

Ampliación de Análisis Numérico

Luis Alvarez León y Javier Sánchez Pérez

Departamento de Informática y Sistemas

Universidad de Las Palmas

Campus de Tafira

35017 Las Palmas, España

Tf: 45.87.10/08

Email: {lalvarez/jsanchez}@dis.ulpgc.es

www: <http://www2.dis.ulpgc.es/~lalvarez/~jsanchez>

Índice General

1	INTRODUCCION	2	
2	METODOS DE OPTIMIZACION PARA LA RESOLUCION DE SISTEMAS DE ECUACIONES.	3	
2.1	PRELIMINARES.	3	
2.1.1	ALGUNOS RESULTADOS ELEMENTALES SOBRE MATRICES.	3	
2.1.2	ALMACENAMIENTO MORSE DE UNA MATRIZ	4	
2.2	MINIMIZACION DE FUNCIONES Y RESOLUCION DE SISTEMAS DE ECUACIONES	5	
2.3	METODO DE GRADIENTE PASO FIJO	5	
2.4	METODO DE GRADIENTE PASO OPTIMO	7	
2.5	METODO DE GRADIENTE CONJUGADO	8	
2.6	PRACTICA 1. METODO DE GRADIENTE CONJUGADO	9	
2.7	PRECONDICIONAMIENTO DEL ALGORITMO DE GRADIENTE CONJUGADO	9	
2.7.1	PRECONDICIONAMIENTO A TRAVES DE LA TRIDIAGONAL DE A.	10	
2.7.2	PRECONDICIONAMIENTO POR FACTORIZACION INCOMPLETA DE CHOLESKY	11	
2.8	PRACTICA 2. PRECONDICIONAMIENTO	11	
2.9	EXTENSION DEL METODO DE GRADIENTE CONJUGADO A MATRICES CUALESQUIERA. METODO DE LA ECUACION NORMAL.	12	
3	TRIANGULACIONES DE DOMINIOS PLANOS	12	
3.1	NOCIONES GEOMETRICAS BASICAS SOBRE TRIANGULOS	12	
3.2	TRIANGULACIONES	14	
3.3	ALMACENAMIENTO DE UNA TRIANGULACION	14	
3.4	REFINAMIENTO UNIFORME DE TRIANGULACIONES	15	
3.5	TRIANGULACIONES DE DELAUNY DE DOMINIOS PLANOS	15	
3.5.1	TRIANGULACIONES DE DELAUNY Y DIAGRAMAS DE VORONOI	16	
3.5.2	PROCEDIMIENTO PARA INCLUIR UN PUNTO EN UNA TRIANGULACION DE DELAUNY.	18	
3.5.3	TRIANGULACIONES DE DELAUNY CON RESTRICCIONES	20	
3.5.4	DISCRETIZACION INICIAL DE UN DOMINIO	21	
3.5.5	ESQUEMA GENERAL DE UN ALGORITMO DE TRIANGULACION SEGUN EL CRITERIO DE DELAUNY.	22	
3.6	PRACTICA 3. TRIANGULACIONES DE DELAUNY	25	
3.7	OPTIMIZACION DEL COSTO COMPUTACIONAL	25	
4	TRIANGULACIONES DE DOMINIOS TRIDIMENSIONALES.	26	
4.1	NOCIONES GEOMETRICAS BASICAS SOBRE TETRAEDROS	26	
4.2	TRIANGULACIONES	27	
4.3	ALMACENAMIENTO DE UNA TRIANGULACION	27	
4.4	TRIANGULACIONES DE DELAUNY DE UN DOMINIO 3-D	28	
5	EL METODO DE ELEMENTOS FINITOS	31	
5.1	INTRODUCCION	31	
5.2	FORMULACION VARIACIONAL DEL PROBLEMA MODELO	34	
5.3	DISCRETIZACION POR ELEMENTOS FINITOS DE ORDEN 1.	35	
5.4	ASPECTOS NUMERICOS EN LA CONSTRUCCION DEL SISTEMA DE ECUACIONES	36	

5.5	PRACTICA 4. METODO DE ELEMENTOS FINITOS	38
5.6	TRIANGULO DE REFERENCIA Y FUNCIONES DE FORMA	39
5.7	ELEMENTOS FINITOS CURVOS	40
5.8	ELEMENTOS FINITOS EN DIMENSIONES DISTINTAS DE 2.	41
5.9	APLICACION DEL METODO ELEMENTOS FINITOS A PROBLEMAS DE EVOLUCION	41
6	SISTEMAS DE ECUACIONES DIFERENCIALES LINEALES	42
6.1	INTRODUCCION	42
6.2	SISTEMAS LINEALES	43
6.2.1	DIAGONALIZACION DE A	47
6.2.2	RESOLUCION DEL CASO MAS GENERAL EN DIMENSION 2	50
6.2.3	CLASIFICACION DE LAS ORBITAS	53
7	SISTEMAS DE ECUACIONES DIFERENCIALES NO LINEALES	58
7.1	PUNTOS CRITICOS	58
7.2	LINEALIZACION	58
7.3	CAMPO DE DIRECCIONES	60
8	ANALISIS NUMERICO DE LOS SISTEMAS	67
8.1	METODOS DE UN PASO	68
8.1.1	METODO DE EULER	68
8.1.2	METODO DE EULER MEJORADO	69
8.1.3	METODO DE RUNGE-KUTTA	70
8.2	METODOS MULTIPASO	71
8.2.1	ESTUDIO DEL ERROR	72
8.3	INTRODUCCION AL PROGRAMA PHASER	74
8.4	PRACTICA 5.	75
8.5	PRACTICA 6.	75
9	BIBLIOGRAFIA BASICA	75
10	APENDICE A: Resumen de los comandos de UNIX	75
11	APENDICE B: Resumen del procesador de texto vi.	76

INTRODUCCION

El presente documento es el texto de referencia de los contenidos de una asignatura de Ampliación de Análisis Numérico orientada a una titulación universitaria en Informática. El enfoque que se pretende dar a la asignatura, en el contexto de unos estudios en informática, hace especial énfasis en los aspectos algorítmicos y numéricos de los temas que vamos a tratar a lo largo del curso. Sin olvidar el rigor matemático en la exposición, se ha intentado minimizar la presentación del aparato matemático necesario

para abordar los diferentes temas que se irán presentando a lo largo del curso. En los libros de referencia presentes en la bibliografía recomendada, se desarrollan con mucha más profundidad de la aquí expuesta, las teorías matemáticas subyacente a los métodos y algoritmos aquí propuestos. Invitamos al lector interesado a acudir a estas referencias para realizar un estudio más profundo de los temas aquí expuestos.

Durante el desarrollo de la asignatura se realizarán una serie de prácticas en ordenador. El lenguaje de programación que se utilizará es el *C*. Se ha elegido este lenguaje por ser un lenguaje de uso muy extendido en la actualidad. Además, habitualmente, en la asignatura de Análisis Numérico básico que precede a ésta, se utiliza el Fortran, con lo que de este modo, el alumno puede tener una visión comparativa de los 2 lenguajes de programación con respecto al cálculo científico. La asignatura posee un directorio en la máquina *serdis.dis.ulpgc.es* del Centro de Cálculo del Departamento de Informática y Sistemas, donde se pueden encontrar programas ejemplos y librerías de funciones que se irán utilizando durante las prácticas. En el texto aparecen explicadas las prácticas de laboratorio que deberán realizarse a lo largo de la asignatura.

Para el buen seguimiento de la asignatura, se recomienda tener cierta soltura en el manejo de los conceptos elementales del análisis numérico, análisis matemático, métodos matemáticos, álgebra lineal, y en la programación de algoritmos.

La materia expuesta en esta documentación, está programada para ser impartida en un cuatrimestre a razón de 4 horas/semana en el aula y 2 horas/semana en el laboratorio informático, lo que hace un total de aproximadamente de 60 horas en aula (3 créditos teóricos) y 30 horas de laboratorio (2 créditos prácticos).

Criterios de Evaluación

La evaluación final de la asignatura constará de los siguientes apartados:

- A1. Prácticas de la asignatura (Hasta 3 puntos)
- A2. Participación en clase (Hasta 6 puntos)
- A3. Examen Escrito (Hasta 10 puntos)

$$\text{Nota final: } A1 + A2 + A3\left(1 - \frac{A1+A2}{10}\right)$$

La idea de utilizar esta fórmula aparentemente compleja para calcular la nota final es la siguiente: por un lado, los puntos obtenidos en los apartados A1 y A2 están consolidados, es decir, el alumno que obtenga, por ejemplo, un 2 en el apartado A1, y un 3 en el apartado A2 tiene como mínimo un 5 en la nota final, por otro lado el examen escrito se utiliza para puntuar en la nota que le resta al alumno para llegar al 10. Es decir, si en el ejemplo

anterior el alumno tuviese en el examen escrito un 4, su nota final sería $2 + 3 + 4(0.5) = 7$.

Valoración del criterio A1 : Se constituirán grupos de trabajo de 4 alumnos, repartidos en 2 puestos de trabajo. Cada grupo presentará, en el momento de la defensa de la práctica, una pequeña planificación de la práctica (como máximo dos páginas), incluyendo:

1. Tareas realizadas por cada puesto de trabajo
2. Estimación del tiempo dedicado por cada persona a la realización de la práctica
3. Parte de incidencias de los principales errores encontrados al depurar el programa, indicando una estimación del tiempo utilizado para localizar y depurar cada error.

Las prácticas 1 y 2 se presentarán conjuntamente con una planificación única para las dos. Las prácticas 5 y 6 se realizarán de forma independiente por cada puesto de trabajo y no se requiere la presentación de una planificación.

La asistencia a las clases prácticas de laboratorio es obligatoria. Las prácticas deberán presentarse obligatoria e inexcusablemente antes del examen escrito siguiendo el calendario que se fijará durante el curso. 2 faltas no justificadas a las clases de laboratorio, o la no entrega de las prácticas durante el calendario previsto, suponen la exclusión del alumno de las convocatorias de exámenes de la asignatura. Los alumnos que asistan regularmente a las clases de laboratorio y entreguen las prácticas siguiendo el calendario previsto obtendrán una puntuación de 2 puntos. Presentar las prácticas fuera de plazo (pero antes del examen escrito de la asignatura) tiene una penalización de 1 punto. No asistir a una clase práctica tiene una penalización de 0.5 puntos. Los alumnos que complementen las prácticas obligatorias con algún trabajo práctico opcional podrán obtener hasta un punto suplementario.

Valoración del criterio A2 : Los alumnos se constituirán en grupos de 4 personas. Durante el desarrollo de las clases el profesor propondrá algunos problemas (o preguntas) para que sean resueltas por los diferentes grupos. Se fijará un tiempo para la contestación a las preguntas, pasado el cual, todos los grupos entregarán las respuestas encontradas. A continuación el profesor realizará algunas preguntas a los grupos sobre las respuestas. Todos los miembros del grupo son responsables de las respuestas presentadas. Si un alumno no contesta adecuadamente a las preguntas del profesor, todo el grupo pierde la puntuación de esta pregunta. La puntuación final en el apartado A2 viene dada por:

$$A2 = 4 \cdot \frac{N^{\circ} \text{ Puntos Contestados}}{N^{\circ} \text{ Puntos Total Asignatura}}$$

Valoración del criterio A3 : Un examen escrito que se puntúa de 0 a 10.

Agradecimientos

Los autores quieren agradecer a Carlos Herrera Canino la colaboración prestada en la elaboración de este documento. Su excelente trabajo como cronista de lo acontecido durante la impartición de la asignatura Análisis Numérico Aplicado en el curso 1996/97 fue el embrión inicial del que salió la idea de realizar este documento.

METODOS DE OPTIMIZACION PARA LA RESOLUCION DE SISTEMAS DE ECUACIONES.

PRELIMINARES.

En este capítulo vamos a recordar brevemente algunos resultados sobre matrices que utilizaremos posteriormente. Presentaremos también un tipo especial de almacenamiento de matrices denominado morse, que es el que utilizaremos en este curso

ALGUNOS RESULTADOS ELEMENTALES SOBRE MATRICES.

Sea A una matriz cuadrada, denotaremos por $|A|$ al determinante de A , y por $\|A\|$ a la norma de la matriz. Siempre que no se diga lo contrario, entenderemos que $\|A\|$ es la norma asociada a la norma vectorial euclídea.

Definición 1 *A continuación se dan algunas definiciones de conceptos relacionados con matrices. A veces un concepto se puede definir de varias formas equivalentes*

- λ es un autovalor de A si existe un autovector x tal que $Ax = \lambda x$
- λ es un autovalor de A si es una raíz del polinomio característico $P(\lambda) = |A - \lambda I|$
- El radio espectral de una matriz es $\rho(A) = \max_i \{|\lambda_i| : \lambda_i \text{ autovalor de } A\}$
- A es definida positiva si para todo vector x no nulo $x^T Ax > 0$
- A es definida positiva si todos sus menores principales son positivos.
- Se denomina condicionamiento de una matriz A asociado a una norma $\|\cdot\|$ al valor $\chi(A) = \|A\| \|A^{-1}\|$

Teorema 1 *Dadas A, B matrices cuadradas, se verifican las siguientes propiedades*

- $|AB| = |A| |B|$

- $|A| = |A^T|$
- A posee inversa A^{-1} si sólo si $|A| \neq 0$
- $\|Ax\| \leq \|A\| \|x\|$ para cualquier vector x .
- $\|AB\| \leq \|A\| \|B\|$
- $|A| = \prod \lambda_i$ donde λ_i son los autovalores de A .
- $A^n \rightarrow 0$ cuando $n \rightarrow \infty$ si sólo si $\rho(A) < 1$

Teorema 2 Dada una matriz A simétrica se verifican las siguientes propiedades

- A posee una base ortonormal de autovectores
- $\|A\| = \rho(A)$
- A es definida positiva si sólo si todos sus autovalores son positivos.
- $\chi(A) = \frac{\max_i |\lambda_i|}{\min_i |\lambda_i|}$

ALMACENAMIENTO MORSE DE UNA MATRIZ

Durante este curso, trabajaremos esencialmente con matrices simétricas, definidas positivas, y que poseen un gran número de ceros en la matriz. Para explotar computacionalmente este hecho, se utiliza un almacenamiento especial de matrices que se denomina almacenamiento morse donde sólo se almacenan los elementos no-nulos de la matriz. Además, este almacenamiento está especialmente diseñado para que la operación matriz por vector (que es la operación básica que vamos a necesitar en el futuro) sea lo más compacta y sencilla posible. Vamos a distinguir el almacenamiento en disco de la matriz en forma de fichero, y el almacenamiento en memoria interna (RAM) del ordenador. Para el almacenamiento en RAM, utilizaremos una estructura en C definida por

```
typedef struct{
    int dimension;
    int nonulos;
    float *valor;
    int *col;
    int *fil;
}morse;
```

donde *dimension* indica la dimensión de la matriz, *nonulos* indica el número de elementos diferentes de cero, **valor* es un puntero donde se almacenan los valores no-nulos de la matriz, **col* es un puntero donde se almacena la columna que ocupa en la matriz cada elemento no-nulo, y por último **fil* es un puntero que determina los tramos del vector **valor* que corresponden a cada fila. Veamos con un sencillo ejemplo un almacenamiento morse de una matriz.

Ejemplo 1 Sea $A = \begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix}$ los campos de la estructura morse serían:

```
dimension = 2
nonulos = 3
valor[0] = 1
valor[1] = 2
valor[2] = 3
col[0] = 0
col[1] = 0
col[2] = 1
fil[0] = 0
fil[1] = 1
fil[2] = 3
```

la información del vector *fil* determina las filas de la siguiente forma: la primera fila corresponde al conjunto de posiciones del vector *valor* que se encuentra entre *fil[0]* y *fil[1]*, de la misma forma, la segunda fila se encuentra entre *fil[1]* y *fil[2]*, etc..

El almacenamiento en disco se realiza a través de un fichero de la siguiente forma: En la primera fila del fichero se escriben la dimensión y el número de elementos no-nulos. A continuación se escribe el vector *valor*, el vector *col* y el vector *fil*. Por ejemplo la forma del fichero que almacena la matriz anterior es:

```
2 3
1.
2.
3.
0
0
1
0
1
3
```

Dentro del fichero *ana.h* existen funciones para leer y escribir matrices morse. En el programa ejemplo *morse1.c* se utilizan estas funciones para crear en disco la matriz morse del ejemplo anterior y un vector. En el programa ejemplo *morse2.c* se utilizan estas funciones para leer desde disco una matriz morse y un vector.

Como ya se ha mencionado anteriormente, el almacenamiento morse, además de economizar memoria está especialmente diseñado para realizar la operación de multiplicar una matriz por un vector. A continuación se escribe una función en C que realiza la multiplicación de una matriz A por un vector x y devuelve la solución en el vector Ax

```
float *MxV(A, x, Ax)
morse A;
float *x, *Ax;
{
```

```

int i, j;
for(i = 0; i < A.dimension; i++) {
    Ax[i] = 0
    for(j = A.fil[i]; j < A.fil[i+1]; j++)
        Ax[i] += A.valor[j] * x[A.col[j]];
}
return(Ax);
}

```

Nótese que el algoritmo es muy compacto. Por otro lado, en cada fila, los elementos de la matriz no necesitan estar ordenados por columnas.

MINIMIZACION DE FUNCIONES Y RESOLUCION DE SISTEMAS DE ECUACIONES

La palabra optimización está relacionada con la búsqueda de mínimos de funciones que se suelen denominar energías. Es decir, optimizar es buscar la mejor solución a un problema, que en términos matemáticos se expresa como el mínimo de una función.

Vamos a ver en primer lugar como la resolución de un sistema de ecuaciones $Au = b$ es equivalente a un problema de optimización, es decir, a buscar el mínimo de una cierta función. Supondremos siempre que la matriz A es simétrica y definida positiva. Sea u la solución del sistema, si definimos la función

$$J(v) = \frac{1}{2}(u - v)^T A(u - v)$$

donde v es un vector cualquiera, tendremos, puesto que A es definida positiva, que $J(v) > 0$ salvo para $v = u$ donde $J(v) = 0$ por tanto la solución u del sistema $Au = b$ es el vector donde alcanza el mínimo la función $J(v)$. Si desarrollamos $J(v)$ obtenemos:

$$J(v) = \frac{1}{2}v^T Av - v^T Au + \frac{1}{2}u^T Au$$

donde hemos utilizado que A es simétrica para asegurar que $v^T Au = u^T Av$. Por otro lado, como $Au = b$, y $\frac{1}{2}u^T Au$ es un término constante respecto a v , el eliminar de la función $J(v)$ el término $\frac{1}{2}u^T Au$ no modifica el lugar donde $J(v)$ alcanza el mínimo. Por tanto podemos establecer el siguiente teorema:

Teorema 3 Si A es una matriz simétrica y definida positiva, y u es la solución de $Au = b$, entonces u es el vector donde alcanza el mínimo el funcional

$$J(v) = \frac{1}{2}v^T Av - v^T b$$

es decir $J(u) < J(v)$ para todo v distinto de u . Recíprocamente, si u es el mínimo del funcional $J(v)$ entonces $Au = b$.

