en [[3],](#_bookmark131) publicado online en [https://burjcdigital.urjc.es](https://burjcdigital.urjc.es/). Las funciones a las que hacemos referencia son:

|  |  |  |  |
| --- | --- | --- | --- |
| metbiseccion.m | metsecante.m | heun.m | ecupoisson.m |
| metnewton1ec.m | metregulafalsi.m | rungekuttao3.m | heaviside.m |
| metpuntofijo.m | eulerexplicito.m | bvpdirichlet.m |  |
| metodoaitken.m | eulerimplicito.m | bvp2cvrobinup.m |  |
| metnewtonsistema.m | cranknicolson.m | ecucalor.m |  |

La mayor parte de ellas son adaptaciones que hemos realizado de las funciones del libro *“C´alculo cient´ıfico con MATLAB y Octave”* de A. Quarteroni, F. Saleri [[1].](#_bookmark129)

5

6 *CAP´ITULO 1. INTRODUCCIO´N*

**Cap´ıtulo 2**

**Resolucio´n de ecuaciones no lineales**

# Problemas resueltos

**Ejercicio 2.1** *Encontrar una aproximaci´on de la ra´ız de la ecuaci´on:*

*x* 1 cos2(*x*) = 0*,*

*−*

2

*con un orden de error igual o inferior a* 2*·*10*−*2 *utilizando el m´etodo de bisecci´on.*

#### Soluci´on.

En primer lugar buscamos un intervalo (*a, b*) satisfaciendo *f* (*a*)*f* (*b*) *<* 0 para *f* (*x*) = *x* 1 cos2(*x*), puesto que garantiza la existencia de al menos una ra´ız en dicho intervalo. Se tiene que *f* (0*.*4) = 0*.*024 y *f* (0*.*5) = 0*.*114, por tanto, tomaremos el intervalo (0*.*4*,* 0*.*5) como intervalo de partida. A continuaci´on, ha- llamos el nu´mero m´ınimo de iteraciones que tenemos que realizar para conseguir que el error sea inferior a 2 *·* 10*−*2,

2

*−*

*−*

*n >* ln((*b − a*)*/*0*.*02) = 2*.*995 = 4*.*321*,*

*{*Ej.T1.Sol.1*}*

ln(2)

por tanto realizaremos 5 iteraciones:

0*.*693

|  |  |  |
| --- | --- | --- |
| *I*0 = (0*.*4*,* 0*.*5) | *x*1 = 0*.*45 | *f* (0*.*45) *>* 0*,* |
| *I*1 = (0*.*4*,* 0*.*45) | *x*2 = 0*.*425 | *f* (0*.*425) *>* 0*,* |
| *I*2 = (0*.*4*,* 0*.*425) | *x*3 = 0*.*4125 | *f* (0*.*4125) *<* 0*,* |
| *I*3 = (0*.*4125*,* 0*.*425) | *x*4 = 0*.*4187 | *f* (0*.*4187) *>* 0*,* |
| *I*4 = (0*.*4125*,* 0*.*4187) | *x*5 = 0*.*4156*.* |  |

Por tanto, la soluci´on es *x*5 = 0*.*4156*.*

7

**Ejercicio 2.2** *Sea dada la funci´on*

*f* (*x*) = *x*3 + 4*x*2 *−* 10*.*

* + 1. *Hallar una aproximaci´on x∗ de la u´nica ra´ız real de la ecuaci´on f* (*x*) = 0*, utilizando el m´etodo de bisecci´on con un error inferior a* 10*−*2*. Hallar tam- bi´en una aproximaci´on mediante 4 iteraciones con el m´etodo de Newton- Raphson, tomando como semilla x*0 = *a, siendo a el extremo izquierdo del intervalo considerado como inicio del m´etodo de bisecci´on.*
    2. *(Octave) Aplicar el algoritmo de la bisecci´on para calcular todas las solu- ciones de la ecuaci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*3 *y un nu´mero m´aximo de 100 iteraciones.*
    3. *(Octave) Aplicar el m´etodo de Newton para resolver la misma ecuaci´on con los mismos valores de tolerancia y nu´mero m´aximo de iteraciones.*

#### Soluci´on.

1. Tomamos como intervalo de partida para iniciar el esquema de bisecci´on el intervalo (*a, b*) = (1*,* 1*.*5), ya que *f* (1) = 5 y *f* (1*.*5) = 2*.*375. Comenzamos a calcular las 6 iteraciones con el m´etodo de bisecci´on necesarias para asegurar un error inferior a 10*−*2 tomando *I*0 = (1*,* 1*.*5) como intervalo de partida:

*−*

*I*0 = (1*,* 1*.*5) *x*1 = 1*.*25 *f* (*x*1) = *−*1*.*796875 *<* 0*,*

*I*1 = (1*.*25*,* 1*.*5) *x*2 = 1*.*375 *f* (*x*2) = 0*.*162109 *>* 0*,*

*I*2 = (1*.*25*,* 1*.*375) *x*3 = 1*.*3125 *f* (*x*3) = *−*0*.*848388 *<* 0*,*

*I*3 = (1*.*3125*,* 1*.*375) *x*4 = 1*.*34375 *f* (*x*4) = *−*0*.*350982 *<* 0*,*

*I*4 = (1*.*34375*,* 1*.*375) *x*5 = 1*.*359375 *f* (*x*5) = *−*0*.*096408 *<* 0*,*

*I*5 = (1*.*359375*,* 1*.*375) *x*6 = 1*.*3671875 *f* (*x*6) = 0*.*323557*.*

La aproximaci´on buscada es por tanto, *x*6 = 1*.*3671875.

Utilizando el m´etodo de Newton, dado por el siguiente esquema iterativo:

*{*Ej.T1.Sol.2*}*

*xn*+1

= *xn*

*f* (*xn*)

*— f′*(*x* ) *,*

*n*

siendo *f* (*x*) = *x*3 + 4*x*2 *−* 10 y *f′*(*x*) = 3*x*2 *−* 8*x*, e inicializado con la semilla

*a* = *x*0 = 1, tenemos que:

*x*1 = 1*.*454545 *x*2 = 1*.*3689004*, x*3 = 1*.*3652366

Como *|x*3 *− x*2*|* = 0*.*0036638 *<* 10*−*2, tomamos *x*3 = 1*.*3652366 como la apro- ximaci´on obtenida por el m´etodo de Newton-Raphson. Se tiene que *f* (*x*3) = 1*.*0877 10*−*4, por lo que adem´as de menos iteraciones, este m´etodo obtiene un residuo menor al de la bisecci´on.

*·*

1. Para resolver el problema con el m´etodo de bisecci´on, utilizando Octave, uti- lizaremos el c´odigo *metbiseccion.m*. Para ello definimos y dibujamos la funci´on:

*>* fecu = @(x) x.^3+4.\*x.^2-10;

*>* ejex =@(x) 0\*x;

*>* I = [-5:0.01:3]; plot(I,f(I),I,ejex(I))

La gr´afica de la funci´on puede verse en la Figura [2.1.](#_bookmark3)

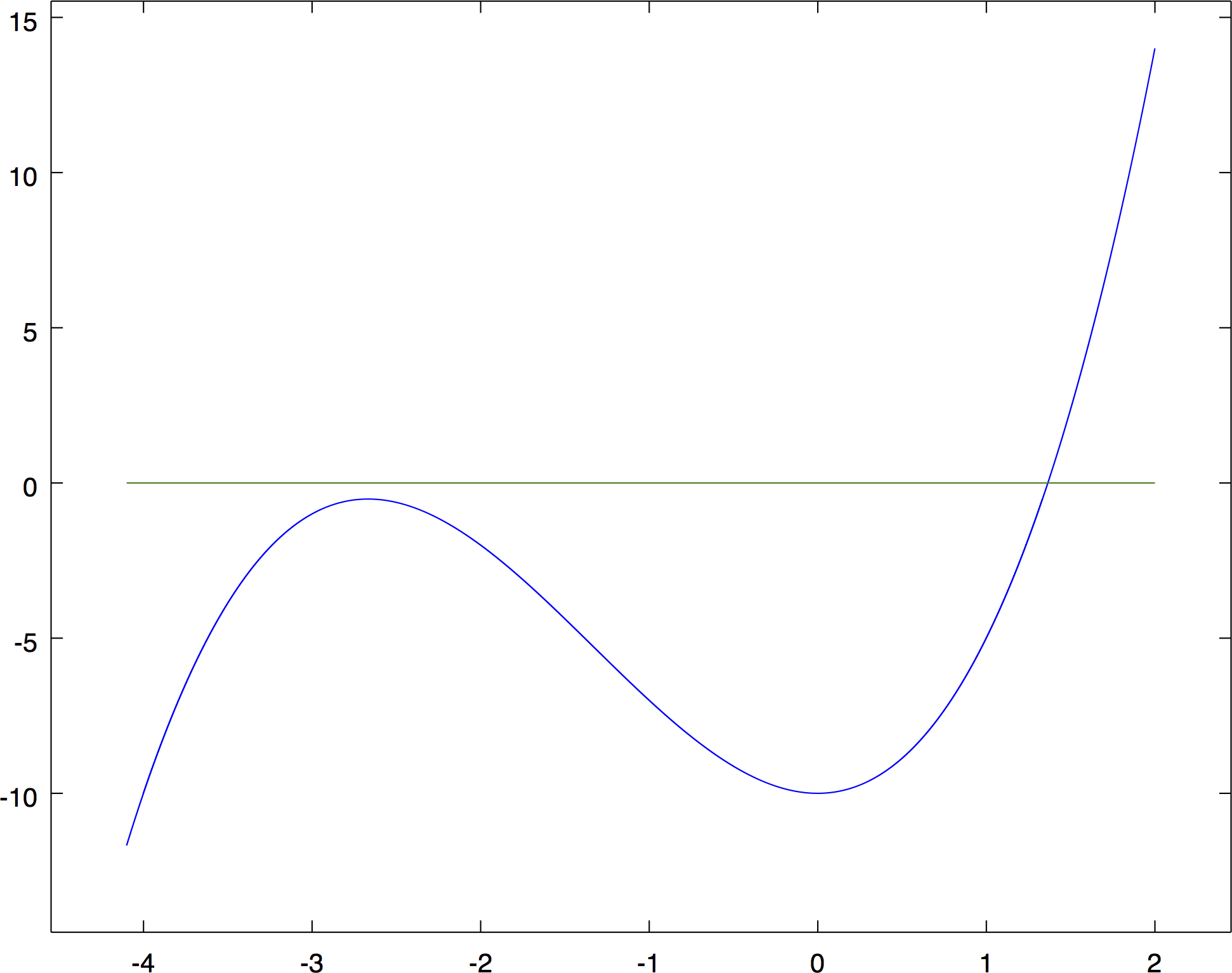


Figura 2.1: Gr´afica de la funci´on *f* (*x*) = *x*3 + 4*x −* 10. *{*FigT1Ej1*}*

Como debemos encontrar un intervalo [*a, b*], tal que *f* (*a*)*f* (*b*) *<* 0, gr´aficamente vemos que podemos probar con *a* = 0 y *b* = 2. En efecto, *f* (*a*)*f* (*b*) = ( 10) 14 *<* 0, por tanto, el intervalo [0*,* 2] nos sirve para iniciar el m´etodo de bisecci´on.

*— ·*

Introducimos:

* a = 0; b = 2; maxitera = 100; errorper = 1e-3;
* [solb,iterab] = metbiseccion(fecu,a,b,errorper,maxitera)

para obtener la soluci´on: *solb* = 1*.*3643 e *iterab* = 10.

1. A continuaci´on, lo resolvemos utilizando el m´etodo de Newton-Raphson. Para ello debemos definir la derivada de la funci´on e introducir un valor como semilla, que en este caso podemos tomar igual a 1:

* dfecu = @(x) 3.\*x.^2+8.\*x
* x0 = 1;

utilizando el punto medio del intervalo de partida en el m´etodo de bisecci´on. Finalmente, ejecutamos el c´odigo *metnewton1ec.m*

* [soln,iteran] = metnewton1ec(fecu,dfecu,x0,errorper,maxitera)

y obtenemos *soln* = 1*.*3652 e *iteran* = 4.

*{*Ej.T1.Sol.3*}*

**Ejercicio 2.3** *Sea dada la funci´on*

*f* (*x*) = *ex −* 1*.*5 *− arctg*(*x*)*.*

*Aplicar el m´etodo de Newton-Raphson para encontrar la ra´ız positiva de la ecua- ci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*3 *y un nu´mero m´aximo de 100 iteraciones.*

#### Soluci´on.

En primer lugar buscamos un intervalo donde sepamos que se encuentra la ra´ız. Siendo *f* (*x*) = *ex −* 1*.*5 *− arctg*(*x*), tenemos que *f* (0) = *−*0*.*5 y *f* (1) = 0*.*43288, por tanto, vamos a elegir como semilla para el esquema *x*0 = 0*.*5.

Se tiene que:

*x*1 = 0*.*8710597917*,*

*x*2 = 0*.*7761330431*,*

*x*3 = 0*.*7677176206*,*

*x*4 = 0*.*7676532699*,*

*x*5 = 0*.*7676532660*,*

*x*6 = 0*.*7676532662*,*

*x*7 = 0*.*7676532662*.*

Como *|x*4 *− x*3*|* = 6*.*4350 *·* 10*−*5 *<* 10*−*3, tomamos como aproximaci´on *x*4 = 0*.*76765. De hecho puede observarse que a partir de *x*4 se mantienen fijas 6 cifras decimales, por lo que *x*4 tiene 6 cifras decimales exactas.

A continuaci´on lo resolvemos utilizando Octave con el m´etodo metnewton1ec.m. Para ello debemos definir *f* (*x*) y su derivada

* + f = inline(’exp(x)-1.5-atan(x)’,’x’);
  + df = inline(’exp(x)-(1+x.^2).^(-1)’,’x’);

e introducir un valor como semilla. En este caso podemos tomar x0 = 0.5, que es el punto medio del intervalo de partida en el m´etodo de bisecci´on. Finalmen- te, ejecutamos el c´odigo metnewton1ec.m

* + [zero1,niter1] = metnewton1ec(f,df,x0,1e-3,100)

y obtenemos el resultado: *zero*1 = 0*.*76765 y *niter*1 = 4. Se deja como ejercicio comprobar que *f* (*x*) tiene una ra´ız negativa en *x* = *−*14*.*101.

*{*Ej.T1.Sol.4*}* **Ejercicio 2.4** *Hallar una aproximaci´on x∗ de la u´nica ra´ız real de la ecuaci´on*

*x*3 + 3*x*2 *−* 8 = 0*, utilizando:*

* 1. *el m´etodo de bisecci´on,*
  2. *el m´etodo de Newton-Raphson,*

*con un error inferior a* 10*−*2 *en ambos casos.*

#### Soluci´on.

1. Comencemos por el m´etodo de bisecci´on. Lo primero que tenemos que hacer es encontrar un intervalo (*a, b*) que contenga la ra´ız, esto es equivalente a que se verifique *f* (*a*)*f* (*b*) *<* 0. Observamos que *f* (1*.*3)*f* (1*.*4) *<* 0, por tanto, podemos tomar *a* = 1*.*3 y *b* = 1*.*4.

Considerando este intervalo de partida, calculamos las iteraciones necesarias para cumplir la condici´on de que el error sea inferior a 10*−*2,

*n >* ln((*b − a*)*/*0*.*01) = 2*.*30

= 3*.*3189*,*

ln(2) 0*.*693

por tanto, realizaremos *n* = 4 iteraciones:

|  |  |  |
| --- | --- | --- |
| *I*0 = (1*.*3*,* 1*.*4) | *x*1 = 1*.*35 | *f* (1*.*35) *<* 0*,* |
| *I*1 = (1*.*35*,* 1*.*4) | *x*2 = 1*.*375 | *f* (1*.*375) *>* 0*,* |
| *I*2 = (1*.*35*,* 1*.*375) | *x*3 = 1*.*3625 | *f* (1*.*3625) *>* 0*,* |
| *I*3 = (1*.*35*,* 1*.*3625) | *x*4 = 1*.*35625*.* |  |

1. Para resolverlo con el m´etodo de Newton, dado por el esquema:

*xn*+1

= *xn*

*f* (*xn*)

*— f′*(*x* ) *,*

*n*

donde *f* (*x*) = *x*3 + 3*x*2 *−* 8 y *f′*(*x*) = 3*x*2 + 6*x*, consideramos como semilla *x*0 = 1*.*35. Y se obtiene que *x*1 = 1*.*3553. Como *x*1 *x*2 = 0*.*0053160 ya es inferior a 10*−*2, tenemos que el m´etodo de Newton-Raphson termina en una sola iteraci´on.

*| − |*

**Ejercicio 2.5** *Hallar una aproximaci´on x∗ de la u´nica ra´ız real de la ecuaci´on f* (*x*) = cos(*x*) *x* = 0*, utilizando el m´etodo del punto fijo con semilla x*0 = 0*.*7 *y realizando 4 iteraciones. Obtener tambi´en una aproximaci´on mediante el m´etodo de bisecci´on tomando, un intervalo de partida v´alido* (*a, b*)*, de modo que adem´as a*+*b* = *x*0*. ¿Qu´e m´etodo obtiene un menor residuo?*

*−*

2

#### Soluci´on.

Comenzamos aplicando el m´etodo del punto fijo, para ello consideramos la fun- ci´on *g*(*x*) = cos(*x*). Entonces el esquema iterativo a seguir ser´a el siguiente: partiendo de una semilla *x*0, el resto de t´erminos de la sucesi´on se obtienen por la relaci´on

*{*Ej.T1.Sol.5*}*

*xn*+1 = *g*(*xn*)*, n ≥* 0*.*

De este modo, tomando *x*0 = 0*.*7 se tiene que:

*x*1 = *g*(*x*0) = cos(0*.*7) = 0*.*76484*, x*2 = *g*(*x*1) = cos(*x*1) = 0*.*72149*, x*3 = *g*(*x*2) = cos(*x*2) = 0*.*75082*, x*4 = *g*(*x*3) = cos(*x*3) = 0*.*73113*.*

Por tanto, el valor aproximado que nos piden es *x*4 = 0*.*73113, con un residuo de *f* (*x*4) = 0*.*01329.

*| |*

Consideremos ahora el intervalo (0*.*6*,* 0*.*8) para comenzar a aplicar el m´etodo de bisecci´on. Primero comprobamos que dicho intervalo es v´alido, es decir, que *f* (0*.*6)*f* (0*.*8) *<* 0. Efectivamente:

*f* (0*.*6) = cos(0*.*6) *−* 0*.*6 = 0*.*825335 *−* 0*.*6 *>* 0*,*

*f* (0*.*8) = cos(0*.*8) *−* 0*.*8 = 0*.*696706 *−* 0*.*8 *<* 0*.*

Adem´as, 0*.*6+0*.*8

2

= 0*.*7. Comenzamos pues a calcular las 4 iteraciones con el

m´etodo de bisecci´on:

*I*0 = (0*.*6*,* 0*.*8) *x*1 = 0*.*7 *f* (*x*1) = 0*.*764842 *−* 0*.*7 *>* 0

*I*1 = (0*.*7*,* 0*.*8) *x*2 = 0*.*75 *f* (*x*2) = 0*.*731688 *−* 0*.*75 *<* 0

*I*2 = (0*.*7*,* 0*.*75) *x*3 = 0*.*725 *f* (*x*3) = 0*.*748499 *−* 0*.*725 *>* 0

*I*3 = (0*.*725*,* 0*.*75) *x*4 = 0*.*7375 *f* (*x*4) = 0*.*740151 *−* 0*.*7375 = 2*.*651 *·* 10*−*3*.*

Se tiene por tanto que el residuo alcanzado por el m´etodo de la bisecci´on (para este reducido nu´mero de iteraciones) es menor.

*{*Ej.T1.Sol.6*}* **Ejercicio 2.6** *Se considera la ecuaci´on f* (*x*) = *x*2 *−* 1 *−* sen(*x*) = 0*.*

* 1. *Probar que dicha ecuaci´on tiene al menos una ra´ız positiva.*
  2. *Encontrar un intervalo en el cual la iteraci´on:*

*xn* = 1 + sen(*xn−*1) = *g*(*xn−*1)*, n* = 1*,* 2*, ...*

*converja para todo x*0 *elegido en dicho intervalo.*

* 1. *Indicar el nu´mero de pasos que hay que realizar para conseguir un error inferior a* 10*−*3*, partiendo de x*0 = *π/*2 *y calcular la aproximaci´on.*

#### Soluci´on.

1. Observamos que *f* (1) = 0*.*8414 y *f* (2) = 2*.*0907, por tanto, podemos ase- gurar que en el intervalo (1*,* 2), la ecuaci´on tiene al menos una ra´ız positiva.

*−*

1. Para ver si en ese intervalo podemos garantizar que el esquema iterativo converge, vamos a estudiar si la derivada, es decir, la funci´on

*′* cos(*x*)

*g* (*x*) = 2 1 + sen(*x*) *,*

se encuentra acotada en valor absoluto por una constante *k <* 1. Puede com- probarse que *g′* es decreciente en el intervalo (1*,* 2), tomando en los extremos los valores *g′*(1) = 0*.*199078 y *g′*(2) = 0*.*150584. Por tanto, podemos tomar *k* = 0*.*2.

*−*

1. Para este valor de *k*, se tiene que el nu´mero de iteraciones obtenido es:

*ϵ*(1*−k*)

 

*x x √ π*

*|* 1 *−* 0 *|*

*n >* ln 

ln(*k*)

 = ln(0*.*001(1 *−* 0*.*2)*/|*

2 *−* 2 *|*)*/* ln(0*.*2) =

ln(0*.*001(1 *−* 0*.*2)*/*0*.*1565)*/* ln(0*.*2) = *−*5*.*2761*/*(*−*1*.*609) = 3*.*2791*,*

es decir *n* = 4, y el valor obtenido es *x*4 = 1*.*40962.

**Ejercicio 2.7** *Considera la ecuaci´on f* (*x*) = 2*x −* sen(*x*) *−* cos(*x*)*.*

* 1. *Demostrar que f* (*x*) *tiene una ra´ız en* [0*,* 1]*.*
  2. *Resolver la ecuaci´on f* (*x*) = 0*, utilizando un m´etodo del punto fijo*

*x*0

*xn*+1 = *g*(*xn*)

*verificando que la funci´on g*(*x*) *elegida es contractiva en las proximidades de la ra´ız y hallando un nu´mero de iteraciones suficiente para que |xi −xi−*1*| <* 10*−*2*.*

#### Soluci´on.

1. Basta comprobar que *f* (*x*) es continua y que *f* (0)*f* (1) *<* 0. En efecto, *f* es continua puesto que los polinomios, sen(*x*) y cos(*x*) son funciones continuas, y cualquier combinaci´on lineal de ellas tambi´en lo es. Adem´as, se tiene que *f* (0) = 1 y *f* (1) = 0*.*61822, por tanto, por el Teorema de Bolzano existe al menos un valor *x∗*, tal que *f* (*x∗*) = 0, es decir, una soluci´on de la ecuaci´on.

*−*

1. Definimos *g*(*x*) = 1 (sen(*x*) + cos(*x*)), que resulta ser contractiva en el in-

*{*Ej.T1.Sol.7*}*

*′*2 1 *′ ′*

tervalo [0*,* 1] ya que *g* (*x*) = 2 (cos(*x*) *−* sin(*x*)), *g* (0) = 0*.*5, *g* (1) = *−*0*.*1505

y *g′′*(*x*) no se anula en [0*,* 1] (en este caso *g′* es mon´otona decreciente en dicho intervalo). Por tanto, puede tomarse como constante de Lipschitz *k* = 0*.*5 *<* 1, con lo cual *g* es contractiva en [0*,* 1]. Eligiendo como semilla el punto medio del intervalo, *x*0 = 0*.*5, se tienen las siguientes iteraciones del punto fijo:

*x*1 = *g*(*x*0) = 0*.*678504*, x*2 = *g*(*x*1) = 0*.*703070*, x*3 = *g*(*x*2) = 0*.*704711*,*

y como el enunciado nos dice que iteremos hasta que se verifique la condici´on *xi xi−*1 *<* 10*−*2, y *x*2 y *x*3 la satisfacen, tomamos como aproximaci´on de la soluci´on, *x* = 0*.*70.

*| − |*

**Ejercicio 2.8** *(Octave) Hallar una aproximaci´on x∗ de la ra´ız m´as pr´oxima a x* = 1 *de la ecuaci´on e−x* sin(*x*) = 0*, con un error inferior a* 10*−*4*, utilizando el m´etodo del punto fijo.*

*−*

*{*Ej.T1.Sol.8*}*

#### Soluci´on.

Tenemos que ejecutar los siguientes comandos:

* g= @(x) asin(exp(-x));
* x0 = 1; errorper= 1e-4; maxiter = 100;
* [sol,itera] = metpuntofijo(g,x0,errorper,maxitera)

El resultado que obtemos es *sol* = 0*.*58850 e *itera* = 23.

Ej.T1.Sol.9

*{ }*

**Ejercicio 2.9** *(Octave) Sea dada la funci´on*

*f* (*x*) = *ex −* 3*x*2*.*

* 1. *Considerando el esquema num´erico asociado al m´etodo de Newton como un esquema de punto fijo, definir la funci´on gN* (*x*) *asociada al m´etodo y utilizar el algoritmo de punto fijo para calcular la soluci´on de la ecuaci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 1000 iteraciones.*
  2. *Define un esquema de punto fijo diferente al considerado en el apartado anterior, para calcular la soluci´on de la ecuaci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 1000 iteraciones.*

#### Soluci´on.

1. En primer lugar debemos definir la funci´on y su derivada, ya que vamos a utilizar el esquema del m´etodo de Newton para definir la funci´on *gN* que aparece en el esquema del punto fijo:

*f* = *ex −* 3*x*2*, f′* = *ex −* 6*x,*

y por tanto,

* gN =@(x) x-(exp(x)-3.\*x.^2)./(exp(x)-6.\*x);

A continuaci´on, debemos dar un valor inicial con el cual arrancar el m´etodo del punto fijo. Para ello primero localizamos un intervalo que contenga a la ra´ız. Por ejemplo, podemos considerar *a* = 0, *b* = 1 ya que *f* (*a*) = 1 y *f* (*b*) = 0*.*28172. Tomamos como semilla el punto medio de este intervalo e introducimos el resto de argumentos que necesitamos para ejecutar el c´odigo *metpuntofijo.m*:

*−*

* a = 0; b = 1;
* x0 = 0.5; errorper = 1e-6; maxitera = 1000;
* [sol,itera] = metpuntofijo(gN,x0,errorper,maxitera)

y obtenemos el resultado *sol* = 0*.*9100076 e *itera* = 5.

1. Utilizamos otra funci´on distinta en el esquema del punto fijo definiendo

*g*2(*x*) = *ex/*3 y el resto de argumentos lo dejamos igual:

* g2 =@(x) sqrt(exp(x)/3);
* [sol2,itera2] = metpuntofijo(g2,x0,errorper,maxitera)

y obtenemos los resultados *sol*2 = 0*.*9100070 e *itera*2 = 17.

**Ejercicio 2.10** *(Octave) Sea la funci´on*

*f* (*x*) = *ex −* 3*x*2*.*

* 1. *Considerando el esquema num´erico asociado al m´etodo de Newton como un esquema de punto fijo, definir la funci´on gN* (*x*) *asociada al m´etodo y utilizar el algoritmo de punto fijo para calcular las soluciones de la ecua- ci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 1000 iteraciones.*
  2. *Utilizar el algoritmo de Aitken para calcular las mismas soluciones con los mismos valores de tolerancia y nu´mero m´aximo de iteraciones. Utiliza las semillas utilizadas en el apartado anterior. Compara en t´erminos de velocidad de convergencia los resultados obtenidos en el apartado anterior.*

*{*Ej.T1.Sol.10*}*

* 1. *Considerar la funci´on*

*g*1(*x*) = ln(3*x*2)*,*

*que define el esquema de punto fijo xn*+1 = *g*1(*xn*)*. Utilizar los algoritmos de punto fijo y de Aitken para resolver la ecuaci´on f* (*x*) = 0*, resuelta en los apartados anteriores. Compara los resultados.*

* 1. *Considerar la funci´on*

*g*2(*x*) = *ex/*3

*que define el esquema de punto fijo xn*+1 = *g*2(*xn*)*. Utilizar los algoritmos de punto fijo y de Aitken para resolver la ecuaci´on f* (*x*) = 0*, resuelta en los apartados anteriores. Compara los resultados.*

#### Soluci´on.

1. Mediante los comandos:

* f =@(x) e.^x-3.\*x.^2;
* eje =@(x) 0.\*x;
* I=[-2:0.1:4]; plot(I,f(I),I,eje(I))

representamos la gr´afica de la funci´on en el intervalo [*−*2*,* 4] y observamos que la ecuaci´on tiene 3 soluciones (ver Figura [2.2).](#_bookmark4) Definimos la funci´on *gN* (*x*) = *x−f* (*x*)*/f′*(*x*) de Newton y ejecutamos el esquema de punto fijo con las semillas

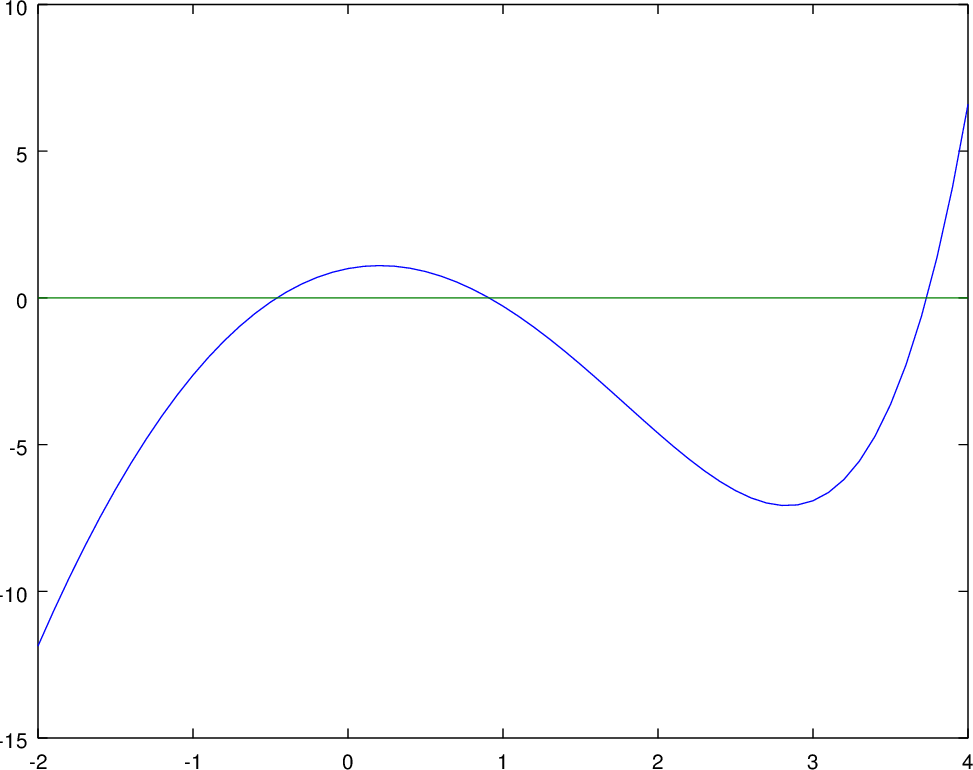


Figura 2.2: Gr´afica de *f* (*x*) = *ex −* 3*x*2 en donde se observan las tres ra´ıces. *{*FigT1Prop1*}*

-2, 1 y 4:

* + gN =@(x) x-(e.^x-3.\*x.^2)./(e.^x-6.\*x);
  + [sol1,itera1] = metpuntofijo(gN,-2,1.e-6,100)
  + [sol2,itera2] = metpuntofijo(gN,1,1.e-6,100)
  + [sol3,itera3] = metpuntofijo(gN,4,1.e-6,100)

y obtenemos las ra´ıces *sol*1 = 0*.*45896, *sol*2 = 0*.*91001 y *sol*3 = 3*.*7331 en 6, 4 y 5 iteraciones respectivamente.

*−*

1. Utilizando el algoritmo de Aitken, con las mismas semillas:

* [sol1a,itera1a] = metodoaitken(gN,-2,1e-6,100)
* [sol2a,itera2a] = metodoaitken(gN,1,1e-6,100)
* [sol3a,itera3a] = metodoaitken(gN,4,1e-6,100)

obtenemos los mismos resultados en 4, 3 y 4 iteraciones respectivamente.

1. Usando ahora *g*1 = ln(3*x*2), los comandos:

* g1 =@(x) log(3.\*x.^2);
* [sol1c,itera1c] = metpuntofijo(g1,-2,1.1e-6,100)
* [sol2c,itera2c] = metpuntofijo(g1,1,1.1e-6,100)
* [sol3c,itera3c] = metpuntofijo(g1,4,1.1e-6,100)

junto con los respectivos comandos para el algoritmo de Aitken, dan como re-

sultados:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Punto Fijo | | Aitken | |
| Semilla | Ra´ız | *niter* | Ra´ız | *niter* |
| *−*2  1 | 3*.*7331  3*.*7331 | 24  28 | 3*.*7331  0*.*91001 | 5  5 |
| 4 | 3*.*7331 | 20 | 3*.*7331 | 4 |

Vemos como el m´etodo del punto fijo converge para todas las semillas utilizadas u´nicamente a la ra´ız *x*1 = 3*.*7331, mientras que el algoritmo de Aitken consigue encontrar tambi´en la ra´ız *x*2.

1. Usando *g*2 obtenemos los resultados:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Punto Fijo | | *Aitken* | |
| Semilla | Ra´ız | *niter* | Ra´ız | *niter* |
| *−*2  1 | 0*.*91001  0*.*91001 | 18  15 | 0*.*91001  0*.*91001 | 4  3 |
| 4 | *Inf* | 7 | 3*.*7331 | 13 |

En este caso el esquema del punto fijo diverge intentando encontrar la ra´ız *x*3, mientras que Aitken consigue una aproximaxi´on en 13 iteraciones.

**Ejercicio 2.11** *Utilizar el m´etodo de Newton-Raphson para encontrar la solu- ci´on del sistema:*

*{*Ej.T1.Sol.11*}*

3*x*2 *− y*2 = 0*,*

3*xy*2 *− x*3 *−* 1 = 0*,*

*partiendo del vector semilla* (1*,* 1)*t y realizando 2 iteraciones del m´etodo.*

#### Soluci´on.

Denotemos por *f*1(*x, y*) = 3*x*2 *y*2 y *f*2(*x, y*) = 3*xy*2 *x*3 1*.* Siguiendo el esquema de Newton-Raphson para sistemas se tiene que:

*— − −*

(*x*

1

*y*1

(*x*

*y*0

*∂f*1 (*x , y* ) *∂f*1 (*x , y* ) *−*1 (*f* (*x , y* )

*∂f*2 (*x*0*, y*0) *∂f*2 (*x*0*, y*0)

*f*2(*x*0*, y*0)

(1

=

0

*−*

*∂x*

0

0

*∂y*

0

0

1

0

0

*∂x*

*∂y*

=

=

*.*

1

(6 *−*2 *−*1 (2

0 6

1

(0*.*612

0*.*833

Ahora calculamos la segunda iteraci´on:

*−*

(*x*

2

*y*2

(*x*

*y*1

*∂f*1 (*x , y* ) *∂f*1 (*x , y* ) *−*1 (*f* (*x , y* )

*∂f*2 (*x*1*, y*1) *∂f*2 (*x*1*, y*1)

*f*2(*x*1*, y*1)

(0*.*612

=

1

*−*

*∂x*

1

1

*∂y*

1

1

1

1

1

*∂x*

*∂y*

=

*−*

=

*.*

0*.*833

(3*.*672 *−*1*.*667 *−*1 (0*.*425

0*.*962 3*.*055

0*.*044

(0*.*504

0*.*085

**Ejercicio 2.12** *(Octave) Utilizar el m´etodo de Newton-Raphson para encontrar una soluci´on aproximada del sistema:*

f

*x*(*x − y*)2 + 1 = 0*,*

*−x − y* + 2*x*2 = 0*,*

*partiendo del vector inicial* (*x*0*, y*0)*t* = (1*,* 1*.*2)*t con un error inferior a* 10*−*5*.*

#### Soluci´on.

La llamada al programa que tenemos que utilizar es:

* [vectorsol,itera] = metnewtonsistema(@fecusistema,...

*>*@jacobiana,x0,errorper,maxitera)

por tanto, vamos a tener que definir los script fecusistema.m

* function F = fecusistema(x,y)
* F(1,1) = x\*(x-y)^2+1;
* F(2,1) = -x-y+2\*x^2;
* end

y el script jacobiana.m

* function J = jacobiana(x)
* J(1,1) = (x-y)^2+2\*x\*(x-y);
* J(1,2) = -2\*x\*(x-y);
* J(2,1) = -1+4\*x;
* J(2,2) = -1;
* end

y definir la condici´on inicial, tolerancia y nu´mero m´aximo de iteraciones como sigue:

* x0 = [1;1.2]; errorper= 1e-5; maxitera = 100;

El resultado que obtenemos es vectorsol(1)=x=-0.4, vectorsol(2)=y=0.95299, en 9 iteraciones.

# Problemas propuestos

*{*Ej.T1.Prop.1*}* **Ejercicio 2.13** *Utilizar el m´etodo de punto fijo para localizar la ra´ız de*

*f* (*x*) = sin(*√x*) *− x,*

*tomando la semilla x*0 = 0*.*5 *e iterando hasta alcanzar una tolerancia de* 0*.*01*. Hacer los c´alculos en radianes.*

*{*Ej.T1.Sol.12*}*

**Ejercicio 2.14** *Sea la funci´on*

1

*f* (*x*) = *x −* 2 sin(*πx*) *−* 4 *.*

1. *Se sabe que la ecuaci´on f* (*x*) = 0*, tiene una de sus ra´ıces en el intervalo* [0*,* 2]*. Calcula una aproximaci´on de dicha ra´ız realizando 5 iteraciones del m´etodo de la secante, utilizando como semillas los puntos situados en los extremos del intervalo. Calcula el residuo que se obtiene.*
2. *Define la funci´on ϕN* (*x*) *asociada al m´etodo de Newton y util´ızala para obtener mediante el m´etodo de punto fijo una aproximaci´on de la misma ra´ız de f* (*x*) = 0*, con una tolerancia de* 10*−*5*. ¿Se obtiene un residuo menor que en el apartado (a)?*
3. *(Octave) Utilizando una tolerancia de* 10*−*8 *y un nu´mero m´aximo de 100 iteraciones, calcula todas las soluciones de la ecuaci´on f* (*x*) = 0*, mediante el esquema de punto fijo del apartado (b) y mediante un esquema de punto fijo asociado a otra funci´on ϕ*(*x*) *distinta de ϕN* (*x*)*.*

*¿Ambos m´etodos producen las ra´ıces buscadas? En caso de que exista di- vergencia, explica a qu´e es debida.*

1. *(Octave) Calcula mediante el m´etodo num´erico que prefieras el valor m´axi- mo de f* (*x*) *en el intervalo* [0*,* 2] *as´ı como el punto donde se alcanza.*

**Ejercicio 2.15** *Sea la funci´on*

*f* (*x*) = (*x −* 2)2 *−* ln(*x*)*.*

1. *Aplicar el m´etodo de Newton-Raphson para calcular la ra´ız de la ecuaci´on*

*f* (*x*) = 0*, dentro del intervalo* [3*,* 4] *con una tolerancia de* 10*−*3*.*

1. *Demuestra que la ra´ız obtenida en el apartado anterior puede calcularse mediante un problema de punto fijo para la funcion*

*ϕ*(*x*) = ln(*x*) + 2*.*

*Utiliza el m´etodo de punto fijo asociado a ϕ*(*x*)*, para obtener una aproxi- maci´on de la ra´ız de f* (*x*) = 0*, con una tolerancia de* 10*−*3 *(usa la misma semilla que en el apartado anterior). ¿Se necesitan m´as o menos iteracio- nes que con el m´etodo de Newton?*

1. *(Octave) Resuelve los apartados (a) y (b) utilizando Octave. ¿Qu´e m´eto- do es m´as r´apido? ¿Qu´e m´etodo produce una aproximaci´on con menor residuo?*
2. *(Octave) Calcula num´ericamente la solucion de la ecuaci´on f* (*x*) = 1*.*5*, en el intervalo* [3*,* 4]*.*

*{*Ej.T1.Prop.2*}*

*{*Ej.T1.Prop.3*}*

*Utiliza en los apartados (c) y (d) una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 100 iteraciones.*

**Ejercicio 2.16** *(Octave) Sea dada la funci´on*

*f* (*x*) = *−*2*x*3 + 5*x* + 2*.*

* 1. *Aplicar el algoritmo de bisecci´on para calcular todas las soluciones de la ecuaci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*3 *y un nu´mero m´aximo de* 100 *iteraciones.*
  2. *Aplicar el m´etodo de Newton-Raphson para resolver la misma ecuaci´on con los mismos valores de tolerancia y nu´mero m´aximo de iteraciones.*

*{*Ej.T1.Prop.4*}*

Ej.T1.Prop.5

*{ }*

**Ejercicio 2.17** *(Octave) Sea dada la funci´on*

*f* (*x*) = *x*4 *−* 1 *− e−xx*2*.*

1. *Considerando el esquema num´erico asociado al M´etodo de Newton como un esquema de punto fijo, definir la funcion ϕN* (*x*) *asociada al m´etodo y utilizar el algoritmo de punto fijo para calcular las soluciones de la ecua- ci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de* 1000 *iteraciones. Utilizar el algoritmo Aitken para calcular las mismas soluciones mediante los mismos par´ametros de entrada (semillas, toleran- cia y nu´mero m´aximo de iteraciones). Comparar los resultados obtenidos en t´erminos de velocidad de convergencia.*
2. *Considerar la funci´on*

*ϕ*1(*x*) = (1 + *e−xx*2)1*/*4*,*

*que define el esquema de punto fijo x* = *ϕ*1(*x*)*. Utilizar los algoritmos de puntofijo y de Aitken para resolver la ecuaci´on f* (*x*) = 0*. Utilizar los mismos par´ametros de entrada (semillas, tolerancia y nu´mero m´aximo de iteraciones) del apartado anterior. Comparar los resultados.*

1. *Considerar la funci´on*

*ϕ*2(*x*) = *x* + *f* (*x*)*,*

*que define el esquema de punto fijo x* = *ϕ*2(*x*)*. Utilizar los algoritmos de puntofijo y de Aitken para resolver la ecuaci´on f* (*x*) = 0*. Utilizar los mismos par´ametros de entrada (semillas, tolerancia y nu´mero m´aximo de iteraciones) del apartado anterior. Comparar los resultados.*

*{*Ej.T1.Prop.6*}* **Ejercicio 2.18** *(Octave) Hallar una aproximaci´on x∗ de la ra´ız m´as pr´oxima a x* = 1*, de la ecuaci´on x*2 1 sin(*x*) = 0*, con un error inferior a* 10*−*4*, utilizando el m´etodo del punto fijo.*

*— −*

Ej.T1.Prop.7

*{ }*

**Ejercicio 2.19** *(Octave) Sea dada la funci´on*

*f* (*x*) = *√x* sin(*x*) *− x*3 + 2*.*

1. *Considerando el esquema num´erico asociado al m´etodo de Newton como un esquema de punto fijo, definir la funcion ϕN* (*x*) *asociada al m´etodo y utilizar el algoritmo de punto fijo para calcular la soluci´on de la ecuaci´on f* (*x*) = 0 *trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 1000 iteraciones.*
2. *Define un esquema de punto fijo diferente al considerado en el apartado anterior justificando la elecci´on, para calcular la soluci´on de la ecuaci´on f* (*x*) = 0*, trabajando con una tolerancia de* 10*−*6 *y un nu´mero m´aximo de 1000 iteraciones.*

**Ejercicio 2.20** *(Octave) Resolver el sistema de ecuaciones no lineales:*

f

*−x − y*3 + *x*2 *−* 2 = 0*,* 1 *−* 2*x* + *y − y*2 = 0*.*

*utilizando el m´etodo de Newton y trabajando con una tolerancia de* 10*−*6*. Utilizar como semilla* (*x*0*, y*0)*t* = (*−*0*.*8*, −*0*.*4)*t.*

**Ejercicio 2.21** *(Octave) Resolver el sistema de ecuaciones no lineales:*

f

ln(*x*2 + *y*2) *−* sin(*xy*) *−* (ln(2) + ln(*π*)) = 0*, ex−y* + cos(*xy*) = 0*.*

*utilizando el m´etodo de Newton y trabajando con una tolerancia de* 10*−*6*. Utilizar como semilla* (*x*0*, y*0)*t* = (2*,* 2)*t.*

**Ejercicio 2.22** *Encuentra el punto de corte de la gr´afica de y* = *x*2 2 *y la gr´afica de y* = *ex para x <* 0*, usando el m´etodo de la bisecci´on con una tolerancia de* 10*−*2*.*

*−*

*{*Ej.T1.Prop.8*}*

*{*Ej.T1.Prop.9*}*

*{*Ej.T1.Prop.10*}*

**Ejercicio 2.23** *(Octave) Dada la funci´on*

*f* (*x*) = *x*7 *− x −* cos(*x*) + 1*,*

*calcular con una tolerancia de* 10*−*5 *y el m´etodo num´erico que prefieras:*

* 1. *Todos sus m´aximos y m´ınimos locales.*
  2. *Todos sus puntos de inflexi´on.*
  3. *Utiliza el algoritmo* metnewtonsistema.m *para calcular los puntos de in- tersecci´on de la gr´afica de f* (*x*) *con la circunferencia x*2 + *y*2 = 1*.*

# Aplicaciones

*{*Ej.T1.Aplic.1*}* **Ejercicio 2.24 (Ca´ıda de presi´on)** *La ca´ıda de presi´on en la circulaci´on de un flujo turbulento en una tuber´ıa recta de secci´on circular constante puede estimarse mediante la expresi´on:*

*{*Ej.T1.Prop.11*}*

∆*p* =

*ρf Lu*2

*,*

2*D*

*donde ρ es la densidad del fluido, L es la longitud de la tuber´ıa, D es el di´ametro de la secci´on, u es la velocidad del fluido y f es el coeficiente de fricci´on de la tuber´ıa. Este coeficiente es, a su vez, proporcional al nu´mero de Reynolds* (*Re* = *Duρ donde µ es la viscosidad del fluido*) *segu´n la relacion f* = *Re−*0*.*25*. Asimismo, en los problemas de tuber´ıas es usual trabajar, no ya en t´erminos de velocidad del fluido, si no de caudal del fluido, siendo el caudal*

*µ*

( *πD*2

*Q* =

*u.*

4

*Con ello, la ca´ıda de presi´on en la tuber´ıa puede expresarse mediante una ley del tipo:*

∆*p*(*x*) = *K*(*x*)*Q*(*x*)1*.*75*,*

*donde* ∆*p*(*x*) *es la p´erdida de presi´on en el punto que dista x unidades de lon- gitud de aqu´el respecto al que se mide la ca´ıda de presi´on, Q*(*x*) *es el caudal en dicho punto y*

 *ρ* ( *µ*

0*.*25 ( 4 1*.*75 

*K*(*x*) =





*ρD πD x.*

2*D* 



*Consid´erese una tuber´ıa de secci´on circular que va del punto P1 al punto P2 y en ´el se divide en dos ramas, una que va al punto P3 y otra que va al punto P4. Designando por Q al caudal que va de P1 a P2, por Q*1 *al que va de P2 a P3,*

*2.3. APLICACIONES* 23

*por Q*2 *al que va de P2 a P4 y por p*1*, p*2*, p*3 *y p*4 *a las presiones en los puntos P1, P2, P3 y P4 respectivamente, las expresiones anteriores, junto a un balance de masa, nos conducen al sistema de ecuaciones no lineales:*

*p*1 *− p*2 = *K*1*Q*1*.*75*,*

*p*2 *− p*3 = *K*2*Q*1*.*75*, p*2 *− p*4 = *K*3*Q , Q* = *Q*1 + *Q*2*.*

1

2

1*.*75

*Si para un fluido y una tuber´ıa concretos se han estimado los valores siguientes:*

*K*1 = 0*.*1170*, K*2 = 0*.*2325*, K*3 = 0*.*5034*,*

*y*

*p*1 = 75 *psi, p*3 = 20 *psi, p*4 = 15 *psi,*

*se desea estimar la presi´on p*2 *existente en el punto P* 2 *as´ı como los caudales Q, Q*1 *y Q*2*, que circulan por cada una de las ramas de la red de tuber´ıas antes descrita.*

**Cap´ıtulo 3**

**Problemas de Valor Inicial**

# Problemas resueltos

**Ejercicio 3.1** *Se considera el problema de valor inicial (PVI):*

f

*y′* = 2*t* cos2(*y*)*, y*(0) = 0*.*

*En el intervalo temporal* [0*,* 0*.*5]*, obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 0*.*5*, considerando una longitud de paso constante h* = 0*.*5*, mediante:*

* + 1. *El esquema num´erico de Euler impl´ıcito utilizando dos iteraciones del m´etodo de Newton-Raphson para resolver la ecuaci´on no lineal que pueda surgir.*
    2. *El m´etodo de Heun.*
    3. *Hallar el error cometido en los apartados anteriores.*

#### Soluci´on.

1. Puesto que el taman˜o de discretizaci´on es *h* = 0*.*5, s´olo tenemos que hacer un paso. En *t*0 = 0, tenemos *y*0 = 0. Para *t*1 = 0*.*5, seguimos el esquema de Euler impl´ıcito y se tiene:

*y*1 = *y*0 + *hf* (*t*1*, y*1) = 0 + 0*.*5 cos2(*y*1)*,* por tanto, la soluci´on *y*1 es ra´ız de la ecuaci´on no lineal

*f* (*x*) = *x* 1 cos2(*x*) = 0*,* 2

*−*

que debemos resolver haciendo dos iteraciones con el m´etodo de Newton-Raphson. Para ello primero debemos elegir una semilla con la que iniciar el esquema. Ob- servamos que en el intervalo *I*0 = (0*.*4*,* 0*.*5) se encuentra la ra´ız, puesto que

25

*{*Ej.T2.Sol.1*}*

*f* (0*.*4)*f* (0*.*5) *<* 0. Realizando 3 iteraciones con el m´etodo de bipartici´on tene- mos *x∗* = 0*.*4125, consiguiendo un error inferior a 0.02. As´ı pues, tomaremos como semilla para el esquema de Newton-Raphson, *x*0 = 0*.*4125. De este modo:

*x*1 = *x*0

*x*2 = *x*1

*f* (*x*0)

*— f′*(*x*0)

*f* (*x*1)

*— f′*(*x*1)

= 0*.*41772151*,*

= 0*.*41771479*.*

Por tanto, tomamos *y*1 = 0*.*4177*.*

1. Ahora lo resolvemos con Heun dado por el siguiente esquema predicci´on correcci´on, teniendo en cuenta que *t*0 = 0, *t*1 = 0*.*5 y *y*0 = 0, tenemos que

*h h*

*y*1 = *y*0 + 2 *f* (*t*0*, y*0) + 2 *f* (*t*1*, y*0 + *hf* (*t*0*, y*0)) = 0*.*25*.*

1. Calculamos la soluci´on exacta integrando la EDO (de variables separadas), para obtener que:

*y*(*t*) = arctan(*t*2)*.*

Por tanto, *y*(0*.*5) = arctan(0*.*25) = 0*.*2449*.* Por u´ltimo, calculamos el error co- metido con cada uno de los m´etodos:

-error con Euler impl´ıcito: *eEI* = *|*0*.*2449 *−* 0*.*4177*|* = 0*.*1728,

-error con Heun: *eH* = *|*0*.*25 *−* 0*.*2449*|* = 0*.*0051.

El m´etodo de Heun ha sido m´as preciso, en cualquier caso hay que tener en cuenta que *h* = 0*.*5 es un paso grande.

Ej.T2.Sol.2

*{ }*

**Ejercicio 3.2** *Sea el Problema de Valor Inicial (PVI)*

*(PVI)* :

f *y′* = 1 *− te−y, t ∈* (0*,* 3]*,*

*y*(0) = 1*,*

*que tiene por soluci´on exacta la funci´on y*(*t*) = ln (*e −* 1)*et* + *t* + 1 *.*

( )

* 1. *Tomando un paso de discretizaci´on de h* = 0*.*5*, utiliza el m´etodo Crank- Nicolson para dar una aproximaci´on de la soluci´on en el instante t* = 0*.*5*, utilizando para resolver las ecuaciones no lineales que aparecen en el proceso dos iteraciones del m´etodo de Newton.*
  2. *(Octave) Tomando ahora un paso de discretizaci´on de h* = 0*.*01*, resuelve el PVI con el m´etodo de Euler Impl´ıcito y el m´etodo de Heun (m´etodo Runge- Kutta de orden 2). Dibuja las dos aproximaciones obtenidas junto con la soluci´on exacta y calcula la aproximaci´on que producen ambos m´etodos en el instante t* = 2*.*23*. ¿Qu´e m´etodo es m´as preciso en ese instante?*
  3. *(Octave) Con el m´etodo m´as preciso en el apartado anterior, calcula en intervalo temporal para el cual las aproximaciones son menores a* 2*.*2*.*
  4. *(Octave) Utiliza el m´etodo de la bisecci´on (con una tolerancia de* 10*−*5*) para calcular el mismo intervalo temporal que en (c) pero esta vez usando la soluci´on exacta y*(*t*)*. ¿Qu´e error ha cometido el m´etodo num´erico usado en el apartado (c)?*

**Soluci´on.** *(a)* Por la condici´on inicial en *t*0 = 0 tenemos que *y*0 = 1. Para la aproximaci´on pedida *y*1 *y*(0*.*5) en *t*1 = 0*.*5 seguimos el esquema de Crank- Nicolson:

*∼*

*h*

*y*1 = *y*0 + 2 [*f* (*t*0*, y*0) + *f* (*t*1*, y*1)] *,*

que hace que *y*1 sea la soluci´on de la ecuaci´on no lineal

*x* + 0*.*125*e−x −* 1*.*5 = 0*.*

Tomando como semilla *x*0 = *y*0 = 1 y realizando dos iteraciones del m´etodo de Newton obtenemos que la aproximaci´on pedida es *y*1 = *x*2 = 1*.*4713. El error cometido es

*e* = *|y*(0*.*5) *− y*1*|* = 0*.*0050442*.*

1. Los comandos necesarios son los siguientes:

* f =@(t,y) 1-t.\*e.^(-y);
* intiempo = [0,3]; valorini = 1;
* h = 0.01; npasos= 3/h;
* [t,yei] = eulerimplicito(f,intiempo,valorini,npasos);
* [t,yhe] = heun(f,intiempo,valorini,npasos);

Dibujamos las aproximaciones obtenidas junto con la soluci´on exacta (ver Fi- gura [3.1)](#_bookmark21) mediante los siguientes comandos:

* sol =@(t) log((e-1)\*e.^t+t+1);
* plot(t,yei,t,yhe,t,sol(t))
* legend(’EI’,’RK2’,’exacta’)

y para las aproximaciones pedidas escribimos:

* punto = 2.23;
* aproxei = yei(find(t==punto))
* aproxhe = yhe(find(t==punto))
* errorei = abs(sol(punto)-aproxei)
* errorhe = abs(sol(punto)-aproxhe)

Se obtiene que la aproximaci´on de Euler Impl´ıcito en *x* = 2*.*23 es de 2*.*9546 con un error de 7*.*7398 10*−*4, mientras que Heun obtiene una aproximaci´on de 2*.*9554 con un error 4*.*6208 10*−*6. El m´etodo de Heun es por tanto m´as preciso en este punto.

*·*

*·*

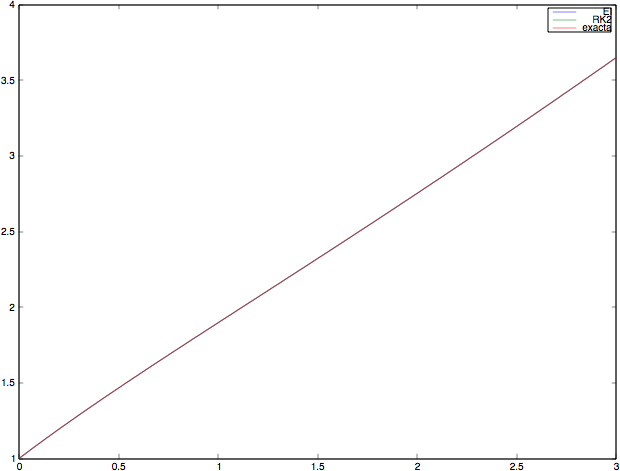


Figura 3.1: Puede observarse como las aproximaciones de Euler Impl´ıcito como

de Heun se ajustan a la gr´afica de la soluci´on exacta. *{*FigEj2.2*}*

1. Para calcular el intervalo pedido mediante el m´etodo de Heun, escribimos:

* vector=t(find(yhe<2.2));
* intervalo = [vector(1),vector(end)]

dando como resultado el intervalo = [0*,* 1*.*35].

1. Obtenemos el intervalo exacto (con una tolerancia de 10*−*5) mediante el m´etodo de la bisecci´on de la siguiente forma:

* d =@(t) log((e-1)\*e.^t+t+1)-2.2;
* a = 1; b = 1.5;
* [sol,itera] = metbiseccion(d,a,b,10^(-5),1000);
* Intervaloex = [0,sol]
* ErrorIntervalo = abs(vector(end)-sol)

Se obtiene el intervalo [0*,* 1*.*35613], por lo que en el apartado anterior se ha cometido un error de 0*.*0061325.

*{*Ej.T2.Sol.3*}* **Ejercicio 3.3** *Se considera el problema de valor inicial:*

f

(*PV I*) *y′* = *yt, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 0*.*5]*, utilizar el esquema num´erico de Euler impl´ıcito para obtener un valor aproximado de la solucion y*(*t*) *en el tiempo t* = 0*.*5*, considerando una longitud de paso constante h* = 0*.*25*. Hallar el error cometido.*

**Soluci´on.** Primero calcularemos la soluci´on anal´ıtica para poder calcular pos- teriormente el error cometido. Se trata de una EDO de variables separadas y

por tanto, se resuelve como sigue,

*dy* = *tdt* =*⇒* ln(*y*) = 0*.*5*t*2 + *C* =*⇒ y*(*t*) = *Ke*0*.*5*t*2 *,*

*y*

donde la constante *K* se tiene que elegir de modo que *y*(0) = 1. Vemos que *K* = 1 y la soluci´on exacta es *y*(*t*) = *e*0*.*5*t*2 . Por u´ltimo, tenemos que *y*(0*.*5) = 1*.*133148*.* Pasemos ahora a calcular el valor aproximado con el m´etodo de Euler impl´ıcito, que viene dado por el siguiente esquema:

*yn*+1 = *yn* + *hf* (*tn*+1*, yn*+1)*.*

Tenemos que realizar dos iteraciones para alcanzar *t* = 0*.*5, ya que *h* = 0*.*25, por tanto,

*y*1 = *y*0 +*hf* (*t*1*, y*1) = *y*

0 +*ht*1*y*1

*y*0 1

*→ y* = (1 *− ht* ) = 1 *−* 0*.*25 *·* 0*.*25 = 1*.*06666*,*

1

1

*y*2 = *y*1 +*hf* (*t*2*, y*2) = *y*

1 +*ht*2*y*2

*y*1 1*.*06666

*→ y* = (1 *− ht* ) = 1 *−* 0*.*25 *·* 0*.*5 = 1*.*219047*.*

2

2

El valor aproximado de *y*(0*.*5) obtenido por el m´etodo de Euler impl´ıcito es

*y*2 = 1*.*219047.

Por u´ltimo, calculamos el error cometido,

*e* = *|y*2 *− y*(0*.*5)*|* = *|*1*.*219047 *−* 1*.*133148*|* = 0*.*112433*.*

**Ejercicio 3.4** *Se considera el problema de valor inicial (PVI):*

f

*y′* = 2*y*1*/*2*, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 0*.*5]*, utilizar el esquema num´erico dado por el m´etodo de Heun, para obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 0*.*5*, considerando una longitud de paso constante h* = 0*.*25*. Hallar el error cometido.*

#### Soluci´on.

Primero calcularemos la soluci´on anal´ıtica para poder calcular posteriormente el error cometido. Se trata de una EDO de variables separadas y por tanto, se resuelve como sigue,

*{*Ej.T2.Sol.4*}*

*dy*

2*√y*

= *dt* =*⇒ √y* = *t* + *C* =*⇒ y*(*t*) = (*t* + *C*)2*,*

donde la constante *C* se tiene que elegir de modo que *y*(0) = 1. Entonces resulta que *C* = 1 y la soluci´on exacta es *y*(*t*) = (*t* + 1)2. Por u´ltimo, tenemos que *y*(0*.*5) = (0*.*5 + 1)2 = 2*.*25*.*

Pasemos ahora a calcular el valor aproximado con el m´etodo de Heun, que viene dado por la siguiente tabla (tipo Runge-Kutta):

0 0 0

1 1 0

0*.*5 0*.*5

Puesto que *h* = 0*.*25, tendremos que hacer dos iteraciones o pasos.

1. Paso 1. *t* = 0*.*25. Tenemos que *t*0 = 0 e *y*0 = *y*(0) = 1. Aplicamos el esquema y se obtiene que:

*t*0*,*1 = *t*0 + 0 *· h* = *t*0 = 0*, t*0*,*2 = *t*0 + 1 *· h* = 0*.*25*,*

*y*0*,*1 = *y*0 + *b*11*hf* (*t*0*,*1*, y*0*,*1) + *b*12*hf* (*t*0*,*2*, y*0*,*2) = *y*0 = 1*,*

*y*0*,*2 = *y*0 + *b*21*hf* (*t*0*,*1*, y*0*,*1) + *b*22*hf* (*t*0*,*2*, y*0*,*2) = *y*0 + *hf* (0*,* 1) = 1*.*5*,*

y finalmente, considerando los pesos de integraci´on *a*1 = 0*.*5 y *a*2 = 0*.*5,

*y*1 = *y*0 + *a*1*hf* (*t*0*,*1*, y*0*,*1) + *a*2*hf* (*t*0*,*2*, y*0*,*2)

= 1 + 0*.*5 *·* 0*.*25*f* (0*,* 1) + 0*.*5 *·* 0*.*25*f* (0*.*25*,* 1*.*5) = 1*.*556186*.*

1. Paso 2. *t* = 0*.*5. Partimos de *t*1 = 0*.*25 e *y*1 = 1*.*556186. Aplicamos el esquema y se obtiene que:

*t*1*,*1 = *t*1 + 0 *· h* = *t*1 = 0*.*25*, t*1*,*2 = *t*1 + 1 *· h* = 0*.*5*,*

*y*1*,*1 = *y*1 + *b*11*hf* (*t*1*,*1*, y*1*,*1) + *b*12*hf* (*t*1*,*2*, y*1*,*2) = *y*1 = 1*.*556186*,*

*y*1*,*2 = *y*1 + *b*21*hf* (*t*1*,*1*, y*1*,*1) + *b*22*hf* (*t*1*,*2*, y*1*,*2) = *y*1 + *hf* (0*.*25*,* 1) = 2*.*179922*,*

y finalmente, considerando los pesos de integraci´on *a*1 = 0*.*5 y *a*2 = 0*.*5,

*y*2 = *y*1 + *a*1*hf* (*t*1*,*1*, y*1*,*1) + *a*2*hf* (*t*1*,*2*, y*1*,*2) = 2*.*237168*.*

El valor aproximado de *y*(0*.*5) obtenido por el m´etodo de Heun es *y*2 = 2*.*237168. Por u´ltimo, calculamos el error cometido,

*e* = *|y*2 *− y*(0*.*5)*|* = *|*2*.*237168 *−* 2*.*25*|* = 0*.*012832*.*

*{*Ej.T2.Sol.5*}* **Ejercicio 3.5** *Se considera el problema de valor inicial (PVI):*

f

*y′* = 2*te−y, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 1]*, utilizar el esquema num´erico dado por la siguiente tabla Runge-Kutta,*

0 0 0

1 1 0

0*.*5 0*.*5

*para obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 1*, consi- derando una longitud de paso constante h* = 0*.*5*. Hallar el error cometido.*

#### Soluci´on.

Primero calcularemos la soluci´on anal´ıtica para poder calcular posteriormente el error cometido. Se trata de una EDO de variables separadas y por tanto, se resuelve como sigue,

*ey y*

*dy* = 2*tdt* =*⇒ e* =

*t*2 2

2 + *C* =*⇒ y*(*t*) = ln(*t*

+ *K*)*,*

donde la constante *C* se tiene que elegir de modo que *y*(0) = 1. Entonces resulta que *C* = *e* y la soluci´on exacta es *y*(*t*) = ln(*t*2 + *e*). Por u´ltimo, tenemos que *y*(1) = ln(1 + *e*) = 1*.*313261*.*

Pasemos ahora a calcular el valor aproximado con el m´etodo num´erico dado por la tabla. Puesto que *h* = 0*.*5, tendremos que hacer dos iteraciones o pasos:

*Paso 1* (*t* = 0*.*5). Tenemos que *t*0 = 0 e *y*0 = *y*(0) = 1. Aplicamos el esquema y se obtiene que:

*t*0*,*1 = *t*0 + 0 *· h* = *t*0 = 0 (*→ c*1 = 0)*, t*0*,*2 = *t*0 + 1 *· h* = 0*.*5 (*→ c*2 = 1)*,*

*y*0*,*1 = *y*0 + *b*11*hf* (*t*0*,*1*, y*0*,*1) + *b*12*hf* (*t*0*,*2*, y*0*,*2) = *y*0 = 1*,*

*y*0*,*2 = *y*0 + *b*21*hf* (*t*0*,*1*, y*0*,*1) + *b*22*hf* (*t*0*,*2*, y*0*,*2) = *y*0 + *hf* (0*,* 1) = 1*,*

y finalmente, considerando los pesos de integraci´on *a*1 = 0*.*5 y *a*2 = 0*.*5,

*y*1 = *y*0 + *a*1*hf* (*t*0*,*1*, y*0*,*1) + *a*2*hf* (*t*0*,*2*, y*0*,*2) = 1*.*091969*.*

*Paso 2* (*t* = 1). Partimos de *t*1 = 0*.*5 e *y*1 = 1*.*091969. Aplicamos el esquema y se obtiene que:

*t*1*,*1 = *t*1 + 0 *· h* = *t*1 = 0*.*5 (*→ c*1 = 0)*, t*1*,*2 = *t*1 + 1 *· h* = 1 (*→ c*2 = 1)*,*

*y*1*,*1 = *y*1 + *b*11*hf* (*t*1*,*1*, y*1*,*1) + *b*12*hf* (*t*1*,*2*, y*1*,*2) = *y*1 = 1*.*091969*,*

*y*1*,*2 = *y*1 + *b*21*hf* (*t*1*,*1*, y*1*,*1) + *b*22*hf* (*t*1*,*2*, y*1*,*2) = *y*1 + *hf* (0*.*5*,* 1*.*091969) = 1*.*259737*,*

y finalmente, considerando los pesos de integraci´on *a*1 = 0*.*5 y *a*2 = 0*.*5,

*y*2 = *y*1 + *a*1*hf* (*t*1*,*1*, y*1*,*1) + *a*2*hf* (*t*1*,*2*, y*1*,*2) = 1*.*317722*.*

El valor aproximado de *y*(0*.*5) obtenido por el m´etodo de Heun es *y*2 = 1*.*317722. Por u´ltimo, calculamos el error cometido,

*e* = *|y*2 *− y*(0*.*5)*|* = *|*1*.*317722 *−* 1*.*313261*|* = 4*.*46093 *·* 10*−*3*.*

**Ejercicio 3.6** *Se considera el problema de valor inicial (PVI):*

*y′* = 2*√y*(2 + 2*t*)*, y*(0) = 1*.*

f

*En el intervalo temporal* [0*,* 0*.*5]*, utilizar el esquema num´erico dado por la si- guiente tabla (tipo Runge -Kutta):*

*{*Ej.T2.Sol.6*}*

0

1

2

1

0

1

2

*−*1

1

6

0 0

0 0

2 0

4 1

6 6

*para obtener un valor aproximado de la solucion y*(*t*) *en el tiempo t* = 0*.*5*, considerando una longitud de paso constante h* = 0*.*5*. Hallar el error cometido.*

#### Soluci´on.

Puesto que *h* = 0*.*5, s´olo necesitaremos hacer un paso para llegar a *t* = 0*.*5, por tanto, siguiendo el esquema, se tiene que el valor aproximado de la soluci´on en *t* = 0*.*5, denotado por *y*1, se calcula como sigue:

*t*0*,*1 = *t*0 = 0*,*

*t*0*,*2 = *t*0 + *h* = 0*.*25*, t*0*,*3 = *t*0 + *h* = 0*.*5*,*

*y*0*,*1 = *y*0 = 1*,*

1

*y*0*,*2 = *y*0 + 2 *hf* (*t*0*,*1*, y*0*,*1) = 2*,*

*y*0*,*3 = *y*0 + (*−h*)*f* (*t*0*,*1*, y*0*,*1) + 2*hf* (*t*0*,*2*, y*0*,*2) = 6*.*07106*,*

y finalmente

*h*

*y*1 = *y*0 + 6 (*f* (*t*0*,*1*, y*0*,*1) + 4*f* (*t*0*,*2*, y*0*,*2) + *f* (*t*0*,*3*, y*0*,*3)) = 5*.*3937*.*

Para hallar el error necesitamos calcular la soluci´on exacta resolviendo la EDO (de variables separadas). Obtenemos que la soluci´on exacta es:

*y*(*t*) = (*t* + 1)4*,*

y por tanto, el valor en *t* = 0*.*5 es *y*(0*.*5) = (1*.*5)4 = 4*.*9223*,* as´ı que el error cometido por el m´etodo num´erico es *e* = *|*4*.*9223 *−* 5*.*0625*|* = 0*.*1402

*{*Ej.T2.Sol.7*}* **Ejercicio 3.7** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′* = (2 *− t*)*y*3*, y*(0) = 1 *.*

2

1. *Aplicar el m´etodo de Heun (*heun.m*) para resolver el PVI anterior en el intervalo temporal* [0*,* 1]*, tomando como paso de discretizaci´on h* = 0*.*01*. Escribir los valores obtenidos para los tiempos t* = 0*.*1*, t* = 0*.*5 *y t* = 1*.*

1

1. *Dicho problema tiene la siguiente soluci´on exacta y*(*t*) =

*Hallar el error cometido por el m´etodo en t* = 1*.*

*√t*2 *−* 4*t* + 4 *.*

1. *Dibujar las gr´aficas de la solucion num´erica y de la soluci´on exacta en un mismo plot.*

#### Soluci´on.

Para resolver el PVI utilizando heun.m tenemos que escribir

* f =@(t,y) (2-t).\*y.^3;
* valorini = 0.5; intiempo = [0 1];
* npasos = 100;
* [soluciont,soluciony] = heun(f,intiempo,valorini,npasos);

Los valores en *t* = 0*.*1, *t* = 0*.*5 y *t* = 1 se corresponden, respectivamente, con *soluciony*(11) = 0*.*52632, *soluciony*(51) = 0*.*66666 y *soluciony*(101) = 0*.*99993. Para calcular el error cometido en *t* = 1, introducimos la soluci´on exacta y el comando:

* exacta =@(t) (sqrt(t.^2-4\*t+4)).^(-1);
* e = abs(exacta(1)-y(101))

dando como resultado 6*.*8550 10*−*5. Para representar gr´aficamente las dos so- luciones (ver Figura [3.2)](#_bookmark22) escribimos

*·*

* plot(soluciont,soluciony,’r’,soluciont,exacta(t),’k’)

**Ejercicio 3.8** *(Octave) Se considera el problema de valor inicial:*

*−*

*y′* = *y* sin(*t*) + cos(*t*)*, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 2]*, utilizar los esquemas num´ericos de Euler expl´ıci- to, Euler impl´ıcito y Crank-Nicolson, para resolver el problema de valor inicial tomando un paso de discretizaci´on h* = 0*.*1*. Sabiendo que la soluci´on exacta es y*(*t*) = *et* + *sin*(*t*)*, dibujar las soluciones obtenidas con los tres m´etodos y la soluci´on exacta en un mismo plot. Comparar los resultados obtenidos. Escribir los valores de las distintas soluciones en t* = 0*.*5 *y t* = 1*.*5*.*

#### Soluci´on.

Para resolver el problema de valor inicial primero tenemos que definir:

*{*Ej.T2.Sol.8*}*

1

0.95

0.9

0.85

0.8

0.75

0.7

0.65

0.6

0.55

0.5

0 0.2 0.4 0.6 0.8 1

Figura 3.2: En negro aparece representada la soluci´on exacta y en rojo la soluci´on num´erica obtenida con Heun. Como puede observarse, los resultados aparecen

solapados. *{*FigEj2.1*}*

* + f = @(t,y) y-sin(t)+cos(t)
  + intiempo = [0 2]; valorini= 1; npasos = 20;

para posteriormente ejecutar los c´odigos eulerexplicito.m (Euler expl´ıcito), eule- rimplicito.m (Euler impl´ıcito) y cranknicolson.m (Crank-Nicolson) y as´ı obtener las soluciones:

* + [t,u1] = eulerexplicito(f,intiempo,valorini,npasos);
  + [t,u2] = eulerimplicito(f,intiempo,valorini,npasos);
  + [t,u3] = cranknicolson(f,intiempo,valorini,npasos);

A continuaci´on definimos la soluci´on exacta para comparar los resultados num´eri- cos obtenidos con cada uno de los tres m´etodos y la soluci´on exacta dibuj´andolos en un mismo plot:

* + solexac = exp(t)+sin(t);
  + figure;
  + plot(t,u1,’ro’,t,u2,’b+’,t,u3,’g\*’,t,solexac,’k^’)

Los resultados aparecen representados en la Figura [3.3.](#_bookmark23) Como puede observar- se, el m´etodo de Crank-Nicolson es el que mejor aproxima la soluci´on exacta, seguido del m´etodo de Euler impl´ıcito.

Finalmente escribimos los valores obtenidos con cada uno de los m´etodos para

*t* = 0*.*5 y *t* = 1*.*5:

*>* t1 = min(find(t>=0.5));

*>* t2 = min(find(t>=1.5));

* + u1(t1), u2(t1), u3(t1)
  + u1(t2), u2(t2), u3(t2)

9

8

7

6

5

4

3

2

1

0 0.5 1 1.5 2

*{*figsem4*}*

Figura 3.3: Con c´ırculos rojos aparecen representados los resultados obtenidos con Euler expl´ıcito, con cruces azules los obtenidos con Euler impl´ıcito, con as- teriscos verdes los correspondientes a Crank-Nicolson y la exacta con tri´angulos negros.

dando como resultado, para *t* = 0*.*5, 2*.*09636, 2*.*16459 y 2*.*1283 respectivamente,

y para *t* = 1, 5*.*25160, 5*.*75853 y 5*.*4825, respectivamente.