Teorema 4 Sea $J'(v)$ el vector gradiente de la función $J(v)$, es decir $J'(v)$ es el vector que tiene por componentes las derivadas parciales de $J(v)$, entonces:

$$J'(v) = Av - b$$

Demostración: $J(v)$ puede expresarse como

$$J(v) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_{ij} v_i v_j - \sum_{i=1}^N v_i b_i$$

entonces, como A es simétrica

$$\frac{\partial J(v)}{\partial v_k} = \sum_{i=1}^N a_{ik} v_k - b_k$$

y por tanto

$$J'(v) = Av - b$$

Ejemplo 2 Resolver el sistema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

es equivalente a minimizar la energía

$$J(x, y) = x^2 - xy + y^2 - 3x$$

METODO DE GRADIENTE PASO FIJO

Este método se basa en partir de una aproximación inicial v^0 del mínimo del funcional $J(v)$, y de manera iterativa se crea una secuencia v^n que converge cuando $n \rightarrow \infty$ hacia el mínimo u del funcional $J(v)$. A partir de un v^n construimos v^{n+1} desplazándonos en la dirección de máximo decrecimiento del funcional $J(v)$. Dicha dirección no es otra que $-J'(v^n)$. Por tanto

$$v^{n+1} = v^n - rJ'(v^n)$$

donde r determina el paso. En este primer método que vamos a estudiar de método de gradiente paso fijo, el paso r es fijo, es decir se fija al principio y no cambia con las iteraciones. El siguiente teorema nos muestra bajo qué condiciones el método converge.

Teorema 5 Sea A una matriz simétrica y definida positiva, y denotemos por $\lambda_{\min} > 0$ y $\lambda_{\max} > 0$ sus autovalores mínimo y máximo, entonces el método de paso fijo converge para cualquier aproximación inicial si

$$0 < r < \frac{2}{\lambda_{\max}}$$

por otro lado el valor óptimo de r para el cual la convergencia es más rápida es

$$r_{opt} = \frac{2}{\lambda_{\max} + \lambda_{\min}}$$

Demostración: Como vimos en la sección anterior $J'(v) = Av - b$. Si denotamos por $e^n = v^n - u$ el error en la etapa n , entonces, como $Au - b = 0$ podemos escribir

$$e^{n+1} = e^n - r(Av^n - b) + r(Au - b) = (Id - rA)e^n$$

aplicando esta igualdad recursivamente obtenemos que

$$e^{n+1} = (Id - rA)^{n+1}e^0$$

por tanto, el método converge si $(Id - rA)^n$ converge hacia cero cuando $n \rightarrow \infty$. Ahora bien, esto es equivalente a que

$$\rho(Id - rA) < 1$$

Por otro lado, los autovalores de la matriz $Id - rA$ son de la forma $1 - r\lambda$ donde λ es autovalor de A . Además, puesto que r es positivo se tiene que

$$1 - r\lambda_{\min} \geq 1 - r\lambda_i \geq 1 - r\lambda_{\max}$$

y por tanto

$$\rho(Id - rA) = \max\{|1 - r\lambda_{\min}|, |1 - r\lambda_{\max}|\}$$

entonces $\rho(Id - rA) < 1$ si sólo si

$$\begin{aligned} -1 &< 1 - r\lambda_{\min} < 1 \\ -1 &< 1 - r\lambda_{\max} < 1 \end{aligned}$$

que es equivalente a

$$\begin{aligned} 2 &> r\lambda_{\min} > 0 \\ 2 &> r\lambda_{\max} > 0 \end{aligned}$$

ahora bien, como los autovalores de A son positivos y $\lambda_{\max} > \lambda_{\min}$ estas condiciones son equivalentes a

$$0 < r < \frac{2}{\lambda_{\max}}$$

que es el primer resultado establecido en el teorema. Por otro lado el valor óptimo de r será aquel que minimice $\rho(Id - rA)$. A partir de la expresión de $\rho(Id - rA)$ dada anteriormente se obtiene fácilmente que dicho valor es mínimo cuando se verifica:

$$1 - r\lambda_{\min} = -(1 - r\lambda_{\max})$$

de donde despejando sale:

$$r_{opt} = \frac{2}{\lambda_{\max} + \lambda_{\min}}$$

además, para este valor de r_{opt} se obtiene

$$\rho(Id - r_{opt}A) = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} = \frac{\chi(A) - 1}{\chi(A) + 1}$$

por tanto, cuanto mejor condicionada esté la matriz A , más rápido converge el método de gradiente de paso fijo.

Criterio de parada del algoritmo. El algoritmo de gradiente paso fijo queda pues de la forma:

$$v^{n+1} = v^n - r(Av^n - b)$$

Los parámetros de entrada del algoritmo son: una aproximación inicial v^0 de la solución, un paso r , un número máximo de iteraciones N_{\max} , y una tolerancia TOL para determinar la convergencia. Los criterios de parada del algoritmo son: haber superado el número de iteraciones máximo, o cumplir el criterio de tolerancia:

$$\|Av^n - b\| < TOL$$

de hecho, como $r > 0$ es una constante positiva, dicho criterio es equivalente a

$$\|v^{n+1} - v^n\| < \frac{TOL}{r}$$

otro criterio alternativo sería utilizar el error relativo, es decir tomar:

$$\frac{\|Av^n - b\|}{\|b\|} < TOL$$

desde el punto de vista de la precisión de las aritméticas de precisión finita con las que trabaja el ordenador, este segundo criterio de error es más correcto.

Ejemplo 3 Vamos a dar 2 iteraciones del método de gradiente paso fijo para resolver el sistema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

como los autovalores de la matriz son 3 y 1, el paso óptimo es

$$r_{opt} = \frac{2}{\lambda_{\max} + \lambda_{\min}} = \frac{1}{2}$$

vamos a partir como aproximación inicial de $(x_0, y_0) = (1, 1)$.

$$J' \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ \frac{1}{2} \end{pmatrix}$$

$$J' \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 2 \\ \frac{1}{2} \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ -1 \end{pmatrix}$$

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} 2 \\ \frac{1}{2} \end{pmatrix} - \frac{1}{2} \begin{pmatrix} \frac{1}{2} \\ -1 \end{pmatrix} = \begin{pmatrix} \frac{7}{4} \\ 1 \end{pmatrix}$$

las iteraciones se van aproximando hacia la solución $(2, 1)$.

METODO DE GRADIENTE PASO OPTIMO

En este caso, y a diferencia del anterior, vamos a optimizar en cada iteración el paso r de tal forma que la función

$$\phi(r) = J(v^n - rJ'(v^n))$$

sea mínima. Es decir, buscamos el paso r tal que J es mínimo sobre la recta que pasa por v^n y tiene como vector director $-J'(v^n)$. La condición de extremo de la función ϕ es

$$\phi'(r) = J'(v^n - rJ'(v^n))^T J'(v^n) = 0$$

por otro lado se verifica que

$$J'(v^n - rJ'(v^n)) = A(v^n - rJ'(v^n)) - b = J'(v^n) - rAJ'(v^n)$$

y por tanto $\phi'(r) = 0$ nos lleva a

$$J'(v^n)^T J'(v^n) - rJ'(v^n)^T AJ'(v^n) = 0$$

de donde despejando r obtenemos

$$r_n = \frac{J'(v^n)^T J'(v^n)}{J'(v^n)^T AJ'(v^n)}$$

nótese que si A es definida positiva entonces el denominador de r_n es 0 si sólo si su numerador es cero.

Por tanto el algoritmo completo de gradiente de paso óptimo se puede escribir como

1. $J'(v^n) = Av^n - b$
2. $r_n = \frac{J'(v^n)^T J'(v^n)}{J'(v^n)^T AJ'(v^n)}$
3. $v^{n+1} = v^n - r_n J'(v^n)$

Criterios de parada del algoritmo. En este caso, los parámetros de entrada del algoritmo son la aproximación inicial v^0 , el número máximo de iteraciones N_{\max} , y la tolerancia TOL . Nótese que comprobar que una matriz es simétrica es inmediato y fácil, pero comprobar que es definida positiva no es tan fácil, y de hecho es mucho más complejo computacionalmente comprobar si la matriz A es definida positiva que resolver el sistema. Por ello, lo que se hace es buscar una solución del sistema utilizando el algoritmo sin comprobar si la matriz es definida positiva, y si el algoritmo no consigue encontrar una solución es que no se puede resolver por este método. Los criterios de parada del algoritmo son:

1. Superado el número máximo de iteraciones (método no converge)
2. $J'(v^n)^T J'(v^n) < TOL^2$ (método converge en la etapa n)
3. $J'(v^n)^T J'(v^n) \geq TOL^2$ y $J'(v^n)^T AJ'(v^n) = 0$ (método no converge)

Como en el caso anterior, también se puede trabajar con el error relativo y escribir como criterio de convergencia

$$\frac{J'(v^n)^T J'(v^n)}{\|b\|^2} < TOL^2$$

Nótese que la operación más costosa computacionalmente para pasar de una iteración a la siguiente es la multiplicación de una matriz por un vector. De hecho hay que hacer 2 multiplicaciones de matriz por vector para pasar de una iteración a la siguiente. Sin embargo, un sencillo argumento, nos permite ahorrarnos una de dichas multiplicaciones. Como $v^{n+1} = v^n - r_n J'(v^n)$ operando obtenemos

$$Av^{n+1} - b = Av^n - b - r_n AJ'(v^n)$$

y por tanto

$$J'(v^{n+1}) = J'(v^n) - r_n AJ'(v^n)$$

como $AJ'(v^n)$ es necesario calcularlo para obtener r_n , podemos utilizar la expresión anterior para calcular $J'(v^{n+1})$ (en lugar de $J'(v^{n+1}) = Av^{n+1} - b$) lo que nos ahorra una de las multiplicaciones matriz por vector.

Ejemplo 4 Vamos a dar 2 iteraciones del método de gradiente paso óptimo para resolver el sistema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

vamos a partir como aproximación inicial de $(x_0, y_0) = (1, 1)$.

$$J' \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$$

$$r_0 = \frac{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} -2 \\ 1 \end{pmatrix}}{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} -2 \\ 1 \end{pmatrix}} = \frac{5}{14}$$

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \frac{5}{14} \begin{pmatrix} -2 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{12}{14} \\ \frac{9}{14} \end{pmatrix}$$

$$J' \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} \frac{12}{14} \\ \frac{9}{14} \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}$$

$$r_1 = \frac{\begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}^T \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}}{\begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}} = \frac{5}{6}$$

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \frac{12}{14} \\ \frac{9}{14} \end{pmatrix} - \frac{5}{6} \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix} = \begin{pmatrix} \frac{53}{28} \\ 1 \end{pmatrix}$$

las iteraciones se van aproximando hacia la solución $(2, 1)$ con más rapidez que con el método de gradiente paso fijo.

METODO DE GRADIENTE CONJUGADO

Este es uno de los métodos más utilizados en la actualidad para resolver los sistemas de ecuaciones que aparecen al aproximar numéricamente la realidad física que nos rodea. La idea de base es mejorar el método de gradiente paso óptimo, optimizando en cada etapa no sólo el paso, sino también la dirección de descenso, es decir, escribiremos

$$v^{n+1} = v^n - r_n d^n$$

donde d^n es una combinación de $J'(v^n)$ y todos los $J'(v^i)$ para $i < n$, es decir

$$d^n = J'(v^n) + \sum_{i=0}^{n-1} \alpha_i J'(v^i)$$

La idea es buscar r_n y α_i en cada etapa de tal forma que la función

$$\phi(\alpha_0, \dots, \alpha_{n-1}, r_n) = J\left(v^n - r_n \left(J'(v^n) + \sum_{i=0}^{n-1} \alpha_i J'(v^i)\right)\right)$$

sea lo más pequeña posible combinando todos los posibles valores de r_n y α_i . Dicho de otra manera el $(\alpha_0, \dots, \alpha_{n-1}, r_n)$ óptimo es el que minimiza la función $\phi(\alpha_0, \dots, \alpha_{n-1}, r_n)$. La condición de extremo de una función de varias variables es que sus derivadas parciales sean 0, de donde obtenemos:

$$\begin{aligned} \frac{\partial \phi}{\partial \alpha_i} &= -r_n J'(v^{n+1})^T J'(v^i) = 0 \\ \frac{\partial \phi}{\partial r_n} &= -r J'(v^{n+1})^T \left(J'(v^n) + \sum_{i=0}^{n-1} \alpha_i J'(v^i)\right) = 0 \end{aligned}$$

de donde obtenemos de forma inmediata que los vectores $J'(v^i)$ forman un sistema ortogonal de vectores.

Dado que el método de gradiente conjugado optimiza al pasar de la etapa n , a la etapa $n+1$ la solución en el espacio afín que pasa por v^n y engendrado por el sistema de vectores $\{J'(v^i)\}_{i=0}^n$ que además son ortogonales entre sí, entonces cuando $n = \dim - 1$ donde \dim es la dimensión del sistema, la familia $\{J'(v^i)\}_{i=0}^n$ contendrá \dim vectores que generarán todo el espacio, y por tanto la solución del sistema u se podrá escribir como

$$u = v^n - r_n \left(J'(v^n) + \sum_{i=0}^{n-1} \alpha_i J'(v^i)\right)$$

dicho de otra manera, la solución del sistema se encuentra de forma exacta en la iteración $n = \dim$ o eventualmente antes si u se puede expresar de la forma anterior para un $n < \dim$. Esto es una propiedad muy interesante pues indica que en un número máximo de iteraciones dado por la dimensión del sistema debemos encontrar la solución del sistema independientemente de la aproximación inicial que puede ser buena o mala.

Encontrar el mínimo de la función $\phi(\alpha_0, \dots, \alpha_{n-1}, r_n)$ puede ser una tarea complicada, sin embargo, como muestra el siguiente teorema, existe una forma algorítmica muy sencilla de calcular las direcciones de descenso d^n , y de calcular v^{n+1} en función de v^n .

Teorema 6 *La secuencia v^n que minimiza en cada etapa la función $\phi(\alpha_0, \dots, \alpha_{n-1}, r_n)$ puede obtenerse a través del algoritmo iterativo siguiente*

1. $r_n = \frac{J'(v^n)^T J'(v^n)}{(d^n)^T A d^n}$
2. $v^{n+1} = v^n - r_n d^n$
3. $J'(v^{n+1}) = J'(v^n) - r_n A d^n$
4. $d^{n+1} = J'(v^{n+1}) + \frac{J'(v^{n+1})^T J'(v^{n+1})}{J'(v^n)^T J'(v^n)} d^n$

Teorema 7 *con $d^0 = J'(v^0) = Av^0 - b$*

Demostración. Dada su complejidad no demostraremos en su totalidad el teorema, nos limitaremos a demostrar algunas conclusiones parciales. Por ejemplo vamos a demostrar que dada la dirección de descenso d^n , el paso óptimo r para buscar el mínimo en esta dirección viene dado por el r_n dado en la expresión 1 del teorema. Efectivamente, razonando como en el método de gradiente paso óptimo, definimos la función

$$\phi(r) = J(v^n - r d^n)$$

buscamos el paso r tal que J es mínimo sobre la recta que pasa por v^n y tiene como vector director $-d^n$. La condición de extremo de la función ϕ es

$$\phi'(r) = J'(v^n - r d^n)^T d^n = 0$$

por otro lado se verifica que

$$J'(v^n - r d^n) = A(v^n - r d^n) - b = J'(v^n) - r A d^n$$

y por tanto $\phi'(r) = 0$ nos lleva a

$$J'(v^n)^T d^n - r (d^n)^T A d^n = 0$$

de donde despejando r obtenemos

$$r_n = \frac{J'(v^n)^T d^n}{(d^n)^T A d^n}$$

por otro lado, como d^n viene dada por la expresión 4 del teorema, y dado que el sistema de vectores $\{J'(v^n)\}$ son ortogonales y además d^{n-1} es una combinación de los vectores $\{J'(v^i)\}_{i=0}^{n-1}$, se tiene que $J'(v^n)^T d^{n-1} = 0$ y por tanto $J'(v^n)^T d^n = J'(v^n)^T J'(v^n)$ de donde se deduce finalmente la expresión 1 del teorema anterior.

Vamos a demostrar también la expresión 3 para el cálculo de $J'(v^n)$. Teniendo en cuenta que $v^n = v^{n-1} - r_{n-1} d^{n-1}$, si multiplicamos por A y restamos b a esta igualdad obtenemos

$$Av^n - b = Av^{n-1} - b - r_{n-1} A d^{n-1}$$

de donde se deduce la expresión 3 del teorema.

Criterios de parada del algoritmo. Los parámetros de entrada del algoritmo son la aproximación inicial v^0 , el número máximo de iteraciones N_{\max} , y la tolerancia TOL . En este caso, sabemos formalmente que el número máximo de iteraciones es la dimensión de la matriz. Sin embargo, numéricamente es posible que si la matriz está mal condicionada, los errores de redondeo pueden hacer necesario hacer más iteraciones que el máximo teórico. Además hay otro pequeño inconveniente si tomamos $N_{\max} = \dim$, la dimensión del sistema, y es que puede darse el caso de que v^{\dim} sea la solución pero que el método no lo detecta pues el criterio de parada detectaría que hemos llegado a la solución al intentar realizar la iteración $\dim + 1$. Los criterios de parada del algoritmo son:

1. Superado el número máximo de iteraciones (método no converge)
2. $J'(v^n)^T J'(v^n) < TOL^2$ (método converge en la etapa n)
3. $J'(v^n)^T J'(v^n) \geq TOL^2$ y $(d^n)^T A d^n = 0$ (método no converge)

Como en el caso anterior, también se puede trabajar con el error relativo y escribir como criterio de convergencia

$$\frac{J'(v^n)^T J'(v^n)}{\|b\|^2} < TOL^2$$

Ejemplo 5 Vamos a dar 2 iteraciones del método de gradiente conjugado para resolver el sistema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

vamos a partir como aproximación inicial de $(x_0, y_0) = (1, 1)$.

$$J' \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$$

$$r_0 = \frac{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} -2 \\ 1 \end{pmatrix}}{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} -2 \\ 1 \end{pmatrix}} = \frac{5}{14}$$

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \frac{5}{14} \begin{pmatrix} -2 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{12}{7} \\ \frac{9}{14} \end{pmatrix}$$

$$J' \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} \frac{12}{7} \\ \frac{9}{14} \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}$$

$$d^1 = \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix} + \frac{\begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}^T \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}}{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} -2 \\ 1 \end{pmatrix}} \begin{pmatrix} -2 \\ 1 \end{pmatrix} =$$

$$\begin{pmatrix} -\frac{15}{196} \\ -\frac{49}{196} \end{pmatrix}$$

$$r_1 = \frac{\begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}^T \begin{pmatrix} -\frac{3}{14} \\ -\frac{3}{7} \end{pmatrix}}{\begin{pmatrix} -15 \\ -\frac{49}{196} \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} -15 \\ -\frac{49}{196} \end{pmatrix}} = \frac{14}{15}$$

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \frac{12}{7} \\ \frac{9}{14} \end{pmatrix} - \frac{14}{15} \begin{pmatrix} -\frac{15}{196} \\ -\frac{49}{196} \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

como vemos, tal y como se esperaba a partir de la teoría el método da la solución exacta en la segunda iteración.