**Ejercicio 3.9** *(Octave) Se considera el problema de valor inicial:*

*−*

*y′* = *ktβy, y*(0) = 1*,*

*donde k* = 4 *y β* = 0*.*25*. Resolver el PVI en el intervalo* [0*,* 4] *utilizando los m´etodos de Euler expl´ıcito y Euler impl´ıcito. Analizar la estabilidad de los dos m´etodos utilizando para ello distintos pasos de discretizaci´on, por ejemplo h*1 = 0*.*5*, h*2 = 0*.*4*, h*3 = 0*.*2 *y h*4 = 0*.*1*.*

#### Soluci´on.

Primero definimos:

* f = @(t,y) -4.\*y.\*t.^(0.25);
* valorini = 1; intiempo = [0 4];

y los intervalos a considerar dependiendo de las longitudes de los pasos de dis- cretizaci´on:

* npasos1 = 8; npasos2 = 10; npasos3 = 20; npasos4 = 40;

A continuaci´on ejecutamos los c´odigos correspondientes a los m´etodos de Euler expl´ıcito (*eulerexplicito.m*) y Euler impl´ıcito (*eulerimplicito.m*) para los distin- tos pasos:

*{*Ej.T2.Sol.9*}*

* + [t1,u1] = eulerexplicito(f,intiempo,valorini,npasos1);
  + [t2,u2] = eulerexplicito(f,intiempo,valorini,npasos2);
  + [t3,u3] = eulerexplicito(f,intiempo,valorini,npasos3);
  + [t4,u4] = eulerexplicito(f,intiempo,valorini,npasos4);
  + [t1,ub1] = eulerimplicito(f,intiempo,valorini,npasos1);
  + [t2,ub2] = eulerimplicito(f,intiempo,valorini,npasos2);
  + [t3,ub3] = eulerimplicito(f,intiempo,valorini,npasos3);
  + [t4,ub4] = eulerimplicito(f,intiempo,valorini,npasos4);

Los resultados obtenidos para el m´etodo de Euler expl´ıcito, aparecen represen- tados en la Figura [3.4](#_bookmark24) (a):

* + figure;
  + plot(t1,u1,’ro’,t2,u2,’b+’,t3,u3,’g\*’,t4,u4,’k^’)

Los resultados obtenidos para el m´etodo de Euler impl´ıcito, aparecen represen- tados en la Figura [3.4](#_bookmark24) (b).

* + figure;
  + plot(t1,ub1,’ro’,t2,ub2,’b+’,t3,ub3,’g\*’,t4,ub4,’k^’)

3 1

2 0.9

0.8

1

0.7

0 0.6

−1 0.5

−2 0.4

0.3

−3

0.2

−4 0.1

−5

0 0.5 1 1.5 2 2.5 3 3.5 4

0

0 0.5 1 1.5

2 2.5 3 3.5 4

*{*segundosem\_1*}*

Figura 3.4: Resultados obtenidos con el m´etodo de Euler expl´ıcito (a) y Eu- ler impl´ıcito (b): con c´ırculos rojos para *npasos* = 8, con cruces azules para *npasos* = 10, con asteriscos verdes para *npasos* = 20 y con tri´angulos negros para *npasos* = 40.

Como puede observarse, el m´etodo de Euler impl´ıcito no presenta problemas de inestabilidad, si bien, los resultados se vuelven m´as precisos a medida que disminuye el paso de discretizaci´on. Sin embargo, el m´etodo de Euler expl´ıcito se vuelve inestable para los taman˜os de discretizaci´on *h* = 0*.*4 y *h* = 0*.*5.

Ej.T2.Sol.10

*{ }*

**Ejercicio 3.10** *(Octave) Se considera el problema de valor inicial (PVI):*



*y′* = 2*y ,*

*t* + 1

 *y*(0) = 1*.*

*En el intervalo temporal* [0*,* 1]*, utilizar los esquemas num´ericos de Euler expl´ıci- to y Euler impl´ıcito, para obtener el valor aproximado de la solucion en t* = 1 *tomando un paso de discretizaci´on h* = 0*.*1*. Sabiendo que la soluci´on exacta es y*(*t*) = (*t* + 1)2*, dibujar las soluciones obtenidas con los dos m´etodos y la solu- ci´on exacta en un mismo plot. Comparar los resultados obtenidos. Escribir los valores de las distintas soluciones en t* = 0*.*5 *y t* = 0*.*8*. Encontrar los intervalos temporales para los cuales, las distintas soluciones num´ericas alcanzan valores superiores a tres.*

#### Soluci´on.

Primero definimos:

* f = @(t,y) 2.\*y.\*(t+1).^(-1);
* valorini =1; intiempo = [0 1]; npasos = 10;

y posteriormente ejecutamos los c´odigos eulerexplicito.m y eulerimplicito.m, pa- ra as´ı obtener las soluciones:

* [t1,u1] =eulerexplicito(f,intiempo,valorini,npasos);
* [t1,v1] = eulerimplicito(f,intiempo,valorini,npasos);

Definimos la soluci´on exacta y dibujamos los resultados obtenidos:

* solexac = inline(’(t+1).^2’,’t’);
* plot(t1,u1,’ro’,t1,v1,’b+’,t1,solexac(t1),’k^’)

que aparecen representados en la Figura [3.5.](#_bookmark25) Puede observarse que con Euler expl´ıcito la soluci´on es aproximada por defecto y que con Euler impl´ıcito por exceso.

A continuaci´on, escribimos los resultados obtenidos con los tres c´odigos (m´eto- dos) para los tiempos *t* = 0*.*5 y *t* = 0*.*8. Calculamos las posiciones donde se encuentran, dentro del mallado t

* pos1 = min(find(t==0.5));
* pos2 = min(find(t==0.8));

Con esto, calculamos las aproximaciones:

* u1(pos1), v1(pos1)
* u1(pos2), v1(pos2)

que son: 2.1818 y 2.3333 respectivamente, para *t* = 0*.*5, y 3.1091 y 3.4000 res- pectivamente, para *t* = 1. Para finalizar, calculamos los tiempos para los cuales las soluciones num´ericas toman valores mayores a 3 con los comandos:

*>* x1 = find(u1>3); IntEE = [t(min(x1)), t(max(x1))]

*>* x2 = find(v1>3); IntEI = [t(min(x2)), t(max(x2))]

4.5

4

3.5

3

2.5

2

1.5

1

0 0.2 0.4 0.6 0.8 1

Figura 3.5: Con c´ırculos rojos aparecen representados los resultados obtenidos con Euler expl´ıcito, con cruces azules los obtenidos con Euler impl´ıcito, con as- teriscos verdes los correspondientes a Crank-Nicolson y la exacta con tri´angulos

negros. *{*segundosem2*}*

y obtenemos el intervalo [0*.*8*,* 1] con Euler expl´ıcito, y [0*.*7*,* 1] con Euler impl´ıcito.

# Problemas propuestos

*{*Ej.T2.Prop.1*}* **Ejercicio 3.11** *Se considera el problema de valor inicial (PVI):*

*y′* = 1 *, ety*





*y*(0) = *√*2*.*

*En el intervalo temporal* [0*,* 0*.*5]*, utilizar el esquema num´erico de Euler impl´ıcito para encontrar un valor aproximado de la soluci´on en t* = 0*.*5 *tomando h* = 0*.*5 *y considerando dos iteraciones del m´etodo de Newton-Raphson para resolver la ecuaci´on no lineal. Calcular el error cometido.*

*{*Ej.T2.Prop.2*}* **Ejercicio 3.12** *Se considera el problema de valor inicial (PVI):*

f

*y′* = *−*2*ty*2*, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 1]*, utilizar el esquema num´erico dado por la siguiente tabla (tipo Runge-Kutta):*

0 0 0

1 1 0

0*.*5 0*.*5

*para obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 1*, consi- derando una longitud de paso constante h* = 0*.*5*. Hallar el error cometido.*

**Ejercicio 3.13** *Se considera el problema de valor inicial (PVI):*

f

*y′* = *−y*2*et, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 1]*, utilizar el esquema num´erico dado por la siguiente tabla Runge-Kutta,*

*{*Ej.T2.Prop.3*}*

0 0 0

1 1 0

0*.*5 0*.*5

*para obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 1*, consi- derando una longitud de paso constante h* = 0*.*5*. Hallar el error cometido.*

**Ejercicio 3.14** *Consid´erese el problema de valor inicial (PVI):*

f

*y′* = 2*y*1*/*2*t, y*(0) = 1*.*

*En el intervalo temporal* [0*,* 1]*, utilizar el esquema num´erico dado por la siguiente tabla (tipo Runge-Kutta):*

*{*Ej.T2.Prop.4*}*

0 0 0

1 1 0

0*.*5 0*.*5

*para obtener un valor aproximado de la soluci´on y*(*t*) *en el tiempo t* = 1*, consi- derando una longitud de paso constante h* = 0*.*5*. Hallar el error cometido.*

**Ejercicio 3.15** *Sea el Problema de Valor Inicial:*

 *y′* = 1 *− y*2*, t ∈* [0*,* 1*.*5]*,*

*{*Ej.T2.Prop.5*}*

*(PVI)* :

 *y*(0) = 0*,*

*que tiene por soluci´on exacta la funci´on y*(*t*) = sin(*t*)*.*

1. *Tomando un paso de discretizaci´on de h* = 0*.*5*, utiliza el m´etodo de Euler impl´ıcito para dar una aproximaci´on de la soluci´on en t* = 1*, utilizan- do para resolver las ecuaciones no lineales que aparecen en el proceso, 3 iteraciones del m´etodo de bisecci´on. Calcula el error cometido en la apro- ximaci´on obtenida.*
2. *(Octave) Resuelve el mismo PVI con Octave utilizando el m´etodo de Crank- Nicolson utilizando un paso de discretizaci´on h* = 0*.*01 *en el intervalo* [0*,* 1*.*5]*. Dibuja la aproximaci´on num´erica junto con la soluci´on exacta y calcula la aproximaci´on que produce el m´etodo en el instante t* = 0*.*73*.*

*¿Qu´e error se comete?*

1. *(Octave) Calcula en intervalo temporal (num´erico) para el cual las apro- ximaciones son mayores que 0.6. ¿Qu´e error se comete en la estimaci´on? (*Nota: Para calcular el intervalo temporal exacto utiliza una tolerancia de 10*−*5*).*
2. *(Octave) Representa la gr´afica del error cometido por el m´etodo en el intervalo [0,1.5]. ¿Cu´al es el error m´aximo que se comete?*

*{*Ej.T2.Prop.6*}* **Ejercicio 3.16** *(Octave) Se considera el problema de valor inicial (PVI):*

*− −*

*y′* = *y*(*y* 1)*,*

*y*(0) = 2*.*

*Aplicar los m´etodos tipo Runge-Kutta dados por los algoritmos heun.m y runge- kuttao3.m para resolver el PVI anterior en el intervalo temporal [0,1], tomando como paso de discretizaci´on h* = 0*.*01*. Escribir los valores obtenidos con los dos m´etodos para los tiempos, t* = 0*.*1*, t* = 0*.*5 *y t* = 1*.*

Ej.T2.Prop.7

*{ }*

**Ejercicio 3.17** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = (1 + 6*t*)*e*1*−y −* 6*, ∀ t ∈* (0*,* 6]*, y*(0) = ln(10) + 1*,*

*cuya soluci´on anal´ıtica es:*

*y*(*t*) = ln (10*e−*6*t* + *t*) + 1*.*

*Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el intervalo temporal* [0*,* 6] *con pasos de discretizaci´on h*1 = 0*.*5*, h*2 = 0*.*15 *y h*3 = 0*.*25*. Discutir el comportamiento de las soluciones obtenidas en t´erminos de estabi- lidad num´erica. Dibujar la soluci´on anal´ıtica junto con las soluciones estables calculadas anteriormente. Calcular anal´ıticamente el instante en el cual se tiene el m´ınimo de concentraci´on as´ı como su valor.*

*{*Ej.T2.Prop.8*}*

**Ejercicio 3.18** *(Octave) Se considera el problema de valor inicial (PVI):*

 *y′*(*t*) = *y − t , ∀ t ∈* (0*,* 5]*,*

*y*

 *y*(0) = 2*,*

*cuya soluci´on anal´ıtica es:*

*y*(*t*) = 1 14*e*2*t* + 4*t* + 2*.*

2

1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 5] *con paso de discretizaci´on h* = 0*.*2*. Utilizar el resul- tado obtenido para proximar el valor de la soluci´on anal´ıtica en el instante t* = 3*.*6 *y calcular el error cometido.*

*¿Sabr´ıas calcular la soluci´on en el instante t* = 3*.*627 *y estimar el error cometido?*

1. *Aplicar los algoritmos de Euler impl´ıcito y Crank-Nicolson para calcular la soluci´on del PVI en las mismas condiciones anteriores (en el intervalo temporal* [0*,* 5] *y con paso de discretizaci´on h* = 0*.*2*).*

*Utilizar los resultados obtenidos para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 3*.*6 *y calcular los errores cometidos determi- nando el m´etodo m´as preciso entre los tres.*

**Ejercicio 3.19** *(Octave) Se considera el problema de valor inicial (PVI):*

*y′*(*t*) = (*√t −* 1)(*y*3 *− y*)*, ∀ t ∈* (0*,* 2]*, y*(0) = 2*,*

f

*cuya soluci´on anal´ıtica es:*

*{*Ej.T2.Prop.9*}*

*y*(*t*) =

1

*.*

1 *−* (3*/*4)*e*(4*/*3)*t*3*/*2 *−*2*t*

1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 2] *con paso de discretizaci´on h* = 0*.*05*.*

*Utilizar el resultado obtenido para proximar el valor de la soluci´on anal´ıti- ca en el instante t* = 1*.*25 *y calcular el error cometido. ¿En que instante se tiene el m´ınimo de concentraci´on? ¿Cu´al es su valor real y el aproximado?*

1. *Aplicar el algoritmo de Runge-Kutta,* heun.m *(Heun), para calcular la soluci´on del PVI en las mismas condiciones anteriores (en el intervalo temporal* [0*,* 2] *y con paso de discretizaci´on h* = 0*.*05*).*

*Utilizar los resultados obtenidos para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 1*.*25 *y calcular los errores cometidos y de- terminar el m´etodo m´as preciso. Dibujar la gr´afica del error del m´etodo determinado anteriormente en cada punto del dominio.*

**Ejercicio 3.20** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = 1 *−* 4*y, ∀ t ∈* (0*,* 10]*,*

*y*(0) = 1*,*

*{*Ej.T2.Prop.10*}*

*cuya soluci´on anal´ıtica es:*

*y*(*t*) =

3 + *e*4*t*

4*e*4*t .*

*Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el intervalo temporal* [0*,* 10] *con pasos de discretizaci´on h*1 = 0*.*1*, h*2 = 0*.*4 *y h*3 = 0*.*5*.*

*Dibujar las soluciones obtenidas junto con la anal´ıtica en una u´nica gr´afica y discutir el comportamiento de las soluciones obtenidas en t´erminos de estabili- dad num´erica.*

*{*Ej.T2.Prop.11*}*

**Ejercicio 3.21** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = *y − y*3*, ∀ t ∈* (0*,* 20]*, y*(0) = 2*,*

*cuya soluci´on anal´ıtica es:*

2*et*

*y*(*t*) = *√*4*e*2*t −* 3 *.*

* 1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 20] *con paso de discretizaci´on h* = 0*.*2*.*

*Utilizar el resultado obtenido para proximar el valor de la soluci´on anal´ıti- ca en el instante t* = 0*.*2 *y calcular el error cometido.*

*b) Aplicar los algoritmos de Euler impl´ıcito y Crank-Nicolson para calcular la soluci´on del PVI en las mismas condiciones anteriores (en el intervalo temporal* [0*,* 20] *y con paso de discretizaci´on h* = 0*.*2*).*

*Utilizar los resultados obtenidos para proximar el valor de la soluci´on anal´ıtica en el instante t* = 0*.*2*, calcular los errores cometidos y deter- minar el m´etodo m´as preciso.*

Ej.T2.Prop.12

*{ }*

**Ejercicio 3.22** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = 1 *− te−y, ∀ t ∈* (0*,* 20]*,*

*y*(0) = 0*,*

*cuya soluci´on anal´ıtica es:*

*y*(*t*) = ln(*t* + 1)*.*

1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 2] *con paso de discretizaci´on h* = 0*.*1*.*

*Utilizar el resultado obtenido para proximar el valor de la soluci´on anal´ıti- ca en el instante t* = 1*.*6 *y calcular el error cometido.*

1. *Aplicar los algoritmos de Runge-Kutta, heun.m y rungekuttao3.m, para calcular la soluci´on del PVI en las mismas condiciones anteriores (en el intervalo temporal* [0*,* 20] *y con paso de discretizaci´on h* = 0*.*1*).*

*Utilizar los resultados obtenidos para proximar el valor de la soluci´on anal´ıtica en el instante t* = 10*.*6*, calcular los errores cometidos y determi- nar el m´etodo m´as preciso.*

**Ejercicio 3.23** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = 10(2 *− y*)*, ∀ t ∈* (0*,* 3]*,*

*y*(0) = 1*,*

*cuya soluci´on anal´ıtica es:*

*{*Ej.T2.Prop.13*}*

*y*(*t*) = 2 *− e−*10*t.*

*Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el intervalo temporal* [0*,* 3] *con pasos de discretizaci´on h*1 = 0*.*3*, h*2 = 0*.*15 *y h*3 = 0*.*05*.*

*Discutir el comportamiento de las soluciones obtenidas en t´erminos de estabi- lidad num´erica. Dibujar la soluci´on anal´ıtica junto con las soluciones estables calculadas anteriormente.*

**Ejercicio 3.24** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = *y −* 1*/y, ∀ t ∈* (0*,* 4]*, y*(0) = 3*,*

*cuya soluci´on anal´ıtica es:*

*{*Ej.T2.Prop.14*}*

*y*(*t*) = 8*e*2*t* + 1*.*

1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 1] *con paso de discretizaci´on h* = 0*.*025*. Utilizar el resultado obtenido para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 0*.*3 *y calcular el error cometido.*
2. *Aplicar los algoritmos de Euler impl´ıcito y Crank-Nicolson para calcular la soluci´on del PVI en las mismas condiciones anteriores (en el intervalo temporal* [0*,* 1] *y con paso de discretizaci´on h* = 0*.*025*). Utilizar los re- sultados obtenidos para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 0*.*3*, calcular los errores cometidos y determinar el m´etodo m´as preciso.*

**Ejercicio 3.25** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = (*t −* 1)(*y − y*2)*, ∀ t ∈* (0*,* 2]*, y*(0) = 1*/*2*.*

*{*Ej.T2.Prop.15*}*

*cuya soluci´on anal´ıtica es:*

*e*(1*/*2)*t*2*−t y*(*t*) = 1 + *e*(1*/*2)*t*2 *−t .*

* 1. *Aplicar el algoritmo de Euler expl´ıcito para calcular la soluci´on en el in- tervalo temporal* [0*,* 2] *con paso de discretizaci´on h* = 0*.*004*.*

*Utilizar el resultado obtenido para aproximar el valor de la soluci´on anal´ıti- ca en el instante t* = 1*.*24 *y calcular el error cometido. Calcular el instante en el que se tiene el m´ınimo de concentraci´on. ¿Cu´al es su valor real y el aproximado?*

* 1. *Aplicar los algoritmos de Runge-Kutta, heun.m y rungekuttao3.m para calcular la soluci´on del PVI en las misma s condiciones anteriores (en el intervalo temporal* [0*,* 2] *y con paso de discretizaci´on h* = 0*.*004*).*

*Utilizar los resultados obtenidos para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 1*.*24 *y calcular los errores cometidos y deter- minar el m´etodo m´as preciso. Dibujar la gr´afica del error de estos m´etodos en cada punto.*

Ej.T2.Prop.16

*{ }*

**Ejercicio 3.26** *(Octave) Se considera el problema de valor inicial (PVI):*

f

*y′*(*t*) = *te*3*t −* 2*y, ∀t ∈* (0*,* 1]*, y*(0) = 0*,*

*cuya soluci´on anal´ıtica es y*(*t*) = 1 *te*3*t −*  1 *e*3*t* + 1 *e−*2*t.*

5

25

25

1. *Aplicar los m´etodos de Euler expl´ıcito, Euler impl´ıcito, Heun y Simpson (rungekuttao3.m) con paso de discretizaci´on h* = 0*.*05 *para aproximar el valor de la soluci´on anal´ıtica en el instante t* = 0*.*95 *y calcular los errores cometidos. ¿Cu´al es el m´etodo m´as preciso? ¿Cu´al es el menos preciso? Justifica tus respuestas.*
2. *Dibuja la gr´afica del error cometido por el m´etodo de Euler impl´ıcito y el m´etodo de Heun en el intervalo* [0*,* 1]*.*

*{*Ej.T2.Prop.17*}* **Ejercicio 3.27** *(Octave) Se considera el problema de valor inicial (PVI):*

*y′*(*t*) = *ty t*2 + 1





*y*(0) = 1*,*

*cuya soluci´on anal´ıtica es:*

*, ∀t ∈* (0*,* 2]*,*

*y*(*t*) = *t*2 + 1*.*

1. *Aplicar los m´etodos de Euler expl´ıcito, Euler impl´ıcito, Heun (*heun.m*) y Simpson (*rungekuttao3.m*) con paso de discretizaci´on h* = 0*.*1 *para apro- ximar el valor de la soluci´on anal´ıtica en el instante t* = 0*.*7 *y calcular los errores cometidos. ¿Cu´al es el m´etodo m´as preciso? ¿Cu´al es el menos preciso? Justifica tus respuestas.*
2. *Dibuja la gr´afica del error cometido por el m´etodo de Euler Expl´ıcito y el m´etodo de Heun en el intervalo* [0*,* 2]*.*

# Aplicaciones

**Ejercicio 3.28 (Capa l´ımite laminar)** *En el estudio de algunas capas l´ımite de flujos laminares, aparece la denominada ecuaci´on de Blasius que es una EDO de tercer orden de la forma:*

*y′′′*(*x*) + *y*(*x*)*y′′*(*x*) = 0*, x ≥* 0*.*

*N´otese que aqu´ı hemos modificado la notaci´on de la variable independiente llam´andola x, pues en esta ocasi´on la variable independiente no es el tiem- po, sino que es una variable adimensional relacionada con la coordenada es- pacial. La ecuaci´on de Blasius se acompan˜a de las tres condiciones iniciales y*(0) = *y′*(0) = 0*, y′′*(0) = *α, donde α es un valor conocido* (*α* = 0*.*47)*. Con ello se puede plantear el problema de valor inicial de Blasius de la forma siguiente:*

*Hallar una funci´on y*(*x*) *verificando:*

f

*y′′′*(*x*) + *y*(*x*)*y′′*(*x*) = 0*, x ≥* 0*,*

*y*(0) = 0*, y′*(0) = 0*, y′′*(0) = *α.*

*Este problema es equivalente al P.V.I. de primer orden:*

*{*Ej.T2.Aplic.1*}*

****

****

*z*1*′* (*x*) = *z*2(*x*)*, x ≥* 0*,*

*z*2*′* (*x*) = *z*3(*x*)*, x ≥* 0*,*

*z*3*′* (*x*) = *−z*1(*x*)*z*3(*x*)*, x ≥* 0*,*

 *z*1(0) = 0*, z*2(0) = 0*, z*3(0) = *α,*

*donde se ha denotado por z*1(*x*) = *y*(*x*)*, z*2(*x*) = *y′*(*x*)*, z*3(*x*) = *y′′*(*x*)*.*

*Abreviadamente el problema anterior puede escribirse entonces como:*

*donde: y:*

{ **z***′*(*x*) = **f** (*x,* **z**(*x*))*, x ≥* 0 [0*.*3*cm*]**z**(0) = *{*0*,* 0*, α}T ,*

**z**(*x*) = *{z*1(*x*)*, z*2(*x*)*z*3(*x*)*}T ,*

**f** (*x,* **z**(*x*)) = *{z*2(*x*)*, z*3(*x*)*, −z*1(*x*)*z*3(*x*)*}T .*

*Se pide calcular y dibujar el perfil de la capa l´ımite y*(*x*)*, resolviendo el PVI asociado al sistema no lineal de EDO.*

*{*Ej.T2.Aplic.2*}*

**Ejercicio 3.29 (Reacci´on qu´ımica)** *Consideraremos ahora un ejemplo que se puede encontrar en el libro de Hanna y Sandall*[1](#_bookmark47)*)*

*En la ingenier´ıa en general, y en la ingenier´ıa qu´ımica en particular, es muy frecuente toparse con problemas de valor inicial regidos por sistemas de ecuacio- nes diferenciales ordinarias. Como bot´on de muestra, baste el siguiente ejemplo en el que se considera una reaci´on qu´ımica no lineal semejante a las que tie- nen lugar en un reactor qu´ımico durante la fase transitoria de una reacci´on a volumen constante de la forma:*

*A* + *B ↔ C → D* + *E,*

*donde A, B, C, D y E representan diferentes compuestos que interaccionan entre s´ı. En una reacci´on qu´ımica como la planteada, la concentraci´on de la especie A, que denotaremos por y*1*, y la de la especie C, que denotaremos por y*2*, se relacionan entre s´ı mediante el sistema de ecuaciones diferenciales ordinarias:*

****

****

*dy*1 *dt*

= *−k*1*y*1(*y*1 *− K*) + *k*2*y*2*,*

*dy*2

*dt*

= *−*(*k*

+ *k* )*y*

+ *k y* (*y*

*— K*)*,*

*donde k*1*, k*2 *y k*3 *son las constantes de reacci´on de las reacciones qu´ımicas A* + *B C, C A* + *B y C D* + *E, respectivamente. Por otra parte, K es una constante que depende de la composici´on inicial de la mezcla que se deja reaccionar. Al sistema anterior debe acompan˜´arsele de las condiciones iniciales*

*→ → →*

2

3

2

1

1

1

*y*(0) *e y*(0)*. Si consideramos el caso en que k*1 = *k*2 = *k*3 = 1*, K* = 0*, y*(0) = 1 *e*

1 2 1

*y*(0) = 0*, el sistema resultante ser´a:*

2

**y***′*(*t*) = **f** (*t,* **y**(*t*))*, t >* 0*,*

**y**(0) = **y**(0)*,*

1Hanna, O.T. y Sandall, O.C. (1995). “Computational Methods in Chemical Engineering”.

Ed. Prentice Hall International Editions.

*donde*

*y*

**y**(*t*) =

*y*1(*t*)

*y*2(*t*)

(

*,* **y***′*(*t*) =

*′*

1

( *y* (*t*)

*y′* (*t*)

2

*,* **y**(0) =

(0)

1

(0)

*y*

2

= (1

( *−y*2(*t*) + *y*2(*t*)

*y*

0

**f** (*t,* **y**(*t*)) =

1

*−*2*y*2(*t*) + *y*1(*t*)

2

*Se pide calcular y dibujar el perfil de las concentraciones* (*y*1(*x*)*, y*2(*x*)) *resol- viendo el PVI asociado al sistema no lineal de EDOs. Comparar la soluci´on obtenida con el valor de la soluci´on exacta de y*1(*t*) *en el instante t* = 0*.*8*, y*1(0*.*8) = 0*.*646234*, tomado de la referencia Hanna y Sandall.*

**Ejercicio 3.30 (Oscilador arm´onico)** *Un l´ıquido de densidad ρ y viscosidad din´amica µ se encuentra en reposo contenido en un tubo de secci´on recta circular de di´ametro constante d y con forma de U. La longitud que ocupa el fluido, medida en el eje del tubo, es denotada por L.*

*En el instante t*0 = 0*, un pist´on golpea una de las superficies libres del fluido im- primi´endole una velocidad V*0*. Como consecuencia de ello, la otra superficie libre comienza a ascender hasta que, por efecto de la fuerza gravitatoria, se detiene su*

*ascenso. En ese momento, el l´ıquido de esa rama del tubo comienza a descender rebasando la posici´on en que inicialmente se encontraba en equilibrio hasta que el efecto de la gravedad sobre la otra rama del fluido detiene el descenso y el l´ıquido vuelve a ascender. Se genera as´ı un movimiento oscilatorio del fluido en torno a su posici´on de equilibrio inicial (que, con el transcurrir del tiempo, volver´a a recuperarse). Si se denota por g a la aceleraci´on gravitatoria y por z*(*t*) *al desplazamiento de la superficie libre del fluido, despreciando el efecto del rozamiento del l´ıquido con las paredes del tubo, el proceso anterior puede modelizarse mediante el problema de valor inicial siguiente:*

f

*z′′*(*t*) + *Az′*(*t*) + *Bz*(*t*) = 0*, t >* 0*,*

*z*(0) = 0*, z′*(0) = *V*0*,*

*{*Ej.T2.Aplic.3*}*

*donde A y B son dos constantes dadas por*

32*µ* 2*g*

*A* = *ρd*2 *, B* = *L .*

*El problema de valor inicial antes planteado puede transformarse en otro regido por un sistema de dos ecuaciones diferenciales de primer orden en la forma:*

**** *y′* (*t*) = *y*2(*t*)*, t >* 0*,*

1

 *y′* (*t*) = *−By*1(*t*) *− Ay*2(*t*)*, t >* 0*,*

 *y*1(0) = 0*, y*2(0) = *V*0*,*

2

*donde se ha denotado por y*1(*t*) = *z*(*t*) *(la funci´on de desplazamiento) y por*

*y*2(*t*) = *z′*(*t*) *(la funci´on de velocidad).*

*Consideremos un fluido para el que µ* = 0*.*001 *Kg/*(*m s*)*, ρ* = 1 *Kgm−*3 *y tomemos como valor de g* = 9*.*81 *ms−*2*, d* = 0*.*2*m, L* = 3*m y V*0 = 1*m/s. Resolver el sistema de ecuaciones (lineal) con paso de integraci´on constante h* = 0*.*05*.*

*¿Sabr´ıas encontrar la soluci´on anal´ıtica de este problema?*

Ej.T2.Aplic.4

*{ }*

**Ejercicio 3.31 (Velocidad Terminal)** *La ecuaci´on diferencial que rige la ve-*

*locidad v de un cuerpo de masa m y ´area proyectada A, que cae en un medio de densidad ρ es*

*dv*

*dt* = *g −*

*ρAv*2

*.*

2*m*

*El cuerpo adquiere su velocidad terminal de ca´ıda cuando no acelera m´as, es decir, cuando la derivada de la velocidad es cero. De acuerdo con la ecuaci´on anterior, se tiene que la velocidad terminal te´orica es*

*v* = 2*mg.*

*f ρA*

*Supongamos que dejamos caer una moneda de un edificio con m* = 0*.*01*kg, A* = 3*.*1416 10*−*4*m−*2 *y ρ* = 1*kg/m*3*. Utiliza Octave para calcular su velo- cidad terminal te´orica y comp´arala con la velocidad terminal que se obtiene al resolver el problema con el m´etodo de Runge-Kuttta dado por el esquema rungekuttao*3*.m.*

*·*

*{*Ej.T2.Aplic.5*}* **Ejercicio 3.32 (Drenaje de un tanque)** *Si se drena el agua desde un tan- que cil´ındrico vertical por medio de una v´alvula en la base, el l´ıquido fluir´a r´apido cuando el tanque est´a lleno y m´as lentamente, conforme se drene. Como resultado, la tasa a la que el nivel de agua disminuye es*

*dy* = *k√y, dt*

*−*

*donde k es una constante que depende de la forma del agujero de drenaje y del*

*´area de la secci´on transversal del tanque. La profundidad del agua y se mide en metros y el tiempo t en minutos.*

*Si k* = 0*.*06*, utiliza Octave para determinar cu´anto tiempo se requiere para va- ciar el tanque si el nivel del fluido se encuentra en un inicio a 3 metros utilizando el m´etodo de resoluci´on num´erica que prefieras y un paso de discretizaci´on de*

*0.5 minutos.*

**Cap´ıtulo 4**

**Problemas de Contorno**

# Problemas resueltos

## Problemas de Transporte estacionarios en dominios unidimensionales

**Ejercicio 4.1** *Se considera el problema de transporte difusivo-convectivo defi- nido por el Problema de Valores de Contorno (PVC):*

f

*−u′′*(*x*) + *u′*(*x*) = *ex, x ∈* (0*,* 1)

*u*(0) = 1*, u*(1) = 0*,*

*cuya soluci´on exacta es u*(*x*) = *−xex* + *ex.*

* + - 1. *Resolver el problema tomando un paso de discretizaci´on h* = 0*.*25*, utili- zando f´ormulas en diferencias finitas centradas, calculando primeramente el sistema asociado al problema en la forma Au*¯ = ¯*b.*
      2. *Calcular el error cometido en los puntos x* = 0*.*5 *y x* = 1*.*
      3. (Octave) *Aplicar el algoritmo bvpdirichlet.m para calcular la soluci´on en* [0*,* 1]*, tomando un paso de discretizaci´on constante h* = 0*.*01*. Calcula la aproximaci´on obtenida en los puntos x* = 0*.*5 *y x* = 1*. ¿Qu´e error comete el m´etodo en dichas aproximaciones?*

*(c)* (Octave) *Dibuja la gr´afica de la aproximaci´on junto con la soluci´on exacta y la gr´afica del error cometido en todo el intervalo* [0*,* 1]*.*

#### Soluci´on.

1. Como *h* = 0*.*25, el mallado consiste en los puntos:

*x*1 = 0*, x*2 = 0*.*25*, x*3 = 0*.*5*, x*4 = 0*.*75*, x*5 = 1*.*

49

*{*Ej.T3.Sol.1*}*

Para *i* = 1 e *i* = 5, dadas las condiciones de contorno, sabemos que *u*1 = 1 y *u*5 = 0, por lo que nos queda por determinar los valores de *u*2, *u*3 y *u*4. Para ellos, utilizando las f´ormulas en diferencias finitas centradas para *u′′*(*xi*) y *u′*(*xi*) obtenemos:

*h*2 2*h*

*−*( *ui−*1 *−* 2*ui* + *ui*+1 ) + ( *ui*+1 *− ui−*1 ) = *exi , i* = 2*,* 3*,* 4*.*

Agrupando t´erminos tenemos que

*−*18*ui−*1 + 32*ui −* 14*ui*+1 = *exi , i* = 2*,* 3*,* 4*.*

La soluci´on buscada se obtiene por tanto resolviendo el sistema lineal *Au*¯ = ¯*b* siguiente:

 1 0 0 0 0

 *u*1

 1 

*−*18 32 *−*14 0 0

 *u*2

*e*0*.*25

 0 *−*18 32 *−*14 0

 *u*3 =  *e*0*.*5 

 0 0 *−*18 32 *−*14 *u*4

0 0 0 0 1

*u*5

*e*0*.*75

0

que da por soluci´on: *u*1 = 0, *u*2 = 0*.*96399, *u*3 = 0*.*82598, *u*4 = 0*.*53077 y *u*5 = 0.

1. El error cometido en *x* = 0*.*5 es *u*(0*.*5) *u*3 = 0*.*0016192, mientras que en

*| − |*

*x* = 1, el error es 0, ya que es conocida la condici´on de contorno.

1. Necesitamos definir:

* a = 0; b = 1; ua = 1; ub = 0; numeronodos=1+((b-a)/h);
* D = 1; V = 1; Q = 0;
* f = @(x) exp(x);

y llamar al programa

* [x,u] = bvpdirichlet(a,b,numeronodos,D,V,Q,f,ua,ub);

Los valores en 0.5 y en 0.75 de la soluci´on num´erica son, respectivamente. u(51) y u(101), dando como aproximaciones 0*.*82436 y 0. La posici´on 51, por ejemplo, puede obtenerse mediante el comando find(x==0.5) o como 0.5/h+1.

Definiendo la soluci´on exacta, podemos calcular el error cometido en *x* = 0*.*5 mediante:

* uexac = @(x) -x.\*exp(x)+exp(x);
* error = abs(u(51)-uexac(0.5))

que da como resultado 2.558724*·*10*−*6. En *x* = 0 el error es nulo.

1. Calculamos el vector con los errores cometidos

*>* verror = abs(u-uexact(x));

y las gr´aficas (ver Figura [4.1)](#_bookmark54) se obtienen mediante los comandos

*>* figure; plot(x,u,x,uexact(x)); legend(’aprox’,’exacta’)

*>* figure; plot(x,verror); title(’ERROR’)

ERROR

1 3e-06

aprox

exacta

2.5e-06

0.8

2e-06

0.6

1.5e-06

0.4

1e-06

0.2

5e-07

0

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

0 0.2 0.4 0.6 0.8 1

0

0 0.2 0.4 0.6 0.8 1

Figura 4.1: Gr´afica de la aproximaci´on obtenida (izquierda) y la gr´afica del error

cometido (derecha). *{*pvc31*}*

Se observa como la aproximaci´on y la soluci´on exacta est´an pr´acticamente so- lapadas y el error es como mucho del orden de 10*−*6.

**Ejercicio 4.2** *Sea dado el Problema de Valores de Contorno (PVC):*

*{*Ej.T3.Sol.2*}*

**** *d*2*u d* r(

1 l

 *−dx*2 + *dx*

****

*x −* 4 *u* = 4*x, x ∈* (0*,* 1)*,*

*u*(0) = 1*, u*(1) = 2*.*

* + 1. *Desarrollar las operaciones indicadas en la ecuaci´on y describir los proce- sos f´ısicos del modelo.*
    2. *Resolver el problema utilizando f´ormulas en diferencias finitas centradas para el t´ermino de difusi´on y descentradas contracorriente para los even- tuales t´erminos de convecci´on, obteniendo un esquema num´erico del tipo:*

*αiui−*1 + *βiui* + *γiui*+1 = *fi, ∀ i* = 2*, . . . , N,*

*definiendo los coeficientes αi, βi y γi, para un gen´erico paso de discreti- zaci´on h.*

* + 1. *Sea h* = 0*.*25*. Escribir los nodos del mallado y calcular una aproximaci´on de la soluci´on en los nodos x*2*, x*3 *y x*4*, resolviendo el sistema asociado al problema en la forma Au*¯ = ¯*b.*
    2. (Octave) *Resuelve el problema para un paso de discretizaci´on de h* = 0*.*01

*y calcula la aproximaci´on que se obtiene en x* = 0*.*261*.*

#### Soluci´on.

1. Desarrollando el t´ermino de divergencia se tiene:

*d* r(*x −* 1 *u*l = (*x −* 1 *du* + *u,*

*dx* 4

luego la ecuaci´on a resolver es:

*d*2*u* (

*−dx*2 +

4 *dx*

1 *du*

*x −* 4

+ *u* = 4*x.*

*dx*

Los procesos f´ısicos del modelo son la difusi´on, la convecci´on y la absorci´on, junto con un t´ermino de forzamiento, en un r´egimen estacionario.

1. Definimos los nodos del mallado: *xi* = (*i* 1)*h*, *i* = 1*, ..., N* + 1*,* y el campo de velocidades en cada uno de ellos:

*−*

1 1

*Vi* = *V* (*xi*) = *xi −* 4 = (*i −* 1)*h −* 4 *.*

Para definir el esquema a contracorriente utilizaremos el par´ametro *ρi* = 1*/*2 + *Vi /*(2*Vi*), de manera que para *i* = 2*, . . . , N* (es decir, para los nodos interiores), se tiene

*| |*

*—* ( *ui−*1 *−* 2*ui* + *ui*+1 + *V* r*ρ* ( *ui − ui−*1 + (1 *− ρ* ) ( *ui*+1 *− ui* l + *u* =

*h*2 *i i h i h i*

4(*i −* 1)*h,*

junto con *u*1 = 1 y *uN*+1 = 2, dados por las condiciones de contorno.

1. Los nodos del mallado son, para *h* = 1*/*4,

*x*1 = 0*, x*2 = 1*/*4*, x*3 = 1*/*2*, x*4 = 3*/*4*, x*5 = 1*.*

El campo de velocidades es:

*V*1 = *−*1*/*4*, V*2 = 0*, V*3 = 1*/*4*, V*4 = 1*/*2*, V*5 = 3*/*4*,*

luego *ρ*3 = *ρ*4 = 1. Utilizando las condiciones de contorno, el sistema lineal asociado al m´etodo es

 1 0 0 0 0

  *u*1 

 1 

*−*16 33 *−*16 0 0  

*u*2

1



  

 0 *−*17 34 *−*16 0  

*u*3

= 2

  

0 0 *−*18 35 *−*16   *u*4   3 

0 0 0 0 1

*u*5

2

cuya soluci´on es *u*1 = 1, *u*2 = 1*.*255, *u*3 = 1*.*526, *u*4 = 1*.*785 y *u*5 = 2.

1. Esta vez utilizamos el algoritmo *bvp2cvrobinup.m* con los comandos:

* a = 0; b = 1;
* h = 0.001; N = ((b-a)/h)-1;
* muf = @(x) 1+0\*x; etaf = @(x) x-1/4; sigmaf = @(x) 1+0\*x;
* f = @(x) 4\*x;
* c11 = 0; c12 = 1; ua = 1;
* c21 = 0; c22 = 1; ub = 2;
* esquema = ’U’;
* [x,u] = bvp2cvrobinup(a,b,N,muf,etaf,sigmaf,f,c11,...
* c12,c21,c22,ua,ub,esquema);
* aprox = u(find(x==0.261))

dando como resultado una aproximaci´on de 1.2678.

**Ejercicio 4.3** *Resolver el siguiente problema de transporte estacionario unidi- mensional:*

*{*Ej.T3.Sol.3*}*

*−u′′*(*x*) + 3*u′*(*x*) + *u*(*x*) = *x* + 1*, x ∈* (0*,* 1)*, u*(0) = 0*, u′*(1) = 0*,*

f

*utilizando un esquema en diferencias finitas centrado y un taman˜o de discreti- zaci´on constante h* = 0*.*5*. Utiliza Octave con un paso de h* = 0*.*05 *para dibujar la aproximaci´on de u*(*x*) *obtenida por el m´etodo y determina los valores x para los que se obtienen aproximaciones mayores a 0.2.*

#### Soluci´on.

Puesto que *x ∈* [0*,* 1] y *h* = 0*.*5, vamos a tener 3 nodos que ser´an

*x*1 = 0*, x*2 = 0*.*5*, x*3 = 1*,*

y tenemos que hallar los valores nodales:

*u*1 *∼ u*(0)*, u*2 *∼ u*(0*.*5)*, u*3 *∼ u*(1)*.*

Por la condici´on de contorno tipo Dirichlet, *u*(0) = 0, tenemos que *u*1 = 0 y por tanto s´olo tenemos que calcular *u*2 y *u*3.

Para calcular *u*2, aplicamos el esquema en diferencias que nos piden en *x*2 y tenemos que:

*—* ( *u*1 *−* 2*u*2 + *u*3 + 3 ( *u*3 *− u*1 + *u*

*h*2

2*h*

= 0*.*5 + 1*.*

2

Teniendo en cuenta que *u*1 = 0 y *h* = 0*.*5, resulta la ecuaci´on,

9*u*2 *− u*3 = 1*.*5*.*

Como en *x*3 tenemos impuesta una condici´on tipo Neumann, tenemos que apli- car el esquema en diferencias en este nodo. Para aplicar el esquema hay que utilizar un nodo ficticio que denotaremos por *xf* con valor nodal *uf* . Entonces, resulta

*—* ( *u*2 *−* 2*u*3 + *uf* + 3 ( *uf − u*2 + *u*

*h*2

2*h*

3

= 1 + 1*.*

Puesto que *u′*(1) = 0, se tiene que su aproximaci´on en diferencias *uf −u*2 tambi´en

2*h*

es cero, *uf −u*2 = 0, y se obtiene

2*h*

1

*−h*2 *u*2

+ 2

*h*2

(

+ 1 *u*3

1

* *h*2 *uf*

= 2*.*

Eliminamos *uf* utilizando la condici´on de contorno *u′*(1) = 0, que implica que

*uf* = *u*2, como hemos visto antes. Por tanto, tenemos que,

2

*−h*2 *u*2

+ 2

*h*2

(

+ 1 *u*3

= 2*,*

de donde se obtiene la ecuaci´on *−*8*u*2 + 9*u*3 = 2. S´olo queda resolver el sistema:

( 9 *−*1 (*u*2 = (1*.*5

*−*8 9

*u*3

2

y obtenemos la soluci´on *u*1 = 0*, u*2 = 0*.*21233 y *u*3 = 0*.*41096.

Resolvemos ahora el problema utilizando Octave tomando *h* = 0*.*05 con los co- mandos:

* a = 0; b = 1;
* h = 0.05; N = (b-a)/h-1;
* muf = @(x) 1+0\*x; etaf = @(x) 3+0\*x; sigmaf = @(x) 1+0\*x;
* f = inline(’x+1’,’x’);
* c11 = 0; c12 = 1; ua = 0;
* c21 = 1; c22 = 0; ub = 0;
* esquema = ’C’;
* [x,u] = bvp2cvrobinup(a,b,N,muf,etaf,sigmaf,...
* f,c11,c12,c21,c22,ua,ub,esquema);
* eje = @(x) 0.2+0\*x;
* figure; plot(x,u,’o-’,x,eje(x))

Las aproximaciones pueden verse en la FIgura [4.2.](#_bookmark55) Para calcular la regi´on pe- dida escribimos:

* vecpos = x(find(u>0.2));
* region = [vecpos(1) vecpos(end)]

que da como soluci´on el intervalo [0*.*55*,* 1].

0.35

0.3

0.25

0.2

0.15

0.1

0.05

0

0 0.2 0.4 0.6 0.8 1

*{*pvc33*}*

Figura 4.2: Gr´afica de las aproximaciones obtenidas (c´ırculos) y la recta *u* = 0*.*2.

## Problemas de Transporte estacionarios en dominios bidimensionales

*{*Ej.T3.Sol.4*}*

**Ejercicio 4.4** *Considera el dominio abierto* Ω *de frontera formada por los lados L*1*, L*2*, L*3 *y L*4*, que se recoge en la Figura* [*4.3.*](#_bookmark57) *Sobre dicho dominio se preten- de resolver mediante un m´etodo en diferencias finitas el problema de contorno siguiente:*

 *−∇ ·* (*∇u*(*x, y*)) + *∇ ·* (*→−***V**(*x, y*)*u*(*x, y*)) + *u*(*x, y*) = 0*, en* Ω*,*

****

 1*−∇u*(*x, y*) + *→−***V**(*x, y*)*u*(*x, y*)l *· →−***n** (*x, y*) = 0*, en L*3 *L*4*,*

**** *u*(*x, y*) = 0*, en L*1 *L*2*.*

*donde →−***V**(*x, y*) *es el campo de velocidades de convecci´on dado por:*

*→−***V**(*x, y*) = (*−y*

*x*

*Se pide escribir las ecuaciones algebraicas a las que conduce el plantear un esquema en diferencias finitas de 5 puntos en cruz para la aproximaci´on del t´ermino difusivo y descentrado contracorriente para la aproximaci´on del t´ermino convectivo, junto a, en su caso, la imposici´on de las correspondientes condiciones de contorno, en los nodos* 9*,* 10 *y* 12 *del mallado dado.*

#### Soluci´on.

En primer lugar operamos en la ecuaci´on:

( )

*−∇ ·* (*∇u*(*x, y*)) + *∇ · →−***V**(*x, y*)*u*(*x, y*) + *u*(*x, y*) = 0*.*

Teniendo en cuenta la definici´on de *→−***V**(*x, y*), obtenemos que:

( *∂*2*u ∂*2*u ∂ ∂*

*−*

*∂x*2 + *∂y*2

+ *∂x* (*−yu*) + *∂y* (*xu*) + *u* = 0*,*

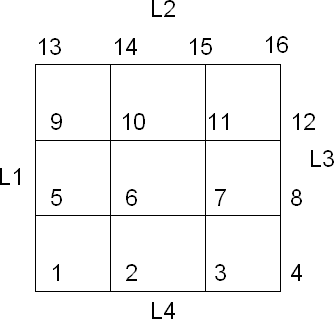


Figura 4.3: Dominio Ω, su frontera *L*1 *L*2 *L*3 *L*4 y correspondiente malla-

do de taman˜o *h* = 1. *{*figura101*}*

( *∂*2*u ∂*2*u ∂u ∂u*

*−*

*∂x*2 + *∂y*2

*— y∂x* + *x∂y* + *u* = 0*.*

Comenzamos hallando la ecuaci´on para el nodo interior 10 de coordenadas (1*,* 2), aplicando el esquema en diferencias que nos dice el enunciado, 5 puntos en cruz para aproximar el operador laplaciano y descentrado contracorriente para el t´ermino convectivo:

*—* ( *u*14 + *u*9 *−* 4*u*10 + *u*6 + *u*11 + (*−*2) ( *u*11 *− u*10 + *u*10 *− u*6 + *u*

12

1

1

= 0*.*

10

La ecuaci´on para el nodo 10 es:

*−*2*u*6 *− u*9 + 8*u*10 *−* 3*u*11 *− u*14 = *−*2*u*6 + 8*u*10 *−* 3*u*11 = 0*,*

puesto que *u*9 = 0 y *u*14 = 0, por la condici´on de contorno.

Para el nodo 12, de coordenadas (3*,* 2), nos damos cuenta de que vamos a ne- cesitar trabajar con un nodo ficticio *A* a su derecha, es decir, localizado en la coordenadas *A* = (4*,* 2). El esquema queda como sigue:

*—* ( *u*16 + *u*11 *−* 4*u*12 + *uA* + *u*8 +(*−*2) ( *uA − u*12 +3 ( *u*12 *− u*8 +*u*

12

1

1

= 0*.*

12

Para eliminar el valor nodal ficticio utilizamos la condici´on de contorno en el nodo 12 y tenemos que como *→−n* 12 = (1*,* 0), entonces:

r*−* ( *∂u, ∂u*

*∂x*

*∂y*

12

*∂x*

+ (*−*2*u*

*,* 3*u*

12

)l *·* (1*,* 0) = 0*,* es decir, ( *∂u*

= *−*2*u .*

12

12

12

Usando la f´ormula en diferencias finitas centradas nos queda que

( *uF − u*12 = *−*2*u .*

1

12

Operando queda:

*uF* = *−u*12*.*

Por tanto, sustituyendo en la ecuaci´on en diferencias que ten´ıamos para el nodo 12, resulta que la ecuaci´on para el nodo 12 es:

*−*4*u*8 *− u*11 + 11*u*12 *− u*16 = *−*4*u*8 *− u*11 + 11*u*12 = 0*,*

puesto que *u*16 = 0, por la condici´on de contorno.

**Ejercicio 4.5** *Sea* Ω *el tri´angulo rect´angulo de v´ertices* (0*,* 0)*,* (4*,* 0) *y* (0*,* 4)*. En*

*´el se considera el mallado que aparece en la Figura* [*4.4*](#_bookmark58) *y el problema siguiente:*

*{*Ej.T3.Sol.5*}*

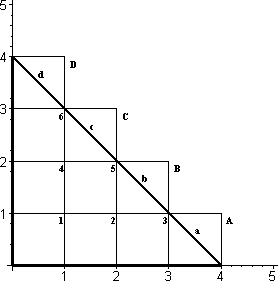


Figura 4.4: Dominio triangular Ω con un mallado de taman˜o *h* = 1. *{*figuraT*}*

**** *−∇ ·* (3*∇u*(*x, y*)) + *∇ ·* (*→−V* (*x, y*)*u*(*x, y*)) + *u*(*x, y*) = *f* (*x, y*)*, en* Ω*,*

 *u*(*x,* 0) = 0*, u*(0*, y*) = 0*,*

(*−*3*∇u*(*x, y*) + *→−V* (*x, y*)*u*(*x, y*)) *· →−n* (*x, y*) = 0*, en L*

****

*donde*

*→−V* (*x, y*) = (*−x* *, f* (*x, y*) = *xy*

*y*

*y L es el lado que une los v´ertices* (4*,* 0) *y* (0*,* 4) *del tri´angulo.*

*Se pide obtener una aproximaci´on del valor que toma la soluci´on en los nodos* 2*,* 3*, y* 6 *del mallado mediante un esquema de diferencias finitas de* 5 *puntos en cruz para el t´ermino difusivo y descentrado contracorriente para el t´ermino convectivo.*

**Soluci´on.** En primer lugar operamos en la ecuaci´on de transporte para obtener:

( *∂*2*u ∂*2*u ∂u ∂u*

*−*3

*∂x*2 + *∂y*2

*— x∂x* + *y ∂y* + *u* = *xy.*

Comenzamos hallando la ecuaci´on para el nodo 6 de coordenadas (2*,* 1), aplican- do el esquema en diferencias que nos dice el enunciado: 5 puntos en cruz para aproximar el operador laplaciano y descentrado contracorriente para el t´ermino convectivo, tomando como paso de discretitaci´on tanto en *x* como en *y*, *h* = 1. Considerando que el valor de *u* en el nodo que est´a debajo del 2 es cero por la condici´on *u*(*x,* 0) = 0, resulta

*−*3 ( *u*5 + 0 *−* 4*u*2 + *u*1 + *u*3 + (*−*2) ( *u*3 *− u*2 + *u*2 *−* 0 + *u*

12

1

1

= 2*,*

2

es decir, la ecuaci´on para el nodo 2 queda:

*−*3*u*1 + 16*u*2 *−* 5*u*3 *−* 3*u*5 = 2*.*

En referencia al nodo 3 de coordenadas (3*,* 1), nos damos cuenta de que vamos a necesitar trabajar con dos nodos ficticios, *A* y *B*. El esquema queda como sigue:

*−*3 ( *u*2 + *uA −* 4*u*3 + 0 + *uB* + (*−*3) ( *uA − u*3 + *u*3 *−* 0 + *u*

12

1

1

= 3*.*

3

N´otese que el valor de *u* en el nodo que est´a debajo del 3, es cero por la condici´on *u*(*x,* 0) = 0. Para eliminar los valores nodales ficticios utilizamos la condici´on de contorno en los puntos *a* = (3*.*5*,* 0*.*5) y *b* = (2*.*5*,* 1*.*5), siendo para ambos la

normal exterior *→−n* = ( *√*1 *, √*1 ). Entonces:

r ( *∂u*

*−*3

*∂x*

2 2

*∂u*

*,*

+ (*−*3*.*5*ua,* 0*.*5*ua*)

*∂y*

*a*

l *→−*

*∂u* 3

*— √*2 *ua* = 0

*∂→−n*

Al aplicar la f´ormula en diferencias descentrada para la derivada normal se tiene

*· n* = 0 =*⇒ −*3

*uA√− ua*

3*ua*

*u*3 + 0

*−*3 2*/*2 *− √*2 = 0*, ua* = 2 *,*

de donde obtenemos:

Por otro lado,

1

*uA* = 4 *u*3*.*

r*−*3

*∂u ∂u*

(

*,*

*∂x ∂y*

+ ( 2*.*5*ub,* 1*.*5*ub*)

*b*

*−*

l

*· →−n* = 0 =*⇒ −*3

*∂u*

*∂→−n*

1

*— √*2 *ub* = 0*.*

Como

*uB − ub*

*ub*

*u*3 + *u*5

queda entonces

*√*2*/*2 = *−* 3*√*2 *, ub* = 2 *,*

5

*uB* = 6 (*u*3 + *u*5)*.*

Por tanto, sustituyendo en la ecuaci´on en diferencias que ten´ıamos para el nodo 3, obtenemos:

57 5

*−*3*u*2 + 4 *u*3 *−* 4 *u*5 = 3*.*

En referencia al nodo 6 de coordenadas (1*,* 3), nos damos cuenta de que vamos a necesitar trabajar con dos nodos ficticios, *C* y *D*. El esquema queda como sigue:

*−*3 ( *uD* + *u*4 + *uA −* 4*u*6 + 0 + *uC* +(*−*1) ( *uC − u*6 +3 ( *u*6 *− u*4 +*u*

12

1

1

= 3*.*

6

Para eliminar los valores nodales ficticios utilizamos la condici´on de contorno en los puntos *c* = (1*.*5*,* 2*.*5) y *d* = (0*.*5*,* 3*.*5), siendo para ambos la normal exterior

*→−n* = ( *√*1 *, √*1 ), entonces:

r*−*3

2 2

*∂u ∂u*

(

*,*

*∂x ∂y*

+ ( 1*.*5*uc,* 2*.*5*uc*)

*c*

*−*

l

*· →−n* = 0 =*⇒ −*3

*∂u*

*∂→−n*

1

+ *√*2 *uc* = 0*.*

Utilizando la f´ormula en diferencias descentradas, se tiene que:

*uC − uc*

*uc*

*u*6 + *u*5

*√*2*/*2 = 3*√*2 *, uc* = 2 *,*

de donde se obtiene:

Por otro lado,

7

*uC* = 12 (*u*6 + *u*5)*.*

r*−*3

*∂u ∂u*

(

*,*

*∂x ∂y*

+ ( 0*.*5*ud,* 3*.*5*ud*)

*d*

*−*

l

*· →−n* = 0 =*⇒ −*3

*∂u*

*∂→−n*

3

+ *√*2 *ud* = 0*.*

Como

*uD√− ud ud*

*u*6 + 0

resulta:

2*/*2 = *√*2 *, ud* = 2 *,*

3

*uD* = 4 *u*6*.*

Finalmente, sustituyendo en la ecuaci´on en diferencias que tendr´ıamos para el nodo 6, resulta que la ecuaci´on para el nodo 6 es:

7

*−*6*u*4 *−* 4 *u*5 +

149

12 *u*6 = 3*.*

**Ejercicio 4.6** *Considera el dominio abierto* Ω *de frontera formada por los lados*

*L*1*, L*2*, L*3 *y L*4 *de la Figura* [*4.5*](#_bookmark59) *sobre el cual se ha efectuado el mallado que*

*{*Ej.T3.Sol.6*}*

*se recoge en la figura, siendo h* = ∆*x* = ∆*y* = 1*. En ´el se pretende resolver mediante un m´etodo en diferencias el problema de contorno siguiente:*

**** *−∇ ·* (2*∇u*(*x, y*)) + *∇ ·* (*→−***V**(*x, y*)*u*(*x, y*)) + *q*(*x, y*)*u*(*x, y*) = 0*, en* Ω*,*

 *u*(*x, y*) = 0*, en L*1 *∪ L*3

1*−*2*∇u*(*x, y*) + *→−***V**(*x, y*)*u*(*x, y*)l *· →−***n** (*x, y*) = 0*, en L*2 *∪ L*4*,*

****

*donde →−***V**(*x, y*) *es el campo de velocidades de convecci´on que se considera dado por*

(

*→−***V**(*x, y*) = *x − y ,*

*y − x*

*siendo adem´as q*(*x, y*) = *y, y →−***n** (*x, y*) *el vector normal unitario exterior en el punto* (*x, y*) *de la frontera de* Ω*.*

*Se pide escribir las ecuaciones algebraicas a las que conduce el plantear un esquema en diferencias finitas de 5 puntos en cruz para la aproximaci´on del t´ermino difusivo y descentrado contracorriente para la aproximaci´on del t´ermino convectivo, junto a, en su caso, la imposici´on de las correspondientes condiciones de contorno (para las que se utilizar´a un esquema centrado cuando se trate de las condiciones de flujo en la frontera), en los nodos* 8*,* 9*,* 10*,* 11 *y* 12 *del mallado anterior.*

1

2

**(2,2)**

**13**

**L3**

**14**

**15 (4,2)**

.5

1

**L4**

**8**

**9**

**10**

**11**

**12 L2**

.5

**1**

0

**2**

1

**3**

2

**L1**

**4**

3

**5**

4

**6**

5

**7**

6

0

*{*Trapecio*}*

Figura 4.5: Dominio trapezoidal Ω con un mallado de taman˜o *h* = 1.

**Soluci´on.** Primero vamos a ver como queda la ecuaci´on

*−∇ ·* (2*∇u*(*x, y*)) + *∇ ·* (*→−***V**(*x, y*)*u*(*x, y*)) + *q*(*x, y*)*u*(*x, y*) = 0*.*

Operamos la parte convectiva para obtener lo siguiente:

*∂ ∂ ∂u ∂u*

*∂x* ((*x − y*)*u*) + *∂y* ((*y − x*)*u*) = (*x − y*) *∂x* + (*y − x*) *∂y* + 2*u,*

y por tanto, tenemos que

*∂u ∂u*

*−*2∆*u* + (*x − y*) *∂x* + (*y − x*) *∂y* + (*y* + 2)*u* = 0*.*

Comenzamos a plantear la ecuaci´on que resulta para el nodo 8 de coordenadas (1*,* 1). Como **V**(1*,* 1) = (0*,* 0), entonces queda:

*−*2∆*u* + 3*u* = 0*.*

Aplicando el esquema en diferencias que nos dice el enunciado necesitamos uti- lizar dos nodos ficticios que llamaremos *F*1 y *F*2, de coordenadas (0*,* 1) y (1*,* 2), respectivamente, de modo que,

*−*2(*uF*2 + 0 *−* 4*u*8 + *u*9 + *uF*1 ) + 3*u*8 = 0*.*

Para eliminar los valores nodales ficticios utilizamos las condici´on de contorno en el lado *L*4. Consideramos los puntos *f*1 = (0*.*5*,* 0*.*5) y *f*2 = (1*.*5*,* 1*.*5) situados en *L*4 y aplcamos la condici´on de contorno en dichos puntos. Siendo el vector velocidad en esos puntos y aplicando la f´ormula en diferencias centradas, como dice el enunciado, tenemos que:

*−*2 ( *∂u*

*∂n*

2

*F*1

*f*1

= 0 =*⇒ −*2 ( *uF*1*√− u*2 = 0 =*⇒ u*

= *u*2

= 0*,*

*−*2 ( *∂u*

*∂n*

*F*2

*f*2

= 0 =*⇒ −*2 ( *uF*2*√− u*9 = 0 =*⇒ u*

= *u*9*.*

Por tanto, la ecuaci´on en el nodo 8 queda:

2

11*u*8 *−* 4*u*9 = 0*.*

A continuaci´on, obtenemos las ecuaciones para los nodos interiores: nodo 9 (2,1), nodo 10 (3,1) y nodo 11 (4,1). Comenzamos por el nodo 9, teniendo en cuenta

que **V**(2*,* 1) = (1*, −*1), obtenemos:

*−*2(*u*13 + *u*3 *−* 4*u*9 + *u*8 + *u*10) + 1(*u*9 *− u*8) + (*−*1)(*u*13 *− u*9) + 3*u*9 = 0*,*

y por las condiciones de contorno, resulta

13*u*9 *−* 2*u*10 = 0*.*

En el nodo 10 se tiene que **V**(3*,* 1) = (2*, −*2) y por tanto,

*−*2(*u*14 + *u*4 *−* 4*u*10 + *u*11 + *u*9) + 2(*u*10 *− u*9) + (*−*2)(*u*14 *− u*10) + 3*u*10 = 0*,*

*−*4*u*9 + 15*u*10 *−* 2*u*11 = 0*.*

En el nodo 11 se tiene que **V**(4*,* 1) = (3*, −*3) y por tanto,

*−*2(*u*15 + *u*5 *−* 4*u*11 + *u*12 + *u*10) + 3(*u*11 *− u*10) + (*−*3)(*u*15 *− u*11) + 3*u*11 = 0*,*

*−*5*u*10 + 17*u*11 *−* 2*u*12 = 0*.*

Finalmente planteamos la ecuaci´on que resulta para el nodo 12 de coordenadas (5*,* 1) con **V**(5*,* 1) = (4*, −*4). Aplicando el esquema en diferencias necesitamos utilizar dos nodos ficticios que llamaremos *F*3 y *F*4, de coordenadas (4*,* 2) y (5*,* 1), respectivamente, de modo que,

*−*2(*uF*3 + *u*6 *−* 4*u*12 + *u*11 + *uF*4 ) + 4(*u*12 *− u*11) + (*−*4)(*uF*3 *− u*12) + 3*u*12 = 0*.*

Para eliminar los valores nodales ficticios utilizamos las condici´on de contorno en el lado *L*2. Consideramos los puntos *f*3 = (4*.*5*,* 1*.*5) y *f*4 = (5*.*5*,* 0*.*5) situados en *L*2 y tenemos que:

*∂u*

(

*−*2 *∂n*

1

+ *√*2 (3*uf*3 *−*3*uf*3 ) = 0 =*⇒ −*2

*f*3

( *uF*3*√− u*11

= 0 =*⇒ uF*3 = *u*11 = 0*,*

*∂u*

2

(

*−*2 *∂n*

1

+ *√*2 (5*uf*4 *−* 5*uf*4 ) = 0 =*⇒ −*2

*f*4

( *uF*4*√− u*5

= 0 =*⇒ uF*4 = *u*5*.*

Por tanto, la ecuaci´on en el nodo 12 queda:

2

*−*12*u*11 + 19*u*12 = 0*.*

Ej.T3.Sol.7

*{ }*

**Ejercicio 4.7** (Octave) *Sea dado el PVC estacionario 2D asociado a la ecua- ci´on de Poisson en el dominio* Ω = (0*,* 1) *×* (0*,* 1)*:*

*−*∆*u* = *f* (*x, y*)*,* (*x, y*) *∈* Ω*,*

*u*(*x, y*) = *g*(*x, y*)*,* (*x, y*) *∈ ∂*Ω*,*

*en donde*

*f* (*x, y*) = 4(cos(*x*2 + *y*2) *−* (*x*2 + *y*2) sin(*x*2 + *y*2))*,*

2 2

*g*(*x, y*) = 2 *−* sin(*x* + *y* )*.*

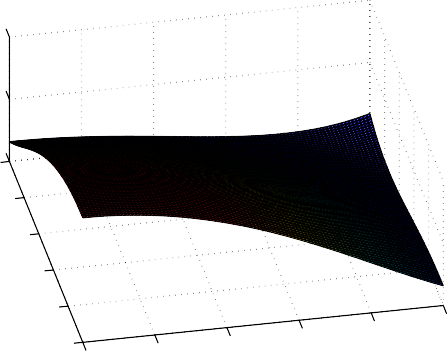
*Aplicar el algoritmo de ecupoisson.m para calcular la soluci´on utilizando como pasos de discretizaci´on* ∆*x* = ∆*y* = 0*.*01*. Dibujar la soluci´on. Sabiendo que la soluci´on exacta es u*(*x, y*) = 2 (*x*2 + *y*2)*, escribir el error cometido en* (*x, y*) = (0*.*24*,* 0*.*76)*.*

*−*

**Soluci´on.** Introducimos los datos:

* a = 0; b = 1; c = 0; d = 1; dx = 0.01; dy = 0.01;
* f = @(x,y) 4.\*(cos(x.^2+y.^2)-(x.^2+y.^2).\*sin(x.^2+y.^2));
* g = @ (x,y) 2-sin(x.^2+y.^2)

N´otese que la definici´on de la soluci´on exacta coincide con la de *g*(*x, y*). Ejecuta- mos *ecupoisson.m* y dibujamos los resultados (aparecen mostrados en la Figura

2

1.5

1

1

0.8

0.6

0.4

0.2

0

0 0.2

0.4 0.6

0.8 1

*{*sol2d\_1*}*

Figura 4.6: Soluci´on num´erica.

[4.6):](#_bookmark60)

* [uh,x,y,er] = ecupoisson(a,c,b,d,dx,dy,f,g,g);
* surf(x,y,uh)

Por u´ltimo hallamos el error cometido en (0*.*24*,* 0*.*76):

* x1 = find(x==0.24); y1 = find(y==0.76); u1 = uh(x1,y1);
* e = u1-g(0.24,0.76)

que da como resultado 1.1333*·*10*−*5.

## Problemas evolutivos en una dimensio´n espacial

**Ejercicio 4.8** *Consideramos el problema evolutivo unidimensional de convec- ci´on siguiente:*

**** 2 *∂u* (*x, t*) *− ∂u* (*x, t*) = 2*u*(*x, t*)*,* 0 *< x ≤* 1*,* 0 *< t ≤* 1*,*

*{*Ej.T3.Sol.8*}*

 *∂t*

****

*∂x*

*u*(0*, t*) = *t,* 0 *< t ≤* 1*, u*(*x,* 0) = (0*.*5 + *x*)(1*.*5 *− x*)*,* 0 *≤ x ≤* 1*.*

*Se pide:*

1. *Construir un esquema en diferencias finitas a partir de una f´ormula pro- gresiva de orden 1 para aproximar la derivada temporal y de una f´ormula regresiva de orden 1 para aproximar la derivada espacial.*
2. *Resolver el problema tomando como paso de discretizaci´on espacial y paso de discretizaci´on temporal* ∆*x* = 0*.*5 *y* ∆*t* = 0*.*5*. ¿Qu´e aproximaci´on se obtiene en el instante t* = 1*?*

**Soluci´on.** *(a)* Primero discretizamos en la variable *t*, y al aplicar la f´ormula progresiva se tiene que

*un*+1(*x*) *un*(*x*)

(

*−*

2

∆*t*

*∂un*(*x*)

=

*∂x*

+ 2*un*

(*x*)*,*

donde *un*(*x*) *∼ u*(*x, n*∆*t*) para *n ≥* 0, que son los pasos en *t* que tenemos que dar para cubrir el intervalo *t ∈* [0*,* 1] y que est´an asociados a los nodos *x*0 = 0*, . . . , xN*+1 = 1. A continuaci´on, discretizamos respecto de la variable *x*, utilizando una f´ormula progresiva para aproximar la derivada,

( *un*+1 *− un*

*i i*

2

*n n*

= *i i−* + 2*un,*

*u − u* 1

∆*t* ∆*x i*

donde *un*(*x*) *u*((*i* 1)∆*x, n*∆*t*), para *i* = 1*, . . . , N* + 1 y *n* 0.

*i*

*∼ − ≥*

En conclusi´on, la f´ormula que tenemos que utilizar para resolver el problema es:

*un*+1 = *un* + 0*.*5 ∆*t* (*un − un*

*i*

*i*

∆*x*

*i*

*i−*1

) + ∆*tun,*

*i*

para *i* = 1*, . . . , N* + 1 y *n ≥* 0.

*(b)* Tomando ∆*x* = 0*.*5 y ∆*t* = 0*.*5 los nodos del mallado son *x*1 = 0, *x*2 = 0*.*5 y *x*3 = 1.

Por la condici´on inicial *u*(*x,* 0) = (0*.*5 + *x*)(1*.*5 *x*), que aparece como dato, se tiene que:

*−*

*u*0 = 0*.*75*, u*0 = 1*, u*0 = 0*.*75*,*

1 2 3

y por la condici´on de contorno *u*(0*, t*) = *t*, se tiene que:

*u*1 = 0*.*5*, u*2 = 1*.*

1 1

El resto de valores se obtienen siguiendo el esquema en diferencias obteniendo las aproximaciones:

*u*1 = (0*.*5*,* 1*.*625*,* 1), en *t* = 0*.*5*,*

*u*2 = (1*,* 3*,* 1*.*1875), en *t* = 1*.*

*{*Ej.T3.Sol.9*}* **Ejercicio 4.9** *Sea dado el Problema de Valor Inicial y de Contorno:*

 *∂u ∂*2*u*

*∂t* = *D∂x*2 + *f* (*x, t*)*, x ∈* (0*, L*)*, t >* 0*,*

****

****

(*PV IC*) = *u*(0*, t*) = *uI* (*t*)*, t >* 0*,*

*u*(*L, t*) = *uD*(*t*)*, t >* 0*,*

****

 *u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*, L*)*,*

*siendo L* = 1*, D* = 1*, uI* (*t*) = *uD*(*t*) = 0*, t >* 0 *(condiciones de contorno de tipo Dirichlet homog´eneas) y dato inicial:*

*∀*

*u*0(*x*) =

2*x x ≤* 1*/*2*,*

2(1 *− x*) *x >* 1*/*2*.*

f

*El t´ermino de forzamiento es f* (*x, t*) = *x* + *t. Considerando una discretizaci´on espacial* ∆*x* = *L/N y un paso temporal* ∆*t* = (1*/*2)(*L*2*/N* 2)*, dividimos el in- tervalo I* = [0*, L*] *en N subintervalos de la forma* [*xi, xi*+1]*, xi* = (*i* 1)∆*x, i* = 1*, ..., N .*

*−*

*Tomando N* = 4*, se pide determinar la expresi´on de la soluci´on del problema en la forma:*

*Au*¯*n*+1 = *Bu*¯*n* + ¯*bn*+1*,*

*calculando expl´ıcitamente las matrices A, B y el vector* ¯*bn*+1 *al que conduce la aplicaci´on de un esquema en diferencias finitas centradas en el espacio y para la parte temporal un m´etodo de tipo:*

* 1. *Euler expl´ıcito,*
  2. *Euler impl´ıcito,*
  3. *Crank-Nicolson.*

*En cada uno de los tres casos, calcular la aproximaci´on de la soluci´on en el instante T* = ∆*t, es decir, calcular u*¯1 *∼ u*(*xi,* ∆*t*)*.*

#### Soluci´on.

Empezaremos calculando el mallado sugerido. Al ser *N* = 4 (nu´mero de sub- intervalos), se tiene ∆*x* = *L/N* = 1*/*4, ∆*t* = (1*/*2)(*L*2*/N* 2) = 1*/*32, lo que asegura estabilidad del m´etodo al cumplirse la condici´on *α* = ∆*t/*(∆*x*)2 = 1*/*2. Los *N* + 1 = 5 nodos del mallado son *xi* = (*i −* 1)∆*x* = (*i −* 1)*/*4, *i* = 1*, . . . ,* 5, siendo *x*1 = 0, *x*5 = 1, los 2 nodos frontera y *x*2, *x*3, *x*4, los *N* 1 = 3, nodos interiores del mallado.

*−*

1. Al aplicar los esquemas de Euler expl´ıcito se tiene:

*un*+1 *− un*

*i i*

∆*t*

= *D*

( *un*

*—* 2*un* + *un*

*i*

siendo *fn*+1 el t´ermino de forzamiento actuando s´olo en los nodos interiores (*i* = 2*,* 3*,* 4) en la forma:

*i*

*i−*1 *i i*+1

(∆*x*)2

+ *fn*+1*,*

*fn*+1 = *f* (*xi, tn*+1) = *xi* + *tn*+1 = (*i −* 1)∆*x* + (*n* + 1)∆*t.*

*i*

Operando obtenemos:

*i*

*i*+1

*i*

*un*+1 = *αDun*

*i*

*i−*1

+ (1 *−* 2*αD*)*un* + *αDun*

+ ∆*tfn*+1, para *i* = 2*,* 3*,* 4*.*

Al sustituir los valores de *α* = 0*.*5 y *D* = 1, la soluci´on general (para todo tiempo) en la forma *Au*¯*n*+1 = *Bu*¯*n* + ¯*bn*+1 ser´a:

*u* *n*+1

1

 0 0 0 0 0  *u* *n*  0 

*u*2 *u*3

 

*u*4

*u*5

1*/*2 0 1*/*2 0 0 *u*2

= 0 1*/*2 0 1*/*2 0 *u*3

1

   

 

0 0 1*/*2 0 1*/*2 *u*4

0 0 0 0 0

*u*5

∆*t*(∆*x* + (*n* + 1)∆*t*)

+ ∆*t*(2∆*x* + (*n* + 1)∆*t*)

∆*t*(3∆*x* + (*n* + 1)∆*t*)

0

de donde *A* = *Id* al ser esquema expl´ıto. Observar que la primera y u´ltima filas de la matriz *B* son nulas para poder imponer las condiciones de contorno a trav´es del vector ¯*bn*+1.