PRACTICA 1. METODO DE GRADIENTE CONJUGADO

Se implementará una función en C que posea como parámetros una matriz morse A , un vector de términos independientes b , una aproximación inicial v_0 , el número máximo de iteraciones N_{\max} , y la tolerancia TOL . La solución se devolverá en el vector v_0 , la función devolverá un valor entero que indicará la iteración donde ha encontrado la solución, en caso de que el método no converga se devuelve -1

Para comprobar que el método funciona se analizarán los siguientes sistemas de ecuaciones

$$1. A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \quad v_0 = \begin{pmatrix} 0 \\ -5 \\ 10 \end{pmatrix}$$

$$2. A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad v_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$3. A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad v_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$4. A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad v_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

5. Los sistemas de dimensión 10, 100 y 1000 que se encuentran en el directorio de la asignatura, y que tienen por solución el vector $(1, \dots, 1)$.

PRECONDICIONAMIENTO DEL ALGORITMO DE GRADIENTE CONJUGADO

La velocidad de convergencia de los algoritmos tipo gradiente depende fuertemente del condicionamiento $\chi(A)$ de la matriz A . Cuanto más cercano a 1 esté $\chi(A)$ más rápido converge el método. La idea de las técnicas de preconditionamiento es sustituir el sistema $Au = b$ por un sistema equivalente $\tilde{A}\tilde{u} = \tilde{b}$, pero donde la matriz \tilde{A} esté

mejor condicionada. Para construir el preconditionamiento primero observamos que si P es una matriz invertible, entonces el sistema $Au = b$ es equivalente al sistema

$$P^T APP^{-1}u = P^T b$$

por tanto si definimos $\tilde{A} = P^T AP$, $\tilde{u} = P^{-1}u$ y $\tilde{b} = P^T b$ entonces el sistema $Au = b$ es equivalente al sistema $\tilde{A}\tilde{u} = \tilde{b}$. Por otro lado es importante resaltar que si A es simétrica y definida positiva, entonces \tilde{A} también es simétrica y definida positiva, y por tanto también se puede aplicar el método de gradiente conjugado descrito en la sección anterior. Vamos a ver a continuación como afecta el preconditionamiento dado por P al algoritmo del método de gradiente conjugado, para ello vamos a introducir la notación $\tilde{d}^n = P^{-1}d^n$ y $\tilde{v}^n = P^{-1}v^n$

El algoritmo de gradiente conjugado aplicado al sistema $\tilde{A}\tilde{u} = \tilde{b}$ es

1. $\tilde{r}_n = \frac{\tilde{J}'(\tilde{v}^n)^T \tilde{J}'(\tilde{v}^n)}{(\tilde{d}^n)^T \tilde{A} \tilde{d}^n}$
2. $\tilde{v}^{n+1} = \tilde{v}^n - \tilde{r}_n \tilde{d}^n$
3. $\tilde{J}'(\tilde{v}^{n+1}) = \tilde{J}'(\tilde{v}^n) - \tilde{r}_n \tilde{A} \tilde{d}^n$
4. $\tilde{d}^{n+1} = \tilde{J}'(\tilde{v}^{n+1}) + \frac{\tilde{J}'(\tilde{v}^{n+1})^T \tilde{J}'(\tilde{v}^{n+1})}{\tilde{J}'(\tilde{v}^n)^T \tilde{J}'(\tilde{v}^n)} \tilde{d}^n$

con $\tilde{d}^0 = \tilde{J}'(\tilde{v}^0) = \tilde{A}\tilde{v}^0 - \tilde{b}$.

Si escribimos ahora los términos que aparecen en el algoritmo en función de la matriz P y las variables v^n y d^n obtenemos

$$\begin{aligned} \tilde{J}'(\tilde{v}^n) &= \tilde{A}\tilde{v}^n - \tilde{b} = P^T APP^{-1}v^n - P^T b = P^T J'(v^n) \\ \tilde{A}\tilde{d}^{n-1} &= P^T APP^{-1}d^{n-1} = P^T Ad^{n-1} \\ \tilde{J}'(\tilde{v}^n)^T \tilde{J}'(\tilde{v}^n) &= (P^T J'(v^n))^T P^T J'(v^n) = \\ &= J'(v^n)^T P P^T J'(v^n) \\ (\tilde{d}^n)^T \tilde{A} \tilde{d}^n &= (P^{-1}d^n)^T P^T APP^{-1}d^n = (d^n)^T Ad^n \end{aligned}$$

por tanto combinando el algoritmo de gradiente conjugado aplicado al sistema preconditionado con las igualdades anteriores llegamos al algoritmo de gradiente conjugado preconditionado dado por

1. $\tilde{r}_n = \frac{J'(v^n)^T P P^T J'(v^n)}{(d^n)^T A d^n}$
2. $v^{n+1} = v^n - r_n d^n$
3. $J'(v^{n+1}) = J'(v^n) - \tilde{r}_n A d^n$
4. $d^{n+1} = P P^T J'(v^{n+1}) + \frac{J'(v^{n+1})^T P P^T J'(v^{n+1})}{J'(v^n)^T P P^T J'(v^n)} d^n$

Teorema 8 con $d^0 = P P^T J'(v^0) = P P^T (Av^0 - b)$.

Nótese que a nivel algorítmico, la única diferencia con respecto al método de gradiente conjugado normal es que en este caso necesitamos calcular en cada iteración el vector $P P^T J'(v^n)$. De hecho se denomina matriz de preconditionamiento a la matriz $C = P P^T$.

Veamos ahora diferentes formas de elegir la matriz C de preconditionamiento. De forma genérica como queremos que $P^T AP$ esté bien condicionada debe cumplirse

$$P^T AP \approx Id$$

es decir que $P^T AP$ esté próxima a la matriz identidad, pues la matriz Id es la mejor condicionada posible. Despejando, esto es equivalente a

$$A \approx (P^T)^{-1} P^{-1}$$

o de forma equivalente

$$A^{-1} \approx P P^T = C$$

es decir, que la matriz C de preconditionamiento es una aproximación de la inversa de A . Ahora bien existe un compromiso entre lo que cuesta computacionalmente calcular C , y lo que se gana en velocidad del algoritmo. Es decir, cuanto más cerca esté C de A^{-1} , más rápido converge el algoritmo, pero más cuesta calcular C . Veamos ahora algunas elecciones de matriz de preconditionamiento C .

PRECONDICIONAMIENTO A TRAVES DE LA TRIDIAGONAL DE A .

En este caso se toma C como la inversa de la matriz tridiagonal de A , es decir, la matriz $A' = (a'_{ij})$ tal que $a'_{ij} = a_{ij}$ si $|i - j| \leq 1$ y $a'_{ij} = 0$ para cualquier otro valor de i, j . Este preconditionamiento produce buenos resultados cuando A y A' son próximas, lo cual se produce cuando la mayoría de los elementos de A que no pertenecen a la tridiagonal son 0. La ventaja de este método es que computacionalmente es muy rápido porque para evaluar el vector $z^n = C J'(v^n)$ basta con resolver el sistema tridiagonal $A' z^n = J'(v^n)$ lo cual se hace muy rápido utilizando el algoritmo de Crout para matrices tridiagonales.

Ejemplo 6 Vamos a dar 1 iteración del método de gradiente conjugado con preconditionamiento por la tridiagonal de A para resolver el sistema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

vamos a partir como aproximación inicial de $(x_0, y_0) = (1, 1)$.

$$J' \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 3 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 1 \end{pmatrix}$$

$$d^0 = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}^{-1} \begin{pmatrix} -2 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$r_0 = \frac{\begin{pmatrix} -2 \\ 1 \end{pmatrix}^T \begin{pmatrix} -1 \\ 0 \end{pmatrix}}{\begin{pmatrix} -1 \\ 0 \end{pmatrix}^T \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \end{pmatrix}} = 1$$

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - 1 \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

por tanto, con este método se consigue la solución exacta en la primera iteración.

PRECONDICIONAMIENTO POR FACTORIZACION INCOMPLETA DE CHOLESKY

El algoritmo de factorización de Cholesky permite descomponer una matriz A simétrica y definida positiva como

$$A = BB^T$$

donde B es una matriz triangular inferior

$$B = \begin{pmatrix} b_{1,1} & 0 & 0 & \dots & 0 \\ b_{2,1} & b_{2,2} & 0 & \dots & \cdot \\ b_{3,1} & b_{3,2} & b_{3,3} & \dots & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot \\ b_{n,1} & b_{n,2} & b_{n,3} & \dots & b_{n,n} \end{pmatrix}$$

De forma general, el algoritmo para calcular B es el siguiente

Para $i = 1, \dots, N$

$$b_{i,i} = \sqrt{\left(a_{i,i} - \sum_{k=1}^{i-1} b_{i,k}^2\right)}$$

Para $j = i + 1, \dots, N$

$$b_{j,i} = \frac{1}{b_{i,i}} \left(a_{j,i} - \sum_{k=1}^{i-1} b_{j,k} b_{i,k}\right)$$

Fin Para j

Fin Para i

El interés de descomponer una matriz A por el método de Cholesky, es que a continuación es muy sencillo resolver el sistema de ecuaciones $Au = b$. Efectivamente, basta descomponer el sistema de la siguiente forma

$$\begin{aligned} Bz &= b \\ B^T u &= z \end{aligned}$$

y ambos sistemas se resuelven rápidamente haciendo un remonte y un descenso.

La factorización incompleta de Cholesky consiste en modificar el algoritmo anterior de tal forma que los b_{ij} sean distintos de cero sólo en el caso de que a_{ij} sea

distinto de 0, es decir cuando calculamos la parte de b_{ji} en el algoritmo precedente hacemos:

$$b_{j,i} = \begin{cases} \frac{1}{b_{i,i}} \left(a_{j,i} - \sum_{k=1}^{i-1} b_{j,k} b_{i,k}\right) & \text{si } a_{ji} \neq 0 \\ 0 & \text{si } a_{ji} = 0 \end{cases}$$

el resultado será una matriz B' que será una aproximación de B que contendrá menos elementos distintos de cero. Nótese que cuantos más elementos nulos tenga la matriz B' más rápido es resolver el sistema $B'(B')^T u = b$.

La matriz de preconditionamiento en este caso es $C = (B'(B')^T)^{-1}$, y como en el caso anterior para calcular el vector $z = CJ'(v^n)$ en cada iteración del método de gradiente conjugado preconditionado se resuelve el sistema $B'(B')^T z = J'(v^n)$.

PRACTICA 2. PRECONDICIONAMIENTO

Se modificará la práctica 1, incluyendo el preconditionamiento por la tridiagonal de A . Para ello se utilizará el algoritmo de Crout para la resolución de sistemas tridiagonales. El caso de sistemas de ecuaciones con matrices A' tridiagonales posee una forma especialmente simple de factorización, vamos a descomponer A' en el producto de dos matrices triangulares de la forma siguiente

$$\begin{aligned} A' &= \begin{pmatrix} a_1 & b_1 & \cdot & 0 \\ c_1 & a_2 & \cdot & 0 \\ 0 & \cdot & \cdot & b_{N-1} \\ 0 & \cdot & c_{N-1} & a_N \end{pmatrix} = LU = \\ &= \begin{pmatrix} l_1 & 0 & \cdot & 0 \\ m_1 & l_2 & \cdot & 0 \\ 0 & \cdot & \cdot & 0 \\ 0 & \cdot & m_{N-1} & l_N \end{pmatrix} \begin{pmatrix} 1 & u_1 & \cdot & 0 \\ 0 & 1 & \cdot & 0 \\ 0 & \cdot & \cdot & u_{N-1} \\ 0 & \cdot & 0 & 1 \end{pmatrix} \end{aligned}$$

los vectores m_i , l_i , y u_i se calculan utilizando el esquema:

$$l_1 = a_1$$

$$u_1 = \frac{b_1}{l_1}$$

Para $i = 2, \dots, N - 1$

$$m_{i-1} = c_{i-1}$$

$$l_i = a_i - m_{i-1} u_{i-1}$$

$$u_i = \frac{b_i}{l_i}$$

Fin Para

$$m_{N-1} = c_{N-1}$$

$$l_N = a_N - m_{N-1} u_{N-1}$$

Finalmente para resolver el sistema $A'u = b$ se resuelven por descenso y remonte los sistemas

$$Lz = b$$

$$Uu = z$$

Se incluirá este algoritmo para preconditionar el método de Gradiente Conjugado. Para comprobar que el método funciona se analizarán los siguientes sistemas de ecuaciones

$$1. A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix} b = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} v_0 = \begin{pmatrix} 0 \\ -5 \\ 10 \end{pmatrix}$$

$$2. A = \begin{pmatrix} 3 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 3 \end{pmatrix} b = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} v_0 = \begin{pmatrix} 0 \\ -5 \\ 10 \end{pmatrix}$$

$$3. A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} v_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$4. A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} v_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$5. A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} b = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} v_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

6. Los sistemas de dimensión 10, 100 y 1000 que se encuentran en el directorio de la asignatura, y que tienen por solución el vector $(1, \dots, 1)$.

Se comprobará de forma general que el número de iteraciones para resolver el sistema es menor al utilizar el preconditionamiento.

EXTENSION DEL METODO DE GRADIENTE CONJUGADO A MATRICES CUALESQUIERA. METODO DE LA ECUACION NORMAL.

Como ya se ha mencionado en varias ocasiones, el método de gradiente conjugado sólo se puede aplicar a matrices simétricas y definidas positivas. Por tanto si la matriz A no es simétrica, o no es definida positiva, no podemos asegurar la convergencia del método y tampoco podemos asegurar que en el caso de que converga, lo haga hacia una solución del sistema. Hay diversas técnicas para ampliar el método de gradiente conjugado, al caso de matrices cualesquiera A , cuya única condición es que sean regulares, esto es que $|A| \neq 0$. Nosotros vamos a presentar aquí la técnica más sencilla, denominada método de la ecuación normal. La idea es muy sencilla, en lugar de estudiar el sistema $Av = b$ estudiamos el sistema equivalente

$$A^T A u = A^T b$$

donde ahora la matriz del sistema es $\tilde{A} = A^T A$, y el vector independiente es $\tilde{b} = A^T b$. Si $|A| \neq 0$, la matriz $A^T A$ es simétrica y definida positiva, y por tanto podemos aplicar el método de gradiente conjugado al sistema $\tilde{A}u = \tilde{b}$, quedando el algoritmo

$$1. \tilde{r}_n = \frac{\tilde{J}'(v^n)^T \tilde{J}'(v^n)}{(\tilde{d}^n)^T \tilde{A} \tilde{d}^n}$$

$$2. v^{n+1} = v^n - \tilde{r}_n \tilde{d}^n$$

$$3. \tilde{J}'(v^{n+1}) = \tilde{J}'(v^n) - \tilde{r}_n \tilde{A} \tilde{d}^n$$

$$4. \tilde{d}^{n+1} = \tilde{J}'(v^{n+1}) + \frac{\tilde{J}(v^{n+1})^T \tilde{J}'(v^{n+1})}{\tilde{J}'(v^n)^T \tilde{J}'(v^n)} \tilde{d}^n$$

$$\text{con } \tilde{d}^0 = \tilde{J}'(\tilde{v}^0) = \tilde{A} \tilde{v}^0 - \tilde{b}.$$

Como en el caso del preconditionamiento, a efectos prácticos, nunca se resuelve el sistema bajo esta forma, pues en este caso, el cálculo $A^T A$ es computacionalmente costoso, y además si A es una matriz morse con pocos elementos no-nulos, la operación $A^T A$ puede hacer que el número de elementos no nulos de la nueva matriz sea mucho más grande que el de A . lo cual aumenta el costo operacional del método. Por eso, al igual que en el caso del preconditionamiento, se modifica el algoritmo anterior para que no aparezca la matriz $A^T A$. Para ello, tendremos en cuenta que $\tilde{J}'(v) = A^T J'(v)$. El algoritmo sin que aparezca explícitamente la matriz $A^T A$ queda de la forma:

$$1. \tilde{r}_n = \frac{(A^T J'(v^n))^T A^T J'(v^n)}{(A \tilde{d}^n)^T A \tilde{d}^n}$$

$$2. v^{n+1} = v^n - \tilde{r}_n \tilde{d}^n$$

$$3. J'(v^{n+1}) = J'(v^n) - \tilde{r}_n A \tilde{d}^n$$

$$4. \tilde{d}^{n+1} = A^T J'(v^{n+1}) + \frac{(A^T J'(v^{n+1}))^T A^T J'(v^{n+1})}{(A^T J'(v^n))^T A^T J'(v^n)} \tilde{d}^n$$

$$\text{con } \tilde{d}^0 = A^T J'(\tilde{v}^0) = A^T (A v^0 - b).$$

A nivel computacional, la principal diferencia con el método de gradiente conjugado normal, es que en este caso, en cada iteración hay que hacer 2 multiplicaciones de matriz por vector. Además la matriz $A^T A$ está, en general, peor condicionada que la matriz A . Por ejemplo, en el caso en que A sea simétrica $\chi(A^T A) = \chi(A)^2$, y por tanto la convergencia es más lenta.

TRIANGULACIONES DE DOMINIOS PLANOS

NOCIONES GEOMETRICAS BASICAS SOBRE TRIANGULOS

Un triángulo $K = (v_0, v_1, v_2)$ en el plano viene determinado por 3 vértices que denotaremos por $v_0 = (x_0, y_0)$, $v_1 = (x_1, y_1)$, y $v_2 = (x_2, y_2)$. A continuación daremos algunos resultados elementales sobre triángulos. Consideraremos siempre que el triángulo no es degenerado, es decir que sus tres vértices no están alineados.