Evaluando el dato inicial en los nodos del mallado se tiene:

*u*¯0 = (*u*1*, u*2*, u*3*, u*4*, u*5)0 = (0*,* 1*/*2*,* 1*,* 1*/*2*,* 0)*,*

y utilizando la f´ormula para *n* = 0, obtenemos que la aproximaci´on pedida es:

*u* 1

1

*u*2

 

*u*3

 

 

 0 0 0 0 0   0 0  0 

 0 

1*/*2



1*/*2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1*/*2 | 0 | 0 |
|  | 1*/*2 | 0 | 1*/*2 | 0 |

  

0*.*0088

0*.*5088

1 + 0*.*0166 = 0*.*5166

 



      

= 0

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 0 | 1*/*2 | 0 |
| 0 | 0 | 0 | 0 |

*u*4 

*u*5

1*/*2 1*/*2

0

0

0*.*0244

0

0*.*5244

0

1. Al aplicar el esquemas de Euler impl´ıcito se tiene:

*n*+1 *n*

*u − u*

*i i* = *D*

∆*t*

Operando obtenemos:

*n*+1 *n*+1 *n*+1 *i−*1 *i i*+1

(∆*x*)2

*u −* 2*u* + *u*

+ *fn*+1*.*

*n*+1

*i*

*n*+1

*n*+1 *n*

*n*+1

*−αDui−*1 + (1 + 2*αD*)*ui − αDui*+1 = *ui* + ∆*tfi ,*

para *i* = 2*,* 3*,* 4 y *n ≥* 0. Tenemos, por tanto, en este caso las matrices

 1 0 0 0 0  0 0 0 0 0

*−αD* 1 + 2*αD −αD* 0 0  0 1 0 0 0

 

*A* = 0 *−αD* 1 + 2*αD −αD* 0



  



*− −*

 

0 0 *αD* 1 + 2*αD αD*

0 0 0 0 1

*, B* = 0 0 1 0 0

0 0 0 1 0

0 0 0 0 0

y el vector ¯*bn*+1 definido por (usando las condiciones de contorno Dirichlet homog´eneas)

 *uI* (*tn*+1)   0 

 ∆*tfn*+1  ∆*tfn*+1

¯  2   2 

*bn*+1 =  ∆*tfn*+1  = ∆*tfn*+1

3   3 

 ∆*tf*

 ∆*tf*

*n*+1

4

*uD*(*tn*+1)

*n*+1

4

0

Al sustituir los valores *α* = 1*/*2, *D* = 1, se tiene que la soluci´on general (para todo tiempo) ser´a:

 1 0 0 0 0

 *u* *n*+1

 0 *n*  0 

*−*1*/*2 2 *−*1*/*2 0 0

1

 

 *u*2

*u*2

∆*tfn*+1

 0 *−*1*/*2 2 *−*1*/*2 0





 ∆*tf*

2

0 0 *−*1*/*2 2 *−*1*/*2 *u* 

3



 



0 0 0 0 1

 *u*3

4

*u*5

= *u*3

4

0

+∆*tfn*+1

*n*+1

4

0

Resolvemos el sistema mediante inversi´on de la matriz *A*, quedando en la forma:











*u* 





*u* *n*+1

1

 

*u*2

*u*3

 

1 0 0 0 0 0

0*.*2679 0*.*5357 0*.*1429 0*.*0357 0*.*0179 *un* + ∆*tfn*+1

   

2

2

   

= 0*.*0714 0*.*1429 0*.*5714 0*.*1429 0*.*0714 *un* + ∆*tfn*+1

   

3

3

4

4

*u*4

*u*5

0*.*0179 0*.*0357 0*.*1429 0*.*5357 0*.*2679 *un* + ∆*tfn*+1

0 0 0 0 1

0

Para obtener la aproximaci´on pedida, tomamos *n* = 0, obteniendo:

*u* 1

1

*u*2

*u*3

 

 

 1 0 0 0 0

  0 

 0 

0*.*2679 0*.*5357 0*.*1429 0*.*0357 0*.*0179 0*.*5088 4365

   

 

= 0*.*0714 0*.*1429 0*.*5714 0*.*1429 0*.*0714 1*.*0166 = 0*.*7285 *,*

     

*u*4

*u*5

0*.*0179 0*.*0357 0*.*1429 0*.*5357 0*.*2679 0*.*5244

0 0 0 0 1

0

0*.*4443

0

que representa una aproximaci´on de la soluci´on exacta *u*(*x,* ∆*t*).

1. Al aplicar el esquema indicado se tiene:

*un*+1 *− un*

*i i*

∆*t*

=

*D* ( *un*

2

* 2*un* + *un*

*un*+1 *−* 2*un*+1 + *un*+1

*i*

para *i* = 2*,* 3*,* 4 y *n ≥* 0. Operando, tenemos que:

*i−*1 *i i*+1

(∆*x*)2

+

*i−*1 *i i*+1

(∆*x*)2

+*fn*+1*,*

1 *n*+1

*n*+1 1

*n*+1 1 *n*

*n* 1 *n*

*n*+1

*−* 2 *αDui−*1 +(1+*αD*)*ui −* 2 *αDui*+1 = 2 *αDui−*1+(1*−αD*)*ui* + 2 *αDui*+1+∆*tfi ,*

de donde se obtienen las matrices

 1 0 0 0 0 

*−αD/*2 1 + *αD −αD/*2 0 0

 

*A* = 0 *−αD/*2 1 + *αD −αD/*2 0



*,*



 

*— −*

0 0 *αD/*2 1 + *αD αD/*2

0 0 0 0 1

 0 0 0 0 0 

*αD/*2 1 *− αD αD/*2 0 0 

*B* = 0 *αD/*2 1 *− αD αD/*2 0



*,*



 

*−*

0 0 *αD/*2 1 *αD αD/*2

0 0 0 0 0

 *uI* (*tn*+1)   0 

 ∆*tfn*+1  ∆*tfn*+1

¯  2   2 

*bn*+1 =  ∆*tfn*+1  = ∆*tfn*+1 *.*

 ∆*tf*

 ∆*tf*

3





3



*n*+1

4

*uD*(*tn*+1)

*n*+1

4

0

La soluci´on general (para todo tiempo) ser´a por tanto:

 1 0 0 0 0

 *u* *n*+1

*−*1*/*4 3*/*2 *−*1*/*4 0 0

1

*u*2

  

 0 *−*1*/*4 3*/*2 *−*1*/*4 0

0 0 *−*1*/*4 3*/*2 *−*1*/*4 *u*4

  

0 0 0 0 1 *u*5

*u*3

=

 0 0 0 0 0  *u* *n*  0 

1

1*/*4 1*/*2 1*/*4 0 0  *u*2

 





2



∆*tfn*+1

=  0 1*/*4 1*/*2 1*/*4 0  *u*3

0 0 1*/*4 1*/*2 1*/*4 *u* 



3





 

4

∆*tfn*+1

0 0 0 0 0

*u*5

+ ∆*tfn*+1

4

0

Al igual que en el apartado anterior, la resoluci´on de este sistema necesita la inversi´on de la matriz *A*. En el caso de la aproximaci´on pedida, tomamos *n* = 0 y se obtiene:

*u*¯1 = (*u*1*, u*2*, u*3*, u*4*, u*5)1 = (0*,* 4496*,* 0*.*6627*,* 0*.*46*,* 0)*,*

que representa una aproximaci´on de la soluci´on exacta *u*(*x,* ∆*t*).

Ej.T3.Sol.10

*{ }*

**Ejercicio 4.10** *Sea dado el Problema de Valor Inicial y de Contorno (PVIC):*

 *∂u ∂*2*u ∂u*

*∂t* = *D∂x*2 *− V ∂x* + *qu* + *f* (*x, t*) *x ∈* (0*, L*)*, t >* 0

****

****

*u*(0*, t*) = *uI* (*t*)*, t >* 0

*u*(*L, t*) = *uD*(*t*) *t >* 0

**** *u*(*x,* 0) = *u*0(*x*) *x ∈* (0*, L*)*,*





*siendo L* = 1*, D* = 1*, V* = 2*, q* = 4 *y suponiendo un forzamiento nulo, es decir f* (*x, t*) 0*. Tenemos por tanto, procesos: difusivo, convectivo y reactivo. Obs´ervese el signo del campo de velocidades V en la EDP del transporte. La for- ma V es coherente con la definici´on, usada a menudo en mec´anica de fluidos, de derivada material:*

*≡*

*−*

*−*

*Du ∂u ∂u*

= + *V ,*

*Dt ∂t ∂x*

*a trav´es de la cual la EDP se escribe:*

*Du ∂*2*u*

*Dt* = *D∂x*2 + *qu.*

*La EDP se complementa con condiciones de contorno de tipo Dirichlet no ho- mog´eneas de tipo transitorio (dependientes del tiempo):*

*uI* (*t*) = 2*t, ∀ t >* 0*, uD*(*t*) = 1*, ∀ t >* 0*,*

*y el dato inicial:*

*u*0(*x*) =

1*, x ≤* 1*/*2*,*

2(1 *− x*)*, x >* 1*/*2*,*

f

*Considerando una discretizaci´on espacial* ∆*x* = *L/N y un paso temporal* ∆*t* = (1*/*2)(*L*2*/N* 2) *dividimos el intervalo I* = [0*, L*] *en N subintervalos de la forma* [*xi, xi*+1]*, xi* = (*i −* 1)∆*x, i* = 1*, ..., N .*

*Tomando N* = 4*, se pide determinar la expresi´on de la soluci´on del problema en la forma*

*Au*¯*n*+1 = *Bu*¯*n* + ¯*bn*+1*,*

*calculando expl´ıcitamente las matrices A, B y el vector* ¯*bn*+1 *al que conduce la aplicaci´on de un esquema en diferencias finitas centradas en el espacio y para la parte temporal un m´etodo de tipo:*

1. *Euler expl´ıcito,*
2. *Euler impl´ıcito,*
3. *Crank-Nicolson.*

*En cada uno de los tres casos, calcular la aproximaci´on de la soluci´on en el instante T* = ∆*t, es decir, calcular u*¯1 *∼ u*(*xi,* ∆*t*)*.*

**Soluci´on.** Empezaremos calculando el mallado sugerido. Al ser *N* = 4 (nu´mero de sub-intervalos) se tiene ∆*x* = *L/N* = 1*/*4, ∆*t* = (1*/*2)(*L*2*/N* 2) = 1*/*32, lo que asegura estabilidad del m´etodo al cumplirse la condici´on:

∆*t* 1

*α* = *D* (∆*x*)2 = 2 *.*

Los *N* + 1 = 5 nodos del mallado son: *xi* = (*i −* 1)∆*x* = (*i −* 1)*/*4, *i* = 1*, . . . ,* 5 siendo *x*1 = 0, *x*5 = 1, los 2 nodos frontera y *x*2, *x*3, *x*4, los *N* 1 = 3, nodos interiores.

*−*

1. Al aplicar los esquemas indicados se tiene:

*i−*1

*i*

(∆*x*)2

*i*+1

*— V*

*i*+1

∆*x*

*i*

+ *qun,*

*un*+1 *− un*

*i*

*i*

= *D*

∆*t*

( *un*

* 2*un* + *un*

( *un*

* *un*

*i*

siendo *L* = 1, *D* = 1, *V* = 2, *q* = 4, *f* (*x, t*) 0 y los par´ametros de discretizaci´on ∆*x* = 1*/*4, ∆*t* = 1*/*32.

*— ≡*

La velocidad es negativa, *V* = 2, luego utilizaremos una f´ormula progre- siva para ir contracorriente con el t´ermino convectivo. Definimos:

*−*

∆*t* 1 ∆*t* 1

*α* = (∆*x*)2 = 2 *β* = ∆*x* = 8 *.*

Operando, obetemos:

*un*+1 = *αDun*

+ (1 *−* 2*αD* + *βV* + *q*∆*t*)*un* + (*αD − βV* )*un ,*

*i i−*1

es decir,

*un*+1 = *αun*

*i i*+1

+ (1 *−* 2*α −* 2*β* + 4∆*t*)*un* + (*α* + 2*β*)*un .*

*i*

Definimos:

*i−*1 *i*

*un*+1 = *λi−*1*un* + *λiun* + *λi*+1*un .*

*i*+1

siendo

*i i−*1 *i*

*i*+1

*λi−*1 = *αD, λi* = 1 *−* 2*αD* + *βV* + *q*∆*t, λi*+1 = *αD − βV,*

para *i* = 2*, . . . , N* (nodos interiores). N´otese que *A* = *Id* (la matriz identi- dad al ser esquema expl´ıto) y *B* es una matriz tridiagonal:

*λ*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |  |
| *i−*1 | *λi* | *λi*+1 | 0 | 0 |
| 0 | *λi−*1 | *λi* | *λi*+1 |  |
| 0 | 0 | *λi−*1 | *λi* | *λi*+1 |
| 0 | 0 | 0 | 0 | 0 |



*B* =

0 





0

Obs´ervese que la primera y u´ltima filas de la matriz *B* son nulas para poder imponer las condiciones de contorno a trav´es del vector ¯*bn*+1, dado por:

 *uI* (*tn*+1)  2(*n* + 1)∆*t*

0

0

¯    

*bn*+1 =



=

0

0

  

 0   0 

*uD*(*tn*+1)

1

Al sustituir los valores *α* = 1*/*2, *β* = 1*/*8, *D* = 1, *V* = *−*2, *q* = 4, se tiene

*λi−*1 = 1*/*2*, λi* = *−*1*/*8*, λi*+1 = 3*/*4*.*

 0 0 0 0 0 

 

1*/*2 *−*1*/*8 3*/*4 0 0 

*B* =  0 1*/*2 *−*1*/*8 3*/*4 0 

0 0 0 0 0

0 0 1*/*2 *−*1*/*8 3*/*4

La soluci´on general (para todo tiempo) ser´a:

*u* *n*+1

1

 0 0 0 0 0  *u* *n*

(*n* + 1)*/*16

*u*2 *u*3

 

*u*4

*u*5

1*/*2 *−*1*/*8 3*/*4 0 0 *u*2

= 0 1*/*2 *−*1*/*8 3*/*4 0 *u*3

1

   

   

0 0 1*/*2 *−*1*/*8 3*/*4 *u*4

0 0 0 0 0

*u*5

0

+ 0 

0

1

Evaluando el dato inicial en los *N* + 1 nodos del mallado, se tiene: (*u*1*, u*2*, u*3*, u*4*, u*5)0 = (1*,* 1*,* 1*,* 1*/*2*,* 0)*.*

Sea *n* = 0, entonces:

*u* 1

1

 0 0 0 0 0   1 

1*/*16

  1*/*2 *−*1*/*8 3*/*4 0 0   1   

*u*2

0

  =  0 1*/*2 *−*1*/*8 3*/*4 0   1  +  

*u*3

0

*u*4

 0 0 1*/*2 *−*1*/*8 3*/*4 1*/*2  0 

*u*5

luego

0 0 0 0 0 0 1

(*u*1*, u*2*, u*3*, u*4*, u*5)1 = (0*.*0625*,* 1*.*1250*,* 0*.*75*,* 0*.*4375*,* 1)*,*

que representa una aproximaci´on de la soluci´on exacta *u*(*xi,* ∆*t*) en los nodos del mallado.

1. Al aplicar los esquemas indicados se tiene:

*i−*1

*i*+1

*i*+1

*i*

*un*+1 *− un*

*i*

*i*

= *D*

∆*t*

*un*+1 *−* 2*un*+1 + *un*+1

*un*+1 *− un*+1

Definimos:

*i*

*i*

(∆*x*)2

*— V*

∆*x*

+ *qun*+1*.*

∆*t* ∆*t*

Operando, obtenemos

*α* = (∆*x*)2 *β* = ∆*x.*

*n*+1

*n*+1

*n*+1 *n*

*−αDui−*1 + (1 + 2*αD − βV − q*∆*t*)*ui −* (*αD − βV* )*ui*+1 = *ui ,*

es decir,

*λi−*1*un*+1 + *λiun*+1 + *λi*+1*un*+1 = *un,*

siendo

*i−*1 *i*

*i*+1 *i*

*λi−*1 = *−αD, λi* = 1 + 2*αD − βV − q*∆*t, λi*+1 = *−*(*αD − βV* )*,* para *i* = 2*, . . . , N* (nodos interiores), de donde:

0 0 0 0 0

 

*λi−*1 *λi λi*+1 0 0





0 0 0 0 0

0 1 0 0 0

 

 

*A* = 0 *λi−*1 *λi λi*+1 0



  



0 0 *λi−*1 *λi λi*+1

0 0 0 0 0

*, B* = 0 0 1 0 0

0 0 0 1 0

0 0 0 0 0

siendo

 *uI* (*tn*+1)  2(*n* + 1)∆*t*

¯    

0

0

*bn*+1 =



=

*.*

0

0

  

 0   0 

*uD*(*tn*+1)

1

*−*

Al sustituir los valores *α* = 1*/*2, *β* = 1*/*8, *D* = 1, *V* = 2, *q* = 4, tenemos que:

*λi−*1 = *−*1*/*2*, λi* = 17*/*8*, λi*+1 = *−*3*/*4*,*

 1 0 0 0 0 

*−*1*/*2 17*/*8 *−*3*/*4 0 0 

*A* = 0 1*/*2 17*/*8 3*/*4 0



*— −*

*.*





0 0 1*/*2 17*/*8 3*/*4

*— −*

0 0 0 0 1

La soluci´on general (para todo tiempo) ser´a:

 1 0 0 0 0

1

*u*2

 *u* *n*+1

*−*1*/*2 17*/*8 *−*3*/*4 0 0

  

*Au*¯*n*+1 =



0 *−*1*/*2 17*/*8 *−*3*/*4 0

  

 0 0 *−*1*/*2 17*/*8 *−*3*/*4 *u*4

*u*3

=

0 0 0 0 1

*u*5

 0 *n*

2(*n* + 1)∆*t*

2(*n* + 1)∆*t*

*u*2  0   *un* 

=

*u*

*.*

*n*

3



*n* ¯*n*+1   

 

+

*u*

= *Bu*¯

+ *b*

= *u*3



0

 

*u*4









0

  2 

0

1

*n*

4

1

Resolviendo el sistema mediante inversi´on de la matriz *A* se tiene:

*u* *n*+1

1

 1 0 0 0 0

 2(*n* + 1)∆*t*

*u*2 0*.*2587 0*.*5*i*75 0*.*1992 0*.*0703 0*.*0527

 

  

*u*

2

*u*3 = 0*.*0664 0*.*1328 0*.*5643 0*.*01992 0*.*1494

*u*

*.*

*u*4 0*.*0156 0*.*0312 0*.*1328 0*.*5175 0*.*3881

 

 

*u*5 0 0 0 0 1

*n*

*n*

3 

*u*

*n*

4

1

Evaluando el dato inicial, obtenemos:

(*u*1*, u*2*, u*3*, u*4*, u*5)0 = (1*,* 1*,* 1*,* 1*/*2*,* 0)*,*

con condiciones de contorno:

*uI* (*t*1) = 2∆*t* = 1*/*16*, uD*(*t*1) = *uD*(∆*t*) = 1*.*

Para *n* = 0, resulta:

*u* 1

1

*u*2

 

*u*3

1 0 0 0 0 1*/*16 0

0*.*2587 0*.*5*i*75 0*.*1992 0*.*0703 0*.*0527 1

  

   

= 0*.*0664 0*.*1328 0*.*5643 0*.*01992 0*.*1494 1

*,*

   

*u*4

*u*5

0*.*0156 0*.*0312 0*.*1328 0*.*5175 0*.*3881  1*/*2 

0 0 0 0 1

1

luego

(*u*1*, u*2*, u*3*, u*4*, u*5)1 = (0*.*0625*,* 0*.*8207*,* 0*.*9502*,* 0*.*8118*,* 1)*,*

que representa una aproximaci´on de la soluci´on exacta *u*(*xi,* ∆*t*) en los nodos del mallado.

1. Al aplicar los esquemas indicados se tiene:

*i−*1

*i*+1

*i*

*un*+1 *− un*

*i*

*i*

=

∆*t*

1 r ( *un*

2

* 2*un* + *un*

( *un*

* *un* l

*i*

1 *un*+1 *−* 2*un*+1 + *un*+1 *un*+1 *− un*+1

2

*i*

*D*

*i*

(∆*x*)2

*— V*

*i*+1

∆*x*

+ *qun*

+

+

*D*

*i−*1

*i*

(∆*x*)2

*i*+1

*— V*

*i*+1

∆*x*

*i*

+ *qun*+1

*.*

Operando, obtenemos

1 *n*+1

*n*+1 1

*n*+1

*−* 2 *αDui−*1 + [1 + *αD −* (*βV/*2) *−* (*q/*2)∆*t*)*ui −* 2 (*αD − βV* ]*ui*+1 =

= 1 *αDun*

+ [1 *− αD −*

*n* 1 *n ,*

2

es decir,

*i−*1

(*βV/*2) + (*q/*2)∆*t*]*ui* + 2 (*αD* + *βV* )*ui*+1

*λi−*1*un*+1 + *λiun*+1 + *λi*+1*un*+1 = *δi−*1*un*

+ *δiun* + *δi*+1*un ,*

*i−*1 *i*

siendo las matrices

*i*+1

*i−*1 *i*

*i*+1

*A* = 





1 0 0 0 0

*λi−*1 *λi λi*+1 0 0

0 *λi−*1 *λi λi*+1 0

0 0 *λi−*1 *λi λi*+1

0 0 0 0 1

 *,*



*δ* 







|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 |
| *i−*1  0  0 | *δi*  *δi−*1  0 | *δi*+1 *δi*  *δi−*1 | 0  *δi*+1 *δi* | 0  0  *δi*+1 |
| 0 | 0 | 0 | 0 | 0 |



*B* =







definidas por los coeficientes:

1 1

*λi−*1 = *−* 2 *αD, λi* = 1 + *αD −* (*βV/*2) *−* (*q/*2)∆*t, λi*+1 = *−* 2 (*αD − βV* )*,*

1 1

*δi−*1 = 2 *αD, δi* = 1 *− αD* + (*βV/*2) + (*q/*2)∆*t, δi*+1 = 2 (*αD − βV* )*,*

para *i* = 2*, . . . , N* (nodos interiores). Al sustituir los valores:

*α* = 1*/*2*, β* = 1*/*8*, D* = 1*, V* = *−*2*, q* = 4*,*

se tiene:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 0 | 0 |
| 1*/*4 | 25*/*16 | *−*3*/*8 | 0 | 0 |

*A* = *−*



0 *−*1*/*4 25*/*16 *−*3*/*8 0 





*— −*

0 0 1*/*4 25*/*16 3*/*8

0 0 0 0 1

y

 

0 0 0 0 0

1*/*4 7*/*16 3*/*8 0 0

*B* = 0 1*/*4 7*/*16 3*/*8 0 0 0 1*/*4 7*/*16 3*/*8

 

0 0 0 0 0

La soluci´on general (para todo tiempo) ser´a:

1

 1 0 0 0 0

 *u* *n*+1

*−*1*/*4 25*/*16 *−*3*/*8 0 0

  

*Au*¯*n*+1 =



0 *−*1*/*4 25*/*16 *−*3*/*8 0

  

 0 0 *−*1*/*4 25*/*16 *−*3*/*8 *u*4

*u*2

0 0 0 0 1 *u*5

*u*3

=

 0 0 0 0 0  *u* *n* 2(*n* + 1)∆*t*

1

1*/*4 7*/*16 3*/*8 0 0 *u*2

   

0



¯ 

= *Bu*¯*n*+*bn*+1 = 0 1*/*4 7*/*16 3*/*8 0 *u*3 0

   

+





 0 0 1*/*4 7*/*16 3*/*8 *u*4  0 

0 0 0 0 0 *u*5 1

Evaluando el dato inicial se tiene:

(*u*1*, u*2*, u*3*, u*4*, u*5)0 = (1*,* 1*,* 1*,* 1*/*2*,* 0)*.*

Sea *n* = 0. Entonces,

 0 0 0 0 0  *u* 0 2∆*t*

1

1*/*4 7*/*16 3*/*8 0 0 *u*2

  

0

 

¯ 

*Bu*¯0 + *b*1 = 0 1*/*4 7*/*16 3*/*8 0 *u*3

   

+

=

0

0 0 1*/*4 7*/*16 3*/*8 *u*4

   

0 0 0 0 0 *u*5

 0 0 0 0 0   1  2∆*t*

0

 0 

1

 2∆*t* 

 

=

1

1*/*4 7*/*16 3*/*8 0 0 1

  

 



0

1*.*0625

= 0 1*/*4 7*/*16 3*/*8 0 1 +

    



  

0 0 1*/*4 7*/*16 3*/*8 1*/*2  0  0*.*5938

0 0 0 0 0

0

1

1

y se tiene, resolviendo el sistema *Au*¯1 = *Bu*¯0 + ¯*b*1 mediante inversi´on de

la matriz *A*

*u* 1 

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 1 0 | 0 | 0 |
|  | 0*.*1667 0*.*6666 | 0*.*1664 | 0*.*0399 |

0  0*.*0625

  

*u*2

*u*3

0*.*0150 1*.*0625

= 0*.*0277 0*.*1109 0*.*6932 0*.*1664 0*.*0624

  

  

 

1



*u*4

*u*5

0*.*0044 0*.*0177 0*.*1109 0*.*666 0*.*25  0*.*5938

0 0 0 0 1

1

luego

(*u*1*, u*2*, u*3*, u*4*, u*5)1 = (0*.*0625*,* 0*.*8980*,* 0*.*8666*,* 0*.*6786*,* 1)*,*

que representa una aproximaci´on de la soluci´on exacta *u*(*xi,* ∆*t*) en los nodos del mallado.

**Ejercicio 4.11** (Octave) *Sea dado el PVIC:*

*{*Ej.T3.Sol.11*}*

****

*∂u ∂*2*u*

*∂t* = *∂x*2

+ *f* (*x, t*)*,*

*donde*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = 1*,*

**** *u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*,* 1)*,*

*f* (*x, t*) = 2*t − x*2 + 10*xt, g*(*x, t*) = *x*2(*x −* 1)*, u*0(*x*) = 0*.*

*Aplicar el algoritmo ecucalor.m para calcular la soluci´on en el intervalo tempo- ral [0,1] con paso de discretizaci´on espacial* ∆*x* = 0*.*05 *y discretizaci´on temporal*

∆*t* = 0*.*02*. Dibujar la soluci´on obtenida para θ* = 0 *(Euler Expl´ıcito). ¿Pue- des justificar la gr´afica obtenida? Determinar el paso de discretizaci´on temporal m´aximo que asegura estabilidad y calcular la soluci´on para ese paso temporal.*

*¿Cu´anto vale la soluci´on en el punto x = 0.85?*

#### Soluci´on.

En primer lugar introducimos los datos del problema que vamos a necesitar para ejecutar *ecucalor.m.*:

* intespacio = [0 1]; intiempo = [0 1]; pasosespacio=20;
* pasostiempo=50; theta = 0; c=1
* u0 = @(x) 0.\*x;
* g =@(t,x) x.^2.\*(x-1);
* f = @(t,x) 2.\*t-x.^2+10.\*x.\*t;

una vez hecho esto, podemos ejecutar el c´odigo:

* + [x,uf] = ecucalor(c,intespacio,intiempo, ...
  + pasosespacio,pasostiempo,theta,u0,g,f);

Representamos el resultado en la Figura [4.7:](#_bookmark62)

* + figure;
  + plot(x,uf)

x 1068

1

0.5

0

−0.5

−1

−1.5

0 0.2 0.4 0.6 0.8 1

*{*tercersembis1*}*

Figura 4.7: Resultados con ∆*x* = 0*.*05 y ∆*x* = 0*.*02.

Como puede observarse hay inestabilidad debido a que el paso temporal es ex- cesivamente grande. En este caso, el m´aximo paso temporal que garantiza la estabilidad ser´a (∆*x*)2 , que es ∆*t* = 0*.*0013, y por tanto se corresponden con

2

800 pasos temporales. Ejecutamos de nuevo el c´odigo con este nuevo paso de discretizaci´on en tiempo:

* + pasosespacio=20, pasostiempo=800;
  + [x,uf1] =ecucalor(c,intespacio,intiempo, ...
  + pasosespacio,pasostiempo,theta,u0,g,f);
  + figure;
  + plot(x,uf1)

Representamos los resultados en la Figura [4.8,](#_bookmark63) y observamos que ahora no apa- recen inestabilidades.

Finalmente calculamos el valor de la soluci´on num´erica en *x* = 0*.*85:

*>* x1=min(find(x>=0.85));

* + uf1(x1)

0.8

0.7

0.6

0.5

0.4

0.3

0.2

0.1

0

0 0.2 0.4 0.6 0.8 1

*{*tercersembis2*}*

Figura 4.8: Resultados con ∆*x* = 0*.*05 y ∆*x* = 0*.*00013.

que resulta ser 0.4496.

**Ejercicio 4.12** (Octave) *Sea dado el PVIC*

*{*Ej.T3.Sol.12*}*

**** *∂u ∂*2*u*

*∂t* = *α∂x*2 + *f* (*x, t*)*, ∀* (*x, t*) *∈* (0*, L*) *×* (0*, T* )*,*

****



*u*(*x, t*) = (*x* 3)2

*−*

2

****



+ *e−t−*(*x−*3)

*, x* = 0*, x* = *L,*

*u*(*x,* 0) = *e−*(*x−*3)2 *, x ∈* (0*, L*)*,*

*que representa un proceso de difusi´on de la concentraci´on de una especie qu´ımi- ca, que se realiza durante un tiempo T* = 20 *segundos, a trav´es de una membrana de L* = 2 *cm de longitud, siendo α* = 0*.*015*, la difusividad de la especie. La fun- ci´on:*

*f* (*x, t*) = 3 *− x*2 + *e−*2*t*(*x −* 1)*,*

*representa un t´ermino de forzamiento que hace que la concentraci´on de la especie crezca de forma no homog´enea y no constante en el tiempo.*

1. *Aplicar el algoritmo ecucalor.m para calcular la solucion en el dominio*

[0*, L*] *y en el intervalo temporal* [0*, T* ]*, con paso de discretizaci´on espacial*

∆*x* = 0*.*05 *y discretizaci´on temporal* ∆*t* = 0*.*02*, para θ* = 0 *(m´etodo de Euler expl´ıcito). Determinar la concentraci´on m´axima y m´ınima de la especie al cabo de T* = 20 *segundos. Calcular la concentraci´on en el punto x* = 1*.*15 *cm. ¿Sabr´ıas calcular cu´anto ha disminuido la concentraci´on en la frontera derecha?*

1. *Manteniendo los mismos pasos de discretizaci´on espacial y temporal, de- terminar la concentraci´on m´axima y m´ınima de la especie al cabo de T* = 2 *minutos. Calcular la concentraci´on en el punto x* = 1*.*15 *cm.*

*Para tiempos grandes, t , la soluci´on del problema anterior se estabiliza a la soluci´on del problema estacionario asociado definido por el PVC:*

f

*→ ∞*

*−αu′′*(*x*) = 3 *− x*2*, ∀ x ∈* (0*, L*)*,*

*u*(0) = 9*, u*(*L*) = 1*,*

*y cuya soluci´on anal´ıtica es u*(*x*) = 9 *−* 1 r 3 *x*2 *−*  1 *x*4 + (4*α −* 7 *x*l*.*

*α*

2

12

3

*Aplicar el algoritmo bvpdirichlet.m para calcular la soluci´on en el mismo inter- valo* (0*, L*)*, L* = 2*, con paso de discretizaci´on h* = 0*.*05*. Se pide:*

1. *Determinar la concentraci´on en el punto x* = 1*.*15 *y el error cometido.*
2. *Determinar el m´aximo y m´ınimo de concentraci´on.*

**Soluci´on.** (a) Los comando necesarios para resolver este apartado son:

* L = 2; intespacio= [0,L];
* T = 20; intiempo = [0,T];
* dx= 0.05; dt = 0.02; pasosespacio=L/dx, pasostiempo=T/dt;
* theta = 0; c = 0.015;
* u0 = @(x) exp(-(x-3).^2);
* f = @(x,t) 3-x.^2+exp(-2\*t).\*(x-1);
* g = @(x,t) (x-3).^2+exp(-t-(x-3).^2);
* [x,uf] = ecucalor(c,intespacio,intiempo,...
* pasosespacio,pasostiempo,theta,mu,u0,g,f);

Calculamos la concentraci´on m´axima y m´ınima

* cmax = max(uf)
* cmin = min(uf)

obteniendo, 37.825 y 1, respectivamente. En el punto *x* = 1*.*15 de la membrana,

*x* = *x*(24) la concentraci´on al cabo de *T* = 20 segundos es uf(24) =29.2232. La disminuci´on de la concentraci´on en la frontera derecha es g(L,0)-g(L,T) = 0.3679.

1. Usando los comandos:

* T2 = 120; intiempo2 = [0,T2];
* pasosespacio=L/dx, pasostiempo2=T2/dt;
* [x2,uf2] = ecucalor(c,intespacio,intiempo2,...
* pasosespacio,pasostiempo2,theta,mu,u0,g,f);

se obtiene que al cabo de 2 minutos, uf2(24) = 59.9829. El m´aximo de con- centraci´on en la membrana despu´es de 2 minutos es max(uf2)= 67.7643 y el m´ınimo es min(uf2) =1.

Dibujamos las gr´aficas (ver Figura [4.9)](#_bookmark64) de las aproximaciones obtenidas en los dos instantes *T* = 20 y *T* = 120 segundos con el comando:

* + - plot(x,uf,x2,uf2); legend(’T=20s’, ’T=120s’)

70

T=20s

T=120s

60

50

40

30

20

10

0

0 0.5 1 1.5 2

Figura 4.9: Aproximaciones en 20 y 120 segundos obtenidas de la concentraci´on

de la especie. *{*Evolutivo1*}*

1. Tenemos que usar el algoritmo *bvpdirichlet.m* de la siguiente forma:
   * + h = 0.05; numeronodos = ((b-a)/h)+1;
     + f = @(x) 3-x.^2;
     + [xh,uh]=bvpdirichlet(0,L,numeronodos,c,0,0,f,9,1);

La concentraci´on estacionaria en el punto *x* = *xh*(24) = 1*.*15 es *uh*(24) = 60*.*7692. Definiendo la soluci´on anal´ıtica como:

* + - uexac = @(x) 9-1/c\*(3/2\*x.^2-1/12\*x.^4+(4\*c-7/3).\*x); obtenemos que el error cometido es abs(uh(24)-uexac(punto)) = 0.0136. (d) max(uh) = 68.5333, y min(uh) =1.

# Problemas propuestos

## Problemas de Transporte estacionarios en dominios unidimensionales

**Ejercicio 4.13** *Considerar el problema de contorno:*

*{*Ej.T3.Prop.1*}*

**** *−u′′*(*x*) + 1 *u′*(*x*) *−* (1 *−* 1

****



*x*

4*x*2

*u*(1) = 1*,*

)*u*(*x*) = 0*, x ∈* [1*,* 4]*,*

*u*(4) = *−*0*.*47*.*

*Tomando un paso de discretizaci´on h* = 1*, plantear un esquema en diferen- cias finitas descentrado a contracorriente y resolver el problema, expresando el resultado en forma de sistema de ecuaciones lineales Au*¯ = ¯*b.*

Ej.T3.Prop.2

*{ }*

**Ejercicio 4.14** *Considerar el problema de contorno:*

****

*u′′*(*x*) *−* (*x* + 1)*u′*(*x*) + 2*u*(*x*) = 2*, x ∈* (0*,* 1)*, u*(0) = 1*,*

****

*u′*(1) = 4*.*

*Resolver el problema tomando un paso de discretizaci´on h* = 1 *, planteando un*

2

*esquema en diferencias finitas descentrado a contracorriente.*

Ej.T3.Prop.3

*{ }*

**Ejercicio 4.15** *Considerar el problema de contorno:*

****

*u′′*(*x*) *− xu′*(*x*) + 2(*x −* 2)*u*(*x*) = 2*, x ∈* (0*,* 1)*, u*(0) = 1*,*

****

*u′*(1) = 2*e*2*.*

*Tomando un paso de discretizaci´on h* = 1 *, se pide:*

2

* + - 1. *Resolver el problema planteando un esquema en diferencias finitas centra- das.*
      2. *Resolver el problema planteando un esquema en diferencias finitas descen- tradas a contracorriente para el t´ermino convectivo.*

*En ambos casos puedes expresar el resultado en forma de sistema de ecuaciones lineales, Au*¯ = ¯*b.*

Ej.T3.Prop.4

*{ }*

**Ejercicio 4.16** (Octave) *Sea dado el Problema de Transporte Difusivo y Con- vectivo definido por el PVC:*

f

*−u′′* + *u′* = *−*2*e−x, ∀x ∈* (0*,* 1)*, u*(0) = 2*, u*(1) = *e* + *e−*1*,*

*cuya soluci´on anal´ıtica es:*

*u*(*x*) = *ex* + *e−x.*

*Aplicar el algoritmo bvpdirichlet.m para calcular la solucion en el intervalo* [0*,* 1]*, con paso de discretizaci´on h* = 0*.*01*.*

1. *Dibujar la soluci´on anal´ıtica junto con la solucion num´erica.*
2. *Determinar los valores m´aximos y m´ınimos de u y su localizaci´on, tanto para la soluci´on num´erica como para la anal´ıtica.*
3. *Dibujar la gr´afica del error cometido por el m´etodo en x ∈* [0*,* 1]*.*

**Ejercicio 4.17** (Octave) *Sea dado el Problema de Transporte Difusivo y Con- vectivo definido por el PVC:*

*−u′′ −* 4*u′* = *−*16*x*3 + 34*x −* 1*, ∀x ∈* (0*,* 2)*, u*(0) = 4*, u*(2) = 2*,*

f

*cuya soluci´on anal´ıtica es:*

*u*(*x*) = *x*4 *− x*3 *−* 3*.*5*x*2 + 2*x* + 4*,*

*siendo u*(*x*) *la concentraci´on de un contaminante.*

1. *Aplicar el algoritmo bvpdirichlet.m para calcular la soluci´on en el intervalo [0, 2] con paso de discretizaci´on h* = 0*.*125*.*
2. *Dibujar la soluci´on anal´ıtica junto con la solucion num´erica.*
3. *Determinar los valores m´aximos y m´ınimos de contaminaci´on as´ı como su localizaci´on tanto para la soluci´on num´erica, como para la anal´ıtica. Calcular los errores cometidos.*
4. *Determinar la regi´on m´axima de seguridad (donde la concentraci´on del contaminante es menor que uno), es decir, u*(*x*) *<* 1*. Utilizar la solucion anal´ıtica.*

*{*Ej.T3.Prop.5*}*

**Ejercicio 4.18** (Octave) *Sea dado el problema de transporte definido por el PVC:*

*{*Ej.T3.Prop.6*}*

*−*4*u′′*(*x*) *−* 6*u′* + 2*u* = 2*x*3 *−* 22*x*2 *−* 4*e−x* + 20*, ∀ x ∈* (0*,* 4)*, u*(0) = 1*, u*(4) = 34 *− e−*4*,*

f

*y cuya soluci´on anal´ıtica en el intervalo* (0*,* 4) *es:*

*u*(*x*) = *x*3 *−* 2*x*2 *− e−x* + 2*.*

*Aplicar el algoritmo bvpdirichlet.m para calcular la soluci´on con paso de discre- tizaci´on h* = 0*.*02*. Adem´as, se pide:*

* 1. *Determinar la temperatura m´ınima en el intervalo* (0*,* 4) *y el error come- tido en su estimaci´on.*
  2. *Utiliza el m´etodo de Newton o el de bisecci´on para determinar la regi´on de la varilla para la cual u*(*x*) *>* 10*. Considera una tolerancia de* 10*−*3 *y un nu´mero m´aximo de 1000 iteraciones.*

## Problemas de Transporte estacionarios en dominios bidimensionales

*{*Ej.T3.Prop.7*}*

**Ejercicio 4.19** *Considera el dominio abierto* Ω *de frontera formada por los lados L*1*, L*2*, L*3 *y L*4 *que se recoge en la Figura* [*4.10.*](#_bookmark76) *Sobre dicho dominio se pretende resolver mediante un m´etodo en diferencias el problema de contorno siguiente:*

 *−∇ ·* (4*∇u*(*x, y*)) + *∇ ·* (*→−***V**(*x, y*)*u*(*x, y*)) + *u*(*x, y*) = 0*, en* Ω*,*

****

 1*−*4*∇u*(*x, y*) + *→−***V**(*x, y*)*u*(*x, y*)l *· →−***n** (*x, y*) = 0*, en L*1 *L*3*,*

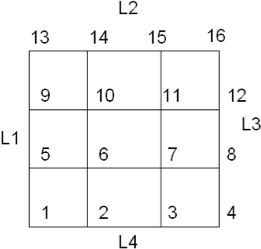
**** *u*(*x, y*) = 0*, en L*2 *L*4*.*

*donde →−***V**(*x, y*) *es el campo de velocidades de convecci´on que se considera dado por:*

*→−***V**(*x, y*) = (*−*1 *.*

1

*Se pide escribir las ecuaciones algebraicas a las que conduce el plantear un esquema en diferencias finitas de 5 puntos en cruz para la aproximaci´on del t´ermino difusivo y descentrado contracorriente para la aproximaci´on del t´ermino convectivo, junto a, en su caso, la imposici´on de las correspondientes condiciones de contorno, en los nodos* 6*,* 8 *y* 9 *del mallado dado.*



*{*figura103*}*

*{*Ej.T3.Prop.8*}*

Figura 4.10: Dominio Ω, su frontera *L*1 *L*2 *L*3 *L*4 y correspondiente ma- llado de taman˜o h=1.

**Ejercicio 4.20** *Considera el dominio abierto* Ω = (0*,* 2) (0*,* 3) *de frontera formada por los lados N , E, S y O. Sobre dicho dominio se pretende resolver mediante un m´etodo en diferencias el problema de contorno siguiente:*

*×*

****  1 *− x*  

*−*∆*u* + div 

****

****

*y −* 2

 *u* + *u* = *x*(*y −* 2) + (*y* + 2)*, en* Ω*,*

*u*(*x,* 0) = 2*x, en S,*

****

**** *u*(2*, y*) = 2(*y* + 2)*, en E,*



 *u*(*x,* 3) = 5*x, en N,*

****

**** 

****

****

 1 *− x* 

*y −* 2

 *→−* 2

*Se pide:*

*−∇u*(*x, y*) +

*u*(*x, y*)

*·* **n** = (*y* + 2)(1 *− x* + *x* )*, en O.*

1. *Desarrollar las operaciones indicadas obteniendo la ecuaci´on del transpor- te que define el modelo. Determinar el r´egimen de flujo, los procesos f´ısicos y el tipo de fluido definidos por el modelo.*

Ayuda*: Recuerda que* div(*V u*) = *u*div(*V* ) + *V u* y *→−***n** es un vector

*· ∇*

unitario y normal a la frontera correspondiente.

1. Desarrollar las operaciones indicadas obteniendo la ecuaci´on que define el flujo en la frontera oeste, *O*, y particularizarla para los nodos 4 y 7 de *O*.
2. Escribir las 6 ecuaciones algebraicas a las que conduce el plantear un es- quema en diferencias finitas de **5 puntos en cruz para la aproximaci´on del t´ermino difusivo y descentrado contracorriente para la apro- ximaci´on del t´ermino convectivo**, junto a, en su caso, la imposici´on de las correspondientes condiciones de contorno, en los nodos interiores 5, 8 y en los nodos 4 y 7 de *O*.
3. Escribir las 6 ecuaciones algebraicas calculadas anteriormente en forma de sistema para el vector de inc´ognitas *u*¯ = (*u*5*, u*8*, u*4*, u*7*, u*4*a, u*7*a*).

*{*Ej.T3.Prop.9*}* **Ejercicio 4.21** *Considera el dominio abierto* Ω*, de frontera formada por los lados L*1*, L*2*, L*3 *y L*4 *que se recoge en la Figura* [*4.11.*](#_bookmark79)

*Sobre dicho dominio se pretende resolver mediante un m´etodo en diferencias el problema de contorno siguiente:*

**** *−*2*∇ ·* (*∇u*(*x, y*)) + *∇ ·* (*→−***V**(*x, y*)*u*(*x, y*)) + 2*u*(*x, y*) = 0*, en* Ω*,*

 *u*(*x, y*) = 0*, en L*1 *L*3*,*

1*−*2*∇u*(*x, y*) + *→−***V**(*x, y*)*u*(*x, y*)l *· →−***n** (*x, y*) = 0*, en L*2 *L*4*,*

****

*donde →−***V**(*x, y*) *es el campo de velocidades de convecci´on que se considera dado por:*

( *−*

*→−***V**(*x, y*) = *x y*

*x* + *y*

*y →−***n** (*x, y*) *es el vector normal unitario exterior en el punto* (*x, y*) *de la frontera de* Ω*. Se pide escribir las ecuaciones algebraicas a las que conduce el plantear un esquema en diferencias finitas de 5 puntos en cruz para la aproximaci´on del t´ermino difusivo y contracorriente para el convectivo, junto a, en su caso, la imposici´on de las correspondientes condiciones de contorno (considerar formulas descentradas para aproximar las derivadas en la direcci´on de la normal), en los nodos* 6*,* 8 *y* 9 *del mallado dado.*

Ej.T3.Prop.10

*{ }*

**Ejercicio 4.22** (Octave) *Sea dado el PVC estacionario 2D asociado a la ecua-*

*ci´on de Poisson en el dominio* Ω = (0*,* 1) *×* (0*,* 1)*:*

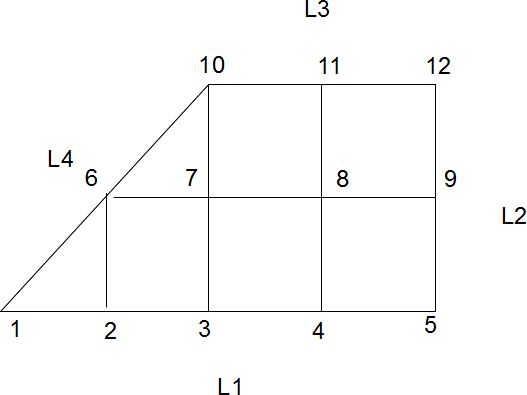
f

*−*∆*u* = *f* (*x, y*)*,* (*x, y*) *∈* Ω*,*

*u*(*x, y*) = *g*(*x, y*)*,* (*x, y*) *∈ ∂*Ω*,*

*donde*

*f* (*x, y*) = 8*π*2 sin(2*πx*) cos(2*πy*)*, g*(*x, y*) = sin(2*πx*) cos(2*πy*)*.*



*{*figura102*}*

Figura 4.11: Dominio Ω correspondiente mallado de taman˜o h=1, estando el nodo 1 localizado en el origen de coordenadas (0*,* 0).

*Sabiendo que la soluci´on exacta es u*(*x, y*) = *g*(*x, y*)*, aplicar el algoritmo* ecupois- son.m *para calcular la soluci´on utilizando como pasos de discretizaci´on* ∆*x* =

∆*y* = 0*.*01*. Dibujar la soluci´on obtenida.*

## Problemas evolutivos en una dimensio´n espacial

**Ejercicio 4.23** *Sea dado el Problema de Valor Inicial y de Contorno:*

*{*Ej.T3.Prop.11*}*

 *∂u* ( 1 *∂*2*u ∂u*

=

*∂t* 4

****

****

*∂x*2 + *∂x −* 4*x, x ∈* (0*,* 2)*, t >* 0

(*PV IC*) = *u*(0*, t*) = *t, t >* 0*,*

*u*(2*, t*) = 8 + *t, t >* 0*,*

**** *u*(*x,* 0) = 2*x*2*, x ∈* (0*,* 2)*.*



*Considerando una discretizaci´on espacial* ∆*x* = 0*.*5 *y un paso temporal* ∆*t* = 0*.*5*,*

*se pide:*

* + - 1. *Determinar la expresi´on de la soluci´on en la forma:*

*Au*¯*n*+1 = *Bu*¯*n* + ¯*bn*+1*,*

*calculando las matrices A, B y el vector* ¯*bn*+1*, obtenidos al aplicar un esquema en diferencias finitas centradas en el espacio para la difusi´on, descentradas contracorriente para la convecci´on y un m´etodo de tipo Euler expl´ıcito para la parte temporal.*

* + - 1. *Calcular finalmente u*1 *≈ u*(*xi,* 0*.*5)*, i* = 1*, ...*5*, es decir, una aproximaci´on de la soluci´on en los nodos xi del mallado en el instante t* = 0*.*5*.*

*i*

*{*Ej.T3.Prop.12*}*

**Ejercicio 4.24** *Sea dado el Problema de Valor Inicial y de Contorno:*

 *∂u ∂u ∂*2*u*

*∂t* + *V ∂x* = *D∂x*2 *, x ∈* (0*,* 1)*, t >* 0

****

(*PV IC*) = **** *u*(0*, t*) = 4 *−* 2*t, t >* 0*,*

*u*(1*, t*) = 5 *−* 2*t, t >* 0*,*

****

**** *u*(*x,* 0) = 4 + *x, x ∈* (0*,* 1)*,*

*siendo V* = 2*, la velocidad del fluido y D* = 1*/*10 *el coeficiente de difusi´on. Considerando una discretizaci´on espacial* ∆*x* = *L/N* = 1*/*5 *y un paso temporal*

∆*t* = 1*/*5*, dividimos el intervalo I* = [0*,* 1] *en N* = 5 *subintervalos de la forma*

[*xi, xi*+1]*, xi* = (*i −* 1)∆*x, i* = 1*, , ...N* + 1*, N* + 1 = 6*.*

*Se pide determinar la expresi´on de la soluci´on en la forma:*

*u*¯*n*+1 = *Au*¯*n* + ¯*bn*+1*,*

*calculando expl´ıcitamente la matriz A y el vector* ¯*bn*+1 *al que conduce la apli- caci´on de un esquema en diferencias finitas centradas en el espacio para la difusi´on, descentradas contracorriente para la convecci´on y un m´etodo de tipo Euler expl´ıcito para la parte temporal.*

*Calcular finalmente u*1 *u*(*xi,* ∆*t*)*, i* = 1*...*6*, es decir, una aproximaci´on de la soluci´on en los nodos del mallado.*

*i*

*≈*

*{*Ej.T3.Prop.13*}*

**Ejercicio 4.25** (Octave) *Sea dado el PVIC:*

****

*∂u ∂*2*u*

*∂t* = *∂x*2

+ *f* (*x, t*)*,*

*donde*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = 2*,*

**** *u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*,* 2)*,*

*f* (*x, t*) = *t* sin(*x*)*, g*(*x, t*) = 0*,*

f

*u*0(*x*) = *x, para* 0 *≤ x ≤* 1*,*

2 *− x, para* 1 *< x ≤* 2*.*

*Aplicar el algoritmo ecucalor.m para calcular la soluci´on en el intervalo temporal [0,1] con paso de discretizaci´on espacial* ∆*x* = 0*.*05 *y discretizaci´on temporal*

∆*t* = 0*.*02*. Dibujar la soluci´on obtenida para θ* = 0*.*5 *en t* = 1*. ¿Cu´anto vale la soluci´on en el punto x = 1?*

Ej.T3.Prop.14

*{ }*

**Ejercicio 4.26** (Octave) *Sea dado el PVIC:*

**** *∂u ∂*2*u*

**** *∂t* = *α∂x*2 + *f* (*x, t*)*, ∀* (*x, t*) *∈* (0*, L*) *×* (0*, T* )*,*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = *L,*

****

*u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*, L*)*,*

*que representa un proceso de difusi´on de la temperatura en una varilla de L* = 4 *cm de longitud, siendo α* = 0*.*08*, la difusividad del material que compone la varilla. La funci´on*

*x*2

*f* (*x, t*) = *x* sin(*t*) + 1 + (*x −* 2)2 *,*

*representa un t´ermino de forzamiento que hace que la temperatura var´ıe de forma no homog´enea y no constante en el tiempo. El comportamiento en la frontera para todo instante t vienen dado por:*

*g*(*x, t*) = *t*3 *− t*2 + *e−t* + 1 + *x*(4 *− x*)*,*

*siendo la distribuci´on inicial de temperaturas de la varilla dada por:*

sin(*x*)

*u*0(*x*) = 1 + 1 + (*x −* 2)2 *.*

1. *Tomando* ∆*x* = 0*.*01 *y* ∆*t* = 0*.*01*, calcula la soluci´on del problema para θ* = 0*.*5 *(Crank-Nicolson), al cabo de 2 segundos. ¿Qu´e temperatura se alcanza en el punto x* = 1*? Compar´ando dicha temperatura con el instante inicial, ¿la varilla en x* = 1 *se ha calentado o enfriado?*
2. *Calcula la temperatura m´ınima en el instante inicial y al cabo de los 2 segundos, as´ı como los lugares de la varilla en donde se alcanzan.*
3. *Resuelve el problema para θ* = 0 *(Euler expl´ıcito), con los mismos valores de* ∆*x y* ∆*t del apartado* (*a*)*. En caso de aparecer inestabilidades determi- nar, manteniendo fijo* ∆*t, el paso de discretizaci´on espacial* ∆*x m´ınimo que asegura la estabilidad y con ese valor volver a aplicar el algoritmo de Euler expl´ıcito. ¿Qu´e temperatura m´ınima se alcanza con este m´etodo?*
4. *Representar en una misma gr´afica el forzamiento f* (*x, t*) *en los instantes t* = 0 *y t* = 1*. Determinar el punto donde el forzamiento es m´aximo en ambos casos.*

**Ejercicio 4.27** (Octave) *Sea dado el PVIC:*

*{*Ej.T3.Prop.15*}*

**** *∂u ∂*2*u*

****

*donde*

*∂t* = *∂x*2 + *f* (*x, t*)*, ∀* (*x, t*) *∈* (0*,* 1) *×* (0*,* 1)*,*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = 1*,*

**** *u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*,* 1)*,*

*f* (*x, t*) = 2*t − x*2 + 10*xt, g*(*x, t*) = *x*2(*x −* 1)*, u*0(*x*) = 0*.*

* 1. *Aplicar el algoritmo ecucalor.m para calcular la soluci´on en el intervalo temporal* [0*,* 1] *con paso de discretizaci´on espacial* ∆*x* = 0*.*05 *y discretiza- ci´on temporal* ∆*t* = 0*.*02*.*

*Dibujar la soluci´on obtenida para θ* = 0 *(Euler expl´ıcito). ¿Puedes jus- tificar la gr´afica obtenida? Determinar el paso de discretizaci´on temporal m´aximo que asegura estabilidad y calcular la soluci´on para ese paso tem- poral.*

*¿Cu´anto vale la soluci´on en el punto x* = 0*.*85*?*

* 1. *Aplicar el algoritmo de Crank-Nicolson para calcular la soluci´on del PVIC en las mismas condiciones. Dibujar la soluci´on calculada y evaluarla en el punto x* = 0*.*85*.*

Ej.T3.Prop.16

*{ }*

**Ejercicio 4.28** (Octave) *Sea dado el PVIC:*

**** *∂u* 1 *∂*2*u*

**** *∂t −* 10 *∂x*2 = *f* (*x, t*)*, ∀* (*x, t*) *∈* (0*, L*) *×* (0*, T* )*,*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = *L,*

****

*u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*, L*)*,*

*que representa un proceso de difusi´on de la temperatura en una varilla de L* = 2*π*

*cm de longitud, donde la funci´on*

*f* (*x, t*) = cos(*x*)*,*

*representa un t´ermino de forzamiento que hace que la temperatura var´ıe de for- ma no homog´enea y no constante en el tiempo. Las condiciones de contorno que definen el comportamiento de la temperatura en la frontera para todo instante t, vienen dadas por:*

*g*(*x, t*) = 1 *−* cos(*x*)*,*

*siendo la distribuci´on inicial de temperaturas de la varilla dada por:*

*u*0(*x*) = sin(*x*)*.*

1. *Calcula la soluci´on del problema con θ* = 0*.*5 *(m´etodo de Crank-Nicolson), para calcular la soluci´on en el dominio* [0*,* 2*π*] *al cabo de 5 segundos, to- mando 100 intervalos en x y un paso de discretizaci´on temporal* ∆*t* = 0*.*5*.*
2. *Determinar la temperatura m´axima y m´ınima de la varilla al cabo de T* = 5

*segundos, as´ı como el lugar de la varilla donde se alcanzan.*

1. *Calcular la temperatura que se alcanza en el punto medio de la varilla, es decir, en x* = *π. Comparando con su temperatura inicial, determina si el punto x* = *π, se ha enfriado o calentado.*

**Ejercicio 4.29** (Octave) *Sea dado el PVIC:*

*{*Ej.T3.Prop.17*}*

**** *∂u* ( 1 *∂*2*u*

=

**** *∂t* 10

*∂x*2 *, ∀* (*x, t*) *∈* (0*,* 10) *×* (0*,* 5)*,*

*donde:*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = 10*,*

**** *u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*,* 10)*,*

*g*(*x, t*) = *−x*(*x −* 10)*, u*0(*x*) = 2*x*2[*H*(*x −* 3) *− H*(*x −* 4)]*,*

*siendo H la funci´on de Heaviside. El dato inicial u*0(*x*) *simula un vertido a lo largo de una carretera entre el km 3 y el km 4.*

1. *Aplicar el algoritmo ecucalor.m para calcular la soluci´on en el intervalo temporal* [0*,* 5]*, con paso de discretizaci´on espacial* ∆*x* = 0*.*1 *y discretiza- ci´on temporal* ∆*t* = 0*.*1*.*

*Dibujar la soluci´on obtenida con θ* = 0 *(Euler expl´ıcito). ¿Puedes justificar la gr´afica obtenida?*

1. *Aplicar el algoritmo del m´etodo Euler Impl´ıcito para calcular la solucion del PVIC en las mismas condiciones anteriores. Dibujar la soluci´on cal- culada.*

*Utilizar los resultados obtenidos para aproximar el valor m´aximo de con- centraci´on del contaminante as´ı como su localizaci´on.*

*Si la concentraci´on del contaminante fuese letal en dosis superiores a* 5*,*

*¿sabr´ıas aproximar la regi´on que habr´ıa que evacuar? ¿Y al cabo de un tiempo T* = 20*? (no olvides mantener las mismas condiciones aunque el tiempo sea mayor).*

**Ejercicio 4.30** (Octave) *Sea dado el PVIC:*

*{*Ej.T3.Prop.18*}*

**** *∂u ∂*2*u*

**** *∂t* = *α∂x*2 + *f* (*x, t*)*, ∀* (*x, t*) *∈* (0*, L*) *×* (0*, T* )*,*

*u*(*x, t*) = *g*(*x, t*)*, x* = 0*, x* = *L,*

****

*u*(*x,* 0) = *u*0(*x*)*, x ∈* (0*, L*)*,*

*que representa un proceso de difusi´on de la temperatura que se realiza durante un tiempo T* = 30 *segundos a trav´es de una varilla de L* = 4 *cm de longitud, siendo α* = 0*.*008*, la difusividad del material que compone la varilla. La funci´on*

*f* (*x, t*) = 4 cos(2*x*) + (*x −* 1)*e−t,*

*representa un t´ermino de forzamiento que hace que la temperatura var´ıe de for- ma no homog´enea y no constante en el tiempo. Las condiciones de contorno que definen el comportamiento de la temperatura en la frontera para todo instante t, vienen dadas por:*

*t*

*g*(*x, t*) = 1 + *t*2 + *x*(*x −* 4)*.*

*La distribuci´on inicial de temperaturas de la varilla dada por el dato inicial:*

*u*0(*x*) =

*x*3 *−* 1*,* 1 *< x <* 3*,*

**** 0*,* 3 *≤ x <* 4*.*

**** 0*,* 0 *< x ≤* 1*,*

*Aplicar el algoritmo ecucalor.m para calcular la soluci´on en el dominio* [0*, L*]*, L* = 4 *y en el intervalo temporal* [0*, T* ]*, T* = 30 *segundos, con paso de discreti- zaci´on espacial* ∆*x* = 0*.*04 *y discretizaci´on temporal* ∆*t* = 0*.*15*. Se pide:*

* 1. *Calcular la soluci´on para θ* = 0 *(m´etodo de Euler expl´ıcito). En caso de aparecer inestabilidades, determinar el paso de discretizaci´on temporal m´aximo que asegura la estabilidad y con este valor volver a aplicar el algoritmo de Euler expl´ıcito determinando la temperatura m´axima de la varilla al cabo de T* = 30 *segundos.*

*Calcular la temperatura en el punto x* = 2*.*36*.*

*Comparando con su temperatura inicial, determinar si el punto x* = 2*.*36*. se ha enfriado o calentado.*

* 1. *Manteniendo los mismos pasos de discretizaci´on espacial y temporal, de- terminar la temperatura m´axima y m´ınima de la varilla al cabo de T* = 2 *minutos y calcula la temperatura en el punto x* = 2*.*36*.*

*Dibuja la gr´afica de la temperatura en la frontera derecha en cada instante*

*t ∈* [0*, T* ]*.*

* 1. *APLICACIONES* 91

*Calcular num´ericamente el valor del m´aximo de temperatura en la frontera derecha y el instante en el cual se alcanza el m´aximo.*

*Para tiempos grandes, t , la soluci´on del problema anterior se estabiliza a la soluci´on del problema estacionario asociado definido por el PVC:*

f

*→ ∞*

*−αu′′*(*x*) = 4 cos(2*x*)*, ∀ x ∈* (0*, L*)*, u*(0) = 0*, u*(*L*) = 0*,*

*y cuya soluci´on anal´ıtica es:*

*u*(*x*) = 1 rcos(2*x*) *−* 1 (cos(8) *−* 1)*x −* 1l *.*

*α*

4

*Aplicar el algoritmo bvpdirichlet.m para calcular la soluci´on en el mismo inter- valo anterior* (0*, L*)*, L* = 4*, con paso de discretizaci´on h* = 0*.*04*. Se pide:*

* 1. *Determinar la temperatura en el punto x* = 2*.*36 *y el error cometido.*
  2. *Da una estimaci´on de la temperatura m´axima en el intervalo* (0*, L*)*, y de- termina el error cometido con respecto al m´aximo de la soluci´on anal´ıtica.*

# Aplicaciones

**Ejercicio 4.31 (Deflexi´on de una placa)** *La deflexi´on w*(*x*) *de una placa rectangular larga y uniformemente cargada, y que se encuentra bajo una fuerza de tensi´on axial, se rige por la ecuaci´on diferencial de segundo orden:*

*{*Ej.T3.Aplic.1*}*

f *w′′*(*x*) *− S w*(*x*) = *−qlx* + *q x*2*,* 0 *≤ x ≤ l,*

*D*

2*D*

2*D*

*w*(0) = 0*, w*(*l*) = 0*,*

*donde S es la fuerza axial, q es la intensidad de la carga uniforme, l es la longitud de la placa y D es la rigidez de deflexi´on de la placa.*

* + 1. *Aproxima la deflexi´on de la placa en intervalos de* 1 *pulgada para los va- lores q* = 200 *lb/plg*2*, S* = 100 *lb/plg, D* = 8*.*8 107 *lb/plg y l* = 50 *plg (*no es necesario que cambies las unidades*).*

*×*

* + 1. *Si la regulaci´on estatal fija el m´aximo de deflexi´on en* 0*.*1*, ¿cumple esta placa la normativa?*

**Ejercicio 4.32 (Deformaci´on de un cilindro)** *La ecuaci´on que analiza las relaciones de esfuerzo-deformaci´on y las propiedades materiales de un cilindro sujeto alternativamente al calentamiento y enfriamiento viene dada por:*

*{*Ej.T3.Aplic.2*}*

1 *∂T*

*∂*2*T* 1 *∂T*

=

4*K ∂t*

*∂r*2 + *r ∂r ,* 0*.*5 *≤ r <* 1*, t >* 0*,*

*en donde T* (*r, t*) *es la temperatura, r es la distancia radial respecto al centro del cilindro, t es el tiempo y K* = 0*.*0025 *es el coeficiente de difusividad.*

*Tomando las condiciones iniciales y de frontera dadas por*

*T* (0*.*5*, t*) = *t, t ≥* 0*, T* (1*, t*) = 100 + 40*t, t ≥* 0*,*

*T* (*r,* 0) = 200(*r −* 0*.*5)*,* 0*.*5 *≤ r ≤* 1*.*

*Se pide:*

* + - 1. *Resolver el problema con los pasos de discretizaci´on* ∆*r* = 0*.*1 *y* ∆*t* = 0*.*5*, utilizando el m´etodo de Euler expl´ıcito para aproximar la parte temporal.*

*Determinar la expresi´on Au*¯*n*+1 = *Bu*¯*n* + ¯*bn*+1 *de la soluci´on*

*general calculando las matrices A, B y el vector* ¯*bn*+1*.*

* + - 1. *Calcular las aproximaciones obtenidas en el instante T* = 0*.*5*.*

**Cap´ıtulo 5**

**Soluciones a los problemas propuestos**

# Resolucio´n de ecuaciones no lineales

**Ejercicio** [**2.13.**](#_bookmark6) *sol* = 0*.*76625, *niter* = 5.

**Ejercicio** [**2.14.**](#_bookmark7) *(a) x*5 = 0*.*88168 con residuo *f* (*x*5) = 0*.*094766.

1. Tomando como semilla *x*0 = 1, la aproximaci´on dentro de la tolerancia pedida es *x*3 = 0*.*89541. El residuo en este caso es *ϕN* (*x*3) *x*3 = 6*.*4071 10*−*13, menor que en el apartado anterior.

*| − | ·*

1. Con *ϕN* (*x*) = *x − f* (*x*)*/f′*(*x*), se obtienen las ra´ıces *z*1 = *−*0*.*82026, *z*2 =

*−*0*.*047530 y *z*3 = 0*.*8954. Tomando *ϕ*1(*x*) = arcsin( ) se obtiene u´nica-

1 *x−*1*/*4

*π* 2

mente la ra´ız *z*2 = *−*0*.*047530.

1. Utilizando el m´etodo de Newton para resolver la ecuaci´on *f′*(*x*) = 0, con la semilla *x*0 = 1*.*5, se obtiene un m´aximo de 3*.*2754 localizado en el punto *x* = 1*.*5509.

**Ejercicio** [**2.15.**](#_bookmark8)*(a)* Tomado *x*0 = 3, obtenemos como aproximaci´on *x*3 = 3*.*0571.

1. Para el m´etodo de punto fijo obtenemos como aproximaci´on dentro de la tolerancia pedida *x*4 = 3*.*0571, por lo que es una iteraci´on m´as lento.
2. El m´etodo de Newton-Raphson produce la aproximaci´on *soln* = 3*.*0571, con un residuo de 4*.*4409 *·* 10*−*16 en 4 iteraciones, mientras que el m´etodo de punto fijo obtiene *solpf* = 3*.*0571, con un residuo de 1*.*0411 *·* 10*−*7 en 7 iteraciones.
3. Utilizando el m´etodo de Newton-Raphson obtenemos el valor *x* = 3*.*6737. **Ejercicio** [**2.16.**](#_bookmark9) (a) Tras dibujar la ecuaci´on se observa que la ecuaci´on tiene tres ra´ıces reales. Utilizando el intervalo [ 2*,* 1], se tiene *x* = 1*.*3193, *itera* = 9. Utilizando el intervalo [ 1*,* 0], se tiene *x* = 0*.*43262, *itera* = 9, y utilizando el intervalo [1*,* 2], se tiene *x* = 1*.*7529, *itera* = 9.

*— −*

*— − −*

1. Calculamos para ello la derivada de la funci´on, obteniendo *f′*(*x*) = *−*6*x*2 +5. Utilizando el m´etodo de Newton con *x*0 = *−*1*.*5 se tiene, *x* = *−*1*.*32, *itera* = 4. Utilizando el el m´etodo de Newton con *x*0 = *−*0*.*5 se tiene, *x* = *−*0*.*43232,

93

*itera* = 3. Utilizando el m´etodo de Newton con *x*0 = 1*.*5 se tiene, *x* = 1*.*7523,

*itera* = 4.

**Ejercicio** [**2.17.**](#_bookmark10)(a) El an´alisis de la gr´afica de la funci´on revela que la ecua- ci´on admite una u´nica soluci´on. Considerando el esquema num´erico asociado al m´etodo de Newton como un esquema de punto fijo, se tiene que:

*f* (*x*)

*ϕN* (*x*) = *x − f′*(*x*) *,*

y el esquema *x* = *ϕN* (*x*). Utilizando el m´etodo de punto fijo con semilla *x*0 = 1, se tiene *x* = 1*.*0875, *itera* = 4. Utilizando el m´etodo de Aitken con la misma semilla, se tiene, *x* = 1*.*0875, *itera* = 3. Ambos m´etodos son convergentes, pero el m´etodo de Aitken es ligeramente m´as r´apido.

* 1. Utilizando el m´etodo de punto fijo con semilla *x*0 = 1, se tiene, *x* = 1*.*0875, *itera* = 6. Con Aitken se alcanza la misma soluci´on, pero en 7 iteraciones, resultando en este caso ligeramente m´as lento.
  2. Utilizando el m´etodo de punto fijo con semilla *x*0 = 1, la sucesi´on no est´a definida (aparece un NaN) en la s´eptima iteraci´on. Con Aitken se alcanza la soluci´on *x* = 1*.*0875, en 7 iteraciones. En este caso el m´etodo de Aitken nos ha proporcionado un m´etodo convergente, mientras que el de puntofijo no ha convergido.

**Ejercicio** [**2.18.**](#_bookmark11) El resultado es *x* = 1*.*4096, *itera* = 5.

**Ejercicio** [**2.19.**](#_bookmark12) *(a) x* = 1*.*47498 en 3 iteraciones.

*√*

*(b)* Definimos *ϕ*2(*x*) = ( *x* sin(*x*) + 2)1*/*3 y obtenemos *x* = 1*.*47498 en 3 itera- ciones.

**Ejercicio** [**2.20.**](#_bookmark13) *x* = *−*0*.*56076, *y* = *−*1*.*10400 e *itera* = 6.

**Ejercicio** [**2.21.**](#_bookmark14) *x* = 1*.*77245, *y* = 1*.*77245 e *itera* = 6.

**Ejercicio***√*[**2.22.**](#_bookmark15) Resolvemos la ecuaci´on *f* (*x*) = *x*2 *−*2*−ex* = 0*√*tomando *a* = *−*2

y *b* = *−* 2. Para esta elecci´on tenemos que hacer log2((*−* 2 + 2)*/*10*−*2) =

5*.*8723 *< k* = 6 iteraciones, obteneniendo *x*6 = *−*1*.*4966. El punto de corte es

(*−*1*.*4966*,* 0*.*23981).

**Ejercicio** [**2.23.**](#_bookmark16) *(a) f* alcanza un m´aximo de 0.89412 y un m´ınimo de -0.39871.

*(b)* El u´nico punto de inflexi´on es (*−*0*.*46311*,* 0*.*56387).

*(b)* Los puntos de intersecci´on son (*−*0*.*62695*,* 0*.*77906) y (0*.*96688*,* 0*.*25522).

**Ejercicio** [**2.24.**](#_bookmark18) Aplicando el algoritmo de Newton-Raphson para sistemas, ob- tenemos los siguientes valores considerando una tolerancia de 10*−*6: *p*2=43.9589, *Q*=24.2662, *Q*1= 14.1355 y *Q*2=10.1307. Como semilla hemos considerado los valores: *p*2=20, *Q*=2, *Q*1= 1 y *Q*2=1.

# Problemas de Valor Inicial

**Ejercicio** [**3.11.**](#_bookmark27) Al tomar *h* = 0*.*5, s´olo tenemos que realizar un paso. La *√*apro-

ximaci´on pedida, *y*1, es soluci´on de la ecuaci´on no lineal *F* (*x*) = *x −* 2 *−*

0*.*3033*/x* = 0*√*, que resuelta con el m´etodo de Newton-Raphson, tomando como

semilla *x*0 =

*y*1 = 1*.*602.

2, da por soluci´on, *x*1 = 1*.*546 y *x*2 = 1*.*602*.* Por tanto, tomamos

*√*

La soluci´on exacta del PVI es *y*(*t*) = 4 *−* 2*e−t*, por tanto, el error cometido es

*e* = *|y*(0*.*5) *− y*1*|* = *|*1*.*669 *−* 1*.*602*|* = 0*.*067*.*

**Ejercicio** [**3.12.**](#_bookmark28) Los nodos temporales utilizados son *t*0 = 0, *t*0*,*1 = 0, *t*0*,*2 = 0*.*5, *t*1 = 0*.*5, *t*1*,*1 = 0*.*5 y *t*1*,*2 = 1. Para el primer paso tenemos que *y*0*,*1 = *y*0*,*2 = 1 y *y*1 = 0*.*75. Para el segundo paso *y*1*,*1 = 0*.*75, *y*1*,*2 = 0*.*46875 y la aproximaci´on pedida es *y*2 = 0*.*499515.

La soluci´on exacta del PVI es *y*(*t*) =

1

*t*2 +1

, por lo que el error cometido por el

m´etodo en *t* = 1 es *e* = *|y*(1) *− y*2*|* = 0*.*000485.

**Ejercicio** [**3.13.**](#_bookmark29) Los nodos temporales utilizados son *t*0 = 0, *t*0*,*1 = 0, *t*0*,*2 = 0*.*5, *t*1 = 0*.*5, *t*1*,*1 = 0*.*5 y *t*1*,*2 = 1. Para el primer paso tenemos que *y*0*,*1 = 1, *y*0*,*2 = 0*.*5 y *y*1 = 0*.*64695. Para el segundo paso *y*1*,*1 = 0*.*64695, *y*1*,*2 = 0*.*302069 y la aproximaci´on pedida es *y*2 = 0*.*412502.

La soluci´on exacta del PVI es *y*(*t*) =  1 = *e−t*, por lo que el error cometido por

*et*

el m´etodo en *t* = 1 es *e* = *|y*(1) *− y*2*|* = 0*.*04462.

**Ejercicio** [**3.14.**](#_bookmark30) Los nodos temporales utilizados son *t*0 = 0, *t*0*,*1 = 0, *t*0*,*2 = 0*.*5, *t*1 = 0*.*5, *t*1*,*1 = 0*.*5 y *t*1*,*2 = 1. Para el primer paso tenemos que *y*0*,*1 = 1, *y*0*,*2 = 1 e *y*1 = 1*.*25. para el segundo paso *y*1*,*1 = 1*.*25, *y*1*,*2 = 1*.*809017 y la aproximaci´on pedida es *y*2 = 2*.*202007.

La soluci´on exacta del PVI es *y*(*t*) = (0*.*5*t*2 + 1)2, por lo que el error cometido por el m´etodo en *t* = 1 es *e* = *|y*(1) *− y*2*|* = 0*.*04793.

**Ejercicio** [**3.15.**](#_bookmark31) *(a)* En la primera iteraci´on se obtiene *y*1 = 0*.*4375 al resolver con bisecci´on la ecuaci´on *y*1 *−* 0*.*5 1 *− y*2 = 0 con un intervalo inicial de [0*,* 0*.*5]. En la segunda iteraci´on se obtiene la aproximaci´on pedida *y*2 = 0*.*78906 al resolver con biseci´on la ecuaci´on *y*2 0*.*5 1 *y*2 0*.*4375 = 0 con el intervalo inicial [0*.*4375*,* 1]. El error cometido es de 0*.*052408.

1

1

*— − −*

1. La gr´afica se encuentra en la Figura [5.1.](#_bookmark95) La aproximaci´on es 0*.*66687 come- tiendo un error de 4*.*5331 *·* 10*−*6.
2. Intervalo aproximado = [0*.*65*,* 1*.*5], Intervalo exacto = [0*.*6435*,* 1*.*5], come- ti´endose un error de 0*.*0064987 en el extremo izquierdo.
3. El error m´aximo es de 4*.*6758 *·* 10*−*6 (ver Figura [5.2).](#_bookmark96)

**Ejercicio** [**3.16.**](#_bookmark32) En la Figura [5.3](#_bookmark97) aparece la soluci´on aproximada junto con la exacta. Puede observarse que los resultados son muy parecidos y en la gr´afica aparecen solapados.

Los resultados obtenidos para cada uno de los m´etodos para *t* = 0*.*1, *t* = 0*.*5

1

CN

exacta

0.8

0.6

0.4

0.2

0

0 0.5 1 1.5 2

Figura 5.1: Aproximaci´on obtenida de *y*(*t*) = sen(*x*) en el intervalo [0*,* 1*.*5] junto

con la recta *y* = 0*.*6. *{*CMFig1*}*

5e-06

4e-06

3e-06

2e-06

1e-06

0

0 0.5 1 1.5 2

Figura 5.2: Gr´afica del error. *{*CMFig2*}*

y *t* = 1 son 1*.*82623, 1*.*43530, 1*.*22542 para *heun.m* y 1*.*82621, 1*.*43527, 1*.*22540

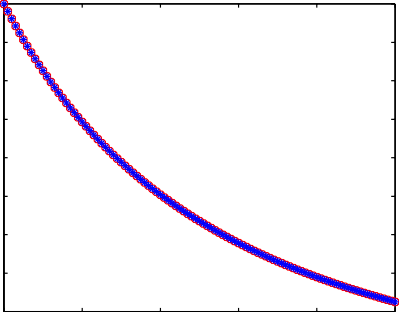
para *rungekuttao3.m* respectivamente.