Teorema 9 El área de un triángulo K en función de las coordenadas de sus vértices viene dada por

$$AREA(K) = \frac{1}{2} ABS \begin{pmatrix} 1 & 1 & 1 \\ x_0 & x_1 & x_2 \\ y_0 & y_1 & y_2 \end{pmatrix}$$

Teorema 10 Existe una única circunferencia circunscrita que pasa por los tres vértices de un triángulo K . El centro (x_c, y_c) y radio R de esta circunferencia vienen dados por las expresiones

$$x_c = \frac{[x_2^2 + y_2^2 - x_0^2 - y_0^2](y_2 - y_1) - [x_2^2 + y_2^2 - x_1^2 - y_1^2](y_2 - y_0)}{2[(x_2 - x_0)(y_2 - y_1) - (y_2 - y_0)(x_2 - x_1)]}$$

$$y_c = \frac{[x_2^2 + y_2^2 - x_1^2 - y_1^2](x_2 - x_0) - [x_2^2 + y_2^2 - x_0^2 - y_0^2](x_2 - x_1)}{2[(x_2 - x_0)(y_2 - y_1) - (y_2 - y_0)(x_2 - x_1)]}$$

$$R = \sqrt{(x_0 - x_c)^2 + (y_0 - y_c)^2}$$

Teorema 11 El radio ρ_K de la circunferencia inscrita en un triángulo K viene dado por

$$\rho_K = \frac{AREA(K)}{\frac{1}{2} (\text{Perímetro de } K)}$$

Teorema 12 Dado un segmento $S = (v_0, v_1)$ en el plano definido por los vértices $v_0 = (x_0, y_0)$, $v_1 = (x_1, y_1)$. La expresión:

$$R_{(v_0, v_1)}(x, y) = (x_1 - x_0)(y - y_0) - (y_1 - y_0)(x - x_0)$$

separa el plano en 3 regiones distintas dadas por: $R_{(v_0, v_1)}(x, y) = 0$ que representa la recta que pasa por el segmento S . $R_{(v_0, v_1)}(x, y) < 0$ que representa el semiplano situado a la derecha del vector S siguiendo la orientación de v_0 hacia v_1 , y por último la zona $R_{(v_0, v_1)}(x, y) > 0$ que representa el semiplano que se encuentra a la izquierda del vector S siguiendo la orientación de v_0 hacia v_1

Teorema 13 Dado un punto cualquiera $v = (x, y)$, y un triángulo cualquiera K , entonces v está en el interior o en la frontera del triángulo $K = (v_0, v_1, v_2)$ si sólo si

$$R_{(v_0, v_1)}(v) \cdot R_{(v_0, v_1)}(v_2) \geq 0$$

$$R_{(v_1, v_2)}(v) \cdot R_{(v_1, v_2)}(v_0) \geq 0$$

$$R_{(v_0, v_2)}(v) \cdot R_{(v_0, v_2)}(v_1) \geq 0$$

Teorema 14 Dados dos segmentos en el plano $S = (v_0, v_1)$ y $S' = (v_2, v_3)$, entonces los dos segmentos se interceptan si sólo si.

$$R_{(v_0, v_1)}(v_2) \cdot R_{(v_0, v_1)}(v_3) < 0$$

$$R_{(v_2, v_3)}(v_0) \cdot R_{(v_2, v_3)}(v_1) < 0$$

Teorema 15 Si los nodos de un triángulo (v_0, v_1, v_2) están orientados siguiendo las agujas del reloj, entonces

$$R_{(v_0, v_1)}(v_2) < 0$$

$$R_{(v_1, v_2)}(v_0) < 0$$

$$R_{(v_2, v_0)}(v_1) < 0$$

Tener los triángulos orientados, agiliza, como veremos en el futuro, algunos de los cálculos que realizaremos sobre los triángulos. Nótese, por ejemplo, que si un triángulo está orientado, entonces un punto v está en su interior si solo si

$$R_{(v_0, v_1)}(v) < 0$$

$$R_{(v_1, v_2)}(v) < 0$$

$$R_{(v_2, v_0)}(v) < 0$$

Un concepto importante en lo que vamos a estudiar en esta sección de triangulaciones es la calidad geométrica de un triángulo. Diremos que un triángulo es de buena calidad geométrica si su forma es parecida a un triángulo equilátero. Para medir cuantitativamente este concepto de calidad de un triángulo definimos la siguiente expresión:

$$Q(K) = \frac{\text{longitud del lado mayor de } K}{\text{radio círculo inscrito en } K}$$

evidentemente, cuanto más pequeño sea el valor de $Q(K)$ mejor aspecto geométrico tendrá un triángulo. El siguiente problema determina el valor óptimo más pequeño posible de $Q(K)$.

Teorema 16 Dado K un triángulo equilátero se cumple: $Q(K) = 2\sqrt{3}$.

Otro criterio de calidad geométrica de un triángulo K es el mínimo de los 3 ángulos que forman las aristas de un triángulo. La mejor situación es de nuevo un triángulo equilátero para el cual el ángulo que forman las aristas es 60 grados.

TRIANGULACIONES

Se considera un dominio Ω en dimensión 2. Una triangulación del dominio Ω es una colección de triángulos cerrados $\{K_i\}_{i=1,\dots,N_t}$, verificando las siguientes condiciones

1. $\Omega = \cup_{i=1,\dots,N_t} K_i$
2. $K_i \cap K_j = \begin{cases} \emptyset \\ \text{una arista completa} \end{cases}$

Consideremos el dominio Ω dado por la figura 1, una

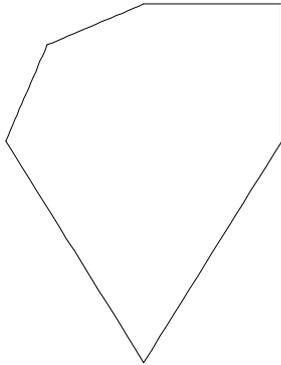


Figura 1:

posible triangulación de Ω viene dada por la colección de triángulos dados en la figura 2,

Obviamente existen muchas triangulaciones distintas del mismo dominio. Para elegir la mejor triangulación un posible criterio es tomar de las posibles triangulaciones, aquella cuyos triángulos tengan mayor calidad geométrica, es decir, sean lo más equiláteros posible. Para medir cuantitativamente la calidad geométrica de un triángulo K se pueden utilizar algunos de los criterios de calidad definidos en la sección anterior.

ALMACENAMIENTO DE UNA TRIANGULACIÓN

Vamos a distinguir dos tipos de almacenamiento. El almacenamiento en disco, y el almacenamiento en RAM. Consideramos como ejemplo modelo la malla de la figura 3 dada por los nodos $v_0 = (-1, -1)$, $v_1 = (-1, 1)$, $v_2 = (1, 1)$, $v_3 = (1, -1)$ y $v_4 = (0, 0)$ y los triángulos $K_0 = (v_0, v_4, v_1)$, $K_1 = (v_2, v_4, v_1)$, $K_2 = (v_2, v_4, v_3)$, $K_3 = (v_0, v_4, v_3)$.

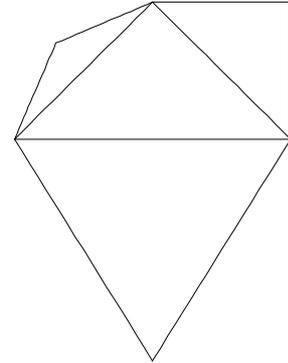


Figura 2:

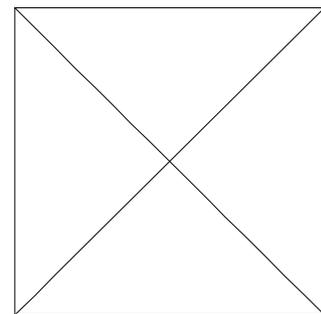


Figura 3:

Almacenamiento en disco. La triangulación anterior se almacena en disco en un fichero que contiene la siguiente información:

```

5 4
-1. -1. 1
-1. 1. 1
1. 1. 1
1. -1. 1
0. 0. 0

0 4 1
2 4 1
2 4 3
0 4 3

3 1 -1
2 0 -1
1 3 -1
0 2 -1

```

La primera línea indica que la triangulación posee 5 nodos y 4 triángulos. Las siguientes 5 líneas indican las coordenadas de los nodos y la última cifra de cada línea indica si el nodo es frontera o interior (*frontera* = 1, 2 *interior* = 0). Cuando estudiemos el método de elementos finitos será necesario distinguir entre 2 tipos de valores frontera por eso los puntos fronteras pueden tener dos valores, por el momento fijaremos dicho valor a 1. Las siguientes 4 líneas determinan los triángulos a través de tripletas de índices de nodos. Es decir, por ejemplo, el triángulo 0 está formado por los nodos de índices 0, 4 y 1. Por último las 4 últimas líneas indican los índices de los triángulos vecinos a uno dado. Por ejemplo el triángulo 0, tiene como vecinos (es decir triángulos con los que comparte arista) los triángulos de índices 1, 3 y -1 el -1 indica que una de las aristas es frontera y no es compartida con otro triángulo. Esta última idea de conservar en memoria las relaciones de vecindad entre triángulos es, como veremos posteriormente, de gran utilidad para acelerar algunos procedimientos relacionados con la triangulación. Además la relación de vecindad se puede ordenar de la siguiente forma: Dado un triángulo con nodos de índices v_0, v_1, v_2 entonces podemos fijar la relación de vecindad del triángulo a través de 3 índices de triángulos t_0, t_1, t_2 , de tal forma que el triángulo t_0 comparte la arista que conecta los nodos v_0 y v_1 con el triángulo inicial, el triángulo t_1 comparte con el inicial la arista v_1v_2 y por último el triángulo t_2 comparte con el inicial la arista v_2v_0 .

Almacenamiento en memoria RAM. En lenguaje C, guardaremos la información de la triangulación utilizando las siguientes estructuras de datos:

```

typedef struct{
    float x,y;
}punto;

```

```

typedef int triangulo[3];

typedef struct{
    int np;
    int nt;
    punto *p;
    int *fr;
    triangulo *t;
    triangulo *v;
}triangulacion;

```

En la estructura *punto* se almacenan las coordenadas de un nodo. *triangulo* define un vector de 3 valores enteros que representan los índices de los nodos que componen el triángulo. La estructura *triangulacion* incluye el número de nodos *np*, el número de triángulos *nt*, un puntero **p* que apunta al bloque de memoria donde están guardadas las coordenadas de los nodos, un puntero **fr* para indicar si un nodo es frontera o interior, un puntero **t* donde se guardan los índices de los nodos que constituyen cada triángulo, y un puntero **v* donde se guardan los índices de los triángulos vecinos a uno dado.

Dentro del fichero *ana.h* existen funciones para leer y escribir triangulaciones. En el programa ejemplo *tri1.c* se utilizan estas funciones para crear en disco la triangulación del ejemplo anterior. En el programa ejemplo *tri2.c* se utilizan estas funciones para leer desde disco una triangulación

REFINAMIENTO UNIFORME DE TRIANGULACIONES

Un refinamiento de una malla (o triangulación) consiste en subdividir la malla inicial en mallas con más nodos y triángulos más pequeños. Una de las propiedades más importantes que se suele exigir a las triangulaciones de dominios es que sus triángulos sean de buena calidad geométrica. Por tanto la técnica de refinamiento de triangulaciones debe respetar o mejorar la calidad geométrica de los triángulos de una malla. La manera más sencilla de refinar una malla respetando la calidad geométrica de los triángulos consiste en dividir cada triángulo en 4, añadiendo como nodos los puntos medios de las aristas, y uniendo estos puntos medios con los vértices del triángulo. Por ejemplo, un refinamiento uniforme de la malla de la figura 4 da lugar a la malla de la figura 5

TRIANGULACIONES DE DELAUNY DE DOMINIOS PLANOS

Definición 2 Una triangulación formada por un conjunto de triángulos K_i se dice de Delauney, si para cada triángulo K_i , el interior del círculo circunscrito no incluye ningún nodo de la triangulación.

El ejemplo de la figura 6 muestra una triangulación de Delaunay formada por tres triángulos

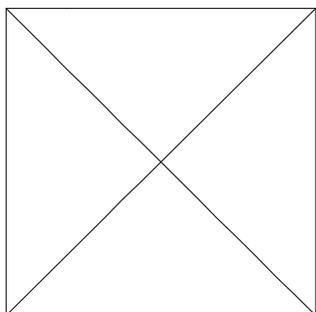


Figura 4:

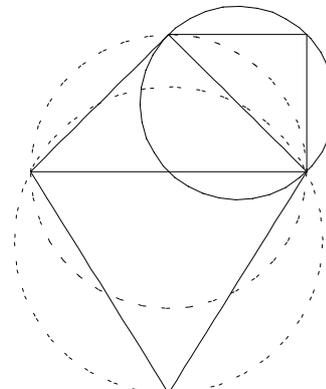


Figura 6:

Evidentemente no todas las triangulaciones son de Delaunay. De hecho como veremos posteriormente, bajo ciertas condiciones la triangulación de Delaunay de un dominio es única.

TRIANGULACIONES DE DELAUNY Y DIAGRAMAS DE VORONOI

Las triangulaciones de Delaunay están íntimamente relacionadas con los denominados diagramas de Voronoi. Dado un conjunto de puntos en el plano (que representan los nodos de la triangulación), asociamos a cada punto v_i el conjunto V_i dado por todos los puntos del plano que están más cerca de v_i que de cualquier otro nodo v_j , es decir

$$V_i = \{v \in R^2 : d(v, v_i) \leq d(v, v_j)\}$$

Llamaremos diagrama de Voronoi a la familia de conjuntos V_i .

La figura 7 representa el diagrama de Voronoi asociado a los nodos de la figura 6, donde las líneas rectas representan la frontera de los conjuntos V_i .

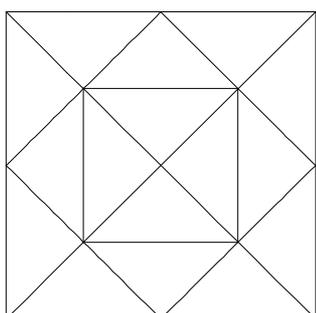


Figura 5:

Definición 3 Un conjunto de nodos se dice que están en posición general si no existe ninguna circunferencia que pase por 4 nodos simultáneamente. Dicho de otra forma, que la circunferencia circunscrita a 3 nodos no pasa por ningún otro nodo.

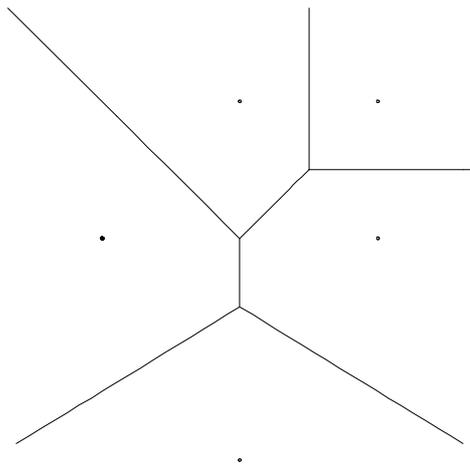


Figura 7:

Evidentemente, los nodos del ejemplo anterior están en posición general. El siguiente teorema pone de manifiesto la relación entre las triangulaciones de Delaunay y los diagramas de Voronoi.

Teorema 17 *Si un conjunto de nodos está en posición general. Existe una única triangulación de Delaunay que contiene a dichos nodos como vértices de triángulos. Además, las aristas de los triángulos están formadas por la unión de los nodos tales que sus respectivos conjuntos de Voronoi V_i tienen una zona de frontera común.*

Obsérvese como en el ejemplo puesto anteriormente se verifica esta relación entre el diagrama de Voronoi y la triangulación de Delaunay. Un colorario inmediato del anterior teorema es la siguiente propiedad que será de utilidad en el futuro

Teorema 18 *Dado un conjunto de nodos en posición general, dos nodos están unidos por una arista de un triángulo de la triangulación de Delaunay si sólo si existe una circunferencia que pasa por esos dos puntos, y que no contiene en su interior ningún otro nodo.*

Los triángulos de una triangulación de Delaunay pueden ser de buena o mala calidad geométrica, en función de la distribución geométrica de los nodos. Sin embargo el siguiente resultado muestra como de todas las triangulaciones posibles, la triangulación de Delaunay es la de mayor calidad geométrica.

Teorema 19 *Sea un conjunto de nodos en posición general, y una triangulación cualquiera T de esos nodos. Vamos a denotar por $\alpha_{\min}(T)$ al mínimo de todos los ángulos formados por las aristas de todos los triángulos de T . Entonces la triangulación de Delaunay verifica que de todas las triangulaciones posibles es la que maximiza $\alpha_{\min}(T)$. Es decir*

$$\alpha_{\min}(T) \leq \alpha_{\min}(T_{\text{Delaunay}}) \quad \forall T.$$

Ejemplo 7 *Vamos a ver lo que sucede si los puntos no están en posición general. El ejemplo de la figura 8 muestra el diagrama de Voronoi de un conjunto de puntos que no están en posición general*

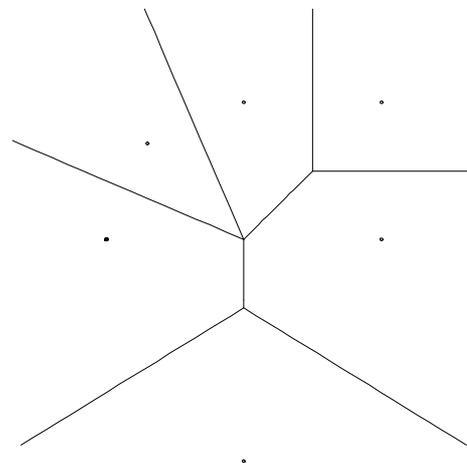


Figura 8:

en este caso hay 4 puntos que están sobre la misma circunferencia. Este hecho se detecta en que hay 4 rectas que se intersectan en un punto. lo cual indica que dicho punto es equidistante a 4 nodos simultáneamente y por tanto, sería el centro de una circunferencia que pasa por los 4 nodos. Si en este caso unimos por aristas los nodos cuyo conjunto de Voronoi V_i tienen una zona de frontera común obtendríamos como resultado la división del dominio dada por la figura 9. Por tanto, no obtenemos una división en triángulos pues nos queda un polihedro en medio. En este caso no podemos encontrar una triangulación a partir del diagrama de Voronoi, y se pierde la unicidad de las posibles triangulaciones utilizando el criterio de Delaunay pues las dos triangulaciones dadas por las figuras 10 y 11 serían de Delaunay

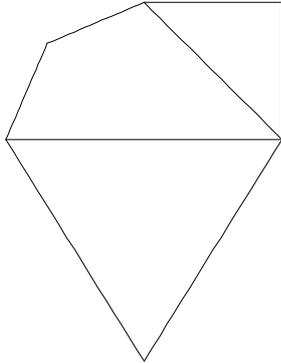


Figura 9:

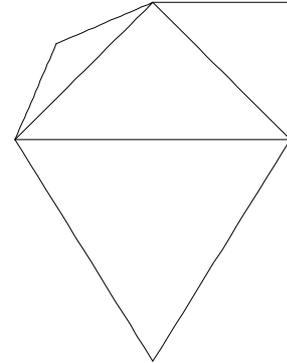


Figura 10:

PROCEDIMIENTO PARA INCLUIR UN PUNTO EN UNA TRIANGULACION DE DELAUNY.