**Ejercicio** [**3.17.**](#_bookmark33) S´olo la soluci´on *y*2 es estable ya que *y*1 oscila de manera explo- siva e *y*3 oscila de manera amortiguada. El l´ımite de estabilidad de este problema es *he <* 1*/*6. Aparecen oscilaciones que son amortiguadas hasta *h <* 1*/*3, luego constantes para *h* = 1*/*3 y finalmente explosivas para *h >* 1*/*3. Para su deter- minaci´on se define el cambio de variable *y*(*t*) = ln(*v*(*t*) + *t*) + 1 y se estudia el problema para *v*(*t*).

El m´ınimo de concentraci´on se da en el instante *t* = (1*/*6) ln(60) = 0*.*68239 y un valor de 0*.*83637.

**Ejercicio** [**3.18.**](#_bookmark34) (a) Puesto que *t*(*m*) = (*m −* 1)*h*, se tiene *t*(19) = 18*h* = 3*.*6 y *uee*(19) = 49*.*892. La soluci´on anal´ıtica es *y*(3*.*6) = 68*.*499, por tanto, el error cometido es *eee* = 18*.*607. Para calcular la soluci´on en el instante *t* = 3*.*627, se define *Nh* = 5000, lo que nos da *h* = 10*−*3 y se ejecuta nuevamente el algoritmo. Se tiene *uee*(3628) = 70*.*245. La soluci´on anal´ıtica es *y*(3*.*627) = 70*.*372, por tanto, el error cometido es *eee* = 0*.*12722.

(b) Puesto que *t*(*m*) = (*m−*1)*h*, se tiene *t*(19) = 18*h* = 3*.*6, con *uei*(19) = 103*.*93

2

1.9

1.8

1.7

1.6

1.5

1.4

1.3

0 0.2 0.4 0.6 0.8 1

Figura 5.3: Con c´ırculos rojos aparecen representados los resultados obtenidos

con Heun y con asteriscos azules los obtenidos con Simpson (rungekuttao3). *{*segundosem3*}*

y *ucn*(19) = 68*.*286. La soluci´on anal´ıtica es *y*(3*.*6) = 68*.*499, por tanto, los errores cometidos son *eei* = 35*.*435 y *ecn* = 0*.*21256.

**Ejercicio** [**3.19.**](#_bookmark35) (a) Puesto que *t*(*m*) = (*m −* 1)*h*, se tiene *t*(26) = 25*h* = 1*.*25 y *uee*(26) = 1*.*2347. La soluci´on anal´ıtica es *y*(1*.*25) = 1*.*2876 por tanto, el error cometido es *eee* = 0*.*052912. El m´ınimo de concentraci´on se da en el instante *t* = 1 con valores *y*(1) = 1*.*2752 y *t*(21) = 20*h* = 1*.*0 con *uee*(21) = 1*.*2271.

En caso de haber resuelto el problema con paso de discretizaci´on *h* = 0*.*004 y pidiendo la evaluaci´on en *t* = 1*.*24, se tendr´ıa: *Nh* = 500, *t*(311) = 310*h* = 1*.*24, *uee*(311) = 1*.*2827, *y*(1*.*24) = 1*.*2866 y *eee* = 0*.*003865.

(b) Puesto que *t*(*m*) = (*m* 1)*h*, se tiene *t*(26) = 1*.*25, con *uhe*(26) = 1*.*2869. La soluci´on anal´ıtica es *y*(1*.*25) = 1*.*2876 y el error cometido es *ehe* = 7*.*009 10*−*4. El m´etodo m´as preciso es el de Heun.

*×*

*−*

**Ejercicio** [**3.20.**](#_bookmark36) Es claro que s´olo *uee*1 es estable ya que *uee*2 oscila inicialmente mientras que *uee*3 oscila de manera constante. El l´ımite de estabilidad de este problema es *he <* 0*.*25, aparecen oscilaciones que son amortiguadas hasta *h <* 0*.*5, luego constantes para *h* = 0*.*5 y finalmente explosivas para *h >* 0*.*5.

**Ejercicio** [**3.21.**](#_bookmark37)(a) Puesto que *t*(*m*) = (*m −* 1)*h*, se tiene *t*(2) = *h* = 0*.*2 y *uee*(2) = 0*.*8. La soluci´on anal´ıtica es *y*(0*.*2) = 1*.*4181, por tanto, el error cometido es *eee*(0*.*2) = 0*.*6181*.*

(b) Puesto que *t*(*m*) = (*m −* 1)*h*, se tiene *t*(2) = *h* = 0*.*2, con *uei*(2) = 1*.*5568 y *ucn* = 1*.*3073 La soluci´on anal´ıtica es *y*(0*.*2) = 1*.*4181 por tanto, los errores cometidos son *eei*(0*.*2) = 0*.*13867 y *ecn*(0*.*2) = 0*.*11080. El m´etodo m´as preciso es el de Crank-Nicolson.

**Ejercicio** [**3.22.**](#_bookmark38) (a) Puesto que *t*(*m*) = (*m−* 1)*h*, se tiene *t*(107) = 106*h* = 10*.*6 y *uee*(107) = 6*.*9828. La soluci´on anal´ıtica es *y*(10*.*6) = 2*.*4510, por tanto, el error cometido es *eee* = 4*.*5318*.*

(b) Puesto que *t*(*m*) = (*m −* 1)*h* se tiene *t*(107) = 10*.*6, con *uhe*(107) = 4*.*0716 y *usim* = 2*.*4210 La soluci´on anal´ıtica es *y*(10*.*6) = 2*.*4510, por tanto, los errores cometidos son *ehe* = 1*.*6205 y *esim* = 0*.*029969. El m´etodo m´as preciso es el de Simpson.

**Ejercicio** [**3.23.**](#_bookmark39)S´olo *uee*3 es estable ya que *uee*2 oscila inicialmente, mientras que *uee*1 oscila de manera explosiva. El l´ımite de estabilidad de este problema es *he <* 0*.*1, aparecen oscilaciones que son amortiguadas hasta *h <* 0*.*2, luego constantes para *h* = 0*.*2 y finalmente explosivas para *h >* 0*.*2. Para la determi- naci´on de la soluci´on exacta se define el cambio de variable *y*(*t*) = 2 *v*(*t*) y se estudia el problema para *v*(*t*).

*−*

**Ejercicio** [**3.24.**](#_bookmark40) (a) Puesto que *t*(*m*) = (*m −* 1)*h*, se tiene *t*(13) = 12*h* = 0*.*3 y *uee*(13) = 3*.*9320. La soluci´on anal´ıtica es *y*(0*.*3) = 3*.*9468, por tanto, el error cometido es *eee* = 0*.*014752.

(b) Puesto que *t*(*m*) = (*m−* 1)*h*, se tiene *t*(13) = 12*h* = 0*.*3, con *uei*(4) = 3*.*9621 y *ucn* = 3*.*9468. La soluci´on anal´ıtica es *y*(0*.*3) = 3*.*9468, por tanto, los errores cometidos son *eei* = 0*.*015335 y *ecn* = 5*.*9019 *×* 10*−*5.

**Ejercicio** [**3.25.**](#_bookmark41) (a) Puesto que *t*(*m*) = (*m−* 1)*h*, se tiene *t*(311) = 310*h* = 1*.*24 y *uee*(311) = 0*.*38373. La soluci´on anal´ıtica es *y*(1*.*24) = 0*.*38433, por tanto, el error cometido es *eee* = 6*.*0292 *×* 10*−*4. El m´ınimo de concentraci´on se da en el instante *t* = 1, con valores *y*(1) = 0*.*37754 y *uee*(251) = 0*.*37706.

(b) Puesto que *t*(*m*) = (*m* 1)*h*, se tiene *t*(311) = 1*.*24, con *uhe*(311) = 0*.*38433 y *usim*(311) = 0*.*38433. La soluci´on anal´ıtica es *y*(1*.*24) = 0*.*38433 y los errores cometidos son *ehe* = 4*.*0405 10*−*8 y *esim* = 3*.*1418 10*−*11. El m´etodo m´as preciso es el de Simpson (rungekuttao3.m).

*−*

*× ×*

**Ejercicio** [**3.26.**](#_bookmark42)(a) Soluci´on exacta 2.5992. El m´as preciso es Simpson y el menos, Euler expl´ıcito.

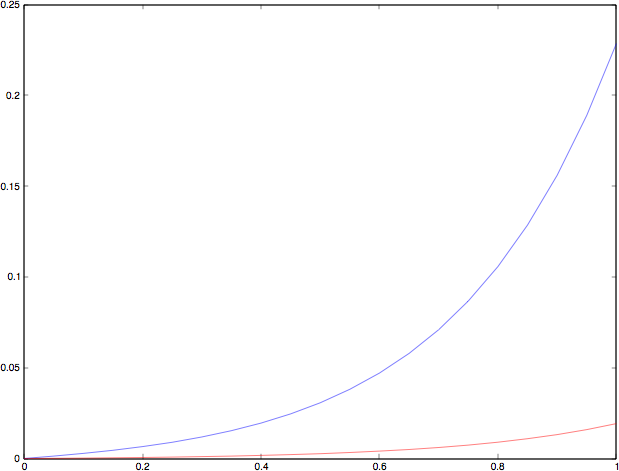
|  |  |  |
| --- | --- | --- |
|  | Aproximaci´on (*pos* = 20) | Error |
| Euler Expl´ıcito | 2.4095 | 0.18964 |
| Euler Impl´ıcito | 2.7882 | 0.18901 |
| Heun | 2.6151 | 0.015927 |
| Simpson (rungekuttao3) | 2.5990 | 1.5213*·*10*−*4 |

(b) La gr´afica del error se muestra en la Figura [5.4:](#_bookmark98)

**Ejercicio** [**3.27.**](#_bookmark43) (a) Soluci´on exacta 1.49. El m´as preciso es Simpson y el menos, Euler impl´ıcito.

|  |  |  |
| --- | --- | --- |
|  | Aproximaci´on (*pos* = 8) | Error |
| Euler Expl´ıcito | 1.4123 | 0.077721 |
| Euler Impl´ıcito | 1.5719 | 0.081884 |
| Heun | 1.4888 | 0.0012153 |
| Simpson (rungekuttao3) | 1.4900 | 3.6444*·*10*−*5 |

(b) La gr´afica del error se muestra en la Figura [5.5:](#_bookmark99)



*{*err1*}*

*{*err2*}*

Figura 5.4: Gr´afica del error del Problema [3.26](#_bookmark42)

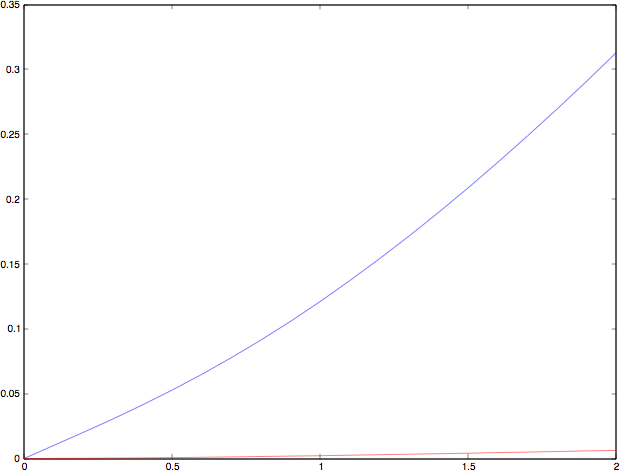
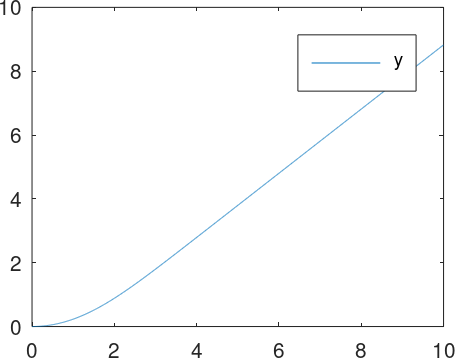


Figura 5.5: Gr´afica del error del Problema [3.27](#_bookmark43)

**Ejercicio** [**3.28.**](#_bookmark45) **Capa l´ımite laminar.** Este ejercicio lo resolvemos utilizando el siguiente c´odigo Octave:

* + - odefun1=@(t,x,y,z) y;
    - odefun2=@(t,x,y,z) z;
    - odefun3=@(t,x,y,z) -x.\*z;
    - inicial1=0;
    - inicial2=0;
    - inicial3=0.47;
    - Nh=1000;
    - tspan=10;
    - n=1001;
    - h=0.01;
    - t=ones(n);
    - x=ones(n);
    - y=ones(n);
    - z=ones(n);
    - x(1)=inicial1;
    - y(1)=inicial2;
    - z(1)=inicial3;
      * t(1)=0;
      * for i=2:n;
      * t(i)=h.\*(i-1);
      * x(i)=x(i-1)+h\*odefun1(t(i-1),x(i-1),y(i-1),z(i-1));
      * y(i)=y(i-1)+h\*odefun2(t(i-1),x(i-1),y(i-1),z(i-1));
      * z(i)=z(i-1)+h\*odefun3(t(i-1),x(i-1),y(i-1),z(i-1));
      * end

El perfil pedido aparece representado en la figura [5.6.](#_bookmark100)



*{*perfil328*}*

Figura 5.6: Gr´afica del perfil. Problema 3.28.

#### Ejercicio [3.29.](#_bookmark46)

*>*odefun1=@(t,x,y) -x.^2+y;

*>*odefun2=@(t,x,y) -2.\*y+x.^2;

*>*inicial1=1;

*>*inicial2=0;

*>*h=0.01;

*>*tspan=0.8;

*>*n=1+(tspan/h);

*>*t=ones(n);

*>*x=ones(n);

*>*y=ones(n);

*>*x(1)=inicial1;

*>*y(1)=inicial2;

*>*t(1)=0;

*>*for i=2:n;

* + - * t(i)=h.\*(i-1);
      * x(i)=x(i-1)+h\*odefun1(t(i-1),x(i-1),y(i-1));
      * y(i)=y(i-1)+h\*odefun2(t(i-1),x(i-1),y(i-1));

*>*end

*>*plot(t,x,’r’,t,y,’b’);

*>*legend(’y1’,’y2’);

*>*x(n),y(n)

Los valores obtenidos tomando un paso de discretizaci´on *h* = 0*.*01, son *y*1 = 0*.*6451 e *y*2 = 0*.*2185. Los perfiles aparecen dibujados en la [figura5.7.](#_bookmark101)

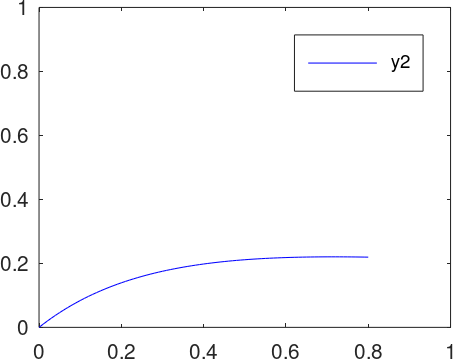
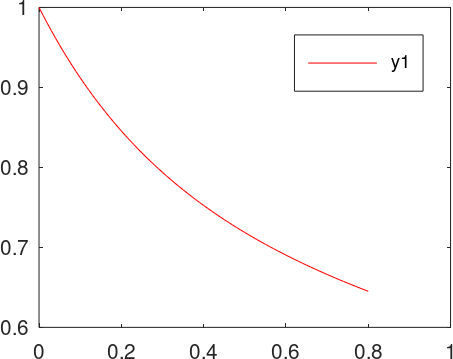


Figura 5.7: Perfiles referentes al ejercicio 3.29. *{*perfil329*}*

#### Ejercicio [3.30.](#_bookmark48)

*>*odefun1=@(t,x,y) y;

*>*odefun2=@(t,x,y) -6.54.\*x-0.8.\*y;

*>*inicial1=0;

*>*inicial2=1;

*>*h=0.05;

*>*tspan=10;

*>*n=1+(tspan/h);

*>*t=ones(n);

*>*x=ones(n);

*>*y=ones(n);

*>*x(1)=inicial1;

*>*y(1)=inicial2;

*>*t(1)=0;

*>*for i=2:n;

* + - t(i)=h.\*(i-1);
    - x(i)=x(i-1)+h\*odefun1(t(i-1),x(i-1),y(i-1));
    - y(i)=y(i-1)+h\*odefun2(t(i-1),x(i-1),y(i-1));
    - end

El resultado pedido aparece representado en la figura [5.8.](#_bookmark102)

**Ejercicio** [**3.31.**](#_bookmark49) La velocidad terminal te´orica es de 24.978, que coincide (con 3 cifras decimales) con la obtenida por el m´etodo Runge-Kutta a partir de los

12.52 segundos, al tomar un paso de discretizaci´on *h* = 0*.*01. La Figura [5.9](#_bookmark103) muestra la velocidad de la moneda segu´n va cayendo.

**Ejercicio** [**3.32.**](#_bookmark50) El tanque se vac´ıa (aproximadamente) a los 56.5 minutos.

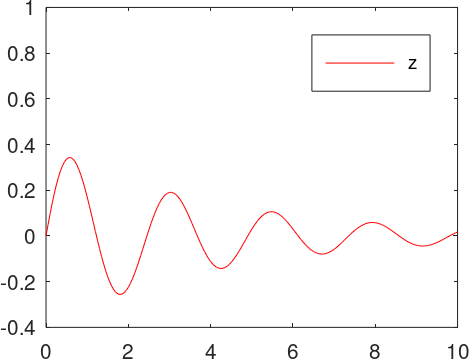


Figura 5.8: Resultado del ejercicio 3.30. *{*perfil330*}*

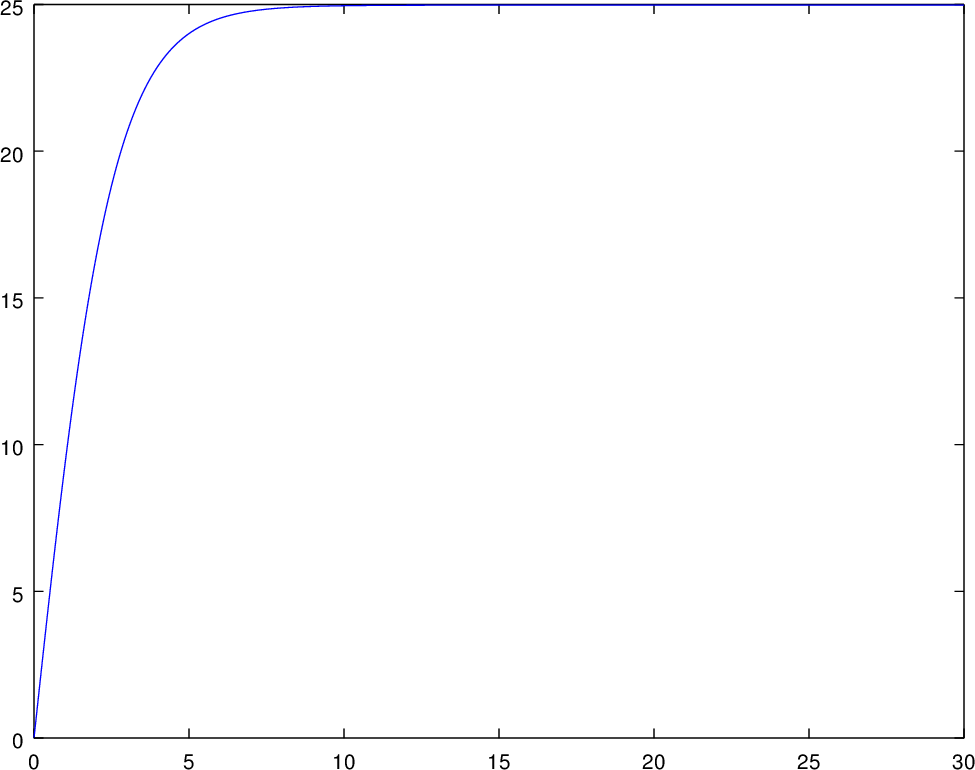


Figura 5.9: Velocidad de la moneda *{*vterm*}*

# Problemas de Contorno

**Ejercicio** [**4.13.**](#_bookmark67) El sistema viene dado por:

 1 0 0 0  *u*1  1 

*−*3*/*2 26*/*16 *−*1 0  *u*2 =  0 

 0 *−*4*/*3 49*/*36 *−*1 *u*3  0 

0 0 0 1

*u*4

*−*0*.*47

que da como soluci´on (*u*1*, u*2*, u*3*, u*4) = (1*,* 1*.*78909*,* 1*.*40727*, −*0*.*47000).

**Ejercicio** [**4.14.**](#_bookmark68) An˜adiendo el nodo ficticio *x*4 = 3*/*2, el sistema que resuelve el problema viene dado por:

 1 0 0 0  *u*1

*−*4 13 *−*7 0  *u*2 =  2 

 0 

 0 *−*4 10 *−*4 *u*3

0 *−*1 0 1

*u*4

10

4

que da como soluci´on (*u*1*, u*2*, u*3) = (0*,* 2*.*7297*,* 4*.*7838), sin considerar la aproxi- maci´on *u*4 en el nodo ficticio.

**Ejercicio** [**4.15.**](#_bookmark69)El mallado es *x*1 = 0, *x*2 = 0*.*5, *x*3 = 1 y el nodo ficticio

*x*4 = 1*.*5.

1. Esquema centrado conduce a:

*−*10 3 0 *u*2

*−*4

 4 *−*10 4 *u*3 = 2*e*2

*−*1 0 1

*u*4

2*e*2

con soluci´on (*u*2*, u*3*, u*4) = (2*.*4411*,* 6*.*3863*,* 17*.*2192).

1. Esquema descentrado conduce a:

*−*11 7*/*2 0 *u*2 *−*9*/*2

 4 *−*10 4 *u*3 =  2*e*2 

*−*1 0 1

*u*4

2*e*2

con soluci´on (*u*2*, u*3*, u*4) = (2*.*2764*,* 6*.*2545*,* 17*.*0545).

**Ejercicio** [**4.16.**](#_bookmark70) Los comandos necesarios son:

*>* f = @(x) -2.\*exp(-x);

*>* [xh,uh] = bvpdirichlet(0,1,101,1,1,0,f,2,exp(1)+exp(-1));

Definimos la soluci´on exacta y dibujamos ambas funciones:

*>* uexact = @(x) exp(x)+exp(-x);

*>* figure; plot(xh,uh,xh,uexact(xh)); legend(’aprox’,’exacta’)

Los resultados aparecen en la Figura [5.10.](#_bookmark105)

3.2

aprox

exacta

3

2.8

2.6

2.4

2.2

2

0 0.2 0.4 0.6 0.8 1

Figura 5.10: En rojo aparecen representados los resultados num´ericos y en verde

la soluci´on exacta. *{*tercersem1*}*

El valor m´aximo es 3.08616 y se alcanza en *x* = 1. El m´ınimo es 1.92256 y se alcanza en *x* = 0*.*27. Los comandos utilizados son:

*>* nmax = max(uh)

*>* nmin = min(uh)

el valor m´ınimo de *u* es , y vemos en qu´e valores de *x* se alcanzan:

ERROR

4e-07

3e-07

2e-07

1e-07

*{*tercersem1\_2*}*

0

0 0.2 0.4 0.6 0.8 1

Figura 5.11: Funci´on error.

*>* xm = find(uh==nmax);

*>* xmin = find(uh==nmin);

*>* locmax = xh(xm)

*>* locmin = xh(xmin)

Finalmente dibujamos el error cometido (v´ease la Figura [5.11).](#_bookmark106)

*>* verror = abs(uh-uexact(xh));

*>* figure; plot(xh,verror); title(’ERROR’)

**Ejercicio** [**4.17.**](#_bookmark71) Antes de ejecutar bvpdirichlet.m, debemos introducir:

*>* a = 0; b = 2; D = 1; V = -4; Q = 0;

*>* f =@(x) -16.\*x.^3+34.\*x-1;

*>* ua = 4; ub = 2; numeronodos = 17;

*>* [xh,uh] = bvpdirichlet(a,b,N,D,V,Q,f,ua,ub);

Para dibujar en un mismo plot la soluci´on num´erica y la exacta, primero debe- mos definir la soluci´on exacta:

*>* solexac = xh.^4-xh.^3-3.5.\*xh.^2+2.\*xh+4;

*>* figure;

*>* plot(xh,uh,’r’,xh,solexac,’g’)

Los resultados aparecen en la Figura [5.12.](#_bookmark107) Ahora calculamos el valor m´aximo, que da 4.32345, y el m´ınimo, 0.72331:

*>* nmax = max(uh)

*>* nmin = min(uh)

y vemos en qu´e valores de *x* se alcanzan:

*>* xm = find(uh>=nmax); xh(xm)

*>* xmin = find(uh<=nmin); xh(xmin) El valor m´aximo se alcanza en *x* = 0*.*25 y el valor m´ınimo se alcanza en *x* = 1*.*625. Tambi´en calculamos el valor m´aximo 4.26953 y el m´ınimo 0.68969 de la soluci´on exacta:

*>* emin = min(solexac)

*>* emax = max(solexac)

y dibujamos el error cometido en la Figura [5.13.](#_bookmark108) *>* figure;

*>* verror = uh-solexac;

4.5

4

3.5

3

2.5

2

1.5

1

0.5

0 0.5 1 1.5 2

*{*sem6*}*

Figura 5.12: En rojo aparecen representados los resultados num´ericos y en verde la soluci´on exacta.

0.08

0.07

0.06

0.05

0.04

0.03

0.02

0.01

0

0 0.5 1 1.5 2

*{*sem6\_2*}*

Figura 5.13: Funci´on error.

* plot(xh,verror)

Finalmente, para la determinaci´on de la regi´on de seguridad se resuelve la ecua- ci´on *u*(*x*) = 1 definiendo una funci´on *f* (*x*) = *u*(*x*) *−* 1 y aplicando dos veces el m´etodo de Newton-Raphson. Se tiene que para *x*0 = 1*.*5 obtenemos *x* = 1*.*4142 en 3 iteraciones y para *x*0 = 2 obtenemos *x* = 1*.*8229 en 4 iteraciones. Una regi´on de seguridad es por tanto, [1*.*5*,* 1*.*75].

#### Ejercicio [4.18.](#_bookmark72)

1. Temperatura m´ınima = 0*.*54209 y en *x* = 1*.*26.

Con soluci´on la soluci´on exacta, el m´ınimo es 0*.*54152, con error = 5*.*7613 10*−*4.

*·*

1. La regi´on es [2*.*9349*,* 4].

**Ejercicio** [**4.19.**](#_bookmark74) La ecuaci´on de Transporte que debemos aproximar es:

( *∂*2*u ∂*2*u ∂u ∂u*

*−*4

*∂x*2 +

*— ∂x* + *∂y* + *u* = 0*.*

*∂y*

Ecuaci´on para el nodo 6: 5*u*2 4*u*5 + 19*u*6 5*u*7 4*u*10 = 0*.*

*— − − −*

Para el nodo 8, an˜adimos el nodo ficticio *A* = (4*,* 1). Utilizando la condici´on de contorno, tenemos que *uA* = 3 *u*8, que sustituido en la ecuaci´on en diferencias

4

obtenida para el nodo 8, da lugar a la ecuaci´on *−*4*u*7 *−*

61 *u*8 *−* 4*u*12 = 0*.*

Para el nodo 9, an˜adimos el nodo ficticio *B* = ( 1*,* 2). Utilizando la condici´on de contorno, tenemos que *uB* = 5 *u*9, que sustituido en la ecuaci´on en diferencias obtenida para el nodo 8, da lugar a la ecuaci´on *−*5*u*5 + 14*u*9 *−* 5*u*10 = 0*.*

4

*−*

4

**Ejercicio** [**4.20.**](#_bookmark75) (a) La ecuaci´on del transporte es:

*−*(*uxx* + *uyy*) + (1 *− x*) *ux* + (*y −* 2)*uy* + *u* = *x*(*y −* 2) + (*y* + 2)

El r´egimen de flujo es estacionario bidimensional, los procesos de transporte son el difusivo y el convectivo. Hay absorci´on y forzamiento. El fluido es un l´ıquido, ya que el campo de velocidades tiene divergencia nula.

* 1. Particularizando en la frontera oeste con **n** = (*−*1*,* 0)*T* , se obtiene la ecuaci´on

*ux −* (1 *− x*) *u* = (*y* + 2)(1 *− x* + *x*2)*.*

En el nodo 4 = (0*,* 1), se tiene la ecuaci´on *ux − u* = 3 y en el nodo 7 = (0*,* 2), se tiene la ecuaci´on *ux − u* = 4.

* 1. La ecuaci´on algebraica que se verifica en el nodo 5 es 6*u*5 *− u*4 *−* 2*u*8 = 10, en el nodo 8 = (1*,* 2), tenemos 5*u*8 *− u*5 *− u*7 = 17, en el nodo 4, obtenemos 7*u*4 *− u*5 *−* 2*u*7 *−* 2*u*4*a* = 3, en el nodo 7 (introduciendo el nodo artificial 7*a* = (*−*1*,* 2)), es 6*u*7 *− u*4 *− u*8 *−* 2*u*7*a* = 4. La ecuaci´on de flujo en el nodo frontera 4 = (0*,* 1) (introduciendo el nodo artificial 4*a* = ( 1*,* 1)), es *u*5 *u*4*a* 2*u*4 = 6, mientras que en la frontera 7 = (2*,* 2) (utilizando diferencias finitas centradas al ser un flujo difusivo), es *u*8 *u*7*a* 2*u*7 = 8.

*— −*

*— − −*

* 1. (d)

 6 *−*2 *−*1 0 0 0   *u*5 



*−*1 5 0 *−*1 0 0   *u*8 

*−*1 0 7 *−*2 *−*2 0   *u*4  =  3 

10

 

17

4

0 *−*1 *−*1 6 0 *−*2 *u*7

   

1 0 *−*2 0 *−*1 0  *u*4*a*

0 1 0 *−*2 0 *−*1

*u*7*a*

 6 

8

y la soluci´on num´erica es (*u*5*, u*8*, u*4*, u*7*, u*4*a, u*7*a*) = (3*,* 4*,* 0*,* 0*, −*3*, −*4).

**Ejercicio** [**4.21.**](#_bookmark77)La ecuaci´on de difusi´on-convecci´on que hay que resolver es

*∂u ∂u*

*— −*

2∆*u* + (*x y*) *∂x* + (*x* + *y*) *∂y* + 4*u* = 0 y las ecuaciones en los nodos pedidos

son:

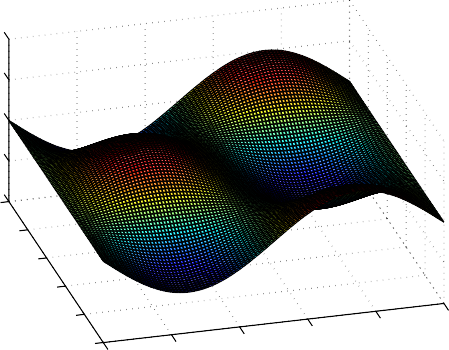
Nodo 6: *−*2*u*7 + 229 *u*6 = 0*.* Nodo 8: *−*4*u*7 + 18*u*8 *−* 2*u*9 = 0*.*

20

Nodo 9: *−*5*u*8 + 15*u*9 = 0*.*

**Ejercicio** [**4.22.**](#_bookmark78) Introducimos los datos:

* a = 0; b = 1; c = 0; d = 1; dx = 0.01; dy = 0.01;
* f = @(x,y) 8.\*pi^2.\* sin (2.\*pi.\*x).\*cos(2.\*pi.\*y);
* g =@(x,y) sin(2.\*pi.\*x).\* cos(2.\*pi.\*y);

2

1

0

−1

−2

1

0.8

0.6

0.4

0.2

0

1

0.8

0.6

0.4

0.2 0

*{*sol2d\_2*}*

Figura 5.14: Soluci´on num´erica.

N´otese que la definici´on de la soluci´on exacta coincide con la de *g*(*x, y*). Ejecu- tamos el c´odigo:

* + - [u,x,y,er] = ecupoisson(a,c,b,d,dx,dy,f,g,g);

y dibujamos los resultados (aparecen mostrados en la figura 5*.*[14.5.14)](#_bookmark109)

* + - surf(x,y,u)

**Ejercicio** [**4.23.**](#_bookmark81) *(a)* La soluci´on general para *n ≥* 0 viene dada por:

*u* *n*+1

1

 0 0 0 0 0  *u* *n*

 *tn*+1 

  *−*0*.*5 1 0*.*5 0 0  *u*2  *−*1 

1

*u*2

*u*3

=

  

+







*u*4

*u*5

0 *−*0*.*5 1 0*.*5 0 *u*3

 0 0 *−*0*.*5 1 0*.*5 *u*4

0 0 0 0 0

*u*5

*−*2

 *−*3 

8 + *tn*+1

*(b)* (*u*1*, u*2*, u*3*, u*4*, u*5)1 = (0*.*5*,* 0*.*5*,* 2*,* 4*.*5*,* 8*.*5).

**Ejercicio** [**4.24.**](#_bookmark82) La soluci´on general (para todo tiempo) es:

*u* *n*+1

1

 0 0 0 0 0 0  *u* *n* 4 *−* 2(*n* + 1)*/*5

*u*2 *u*3

 

*u*5

*u*4

*u*6

5*/*2 *−*2 1*/*2 0 0 0 *u*2

+

1

 0 5*/*2 *−*2 1*/*2 0 0 *u*3  =

   







0

0 0 5*/*2 *−*2 1*/*2 0 *u*4

   



0 0 0 5*/*2 *−*2 1*/*2 *u*5 

0 0 0 0 0 0

*u*6

0 

0 

5 *−* 2(*n* + 1)*/*5

0

Con el dato inicial (*u*1*, u*2*, u*3*, u*4*, u*5*, u*6)0 = (4*,* 4*.*2*,* 4*.*4*,* 4*.*6*,* 4*.*8*,* 5), se tiene que (*u*1*, u*2*, u*3*, u*4*, u*5*, u*6)1 = (3*.*6*,* 3*.*8*,* 4*,* 4*.*2*,* 4*.*4*,* 4*.*6).

**Ejercicio** [**4.25.**](#_bookmark83) Puesto que nos pide utilizar el c´odigo ecucalor.m, debemos in- troducir los datos siguientes comandos:

* + - * intespacio = [0 2]; intiempo = [0 1]; pasosespacio=40; pasostiempo=50;
      * theta = 0.5; c=1;

El dato inicial,

* + - * u0 = @(x) x.\*(heaviside(x)-heaviside(x-1))+...
      * (2-x).\*(heaviside(x-1)-heaviside(x-2));

utilizando la funci´on *heaviside.m*. El dato de contorno y el forzamiento:

* + - * g = @(t,x) 0.\*x.\*t;
      * f = @(t,x) t.\*sin(x);

Finalmente, podemos ejecutar el c´odigo:

* + - * [x,u] = ecuacalor(c,intespacio,intiempo,pasosespacio,...
      * pasostiempo,theta,u0,g,f);

y dibujar la soluci´on (Figura [5.15):](#_bookmark110)

* + - * plot(x,u)

0.4

0.35

0.3

0.25

0.2

0.15

0.1

0.05

0

−0.05

0 0.5 1 1.5 2

*{*tercersem2*}*

Figura 5.15: Soluci´on num´erica.

Por u´ltimo, calculamos lo que vale la soluci´on en *x* = 1, que es 0.35166, mediante el comando:

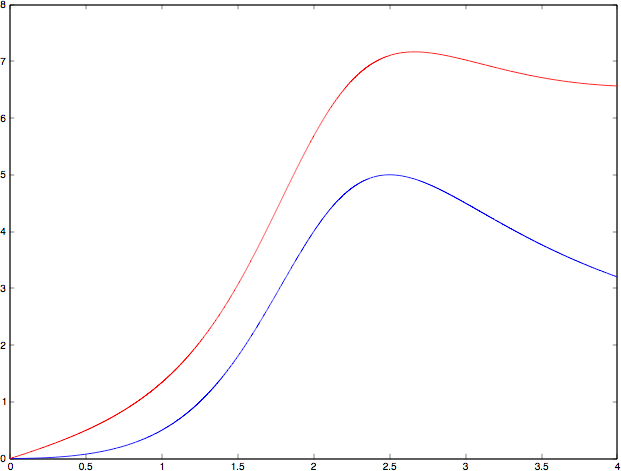
* + - x1 = min(find(x>=1))
    - u(x1)

**Ejercicio** [**4.26.**](#_bookmark84) (a) La temperatura en *x* = 1 es de 4.4188. En el instante inicial es de 1.4207, por tanto, se ha calentado.

1. M´ınimo inicial = 0*.*84864 en *x* = 4, m´ınimo a los 2 segundos = 2*.*5454 en

*x* = 0*.*39*.*

1. ∆*xmin* = 0*.*04. Temperatura m´ınima = 2*.*5481*.*
2. En *t* = 0, forzamiento m´aximo en *x* = 2*.*5. En *t* = 1, forzamiento m´aximo en *x* = 2*.*662.

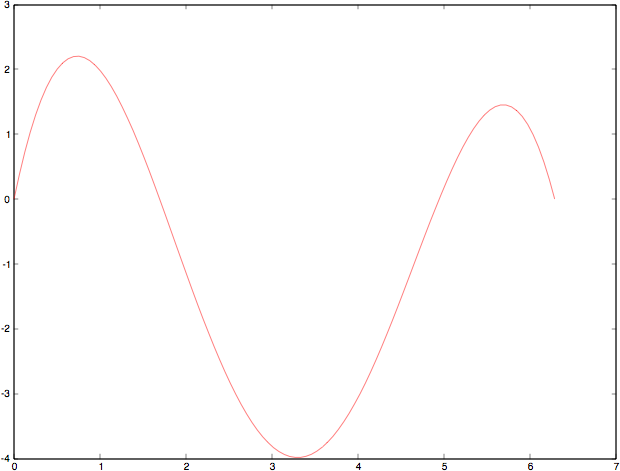


**Ejercicio** [**4.27.**](#_bookmark85)(a) Puesto que *α* = *µ* ∆*t* 2 = 8 la soluci´on es inestable y aparecen oscilaciones. Si definimos ∆*t* = 0*.*00125 entonces *α* = 0*.*5 y desaparecen las oscilaciones. Calculando la soluci´on, con ∆*x* = 0*.*05, se tiene *x*(18) = 0*.*85 y *u*(18) = 0*.*44957058.

(∆*x*)

1. Se define *θ* = 0*.*5, para utilizar el m´etodo de Crank-Nicolson en el algoritmo ecucalor.m. Para la soluci´on se tiene *x*(18) = 0*.*85 y *u*(18) = 0*.*44957077.

**Ejercicio** [**4.28.**](#_bookmark86) (a) El perfil de la soluci´on es el siguiente:



(b) M´aximo = 2*.*2036 en *x* = 0*.*75398, m´ınimo = *−*3*.*9834 en *x* = 3*.*2673,

1. Temperatura en *x* = *π* es de 3*.*9381 (*pos* = 51).

*−*

La diferencia con el instante inicial es de 3*.*9381 grados, por lo que se ha enfriado.

**Ejercicio** [**4.29.**](#_bookmark87)(a) Puesto que *α* = *µ* ∆*t* 2 = 1*,* la soluci´on es inestable y

(∆*x*)

aparecen oscilaciones.

(b) Se define *θ* = 1, para utilizar el m´etodo de Euler impl´ıcito en el algoritmo ecucalor.m. La representaci´on gr´afica muestra que han desaparecido las oscila- ciones. El m´aximo de la concentraci´on es *u* = 9*.*50785 y se encuentra en el km *x* = 3*.*6.

La regi´on a evacuar es *x*(25 : 48) que corresponde a los km 2*.*4 4*.*7.

*−*

Al cabo de *T* = 20 ya no hay concentraci´on superior a 5 y toda la regi´on es segura.

**Ejercicio** [**4.30.**](#_bookmark88)(a) El coeficiente de estabilidad es *α* ∆*t* 2 = 0*.*75 *>* 0*.*5, por tanto, la soluci´on presenta inestabilidades. El coeficiente m´aximo es de 0*.*1, con el cual se obtiene una temperatura m´axima al cado de 30 segundos de 86.526 grados.

(∆*x*)

En el punto *x* = 2*.*36 la temperatura es de 10*.*724 grados.

La diferencia entre la temperatura en el instante inicial y al cabo de 30 segundos es de 1*.*4201 grados, por lo que la temperatura en este punto se ha enfriado.

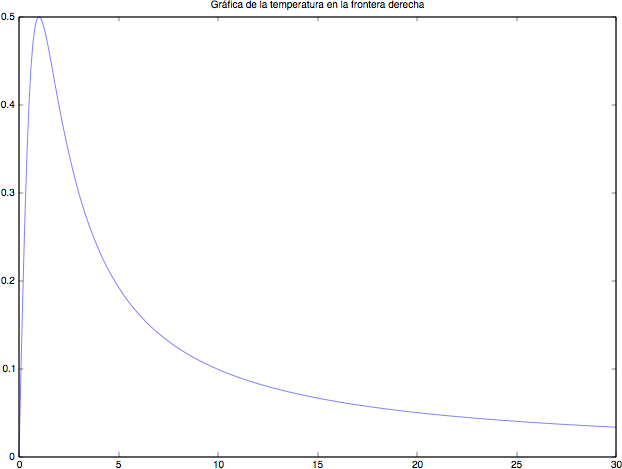
1. Al cabo de 2 minutos la temperatura m´axima es de 133*.*07 grados, la tem- peratura m´ınima es de 141*.*74 y la temperatura en el punto *x* = 2*.*36 es de 2*.*6361 grados.

*−*

La temperatura en la frontera derecha viene dada por la funci´on *g*(4*, t*) = *t* 2 ,

1+*t*

que tiene por gr´afica la Figura [5.16](#_bookmark111)



*{*TFront*}*

Figura 5.16: Temperatura en la frontera derecha.

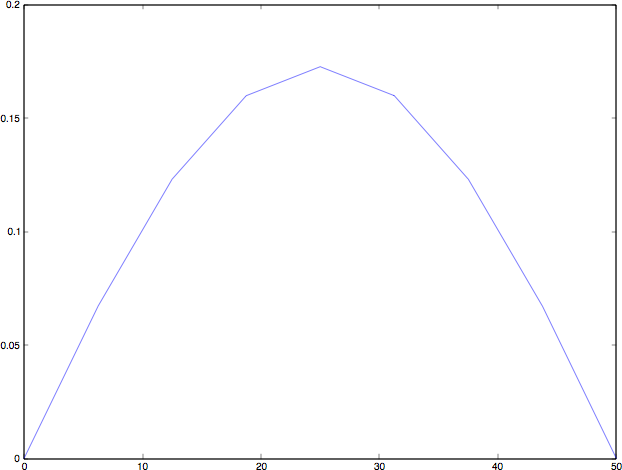
La temperatura m´axima en la frontera derecha es de 0*.*5 grados y se alcanza al cabo de 1 segundo.

1. La temperatura es de 39*.*589 grados y el error cometido es de 0*.*021110.

*−*

1. El m´aximo num´erico es de 113*.*76, mientras que el m´aximo de la soluci´on anal´ıtica es de 113*.*70, por lo que hemos cometido un error de 0*.*060658 grados.

**Ejercicio** [**4.31.**](#_bookmark90)La aproximaci´on de la funci´on se muestra en la Figura [5.17.](#_bookmark112) Alcanza valores por encima de 0.1, por lo que la placa no cumple la normativa.



*{*deflec*}*

Figura 5.17: Deflexi´on producida sobre la placa.

**Ejercicio** [**4.32.**](#_bookmark91) *(a)* Por el contorno *Tn*+1 = *tn*+1 y *Tn*+1 = 100 + 40*tn*+1.

1

Para *i* = 2*, . . . ,* 5 se tiene *Tn*+1 = (0*.*5 *−* 0*.*05 )*Tn*

*i*

*ri*

*i−*1

6

+ 0*.*05 *Tn* + 0*.*5*Tn* .

*ri*

*i*

*i*+1

 0 0 0 0 0 0  

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 5*/*12 | 1*/*12 | 1*/*2 | 0 | 0 | 0 |  |  | 0 |  |
|  | 0  0  0 | 6*/*14  0  0 | 1*/*14  7*/*161*/*16  0 | 1*/*2  1*/*2  8*/*18 | 0  0  1*/*18 | 0  | |  | 0  0  0 |  |
|  | 0 | 0 | 0 | 0 | 0 | 0 |  | 100 | + 10*tn* | +1 |

*B* = , *b* =



 1*/*2 

*tn*+1 

y



*A* = *Id*.

*(b) u*¯1 = (0*.*5*,* 21*.*6666*,* 41*.*42856*,* 61*.*25*,* 81*.*1112*,* 120)*.*

**Cap´ıtulo 6**

**Ap´endice A. Comandos de Octave/Matlab**

* 1. **Funciones**
     1. **Definici´on**

**Funciones en linea**. Se definen como cadenas de caracteres con el co- mando inline:

* + - * f = inline(’x.^2-3\*x+2’,’x’) Define la funci´on *x*2 3*x* + 2

*−*

como funci´on de *x*.

* + - * g = inline(’sin(x)’,’x’) Define sin(*x*) como funci´on de *x*.
      * h = inline(’y./sin(x)’,’x,y’) Define la funci´on *h*(*x*) = *y/* sin(*x*)

como funci´on de *x* e *y*.

Como la funci´on puede ser evaluada en vectores o matrices, debemos in- dicar que las operaciones , ˆ,*/* tienen que hacerse t´ermino a t´ermino. Es decir, **se escribe un punto antes de , ˆ y** */* cuando se vayan a evaluar vectores. Por ejemplo, si hubi´eramos definido:

*∗*

*∗*

* + - * f = inline(’x^2-3\*x+2’,’x’)

entonces el comando f(34) funcionar´ıa pero f([1,2,3,4]) no, ya que la multiplicaci´on [1*,* 2*,* 3*,* 4]2 = [1*,* 2*,* 3*,* 4] [1*,* 2*,* 3*,* 4] solo puede hacerse si es t´ermino a t´ermino.

*·*

El punto antes de estas operaciones no es necesario si estamos haciendo operaciones entre nu´meros 1*/*3*,* sin(3) log(5)*, . . .* etc. o si por ejemplo multiplicamos 3\*x ya que la multiplicaci´on por escalar ya se hace t´ermino a t´ermino.

*∗*

**Funciones an´onimas**. Las funciones an´onimas no usan cadenas de ca- racteres y en este caso se utiliza la arroba @ seguida por las variables de la funci´on entre par´entesis:

113

* f = @(x) x.*∧*2-3\*x+2;
* g = @(x) sin(x).
* h = @(x,y) y./sin(x);

*Matlab* recomienda el uso de funciones an´onimas frente a las inline.

Tambi´en podemos definir funciones utilizando el comando

function *. . .* endfunction, como en el siguiente ejemplo:

* function y = fun(x)
* y = x.*∧*2-1+exp(x);
* endfunction

Este tipo de funciones suele guardarse en un archivo .m (en el directorio de trabajo) de forma separada al que se le llama cada vez que queramos utilizar la funci´on.

* + 1. **Gr´aficas**

Es importante tener en cuenta que **Octave siempre dibuja vectores** de la siguiente forma:

1. Sean dos vectores de la misma longitud: uno *x* = (*x*1*, x*2*, . . . , xn*) con las coordenadas del eje *x* y otro *y* = (*y*1*, y*2*, . . . , yn*) con las coordenadas del eje *y*.
2. El comando plot(x,y) construye los pares (*xi, yi*) y dibuja la funci´on mediante **interpolaci´on** de los pares (*xi, yi*).

#### Ejemplos:

Para dibujar los puntos (*−*1*,* 2)*,* (1*,* 4)*,* (2*,* 8)*,* (3*,* 1) escribimos

* + x = [-1, 1, 2, 3];
  + y = [2, 4, 8, 1];
  + plot(x,y)

En el comando plot(x,y) podemos an˜adir **atributos** de color o forma a continuaci´on de la pareja de vectores. Algunos de estos atributos son:

|  |  |  |  |
| --- | --- | --- | --- |
| ’b’ | - blue | ’o’ | - dibuja c´ırculos |
| ’r’ | - red | ’x’ | - dibuja cruces |
| ’g’ | - green | ’-’ | - dibuja un recta entre dos puntos sucesivos. |
| ’k’ | - black |  |  |

Prueba por ejemplo con los siguientes comandos:

* + plot(x,y,’r’)
  + plot(x,y,’or’)
  + plot(x,y,’g-’)

*6.1. FUNCIONES* 115

Para dibujar la **gr´afica de una funci´on** en un intervalo [a,b], tenemos que crear un vector con numerosos puntos entre *a* y *b* con los comandos:

* [a:n:b] Crea un vector con entradas desde a hasta b, con pasos de n en n.
* linspace(a,b,n) Crea un vector con *n* entradas equiespaciadas entre a y b.

**Ejemplo:** La gr´afica de la funci´on *f* (*x*) = *x*2 + *x* + 1, en el intervalo [ 2*,* 2], se dibujar´ıa de la siguiente forma:

*−*

7

6

5

* + f = inline(’x.^2+x+1’,’x’);
  + I = [-2:0.01:2];

4

* + plot(I,f(I))

3

2

1

0

-2 -1 0 1 2

Prueba a dibujar *f* (*x*) usando el intervalo J=[-2:1:2] ¿Qu´e ocurre?

* + 1. **Presentaci´on de gr´aficas**

Para **dibujar varias funciones en la misma ventana** se van an˜adiendo al comando plot de dos en dos:

2

1

* f = inline(’x.\*sin(x)’,’x’); 0
* g = inline(’sin(x)./x’,’x’); -1
* I = [-5:0.01:5]; -2
* plot(I,f(I),I,g(I)) -3

-4

-5

-6 -4 -2 0 2 4 6

Despu´es de cada ”pareja de vectores”se pueden incluir atributos de color, forma, etc. Prueba con los siguientes comandos:

* I = [-5:0.25:5];
* plot(I,f(I),’b’,I,g(I)),’ok’)

Para **dibujar distintas gr´aficas en distintas ventanas** se usa el co- mando figure antes de cada plot:

* figure(1)
* f = inline(’x.\*sin(x)’,’x’);
* figure(2)
* g = inline(’sin(x)./x’,’x’);

Para **dibujar varias subventanas en la misma imagen** se utiliza el comando subplot. La idea es crear una matriz de *m* filas por *n* columnas en donde incluir las gr´aficas e ir indicando d´onde incluir cada una con un

´ındice. Por ejemplo,

2

* + f = inline(’x.\*sin(x)’,’x’); 1

0

* + g = inline(’sin(x)./x’,’x’); -1

-2

* + I = [-5:0.001:5];
  + subplot(2, 1, 1)
  + plot(I,f(I));
  + subplot(2, 1, 2)
  + plot(I,g(I));

-3

-4

-5

1

0.8

0.6

0.4

0.2

0

-0.2

-6 -4 -2 0 2 4 6

-0.4

-6 -4 -2 0 2 4 6

Podemos an˜adir despu´es de cada plot comandos para incluir un t´ıtulo, nombres en los ejes, leyendas, etc.

* f1 = inline(’cos(x)’,’x’);
* f2 = inline(’sin(x)’,’x’);
* I = [-5:0.01:5];
* plot(I,f1(I),I,f2(I))
* title(’FUNCIONES SENO Y COSENO’)
* legend(’COS(x)’,’SEN(x)’)
* xlabel(’EJE X’); ylabel(’EJE Y’)

1

0.5

0

EJE Y

-0.5

FUNCIONES SENO Y COSENO

-1

COS(X)

SEN(X)

-6 -4 -2 0 2 4 6

EJE X

* 1. **Comandos y funciones u´tiles**

clear all**,** close all**,** clf**,** clc **y** warning off

|  |  |
| --- | --- |
| clear all | Limpia las variables almacenadas hasta ese momento. |
| clear all | Cierra todas las gr´aficas abiertas hasta ese momento. |
| clf | Limpia (sin cerrar) las gr´aficas abiertas hasta ese momento. |
| clc | Limpia la ventana de comandos. |
| warning off | Si existen *warnings* no los muestra. |

Suele ser conveniente utilizar algunos de estos comandos como las primeras l´ıneas de nuestro c´odigo (en especial clear all).

**Salida de resultados. Los comandos** disp **y** printf

El comando disp(’...’) presenta por la ventana de comandos el texto incluido entre las comillas simples. Se puede combinar con alguna variable despu´es de una coma para presentar (en la l´ınea siguiente) un resultado. Por ejemplo,

* a = 2.33;
* disp(’La soluci´on es’), a

que muestra por pantalla:

*La soluci´on es*

*a* = 2*.*3300

El comando printf es m´as completo y permite combinar texto y variables de manera m´as eficiente. Un ejemplo de su uso es

* a = 2.33;
* printf(’La soluci´on es %f *\*n’,a)

que muestra por pantalla: *La soluci´on es a* = 2*.*3300

El s´ımbolo % indica la posici´on en la que queremos que se situ´e la variable *a*, la letra f le da un formato (ver la tabla para las distintas posibilidades) y el s´ımbolo *\*n hace que Octave realice un salto de l´ınea al finalizar.

|  |  |
| --- | --- |
| %f | nu´mero en coma flotante (decimal) |
| %e | notaci´on cient´ıfica |
| %g | como %f o %e pero sin ceros a la derecha |
| %i | nu´mero entero |
| %c | caracter |
| %s | cadena de caracteres (string) |

Otros comandos u´tiles que se pueden incluir dentro del comando son:

|  |  |
| --- | --- |
| %n | nueva l´ınea |
| %t | tabular |
| %b | elimina un espacio |

Se pueden incluir varias variables en el comando printf, teniendo en cuenta que el orden de los %, coincide con el orden de las variables separadas por comas al final del comando. Por ejemplo,

* niter = 9;z = log(10);
* printf(’Al cabo de %i iteraciones el resultado es %g *\*n’,niter,z)

devolviendo en la ventana de comandos: *Al cabo de 9 iteraciones el resultado es 2.30259.*

**Los comandos** max **y** min

Dado un vector cualquiera *u*, los comandos de Octave max y min devuelven los valores m´aximo y m´ınimo que contiene un vector.

Por ejemplo, si definimos el vector *u* = (2*,* 3*,* 4*,* 5*,* 6*,* 5*,* 4*,* 3*,* 2*,* 1), entonces max(u)

devuelve 6 y min(u) devuelve 1:

* u = [2,3,4,5,6,5,4,3,2,1]
* max(u)
* min(u)

Si adem´as incluimos dos par´ametros de salida de la forma

* + [M,posM] = max(u)
  + [m,posm] = min(u)

en las variables M y m se almacenan el valor m´aximo y el m´ınimo de u respecti- vamente, y en posM y posm se almacenan las posiciones en el vector u donde se alcanzan.

#### C´alculo de posiciones. El comando find.

Dado un vector cualquiera *u*, el comando find devuelve la posici´on (dentro del vector) en la que se encuentra uno o varios valores de *u*. Es importante tener en cuenta que en Octave *los vectores empiezan en la posici´on 1*.

Por ejemplo, si tomamos de nuevo el vector *u* = (2*,* 3*,* 4*,* 5*,* 6*,* 5*,* 4*,* 3*,* 2*,* 1) y escri- bimos:

* + find(u==6)
  + find(u==4)

el primer comando devuelve el valor (posici´on) 5 y el segundo comando devuelve los valores 3 y 7, que son en efecto las posiciones en las que se encuentra el 4 dentro de *u*.

El comando find permite tambi´en recuperar las posiciones donde los valores del vector verifican alguna condici´on *>, <, >*=*, <*=*,* =.

*∼*

Por ejemplo, el comando

* + find(u>=4)

devuelve las posiciones 3*,* 4*,* 5*,* 6 y 7.

**Forma alternativa de calcular posiciones.** En la mayor parte de los pro- blemas que trataremos, las bu´squedas de posiciones las realizaremos en vectores que representan los mallados:

*t*0 *< t*1 *< t*2 *< . . . < tm < . . . < tN−*1 *< tN ,*

que consideramos para resolver de manera num´erica los problemas que se nos plantean. En este caso, si tomamos como constante el paso de discretizaci´on *h* (constante), podemos recuperar todos los puntos del mallado dentro del vector *t*, en la posici´on *m*, mediante la f´ormula *tm* = *t*0 + (*m* 1)*h*, la cual nos sirve para calcular la posici´on *m* de la forma:

*−*

*m* = *tm − t*0 + 1*. h*

Por ejemplo, si queremos calcular la aproximaci´on obtenida a partir del m´etodo de Euler expl´ıcito en un intervalo [1*,* 3] en el instante *t* = 1*.*6 habiendo tomado un paso de discretizaci´on *h* = 0*.*1, tenemos que:

*m* = 1*.*6 *−* 1 + 1 = 7*.*

0*.*1

#### Funciones definidas a trozos. La funci´on Heaviside.

Las funciones definidas a trozos vamos a implementarlas usando la *funci´on Hea- viside* o funci´on escal´on:

*heaviside*(*x*) =

0*, x <* 0*,*

1*/*2*, x* = 0*,*



 1*, x >* 0*.*

Para utilizarla en Octave usamos la funci´on heaviside.m, la cual podremos combinar para construir todo tipo de funciones definidas a trozos.

Por ejemplo, prueba a dibujar las siguientes funciones en el intervalo [0*,* 5], usan- do Octave:

* f = inline(’heaviside(x-2)’,’x’)
* f = inline(’heaviside(x-2)-heaviside(x-4)’,’x’)
* f = inline(’exp(x).\*(heaviside(x-2)-heaviside(x-4))’,’x’)

**Ejercicio:** Dibuja con Octave la siguiente funci´on definida a trozos:



0*, x <* 3*,*

*x*2 8*,* 3 *x <* 4*,*

*— ≤*

 *e−x*+4*, x ≥* 4*.*

8

6

4

2

0

0 2 4 6 8

* 1. **Ecuaciones no lineales**

Si queremos aproximar la ra´ız de la ecuaci´on *f* (*x*) = 0, con una tolerancia errorper y en un nu´mero m´aximo de iteraciones, maxitera, definiremos la funci´on *f* como una funci´on inline (o an´onima) con el comando:

*>* fecu = inline(’...’,’x’) o *>* fecu =@(x) ...

y eligiremos alguno de los siguientes m´etodos:

**M´etodo de la biseccio´n**

Se eligen *a* y *b* de manera que *f* contenga una u´nica ra´ız en el intervalo [*a, b*] y se ejecuta el comando:

* [sol,itera] = metbiseccion(fecu,a,b,errorper,maxitera)

**M´etodo de Newton.**

Se define la derivada dfecu de *fecu* como otra funci´on inline, se elige el valor inicial x0 y se ejecuta el comando:

* [sol,itera] = metnewton1ec(fecu,dfecu,x0,errorper,maxitera)

**M´etodo de la secante.**

Se eligen los valores iniciales x0, x1, y se ejecuta el comando:

* [sol,itera] = metsecante(fecu,x0,x1,errorper,maxitera)

**M´etodo de ‘regula falsi”.**

Se eligen *a* y *b* de manera que *f* contenga una u´nica ra´ız en el intervalo [*a, b*] y se ejecuta el comando:

* 1. *SISTEMAS DE ECUACIONES NO LINEALES.ME´TODO DE NEWTON-RAPHSON.*121
  + [sol,itera] = metregulafalsi(fecu,a,b,errorper,maxitera)

En la **salida** de todos ellos, sol contiene la aproximaci´on buscada e itera es el nu´mero de iteraciones que ha tardado el m´etodo en alcanzar sol.

**M´etodo del punto fijo y m´etodo de Aitken.**

Si queremos resolver el problema de punto fijo, *g*(*x*) = *x*, primeramente se define

*g* como una funci´on inline (o an´onima) con el comando:

* g = inline(’...’,’x’) o
* g =@(x) ...

Se elige el valor inicial *x*0 y se ejecuta uno de los comandos:

* [sol,itera] = metpuntofijo(g,x0,errorper,maxitera)
* [sol,itera] =metodoaitken(g,x0,errorper,maxitera)

dependiendo de si queramos aplicar el m´etodo del punto fijo o el m´etodo de aceleraci´on de Aitken, respectivamente.

En la **salida** de ambos, sol contiene la aproximaci´on buscada, e itera el nu´mero de iteraciones que ha tardado el m´etodo en alcanzar sol.

* 1. **Sistemas de Ecuaciones no lineales. M´etodo de Newton-Raphson.**

Sea el sistema de ecuaciones no lineales:





*f*1(*x*1*, x*2*, . . . , xn*) = 0*,*

.

*fn*(*x*1*, x*2*, . . . , xn*) = 0*.*

Para encontrar aproximaciones de un vector soluci´on por el m´etodo de Newton-

Raphson, se toma como dato inicial el vector columna **x**

= (*x*(0)*, . . . , x*(0))*T ∈*

R*n*, y creamos 2 archivos .m auxiliares:

**0** 1 *n*

1. Archivo fecusistema.m que contiene al vector columna **f** = (*f*1*, . . . , fn*)*T* ,
2. Archivo jacobiana.m que contene la matriz Jacobiana *J***f** ,

donde las variables *x*1*, . . . , xn* se denotan por *x*(1)*, . . . , x*(*n*). y se ejecuta el comando:

* + [vectorsol,itera] = metnewtonsistema(@fecusistema,@jacobiana,...
  + vectorx0,errorper,maxitera)

**Ejemplo:** Sea el sistema de ecuaciones no lineales:

*x*2 *−* 10*x* + *y*2 + 8 = 0*, xy*2 + *x −* 10*y* + 8 = 0*.*

Tomando como vector semilla **x0** = (0*.*5*,* 0*.*5)*T* , tenemos que

(*x*2 *−* 10*x* + *y*2 + 8 (2*x −* 10 2*y*

*f* =

2

*xy* + *x −* 10*y* + 8

y *J***f** (**x**) =

*y*2 + 1 2*xy −* 10

por lo que creamos los archivos fecusistema.m y jacobiana.m de la siguiente forma:

Archivo fecusistema.m

* function F=fecusistema(x,y)
* F(1,1) = x*∧*2-10*∗*x+y*∧*2+8;
* endfunction
* F(2,1) = x*∗*y*∧*2+x-10*∗*y+8;

Archivo jacobiana.m

. *>* function J=jacobiana(x,y)

* J(1,1) = 2*∗*x-10;
* J(1,2) = 2*∗*y;
* J(2,1) = y*∧*2+1;
* J(2,2) = 2*∗*x*∗*y-10;
* endfunction

Ahora el sistema se resuelve mediante los comandos:

* + - vectorx0 = [0.5;0.5]; errorper=1.e-6; maxiter=1000;
    - [vectorsol,itera] = metnewtonsistema(@fecusistema,@jacobiana,...
    - vectorx0,errorper,maxitera)
  1. **Problemas de Valor Inicial**

Para resolver un Problema de Valor Inicial (PVI) de la forma

*y′* = *f* (*t, y*)*, t ∈* [*t*0*, tN* ]

*y*(*t*0) = *y*0*,*

primeramente definimos la funci´on *f* (*t, y*) en las variables *t*, *y*, como una funci´on inline o an´onima, el intervalo en el que se presenta el problema [*t*0*, tN* ], y la condici´on inicial *y*0 como:

* f = inline(’...’,’t,y’)
* intiempo = [t0,tN]
* y0=valorini = ...

Despu´es se especifica el nu´mero de intervalos, npasos, del mallado utilizado, es decir, el nu´mero de intervalos en los que se divide el intervalo [*t*0*, tN* ], calculado a partir del paso de discretizaci´on *h* (que vamos a tomar siempre constante). En otras palabras:

* npasos =

tN-t0 h

y resolvemos el PVI por medio de alguno de los siguientes m´etodos:

**M´etodo de Euler Expl´ıcito**

* [solt,soly] = eulerexplicito(f,intiempo,valorini,npasos)

**M´etodo de Euler Impl´ıcito**

* [solt,soly] =eulerimplicito(f,intiempo,valorini,npasos)

**M´etodo de Crank-Nicolson**

* [solt,soly] = cranknicolson(f,intiempo,valorini,npasos)

**M´etodo de Heun**

*>*[solt,soly] = heun(f,intiempo,valorini,npasos)

**M´etodo de Simpson**

* [solt,soly] = rungekuttao3(f,intiempo,valorini,npasos)

Para todos los m´etodos, la **salida** consiste en el vector solt, que contiene los *npasos* + 1 puntos del mallado en los que se ha dividido el intervalo [*t*0*, tN* ], y en el vector soly, que contiene las aproximaciones a la soluci´on exacta *y*(*t*) en los puntos del mallado.

Una vez ejecutado el m´etodo podemos dibujar la aproximaci´on mediante el co- mando plot(solt,soly)

El algoritmo heun.m contiene el m´etodo de Heun, mientras que el algoritmo rungekuttao3.m contiene el m´etodo de Runge-Kutta de orden 3, denominado de Simpson, dado por la tabla:

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |
| 1*/*2 | 1*/*2 | 0 | 0 |
| 1 | *−*1  1*/*6 | 2  4*/*6 | 0  1*/*6 |

* 1. **Problemas de Contorno**
     1. **Problemas de transporte estacionario 1-dimensionales**

#### Coeficientes constantes y condiciones frontera tipo Dirichlet.

El siguiente problema de transporte:



*−Du′′*(*x*) + *V u′*(*x*) + *Qu*(*x*) = *f* (*x*)*, x ∈* (*a, b*) *u*(*a*) = *ua,*

 *u*(*b*) = *ub,*

se resuelve mediante el comando:

* [xh,uh] = bvpdirichlet(a,b,numeronodos,D,V,Q,f,ua,ub)

en donde:

-D, V, Q: constantes;

-f: funci´on inline o an´onina;

-numeronodos =N+2=1 + *b−a* : nu´mero de nodos, siendo *h* es el paso de discretizaci´on;

*h*

-xh: vector que contiene los nodos de la discretizaci´on xh = (*x*1*, . . . , xN*+2);

-uh: vector que contiene las aproximaciones num´ericas uh = (*u*1*, . . . , uN*+2);

#### Coeficientes no constantes y condiciones frontera generales.

El siguiente problema de transporte:



*−D*(*x*)*u′′*(*x*) + *V* (*x*)*u′*(*x*) + *Q*(*x*)*u*(*x*) = *f* (*x*)*, x ∈* (*a, b*) *c*11*u′*(*a*) + *c*12*u*(*a*) = *ua,*



*c*21*u′*(*b*) + *c*22*u*(*b*) = *ub,*

se resuelve mediante el comando:

* [x,u] = bvp2cvrobinup(a,b,N,D,V,Q,f,c11,c12,...

*>*c21,c22,ua,ub,esquema)

donde:

-D, V, Q: funciones inline o an´onimas;

-f: funci´on inline o an´onina;

-N: *nu´mero de nodos internos*,

es decir, *N* = *b−a* 1 en donde *h* es el paso de discretizaci´on;

*h*

*−*

-esquema : ’C’ para usar f´ormulas centradas;

-’U’ para usar un esquema, a contracorriente (*upwind*), para aproximar el t´ermino convectivo;

-xh: vector que contiene los nodos espaciales, xh = (*x*1*, . . . , xN*+2);

-uh: vector que contiene las aproximaciones num´ericas uh = (*u*1*, . . . , uN*+2);

* + 1. **Problemas de difusi´on evolutiva 1-dimensionales**

El siguiente Problema de Valor Inicial y de Contorno (PVIC):

**** *∂u*

****

****

*∂t* = *c∂x*2 + *f* (*x, t*)*, ∀* (*x, t*) *∈* (*a, b*) *×* (0*, T* )*,*

*u*(*x, t*) = *g*(*x, t*)*, x* = *a, x* = *b,*

*∂*2*u*

*u*(*x,* 0) = *u*0(*x*)*, x ∈* (*a, b*)*,*

se resuelve mediante el comando:

* [x,uf] = ecucalor(c,intespacio,intiempo,...
* pasosespacio,pasostiempo,theta,u0,g,f)

donde:

-intespacio = [a,b] es el intervalo espacial;

-intiempo = [0,T] es el intervalo temporal;

-pasosespacio y pasostiempo son el nu´mero de intervalos espaciales y temporales;

-theta = *θ* indica el *θ*-m´etodo de la parte temporal (*θ* = 0 EE, *θ* = 0*.*5 CN, *θ* = 1 EI);

-c: es una constante positiva;

-u0,g,f: son funciones inline o an´onimas.

En la **salida**, Octave almacena en el vector x = (*x*1*, x*2*, . . . , xN*+2), los puntos

del mallado espacial y en uf = (*uT , uT , . . . , uT*

), los valores de la aproxima-

1 2

ci´on buscada en el instante *t* = *T* .

*N* +2

∆*t* 1

**Observaci´on.** El m´etodo es **estable** s´ı y s´olo s´ı *⇐⇒ c*(1 *−* 2*θ*)(∆*x*)2 *≤* 2 *,*

donde ∆*t* y ∆*x*, son los taman˜os de discretizaci´on en tiempo y espacio, respec- tivamente.

* + 1. **Problemas de transporte estacionario 2-dimensionales**

El siguiente problema de transporte (tipo Poisson):

f

*−*∆*u*(*x, y*) = *f* (*x, y*)*, ∀* (*x, t*) *∈* Ω(*a, b*) *×* (*c, d*)*, u*(*x, y*) = *cc*(*x, y*)*, ∀* (*x, y*) *∈ ∂*Ω*,*

se resuelve mediante los comandos:

* [u,x,y] = ecupoisson(a,b,c,d,dx,dy,f,cc)
* [u,x,y,error] = ecupoisson(a,b,c,d,dx,dy,f,cc,solexac)

dependiendo de si la soluci´on exacta es conocida o no. Los argumentos son los siguientes:

-dx es el taman˜o de discretizaci´on en *x*;

-dy es el taman˜o de discretizaci´on en *y*;

-f es una funci´on inline, con el forzamiento externo;

-cc es una funci´on inline, con la condici´on de contorno;

-solexac es una funci´on inline, con la soluci´on exacta (si es conocida);

-error muestra el *error nodal relativo* entre *u* y *solexac*

(error nodal = m´ax*i,j |u*(*xi, yj*) *− uij|/* m´ax*i,j |u*(*xi, yj*)*|*).

Para mostrar la soluci´on utilizamos el comando mesh(x,y,u).

**Cap´ıtulo 7**

**Ap´endice B. Fo´rmulas en diferencias finitas**

* 1. **Fo´rmulas en diferencias finitas 1-dimensionales**

Sea el intervalo Ω = [0*, L*] y tomemos un mallado unidimensional *{xi}i*=1*,...,N*+1 con *x*1 = 0 y *xN*+1 = *L*, donde consideramos un paso de discretizaci´on constante *h*.

Si nos centramos en un punto *xi* del mallado

*xi−*1 *xi xi*+1

tenemos las siguientes **f´ormulas en diferencias finitas**:

1. Para aproximar el valor de *u′*(*x*) = *du* (*x*) en el nodo *x* = *x*

*dx i*

tenemos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| F´ormula | Error de Truncamiento | Esquema | | |
| **Progresiva** *u′*(*x* ) *≈ ui*+1 *− ui*  *i h*  **Regresiva** *u′*(*x* ) *≈ ui − ui−*1  *i h*  **Centrada** *u′*(*x* ) *≈ ui*+1 *− ui−*1  *i* 2*h* | *Etr* = *O*(*h*) | *−*1 | +1 | *h* |
| *Etr* = *O*(*h*) | *−*1 +1 |  | *h* |
| *Etr* = *O*(*h*2) | *−*1 | +1 | 2*h* |

1. Para aproximar el valor de *u′′*

*d*2*u*

(*x*) = *dx*2 (*x*) en el nodo *x* = *xi* tenemos:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| F´ormula | | | Error de Truncamiento | Esquema | | | |
| **Centrada** | *u′′*(*xi*) *≈* | *ui−*1 *−* 2*ui* + *ui*+1  *h*2 | *Etr* = *O*(*h*2) | +1 | *−*2 | +1 | *h*2 |

127

**7.2. F´ormulas en diferencias finitas bidimensio- nales**

Sea un dominio rectangular Ω = [0*, Lx*] [0*, Ly*] y el mallado (*xi, yj*) con pasos de discretizaci´on constantes *h* = ∆*x* y *k* = ∆*y*. Si nos centramos en un nodo *C* = (*xi, yj*) con los puntos “vecinos”: *N* , *S*, *E* y *W* , como muestra la Figura, tenemos las siguientes **f´ormulas en diferencias finitas**:

*× { }*

*N* =(*xi, yj*+1)

*C* = (*xi, yj* )

(*xi−*1*, yj* ) = *W*

*E* = (*xi*+1*, yj* )

*S* = (*xi, yj−*1)

1. Para aproximar las derivadas parciales de primer orden:

*∂u ∂u*

*ux*(*x, y*) = *∂x* (*x, y*) y *uy*(*x, y*) = *∂y* (*x, y*) en *C* = (*xi, yi*), tenemos:

|  |  |  |
| --- | --- | --- |
| F´ormula | Aproximaci´on *ux* | Error |
| **Progresiva Regresiva**  **Centrada** | *uE − uC*  *ux*(*C*) *≈ h*  *uC − uW*  *ux*(*C*) *≈ h*  *uE − uW*  *ux*(*C*) *≈* 2*h* | *Etr* = *O*(*h*) *Etr* = *O*(*h*)  *Etr* = *O*(*h*2) |

|  |  |  |
| --- | --- | --- |
| F´ormula | Aproximaci´on *uy* | Error |
| **Progresiva**  **Regresiva Centrada** | *uN − uC*  *uy*(*C*) *≈ k*  *uC − uS*  *uy*(*C*) *≈ k*  *uN − uS*  *uy*(*C*) *≈* 2*k* | *Etr* = *O*(*k*)  *Etr* = *O*(*k*) *Etr* = *O*(*k*2) |

1. Para aproximar *uxx*(*x, y*) =

*∂*2*u*

*∂x*2 , *uyy*(*x, y*) =

*∂*2*u*

*∂y*2 , *uxy*(*x, y*) =

*∂*2*u*

en

*∂x∂y*

* 1. *FO´RMULAS EN DIFERENCIAS FINITAS BIDIMENSIONALES* 129

*C* = (*xi, yi*):

|  |  |  |
| --- | --- | --- |
| F´ormula | Aproximaci´on *ux* | Error |
| **Centrada**  **Centrada Centrada** | *uW −* 2*uC* + *uE*  *uxx*(*C*) *≈ h*2  *uS −* 2*uC* + *uN*  *uyy*(*C*) *≈ k*2  *uNE − uSE* + *uSW − uNW*  *uxy*(*C*) *≈* 4*hk* | *Etr* = *O*(*h*2) *Etr* = *O*(*k*2)  3 3  *Etr* = *O*( *h , h*2*, hk, k*2*, k* )  *k h* |

**Bibliograf´ıa**

1. A. Quarteroni, F. Saleri.. *C´alculo cient´ıfico con MATLAB y Octave.* Sprin- ger, 2006.
2. E. Schiavi, A. I. Mun˜oz y C. Conde. *M´etodos Matem´aticos para los grados en ingenier´ıa. Primera parte: Teor´ıa.* Dykinson S.L., 2012.
3. E. Schiavi, A. I. Mun˜oz y A. Nolla. Los c´odigos empleados aparecen desa- rrollados el texto: *C´odigos en Octave/Matlab utilizados en el libro M´etodos Matem´aticos aplicados a la Ingenier´ıa. Ejercicios y problemas resueltos*. Pu- blicados en htpps://burjcdigitalurjc.es.

131