Vamos ahora a abordar el problema de la construcción de la triangulación de Delaunay de una nube de puntos. En primer lugar abordaremos un problema más simple, que como veremos posteriormente será la clave para abordar el problema completo. Este problema más simple es: Dada una triangulación de Delaunay, $T = \{K_i\}$ donde K_i representan los triángulos y un punto v cualquiera que pertenece a alguno de los triángulos K_i . Pretendemos añadir el punto v como un nuevo nodo de la triangulación y modificar convenientemente los triángulos de tal forma que la nueva triangulación $T' = \{K'_i\}$ contenga al punto v como un nuevo nodo y sea a su vez una triangulación de Delaunay.

Definición 4 Llamaremos *cavidad de un punto v respecto a la triangulación T* a la región determinada por todos los triángulos K_i tales que el círculo circunscrito de K_i contiene a v .

Si queremos que la nueva triangulación T' sea también de Delaunay, evidentemente todos los triángulos K_i de la cavidad de v deben desaparecer en la nueva triangulación, y en su lugar aparecerá otra configuración de triángulos sustituyendo a los K_i eliminados. Por tanto la resolución del problema pasa por tres fases

1. Se añade el nodo v a la lista de nodos
2. Se eliminan todos los triángulos K_i de la cavidad de v
3. Se añaden nuevos triángulos para cubrir de nuevo la cavidad de v .

Por ejemplo, consideremos la triangulación dada en la figura 12, a la cual queremos añadir el punto representado en la figura. Como puede apreciarse hay tres triángulos cuyo círculo incluye al punto. La nueva triangulación T' quedaría como muestra la figura 13, donde hemos eliminado 3 triángulos y añadido 5 nuevos triángulos. Nótese que los nuevos triángulos están formados uniendo el nuevo punto v con los vértices de las aristas de los anteriores triángulos que son frontera de la cavidad de v . Una arista es frontera de la cavidad de v si pertenece a un sólo triángulo de la cavidad, si por el contrario la arista pertenece a dos triángulos de la cavidad entonces es interior a la cavidad

Por tanto un sencillo algoritmo para construir T' a partir de T y v sería el siguiente:

1. Se añade el nodo v a la lista de nodos
2. Se seleccionan los triángulos K_i de la cavidad de v
3. Para cada arista de un triángulo de la cavidad se comprueba si la arista está en otro triángulo de la cavidad. Si la arista sólo está en un triángulo se añade a T' el triángulo formado por el vértice v y los dos vértices de la arista
4. Se eliminan los triángulos de la cavidad de v .

Teorema 20 Si $T = \{K_i\}$ es una triangulación de Delaunay, y la familia de puntos dados por los nodos de T y el punto v están en posición general, entonces la triangulación T' determinada por el procedimiento anterior es de Delaunay.

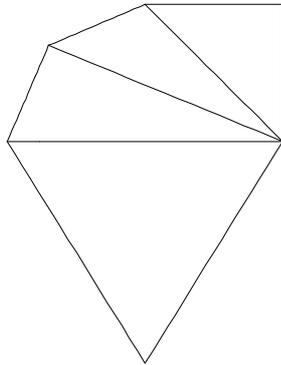


Figura 11:

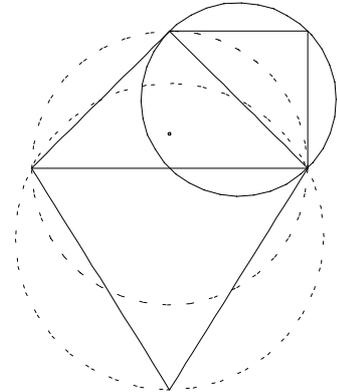


Figura 12:

Este teorema muestra que de forma teórica el procedimiento anterior funciona correctamente en el caso de puntos en posición general. En la práctica puede haber situaciones donde falle y dé lugar a triangulaciones incorrectas donde se solapen triángulos debido principalmente a los errores de redondeo en los cálculos. Veamos que tipo de fallos pueden aparecer con un ejemplo: Consideremos la triangulación de la figura 14.

Donde queremos añadir el punto representado por un pequeño círculo en la figura. En la práctica, debido a los errores de redondeo que se producen al trabajar con números reales en el ordenador, puede suceder que al hacer el test para comprobar si un triángulo K_i está en la cavidad de v , detectemos que el triángulo central no esté en la cavidad, pero sin embargo que el triángulo superior izquierdo sí esté en la cavidad. Si ello se produce, entonces al aplicar el algoritmo precedente obtendríamos una triangulación incorrecta.

Para eliminar este tipo de errores vamos a modificar ligeramente el concepto de cavidad de un punto v a través de un criterio de vecindad, siguiendo los siguientes pasos:

1. Calculamos un triángulo K_i que contenga al punto v (Nótese que v puede estar en dos triángulos en el caso en que caiga exactamente en una arista de la triangulación), y añadimos K_i a la cavidad. Inicialmente la cavidad estará formada por un único triángulo K_i .
2. Añadimos a la cavidad los triángulos vecinos a K_i (es decir que comparten una arista) tal que su círculo circunscrito incluya al punto v .
3. Hacemos el proceso anterior recursivo, es decir, añadimos recursivamente a la cavidad los triángulos vecinos

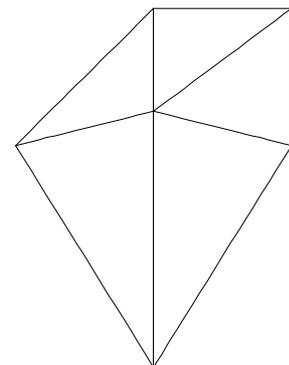


Figura 13:

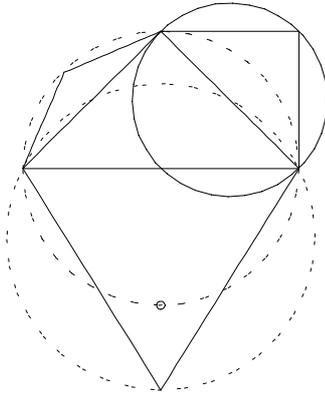


Figura 14:

a triángulos ya presentes en la cavidad. tales que su círculo circunscrito incluya a v .

Evidentemente al terminar este procedimiento obtendremos una cavidad por vecindad que podrá ser en algunos casos más pequeña que la cavidad normal de v sin tener en cuenta la vecindad. Justamente en el ejemplo anterior la cavidad calculada de esta forma no incluye al triángulo superior izquierda que está desconectado (por vecindad) del triángulo que incluye al punto. Por tanto si utilizamos esta nueva cavidad en el algoritmo inicial para introducir un punto el resultado sería una triangulación correcta. Nótese que en términos teóricos, esta nueva triangulación ya no sería de Delaunay, aunque está muy próxima a una triangulación de Delaunay, y de hecho la llamaremos triangulación de Delaunay. Si queremos a toda costa que la triangulación resultante sea de Delaunay en sentido estricto entonces, no podemos añadir puntos v tales que la cavidad por vecindad y la cavidad normal sean distintas. Lo que podemos hacer en este caso es variar ligeramente las coordenadas de v hasta encontrar que la cavidad por vecindad y la normal coincidan. Nótese que un pequeño cambio en las coordenadas de v en el ejemplo anterior puede modificar totalmente la cavidad.

TRIANGULACIONES DE DELAUNY CON RESTRICCIONES

Con frecuencia, como veremos posteriormente, es necesario imponer a una triangulación, no sólo que contenga a un conjunto de nodos, sino además que un conjunto de segmentos esté presente en la triangulación a través de

aristas de triángulos. Supondremos siempre que los interiores de los segmentos no se intersectan entre sí. La restricción que imponemos a la triangulación es justamente que deben aparecer en la triangulación estos segmentos a través de las aristas. Nuestro dato inicial será un conjunto de segmentos. Si tomamos como nodos los vértices de estos segmentos y calculamos la triangulación de Delaunay asociada, puede que esta triangulación contenga a los segmentos como aristas de triángulos o puede que no. El siguiente teorema nos muestra una condición que asegura que un segmento sea una arista de una triangulación de Delaunay.

Teorema 21 *Dados 2 nodos v_i y v_j de una triangulación. Si existe una circunferencia que pasa por estos dos nodos y que no contiene en su interior ningún otro nodo de la triangulación, entonces el segmento que une v_i con v_j es una arista de algún triángulo de la triangulación de Delaunay aplicada al conjunto de nodos*

La forma más sencilla de evaluar algorítmicamente este criterio viene dada por el siguiente teorema:

Teorema 22 *Dados 2 nodos v_i y v_j , si el interior de la circunferencia que pasa por los puntos v_i y v_j y tiene como centro el punto medio $v_m = \frac{v_i+v_j}{2}$ no contiene ningún otro nodo, entonces el segmento que une v_i con v_j es una arista de algún triángulo de la triangulación de Delaunay aplicada al conjunto de nodos.*

Utilizando estos resultados, resulta sencillo construir un procedimiento que asegure que los segmentos iniciales están en la triangulación. Para construir dicho procedimiento, realizaremos las siguientes etapas:

1. Recorro todos los segmentos. Para cada segmento compruebo si cumple la condición del teorema anterior. En el caso en que un nodo v_k esté incluido en la circunferencia de centro $\frac{v_i+v_j}{2}$, entonces cálculo la circunferencia que pasa por los puntos v_i, v_j y v_k , y compruebo si todos los demás nodos caen fuera de la esta circunferencia. En el caso en que algún nodo caiga dentro, añado como nodo a la triangulación el punto medio del segmento que une v_i y v_j , y sustituyo el segmento original por sus dos segmentos mitad.
2. Si al terminar de recorrer los segmentos, algún segmento se ha modificado en la etapa 1, vuelvo a empezar de nuevo la etapa 1

Si el interior de los segmentos no se intersecta entre sí, el procedimiento anterior converge en un número finito de etapas, es decir subdividiendo progresivamente los segmentos llegará un momento en que todos cumplan la

condición necesaria para que los segmentos estén representados como aristas de la triangulación. Nótese que con este procedimiento aseguramos que los segmentos iniciales estén presentes en la triangulación no como la arista de un único triángulo (lo cual es imposible de imponer), sino como la unión de un conjunto de aristas.

DISCRETIZACION INICIAL DE UN DOMINIO

Dado un dominio Ω que queremos triangular. La primera cuestión a resolver es cómo podemos describir Ω de una manera algorítmica. La manera habitual de proceder es describir Ω como una colección de polígonos orientados que determinan su frontera. En general necesitaremos tantos polígonos como componentes conexas tenga la frontera de Ω . En general, los polígonos se construyen recorriendo la frontera de Ω y tomando puntos consecutivos sobre dicha frontera, por convenio, para distinguir el exterior del interior de Ω , asumimos que recorremos la frontera en el sentido de las agujas del reloj. El interior de Ω va quedando a la derecha de cada segmento del polígono siguiendo la dirección de recorrido.

Ejemplo 8 Consideremos el dominio Ω dado en la figura 15 por la banda interior que queda entre los círculos concéntricos

$$\begin{aligned}x^2 + y^2 &= 1 \\x^2 + y^2 &= 4\end{aligned}$$

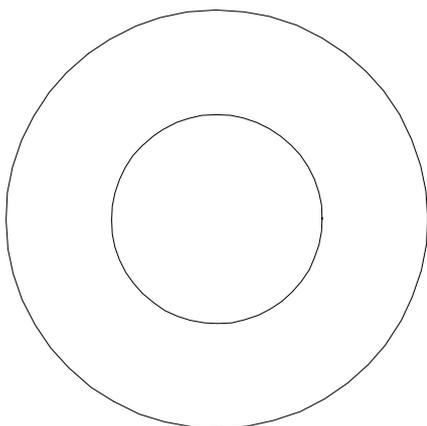


Figura 15:

una posible discretización de este dominio vendría dada por los polígonos

$$P_1 = \left\{ 2 \cos \left(\frac{2\pi k}{N} \right), 2 \sin \left(\frac{2\pi k}{N} \right) \right\}_{k=N, \dots, 0}$$

$$P_2 = \left\{ \cos \left(\frac{2\pi k}{N} \right), \sin \left(\frac{2\pi k}{N} \right) \right\}_{k=0, \dots, N}$$

donde N representa el número de puntos que tomamos sobre las circunferencias al recorrer la frontera. Nótese que la circunferencia exterior la recorreremos en el sentido de las agujas del reloj y la circunferencia interior en el sentido contrario. Nótese además que el primer punto y el último coinciden para indicar que se trata de polígonos cerrados.

Para almacenar en disco la familia de polígonos que determinan la frontera de un dominio Ω se utilizará un fichero que contiene en la primera línea el número de polígonos, a continuación en la segunda línea aparece el número de puntos del primer polígono (se tomarán polígonos cerrados, es decir el primer y último punto coinciden) a continuación en las siguientes líneas aparecen las coordenadas de los vértices del primer polígono, cada punto tiene asociado además un valor entero que será de utilidad para distinguir dos tipos de valores frontera cuando estudiemos el método de elementos finitos, de momento no prestaremos atención a este valor al cual daremos en el ejemplo el valor 1. Una vez terminado el primer polígono se procede de forma análoga con los polígonos siguientes. Por ejemplo si en el ejemplo anterior tomamos $N = 4$ para describir los polígonos P_1 y P_2 el fichero para almacenar P_1 y P_2 sería el siguiente:

```
2
5
2. 0. 1
0. -2. 1
-2. 0. 1
0. 2. 1
2. 0. 1
5
1. 0. 1
0. 1. 1
-1. 0. 1
0. -1. 1
1. 0. 1
```

Para almacenar los polígonos en memoria RAM se utilizará la siguiente estructura:

```
typedef struct{
    int npol;
    /* Número de polígonos */
    int *dimpol;
    /* Número de puntos de cada polígono */
    float ** x;
    /* Coordenadas x de los polígonos */
    float ** y;
    /* Coordenadas y de los polígonos */
    int ** fr;
    /* flat de frontera de cada punto */
}pol
```

De tal forma que en el ejemplo anterior tendríamos

$$\begin{aligned} npol &= 2 \\ \dim pol[0] &= 5 \\ \dim pol[1] &= 5 \end{aligned}$$

$$\begin{array}{lll} x[0][0] = 2 & y[0][0] = 0 & fr[0][0] = 1 \\ x[0][1] = 0 & y[0][1] = -2 & fr[0][1] = 1 \\ x[0][2] = -2 & y[0][2] = 0 & fr[0][2] = 1 \\ x[0][3] = 0 & y[0][3] = 2 & fr[0][3] = 1 \\ x[0][4] = 2 & y[0][4] = 0 & fr[0][4] = 1 \\ x[1][0] = 1 & y[1][0] = 0 & fr[1][0] = 1 \\ x[1][1] = 0 & y[1][1] = 1 & fr[1][1] = 1 \\ x[1][2] = -1 & y[1][2] = 0 & fr[1][2] = 1 \\ x[1][3] = 0 & y[1][3] = -1 & fr[1][3] = 1 \\ x[1][4] = 1 & y[1][4] = 0 & fr[1][4] = 1 \end{array}$$

Dentro del fichero *ana.h* existen funciones para leer y escribir polígonos. En el programa ejemplo *pol1.c* se utilizan estas funciones para crear en disco los 2 polígonos del ejemplo anterior. Se permite al usuario elegir el número de puntos para describir las circunferencias exteriores e interiores. En el programa ejemplo *pol2.c* se utilizan estas funciones para leer desde disco una familia de polígonos.

ESQUEMA GENERAL DE UN ALGORITMO DE TRIANGULACION SEGUN EL CRITERIO DE DELAUNY.

Hasta ahora hemos visto algunos procedimientos para incluir un nodo en una triangulación de Delaunay, asegurar que un conjunto de segmentos esté en la triangulación de Delaunay, y la manera de discretizar un dominio

Los datos de entrada a un algoritmo para triangular un dominio Ω es un conjunto de polinomios orientados que determinan una aproximación de la frontera de Ω , y un parámetro que llamaremos L_{\max} que determina la longitud máxima permitida para las aristas de la triangulación resultante. Normalmente L_{\max} se toma en base a la longitud de los segmentos de los polígonos que determinan la frontera. También se puede incluir como parámetro un valor Q_{\min} que indicará cual es el mínimo de calidad geométrica que imponemos a los triángulos.

El resultado final del algoritmo sería una triangulación del interior de Ω que incluirá, por supuesto, nodos interiores, y formada por triángulos de buena calidad geométrica, según los criterios vistos anteriormente. Las fases principales por las que el algoritmo debe pasar son

1. Se recorren todos los segmentos de los polígonos que determinan la frontera del dominio. Si la longitud L de un segmento dado por dos vértices v_i, v_j es mayor que L_{\max} , se toma N el número entero más próximo a L/L_{\max} y se calculan N puntos dados por

$$x_k = v_i + \frac{k}{N+1} (v_j - v_i) \quad k = 1, \dots, N$$

y se sustituye el segmento $\overline{v_i v_j}$ por los segmentos $\overline{v_i x_1}, \overline{x_1 x_2}, \overline{x_2 x_3}, \dots, \overline{x_N v_j}$. Si v_i y v_j tienen ambos el flat de frontera igual a 2, entonces a los x_i se les pone el flat de frontera a 2. En caso contrario se pone el flat de frontera a 1.

2. Se determina un rectángulo que incluya en su interior a todos los vértices de los segmentos. Dicho rectángulo se tomará lo suficientemente grande como para que los vértices de dicho rectángulo no estén incluidos nunca en el círculo que pasa por cada segmento de los polígonos iniciales y tiene como centro el punto medio del segmento
3. Se refinan los segmentos de los polígonos según el criterio de la sección anterior para asegurar que dichos segmentos estarán en la triangulación
4. Se van introduciendo utilizando el criterio de Delaunay todos los puntos (con su flat de frontera) de los polígonos de la frontera en la triangulación inicial formada por los dos triángulos que resultan de dividir el rectángulo inicial.
5. La triangulación resultante será una triangulación del rectángulo inicial que contendrá todos los polígonos de la frontera de Ω . A continuación se eliminan los nodos que corresponden a los vértices del rectángulo inicial, y también se eliminan todos los triángulos que tuviesen como vértice uno de esos nodos.
6. Se recorren los triángulos y se comprueba si tienen como arista a alguno de los segmentos del polígono inicial, en este caso se comprueba si el otro vértice del triángulo queda a la izquierda del segmento siguiendo la orientación con la que ha sido definido el polígono, en cuyo caso se eliminará dicho triángulo. Posteriormente se eliminan recursivamente los triángulos vecinos de los que ya han sido eliminados y que además no poseen una arista que sea segmento del polígono inicial.
7. Se recorren todos los triángulos, si la longitud L de una arista dada por dos vértices v_i, v_j es mayor que L_{\max} , se toma N el número entero más próximo a L/L_{\max} y se introducen como nuevos nodos N puntos dados por

$$x_k = v_i + \frac{k}{N+1} (v_j - v_i) \quad k = 1, \dots, N$$

a los nuevos nodos, como son interiores se les asigna el flat de frontera a 0.

8. Si al recorrer todos los triángulos en la etapa 7 se ha modificado alguno se vuelve a empezar la etapa 7.
9. Se recorren los triángulos K_i y se calcula el test de calidad $Q(K_i)$. Si $Q(K_i) > Q_{\min}$ entonces se toma la arista de longitud mayor del triángulo y se introduce como nuevo nodo su punto medio siguiendo el criterio de Delaunay.

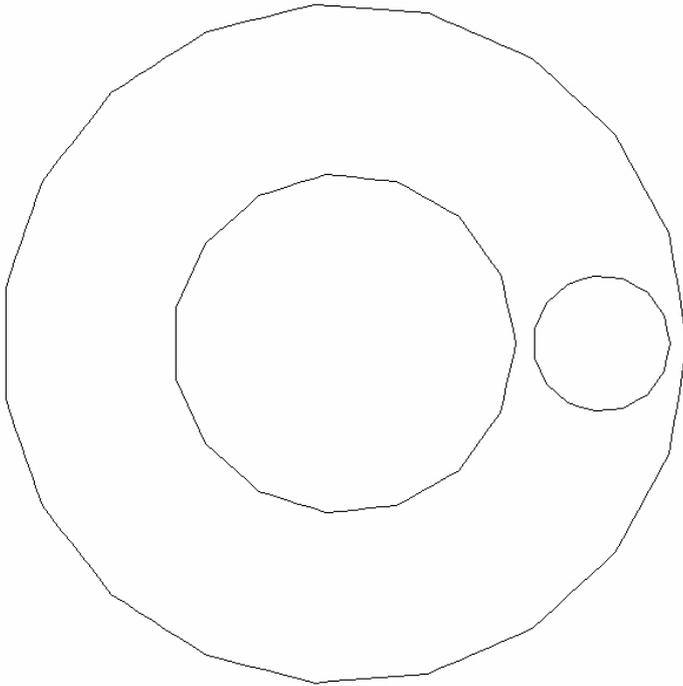


Figura 16: Poligonos Iniciales. El dominio está formado por la zona que se encuentra entre las dos circunferencias concéntricas.

10. Si al recorrer todos los triángulos en la etapa 9 alguno ha sido modificado, entonces se vuelve a empezar la etapa 9.

Las etapas 9 y 10 son opcionales, hay que tener en cuenta que si se toma Q min muy pequeño se corre el riesgo de entrar en un bucle infinito. Además normalmente los refinamientos de la etapa 7 – 8 dan buenos resultados a nivel del aspecto geométrico de los triángulos.

Nota: Una observación importante es que si el conjunto de nodos resultante está en posición general, entonces la triangulación de Delaunay asociada es única, y por tanto no depende del orden en que se introduzcan los nodos en el rectángulo inicial.

A continuación se presentarán algunos resultados numéricos de triangulaciones. En las figuras ??, ??, ?? y ?? se encuentra un ejemplo de las diferentes etapas por las que pasa el algoritmo de triangulación en el caso de un dominio formado por el interior de dos circunferencias concéntricas e incluyendo un hueco circular.

En las figuras 20, 21, 22, 23, se muestran las fases por las que pasa el algoritmo de triangulación a partir de un polígono que determina el contorno exterior de la isla de

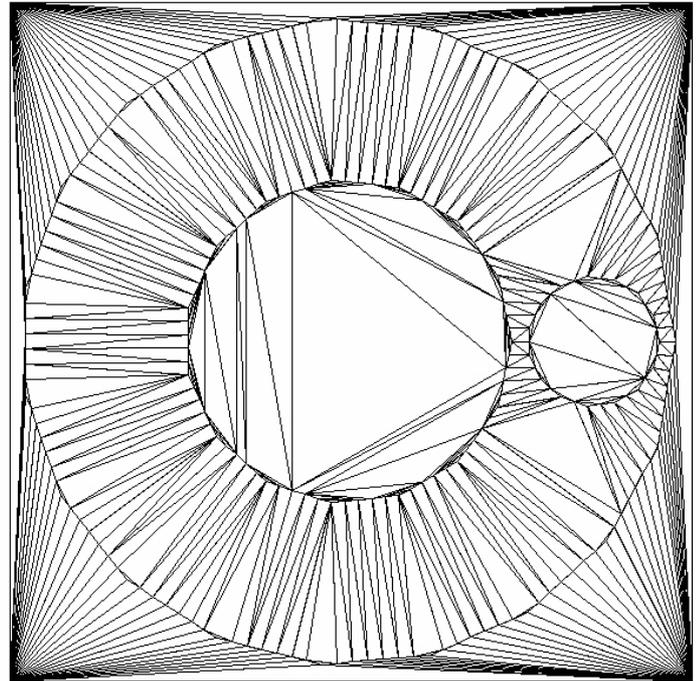


Figura 17: Triangulación después de las etapas 1-2-3-4 del proceso de triangulación de Delaunay

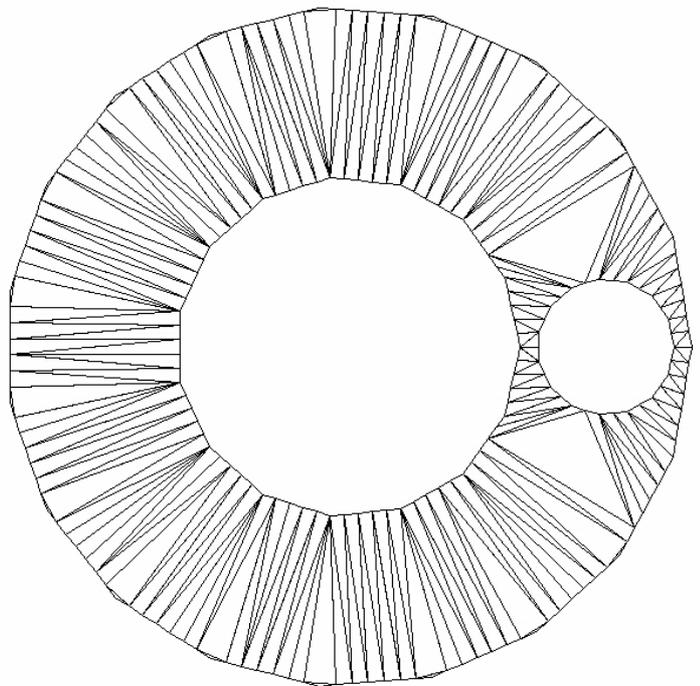


Figura 18: Triangulación después de la etapa 6

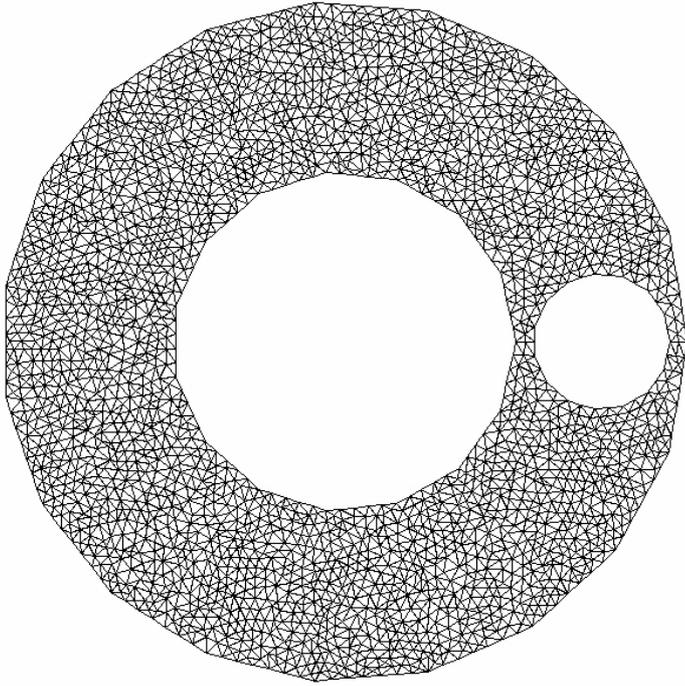


Figura 19: Triángulación después de las etapas 7-8

Gran Canaria. En la figura 24 se muestra un resultado de la triángulación obtenida utilizando un valor más pequeño de la variable L_{\max} .

Nota: Un problema que puede plantearse al aplicar las etapas 7 y 8 se produce cuando el valor de L_{\max} es muy pequeño respecto al tamaño de los triángulos presentes en la triángulación, pues al intentar dividir de una sola vez en trozos muy pequeños los triángulos muy grandes, la calidad geométrica final de los triángulos resultantes empeora. Una técnica para evitar esto consiste, y que ha sido utilizada en los ejemplos mostrados, consiste aplicar las etapas 7 y 8 de forma recursiva empezando primero por un valor grande de L_{\max} y luego ir refinando la triángulación con valores cada vez más pequeños de L_{\max} hasta llegar a la triángulación final. Concretamente basta con tomar

$$L_{\max}^n = 2^n L_{\max}$$

y realizar las etapas 7 – 8 con L_{\max}^n sucesivamente para $n = N, \dots, 0$ donde N se elige en función del tamaño de los triángulos iniciales, en los ejemplos mostrados se ha tomado $N = 4$.

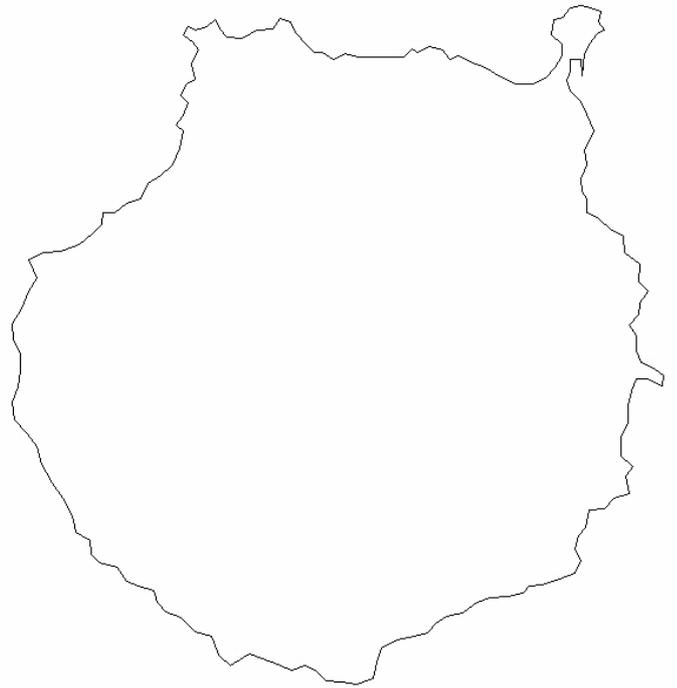


Figura 20: Polígono inicial dado por el contorno de la isla de Gran Canaria

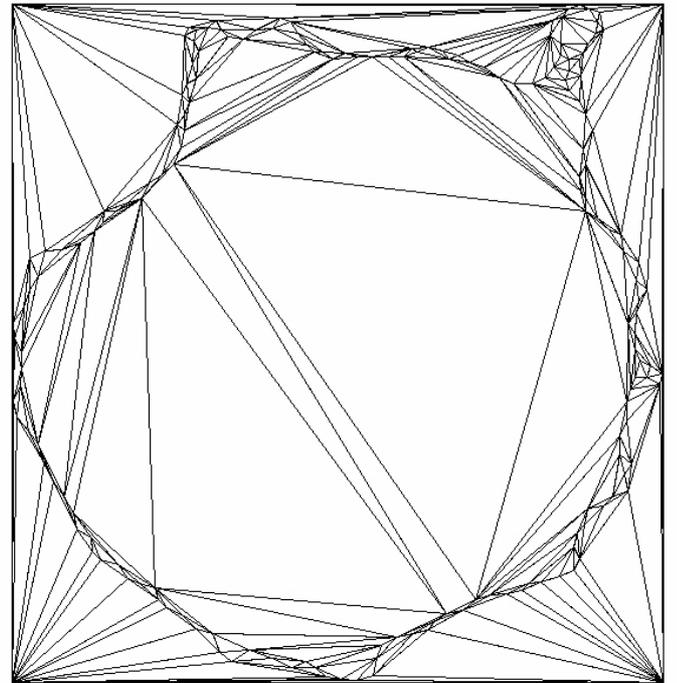


Figura 21: Triángulación después de las etapas 1-2-3-4.

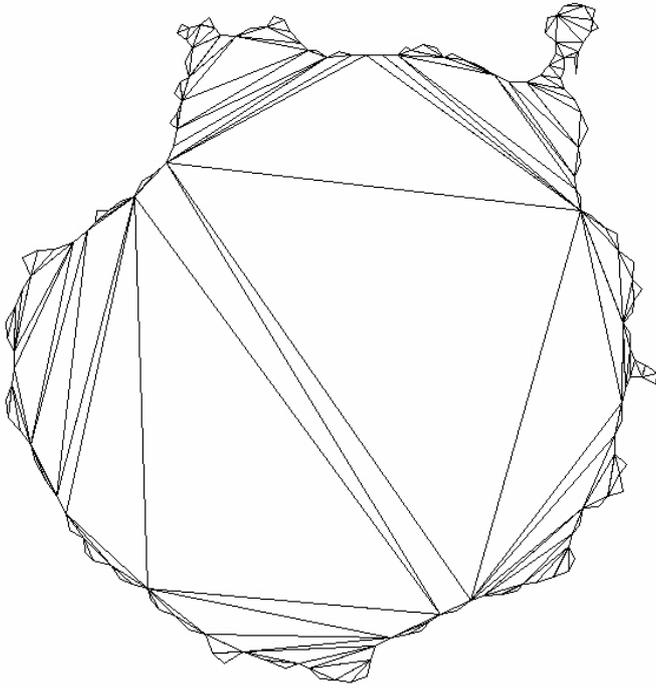


Figura 22: Triangulación después de las etapas 5-6.

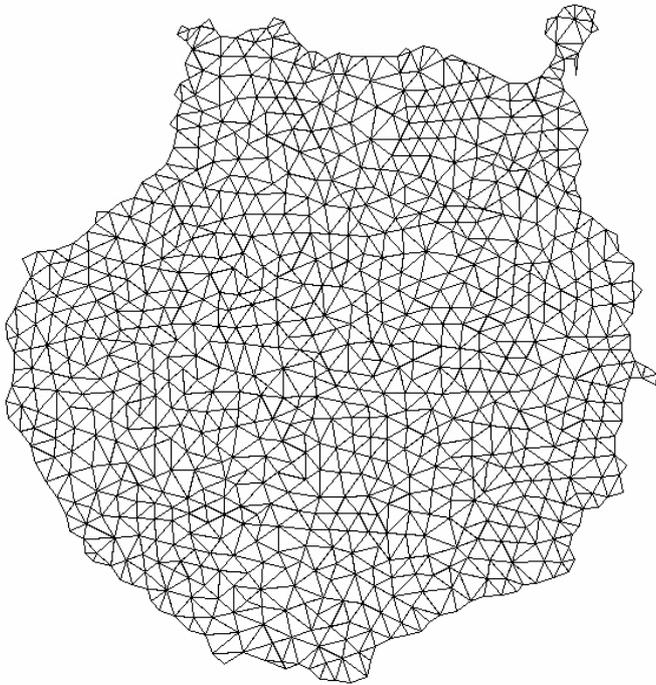


Figura 23: Triangulación después de las etapas 7-8.

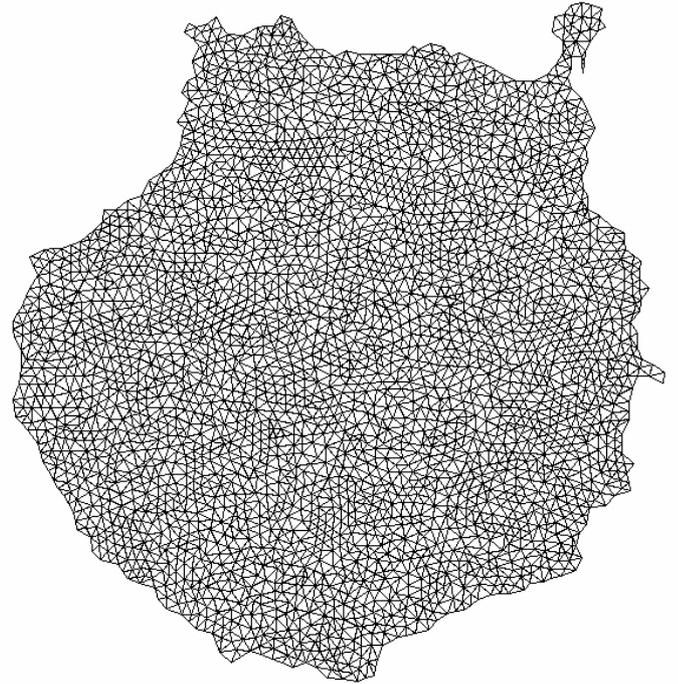


Figura 24: Triangulación utilizando un valor de L max más pequeño que en la figura anterior

PRACTICA 3. TRIANGULACIONES DE DELAUNY

En esta práctica se implementará el método de Delaunay para triangular dominios planos. Dado que la mejor manera de comprobar que el método funciona correctamente es visualizando en pantalla las triangulaciones, se utilizará una función de visualización que se encuentra en el directorio de la asignatura.

Para implementar la práctica se irán construyendo de forma modular diferentes funciones en C para realizar las diferentes etapas del algoritmo definido en la sección precedente. Habrá que realizar como mínimo los siguientes tests:

1. Realizar una triangulación del dominio de la figura 15 tomando un número diverso de puntos sobre cada circunferencia.
2. Tomar el polígono que determina la costa de Gran Canaria del directorio de la asignatura y calcular un mallado de la isla

OPTIMIZACION DEL COSTO COMPUTACIONAL

El procedimiento que acabamos de describir para realizar el mallado de un dominio requiere un gran número de cálculos que hacen computacionalmente costoso el procedimiento cuando el número de nodos es grande. Las técnicas

para acelerar el procedimiento se basan en la utilización de la información de vecindad.

La información de vecindad de la que disponemos es la que a cada triángulo asocia los índices en la lista de triángulos de los triángulos que comparten una arista con él. Veamos las ventajas y los inconvenientes de utilizar dicha información de vecindad.

En primer lugar analizaremos el procedimiento de introducir un nuevo punto en una malla utilizando el criterio de Delauny. Lo primero que debemos hacer es buscar un triángulo que contenga al punto. Para ello, en principio debemos recorrer toda la lista de triángulos hasta encontrar uno que lo contenga. Este proceso se puede agilizar de la siguiente manera. Se toma un primer triángulo de forma aleatoria, se toma también de forma aleatoria uno de los tres nodos del triángulo. Se recorren las aristas del triángulo a partir de ese nodo, si el punto v y el vértice opuesto a una arista están en lados distintos de la arista (es decir $R_{arista}(v)R_{arista}(v_{opuesto}) < 0$, si los triángulos están orientados basta con preguntar $R_{arista}(v) > 0$) entonces nos movemos al triángulo vecino que comparte dicha arista, tomamos de nuevo un nodo de forma aleatoria y volvemos a recorrer las aristas, y así sucesivamente nos vamos desplazando de triángulo en triángulo. De esta forma los triángulos estarán cada vez más cerca del punto v . Cuando para un triángulo ninguna arista cumpla la condición anterior, dicho triángulo será el triángulo buscado que incluye el punto. Si el dominio Ω tiene agujeros o concavidades, este procedimiento puede bloquearse al intentar pasar a un triángulo vecino inexistente. En dicho caso se volvería a empezar tomando otro triángulo inicial de forma aleatoria.

Nótese que si la relación de vecindad está ordenada, el paso de un triángulo a otro en el procedimiento anterior es mucho más rápido que si la relación de vecindad no está ordenada.

También resulta de gran utilidad la relación de vecindad para encontrar la cavidad por vecindad de un punto, pues a través de las relaciones de vecindad recorro rápidamente los triángulos vecinos para hacer el test del círculo circunscrito y encontrar dicha cavidad por vecindad.

Las objeciones a la utilización de los criterios de vecindad es que requieren un espacio de memoria suplementaria para almacenar las relaciones de vecindad, y además, al introducir un nuevo punto, desaparecen y aparecen nuevos triángulos y por tanto tendremos que actualizar las relaciones de vecindad. Sin embargo ello no es muy complicado puesto que los únicos triángulos donde hay que actualizar a los vecinos son: por un lado en los nuevos triángulos, para los cuales tenemos que definir sus vecinos, y en los triángulos vecinos a los triángulos que desaparecen donde habrá que reemplazar el índice del triángulo que desaparece, por el índice del nuevo triángulo que lo sustituye. Por tanto

el número de triángulos afectados a los que hay que actualizar las relaciones de vecindad va a ser pequeño, y por tanto su costo computacional bajo. Por tanto como podemos observar, el costo computacional de definir y gestionar las relaciones de vecindad es insignificante respecto a las grandes ventajas que se obtienen en términos de aumento de velocidad del algoritmo

TRIANGULACIONES DE DOMINIOS TRIDIMENSIONALES.

En esta sección, prestaremos especial atención a las diferencias en las técnicas desarrolladas en el capítulo anterior para dominios bidimensionales al pasar a dominios tridimensionales. En primer lugar, en $3 - D$ los elementos de referencias básicos son los tetraedros. Empezaremos recordando algunas nociones básicas sobre tetraedros.

NOCIONES GEOMETRICAS BASICAS SOBRE TETRAEDROS

Un tetraedro $T = (v_0, v_1, v_2, v_3)$ en dimensión 3 viene dado por 4 vértices no coplanarios $v_0 = (x_0, y_0, z_0)$, $v_1 = (x_1, y_1, z_1)$, $v_2 = (x_2, y_2, z_2)$, $v_3 = (x_3, y_3, z_3)$. A continuación expondremos algunos resultados de interés sobre tetraedros.

Teorema 23 *El volumen de un tetraedro T viene dado por la expresión:*

$$Vol(T) = \frac{1}{6} ABS \begin{pmatrix} 1 & 1 & 1 & 1 \\ x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ z_0 & z_1 & z_2 & z_3 \end{pmatrix}$$

Teorema 24 *Existe una única esfera circunscrita que pasa por los cuatro vértices de un tetraedro T la ecuación de esfera, viene dada en forma implícita como:*

$$\begin{vmatrix} l_0^2 - l^2 & l_1^2 - l_0^2 & l_2^2 - l_0^2 & l_3^2 - l_0^2 \\ x_0 - x & x_1 - x_0 & x_2 - x_0 & x_3 - x_0 \\ y_0 - y & y_1 - y_0 & y_2 - y_0 & y_3 - y_0 \\ z_0 - z & z_1 - z_0 & z_2 - z_0 & z_3 - z_0 \end{vmatrix} = 0$$

donde $l^2 = x^2 + y^2 + z^2$ y $l_i^2 = x_i^2 + y_i^2 + z_i^2$.

Teorema 25 *El radio ρ_K de la esfera inscrita en un tetraedro viene dado por*

$$\rho_T = \frac{3 \cdot Vol(T)}{S_1 + S_2 + S_3 + S_4}$$

donde S_i representa la superficie de una cara del tetraedro

Teorema 26 Dados tres puntos en el espacio no colineales (v_0, v_1, v_2) definidos por los vértices $v_0 = (x_0, y_0, z_0)$, $v_1 = (x_1, y_1, z_1)$ y $v_2 = (x_2, y_2, z_2)$. La expresión:

$$P_{(v_0, v_1, v_2)}(x, y, z) = \begin{vmatrix} x - x_0 & y - y_0 & z - z_0 \\ x_1 - x_0 & y_1 - y_0 & z_1 - z_0 \\ x_2 - x_0 & y_2 - y_0 & z_2 - z_0 \end{vmatrix}$$

separa el espacio en 3 regiones distintas dadas por: $P_{(v_0, v_1, v_2)}(x, y, z) = 0$ que representa el plano que pasa por los tres puntos v_0, v_1 y v_2 , $P_{(v_0, v_1, v_2)}(x, y, z) > 0$ que representa el semiespacio hacia el que apunta la normal al plano siguiendo la regla del sacacorchos para el producto vectorial, y por último el semiespacio $P_{(v_0, v_1, v_2)}(x, y, z) < 0$ que es el semiespacio opuesto al anterior.

Teorema 27 El área de un triángulo K en el espacio formado por los vértices $v_0 = (x_0, y_0, z_0)$, $v_1 = (x_1, y_1, z_1)$ y $v_2 = (x_2, y_2, z_2)$ viene dado por la expresión

$$AREA(K) = \frac{\|e_1 \times e_2\|}{2}$$

donde $e_1 = (x_1 - x_0, y_1 - y_0, z_1 - z_0)$, $e_2 = (x_2 - x_0, y_2 - y_0, z_2 - z_0)$, y $e_1 \times e_2$ denota el producto vectorial de los vectores e_1 y e_2 .

Teorema 28 Dado un punto cualquiera (x, y, z) , y un tetraedro cualquiera T , entonces (x, y, z) está en el interior o en la frontera del tetraedro $T = (v_0, v_1, v_2, v_3)$ si sólo si

$$\begin{aligned} P_{(v_0, v_1, v_2)}(x, y, z) \cdot P_{(v_0, v_1, v_2)}(v_3) &\geq 0 \\ P_{(v_0, v_1, v_3)}(x, y, z) \cdot P_{(v_0, v_1, v_3)}(v_2) &\geq 0 \\ P_{(v_0, v_3, v_2)}(x, y, z) \cdot P_{(v_0, v_3, v_2)}(v_1) &\geq 0 \\ P_{(v_3, v_1, v_2)}(x, y, z) \cdot P_{(v_3, v_1, v_2)}(v_0) &\geq 0 \end{aligned}$$

Teorema 29 Dado un triángulo $K = (v_0, v_1, v_2)$, en el espacio y un segmento $S = (v_3, v_4)$ entonces el segmento intersecta al triángulo si sólo si.

$$\begin{aligned} P_{(v_0, v_1, v_2)}(v_3) \cdot P_{(v_0, v_1, v_2)}(v_4) &< 0 \\ P_{(v_0, v_1, v_3)}(v_2) \cdot P_{(v_0, v_1, v_3)}(v_4) &> 0 \\ P_{(v_0, v_3, v_2)}(v_1) \cdot P_{(v_0, v_3, v_2)}(v_4) &> 0 \\ P_{(v_3, v_1, v_3)}(v_0) \cdot P_{(v_3, v_1, v_3)}(v_4) &> 0 \end{aligned}$$

Al igual que para los triángulos también existen medidas para cuantificar la calidad geométrica de un tetraedro. Una medida de calidad geométrica del tetraedro comúnmente utilizada es la siguiente:

$$Q(T) = \frac{\text{longitud arista mayor}}{\text{radio esfera inscrita}}$$

Nota: El concepto intuitivo de calidad geométrica de un tetraedro es mucho más complejo que el caso de un triángulo. Por ejemplo, dada una circunferencia y tres puntos sobre ella, entonces si el área del triángulo formado por los tres puntos que se mueven en la circunferencia tiende hacia 0, entonces la longitud de alguno de los lados del triángulo tiende hacia 0. Sin embargo, en el caso de una esfera, si tomamos 4 puntos sobre la esfera, puede ser que el volumen del tetraedro sea muy pequeño (tan pequeño como se quiera) sin que ninguna de las aristas del tetraedro tienda hacia 0.

Ejemplo 9 Consideremos el tetraedro T dado por los siguientes vértices $v_0 = (0, 0, l\frac{\sqrt{2}}{\sqrt{3}})$, $v_1 = (l\frac{1}{\sqrt{3}}, 0, 0)$, $v_2 = (-l\frac{1}{2\sqrt{3}}, l\frac{1}{2}, 0)$, y $v_3 = (-l\frac{1}{2\sqrt{3}}, -l\frac{1}{2}, 0)$. Se puede comprobar fácilmente que la longitud de las aristas del tetraedro son de tamaño l , y por tanto el área de las caras del tetraedro es $S = l^2 \frac{\sqrt{3}}{4}$

El volumen de T viene dado por:

$$Vol(T) = \frac{1}{6} ABS \begin{vmatrix} 1 & 1 & 1 & 1 \\ 0 & l\frac{1}{\sqrt{3}} & -l\frac{1}{2\sqrt{3}} & -l\frac{1}{2\sqrt{3}} \\ 0 & 0 & l\frac{1}{2} & -l\frac{1}{2} \\ l\frac{\sqrt{2}}{\sqrt{3}} & 0 & 0 & 0 \end{vmatrix} = \frac{\sqrt{2}}{12} l^3$$

El radio de la esfera inscrita siguiendo la fórmula dada anteriormente es: $\rho_T = \frac{\frac{\sqrt{2}}{4} l^3}{l^2 \sqrt{3}}$, y por tanto la medida de calidad $Q(T)$ para este tetraedro da:

$$Q(T) = \frac{4\sqrt{3}}{\sqrt{2}} = 2\sqrt{6}$$

Del ejemplo anterior deducimos que la calidad geométrica no depende del tamaño del tetraedro y que además el valor óptimo para $Q(T)$ es $2\sqrt{6}$.

TRIANGULACIONES

Se considera un dominio Ω en dimensión 3. Una triangulación del dominio Ω es una colección de tetraedros cerrados $\{T_i\}_{i=1, \dots, Nt}$, verificando las siguientes condiciones

1. $\Omega = \cup_{i=1, \dots, Nt} T_i$
2. $T_i \cap T_j = \begin{cases} \emptyset \\ \text{una cara completa} \end{cases}$

ALMACENAMIENTO DE UNA TRIANGULACION

Hay pocas diferencias con respecto al caso bidimensional, sólo debemos tener en cuenta que en este caso cada nodo tiene asociadas tres coordenadas, en lugar de 2, los tetraedros vienen dados por 4 índices de la tabla de nodos, y para

las relaciones de vecindad necesitamos para cada tetraedro 4 índices de la tabla de tetraedros que indican cuáles son los tetraedros que comparten una cara con uno dado

TRIANGULACIONES DE DELAUNY DE UN DOMINIO 3-D

Por analogía con el caso bidimensional, diremos que una triangulación de un conjunto de nodos $3 - D$ es de Delauny si el interior de la esfera circunscrita que pasa por un tetraedro de la triangulación no contiene ningún nodo del dominio. Podemos asociar a un conjunto de puntos $3 - D$ un diagrama de Voronoi, al igual que en caso bidimensional, y se cumplen las mismas propiedades que en $2 - D$. Como en el caso bidimensional necesitamos en primer lugar un procedimiento para incluir un punto en una triangulación de Delauny dada. Este procedimiento sigue los mismos pasos que en $2 - D$, es decir, dado un nodo v que queremos incluir en la triangulación, seguiremos los siguientes pasos:

1. Calculamos un tetraedro T_i que contenga al punto v , y añadimos T_i a la cavidad. Inicialmente la cavidad estará formada por un único tetraedro T_i .
2. Añadimos a la cavidad los tetraedros vecinos a T_i (es decir que comparten una cara) tal que su esfera circunscrita incluya al punto v .
3. Hacemos el proceso anterior recursivo, es decir, añadimos recursivamente a la cavidad los tetraedros vecinos a tetraedros ya presentes en la cavidad. tales que su esfera circunscrita incluya a v

Con respecto a las restricciones que podemos imponer a una triangulación tenemos, en primer lugar la restricción de que un segmento $3 - D$ esté representado en la triangulación a través de las aristas de los tetraedros. Esto se puede conseguir, al igual que en $2 - D$, subdividiendo el segmento de tal forma que al final, para cada segmento, la esfera de centro el punto medio de los segmentos y radio la mitad de la longitud del segmento no incluya ningún nodo de la triangulación. En segundo lugar, también se pueden imponer restricciones del tipo de que un conjunto de triángulos $3 - D$ estén representados en la triangulación a través de caras de tetraedros, pero esta restricción es mas compleja de analizar, y no la abordaremos en este curso.

Donde si que hay cambios importantes respecto a $2 - D$ es en la descripción de la frontera de un dominio Ω . En $3 - D$ es evidente que ya no podemos describir la frontera como una unión de polígonos orientados. Vamos a dar 3 posibles formas para describir la frontera de un dominio tridimensional:

1. Un conjunto de triángulos $3 - D$ orientados en el siguiente sentido, para cada triángulo dado por los vértices v_1, v_2 y v_3 , el producto vectorial $(v_2 - v_1) \times (v_3 - v_1)$ determina una dirección normal que apunta hacia el exterior del dominio Ω . De esta forma, a través del orden en que se dan los vértices de los triángulos, podemos deducir a que lado del triángulo se encuentra el interior del dominio y que lado el exterior del dominio
2. Un conjunto de nodos situados sobre la frontera de Ω , y para cada nodo un punto $3 - D$ que representa un punto del exterior de Ω desde donde es visible el nodo.
3. Un conjunto de segmentos $3 - D$, y para cada vértice de cada segmento, un punto del exterior del dominio desde donde es visible el vértice.

A la forma que prestaremos más atención es a la forma 2 que es la manera más natural de describir una frontera $3 - D$ cuando se trata de objetos reales. Por ejemplo, cuando nosotros observamos un objeto $3 - D$ realmente tenemos 2 informaciones importantes, la primera es lo que vemos del objeto, y otra desde donde lo vemos. Normalmente es necesario mirar un objeto desde diferentes puntos de vista para tener una panorámica global del objeto. Hay dos situaciones concretas reales que nos dan una información de este tipo. En primer lugar si apuntamos con un rayo laser el objeto y vamos moviendo el laser alrededor del objeto, la información de la que disponemos para cada posición donde pongamos el laser es el punto donde el rayo laser toca el objeto, y el punto desde donde sale el rayo laser. Otra situación real donde aparece esta información es en la visión estereográfica. En este caso tenemos dos imágenes distintas de una misma escena (por ejemplo nuestra visión binocular es un prototipo de visión estereográfica). En este caso, a partir de las dos imágenes es posible reconstruir la posición $3 - D$ de la parte del objeto visible por las 2 cámaras. En este caso, para cada punto de la frontera tenemos dos puntos exteriores (los focos de las cámaras) desde donde el punto es visible.

Como en el caso bidimensional, para realizar el malla del objeto tridimensional necesito introducir el conjunto de nodos en un dominio auxiliar inicial dado, en este caso por un cubo. Para ello se calculan las coordenadas mínimas y máximas en las tres variables del conjunto de nodos de la frontera y a partir de esta información se determina el cubo inicial. Para evitar al máximo las posibles interacciones de la nube de puntos con los vértices del cubo, dicho cubo inicial se toma estrictamente mayor que el cubo mínimo dado por los máximos y mínimos de las coordenadas.

A partir del cubo inicial, lo primero que tenemos que hacer es construir una triangulación por tetraedros del cubo inicial. Supongamos que los vértices del cubo inicial vienen dados por

$$\begin{aligned} v_1 &= (-C + x_{\min}, -C + y_{\min}, -C + z_{\min}) \\ v_2 &= (C + x_{\max}, -C + y_{\min}, -C + z_{\min}) \end{aligned}$$

$$\begin{aligned}
v_3 &= (-C + x_{\min}, C + y_{\max}, -C + z_{\min}) \\
v_4 &= (C + x_{\max}, C + y_{\max}, -C + z_{\min}) \\
v_5 &= (-C + x_{\min}, -C + y_{\min}, C + z_{\max}) \\
v_6 &= (C + x_{\max}, -C + y_{\min}, C + z_{\max}) \\
v_7 &= (-C + x_{\min}, C + y_{\max}, C + z_{\max}) \\
v_8 &= (C + x_{\max}, C + y_{\max}, C + z_{\max})
\end{aligned}$$

donde los subíndices min y max indican los máximos y mínimos de las coordenadas de la nube de puntos. C es una constante positiva que se sirve para alejar los vértices del cubo de la nube de puntos.

La triangulación del cubo inicial requiere 6 tetraedros que se construyen de la siguiente forma:

$$\begin{aligned}
T_1 &= (v_1, v_2, v_3, v_5) \\
T_2 &= (v_4, v_2, v_3, v_5) \\
T_3 &= (v_4, v_7, v_3, v_5) \\
T_4 &= (v_7, v_5, v_4, v_8) \\
T_5 &= (v_4, v_2, v_5, v_8) \\
T_6 &= (v_6, v_2, v_8, v_5)
\end{aligned}$$

Con la presentación que hemos realizado en esta sección ya estamos en condiciones de dar un procedimiento completo para determinar el mallado de un dominio $3-D$. Los datos de entrada del procedimiento será la descripción de la frontera del dominio Ω que supondremos expresada en la forma 2. También utilizaremos como parámetro L_{\max} que determina la longitud máxima de los aristas de los tetraedros y Q_{\min} que determinará la calidad geométrica mínima de los tetraedros.

1. Se determina un cubo inicial y su mallado inicial a través de tetraedros que incluya en su interior a todos los nodos.
2. Se van introduciendo utilizando el criterio de Delauny todos los puntos de la frontera de Ω en la triangulación inicial.
3. La triangulación resultante será una triangulación del cubo inicial que contendrá todos los puntos de la frontera de Ω . A continuación se eliminan los nodos que corresponden a los vértices del cubo inicial, y también se eliminan todos los tetraedros que tuviesen como vértice uno de esos nodos.
4. Se recorren los puntos de la frontera de Ω y se eliminan los tetraedros tales que alguna de sus caras interseque al segmento que une el punto en cuestión, con el punto desde donde es visible dicho punto, es decir elimino los tetraedros que no me dejan ver el punto. Además si el tetraedro contiene al punto como uno de los vértices, hay que comprobar también si el punto desde donde es visible está incluido dentro del tetraedro. En cuyo caso se eliminaría.

5. Se recorren todos los tetraedros, si la longitud de una arista es mayor que L_{\max} , se introduce como nuevo nodo, el punto medio de la arista utilizando el criterio de Delauny
6. Si al recorrer todos los tetraedros en la etapa 5 se ha modificado alguno se vuelve a empezar la etapa 5.
7. Se recorren los tetraedros T_i y calcula el test de calidad $Q(T_i)$. Si $Q(T_i) > Q_{\min}$ entonces se toma la arista de longitud mayor del tetraedro y se introduce como nuevo nodo su punto medio siguiendo el criterio de Delauny.
8. Si al recorrer todos los tetraedros en la etapa 7 alguno ha sido modificado, entonces se vuelve a empezar la etapa 7.

Las etapas 7 y 8 son opcionales, hay que tener en cuenta que si se toma Q_{\min} muy pequeño se corre el riesgo de entrar en un bucle infinito. Además normalmente los refinamientos de la etapa 5 – 6 dan buenos resultados a nivel del aspecto geométrico de los tetraedros.

Para ilustrar la técnica de triangulación de Delauny en $3-D$ vamos a utilizar un conjunto de puntos generados por la reconstrucción $3-D$ de la geometría de la cara de una persona a partir de un par estéreo. Concretamente las imágenes de partida son el par estéreo dado por las figuras 25 y 26. A partir de estas 2 imágenes (utilizando técnicas de visión estéreo) se calcula una nube de puntos $3-D$ que corresponden a la cara de la pesona, en este caso Javier Sánchez. Esta nube de punto es la etapa inicial de nuestro algoritmo, en la figura 27 se representa la triangulación del cubo inicial con la inclusión de la nube de puntos. En la figura 28, se encuentra el resultado de la triangulación después de eliminar los tetraedros que tienen algún vértice del cubo inicial, en las figuras 29, 30 y 31 se muestran 3 vistas distintas de la triangulación después de aplicar el algoritmo de visibilidad de la etapa 4. En este caso los puntos desde donde son visibles los nodos de la triangulación son los focos de las cámaras.

Una aplicación importante de los mallados de dominios $3-D$ es la simulación de escenas reales en animación por ordenador. Para simular en el ordenador una escena como puede ser un barco navegando en el mar, necesito tener una descripción completa de la estructura exterior del barco en términos de triángulos. $3-D$, es decir necesito un mallado de la estructura del barco, que es justamente la información que suministra el procedimiento de mallado que acabamos de describir. Además, en este caso, para



Figura 25: Primera imagen del par estéreo.

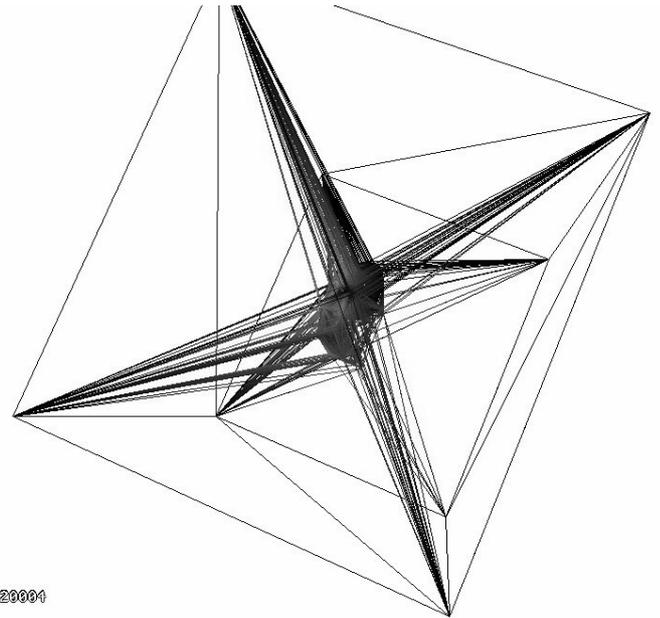


Figura 27: Triangulación 3 - D del cubo inicial después de las etapas 1 - 2 - 3.

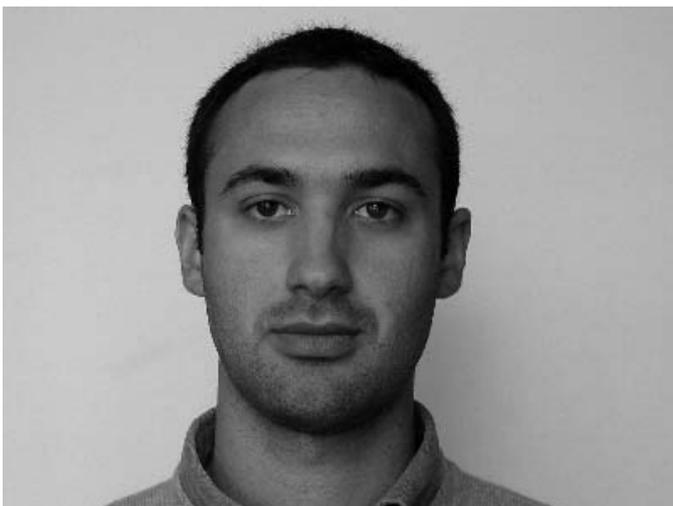


Figura 26: Segunda imagen del par estéreo

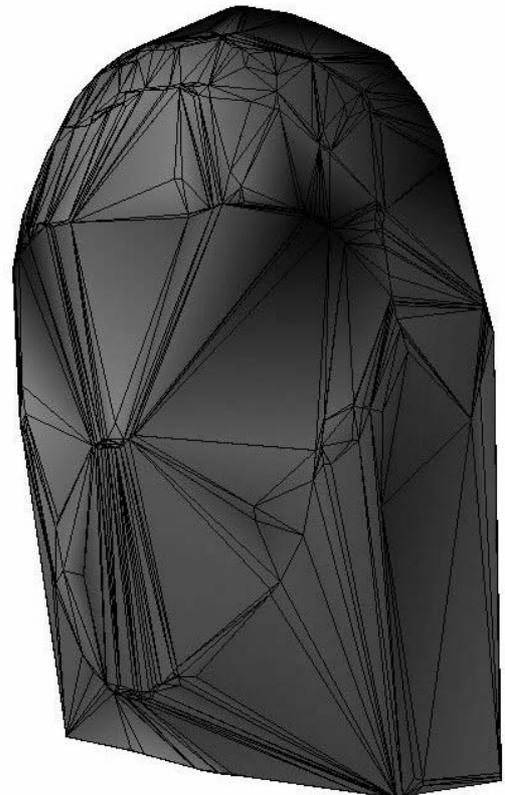


Figura 28: Envoltura Convexa de la familia de puntos después de eliminar los tetraedros que contienen algún vértice del cubo.

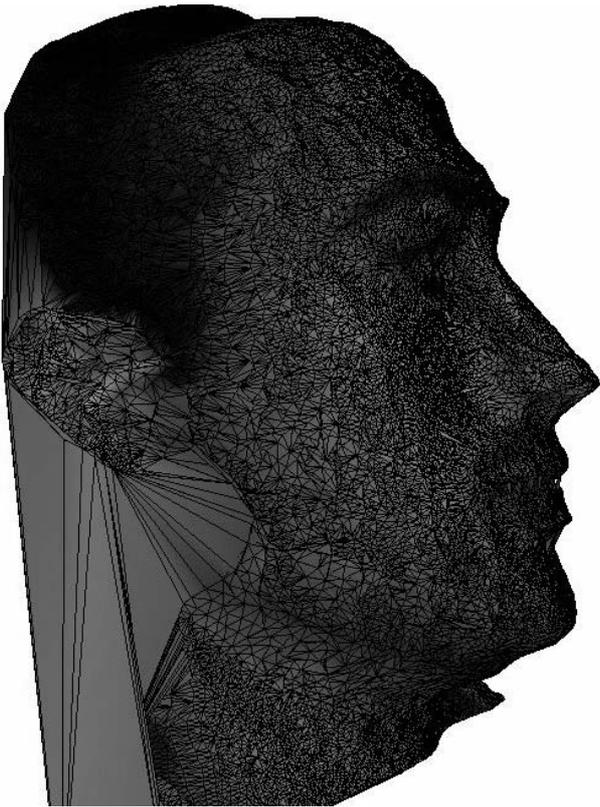


Figura 29: Vista de la triangulación después de la etapa 4.



Figura 30: Vista de la triangulación después de la etapa 4.



Figura 31: Vista de la triangulación después de la etapa 4.

visualizar el barco navegando no necesito guardar la información de todos los tetraedros. De hecho sólo necesito la información de los triángulos de los tetraedros formados por caras exteriores al dominio. Ello es así dado que las caras de los tetraedros interiores al dominio nunca son visibles desde el exterior.

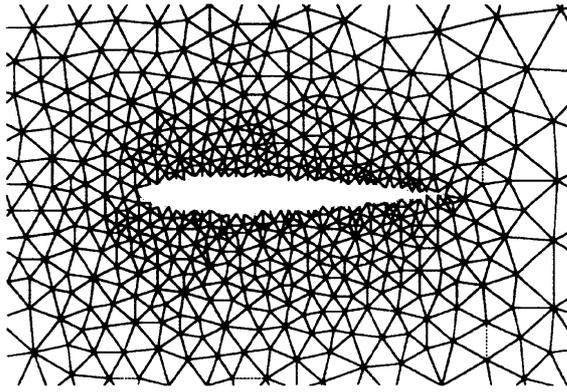
La utilización de las relaciones de vecindad, al igual que en el caso de los dominios $2 - D$ permiten acelerar considerablemente la velocidad de los algoritmos. Además la forma de gestionar las relaciones de vecindad sigue los mismos pasos que en $2 - D$

EL METODO DE ELEMENTOS FINITOS .

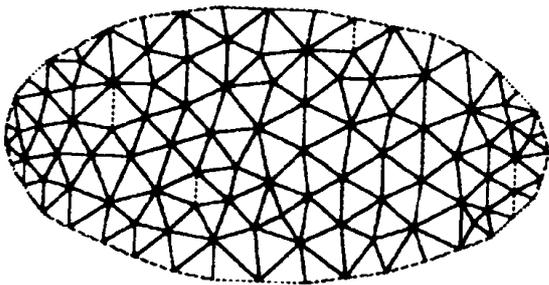
INTRODUCCION

El método de elementos finitos es una técnica que permite simular numéricamente en el ordenador el comportamiento de un sistema físico. De entre las numerosas aplicaciones en ingeniería y física que tiene este método, caben destacar las siguientes

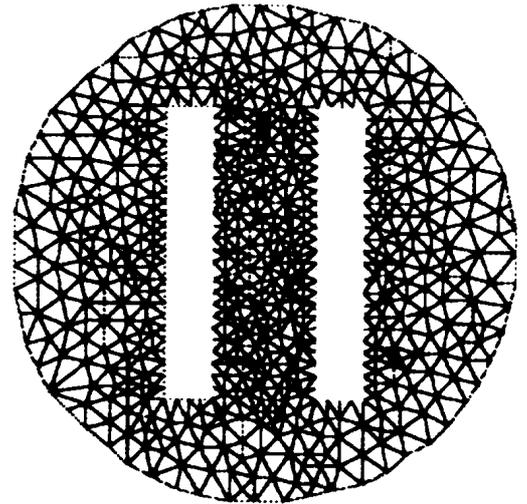
1. **Simulación del comportamiento aerodinámico de un perfil de ala de avión.** Una de las primeras aplicaciones del método de elementos finitos fue desarrollada por científicos americanos de la empresa BOEING para simular en el ordenador el comportamiento aerodinámico de un ala de avión. En este caso se estudian los cambios de presión generados en el aire cuando el ala lo atraviesa. En la siguiente figura se muestra un ejemplo del perfil de ala denominado NACA0012



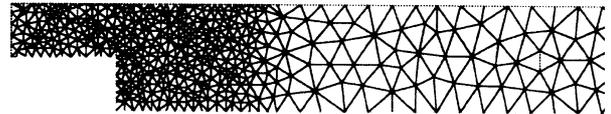
2. **Problemas de elasticidad.** Un ejemplo de aplicación clásica del método de elementos finitos es la simulación en ordenador de las deformaciones elásticas de un objeto sometido a algún tipo de presión. Un ejemplo particularmente sencillo es el estudio de la deformación elástica de una superficie fija a sus bordes (por ejemplo la piel de un tambor), bajo la acción de una presión exterior. En la siguiente figura se muestra un ejemplo de dominio que determina la superficie elástica a estudiar



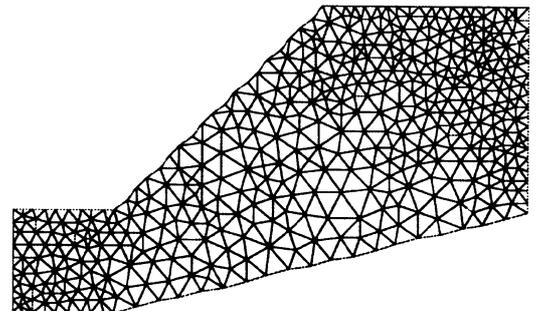
3. **Electrostática.** Otro ejemplo interesante de aplicación del método de elementos finitos es el cálculo del potencial electrostático generado por un condensador. En la siguiente figura se muestra un ejemplo de condensador.



4. **Mecánica de fluidos.** El estudio del movimiento de un fluido es otro de los grandes campos de aplicaciones del método de elementos finitos. Un problema clásico a analizar es el estudio del comportamiento de un fluido al atravesar un escalón descendente. La siguiente figura muestra un ejemplo de discretización de un dominio que simula un escalón por el que caerá un fluido.



5. **Acústica.** Otro ejemplo de aplicación interesante del método de elementos finitos es la simulación de la propagación de las ondas acústicas en un recinto cerrado. Este problema es de gran interés para el diseño de auditorios. En la siguiente figura se presenta una sección transversal de un auditorio donde el escenario estaría en la parte baja de la figura.



Para simular numéricamente el comportamiento de los sistemas físicos anteriormente descritos es necesario establecer en todos los casos unas condiciones iniciales (es decir los datos de entrada) que determinarán la magnitud física que queremos analizar. Dichas condiciones son:

1. Establecer un dominio Ω donde se va a estudiar el problema. En discreto, dicho dominio vendrá representado por una colección de triángulos y nodos, tal y como hemos visto en el capítulo anterior.
2. Un modelo físico, que habitualmente viene dado por una ecuación diferencial que determina el comportamiento de la solución del problema
3. Unas condiciones de contorno sobre la frontera del dominio, que determinan el comportamiento de la solución en dicha frontera
4. Un conjunto de funciones o magnitudes que determinan la influencia de los elementos exteriores al sistema (por ejemplo, la presión exterior ejercida sobre la malla elástica en el caso del problema de elasticidad).

A partir de estos datos de entrada, se simulará numéricamente la solución del problema. Habitualmente la simulación suministra un valor aproximado de la solución en los nodos de la triangulación.

Dada la orientación de este curso hacia alumnos de ingeniería en informática, para presentar el método de elementos finitos, vamos a partir de un problema modelo, que iremos desarrollando haciendo especial hincapié en los aspectos numéricos y algoritmos del problema, y pasando de puntillas sobre los aspectos más teóricos de modelización matemática de los problemas presentados. El problema modelo que vamos a desarrollar es el modelo de elasticidad donde se estudia la deformación de una superficie elástica bajo la presión de una fuerza en un dominio Ω . En este caso la incógnita a calcular será una función $u(x, y)$ que determina la amplitud de la deformación de la superficie en el punto $(x, y) \in \Omega$, en la dirección perpendicular a Ω . La ecuación en derivadas parciales que modeliza el fenómeno es

$$-div(k(x, y)\nabla u) + a(x, y)u = f$$

donde $f(x, y)$ representa la presión exterior ejercida sobre la malla elástica, la función $k(x, y) > 0$, determina el coeficiente de elasticidad en cada punto de la malla, que indica la resistencia de la malla a ser deformada, así por ejemplo un valor grande de $k(x, y)$ indica una fuerte resistencia a deformarse, y por el contrario, un valor pequeño de $k(x, y)$ indica que en ese punto la malla se deforma con facilidad. La función $a(x, y) \geq 0$ es un coeficiente de absorción interno que indica la absorción de energía que la malla realiza en el proceso de deformación. Así por ejemplo, un valor grande de $a(x, y)$ indica que en esa zona la malla absorbe mucha energía al deformarse, lo cual redundará en una

menor deformación, y por el contrario un valor de $a(x, y)$ pequeño indica poca absorción de energía, y por tanto, mayor deformación elástica.

El operador diferencial $\nabla u(x, y)$ determina el vector de derivadas parciales:

$$\nabla u(x, y) = \left(\begin{array}{c} \frac{\partial u}{\partial x}(x, y) \\ \frac{\partial u}{\partial y}(x, y) \end{array} \right)$$

el operador diferencial $div(\cdot)$ actúa sobre una función vectorial $F(x, y) = (F_1(x, y), F_2(x, y))$ y viene definido por

$$div(F(x, y)) = \frac{\partial F_1}{\partial x}(x, y) + \frac{\partial F_2}{\partial y}(x, y)$$

es decir que

$$div(k\nabla u)(x, y) = \frac{\partial(k\frac{\partial u}{\partial x})}{\partial x}(x, y) + \frac{\partial(k\frac{\partial u}{\partial y})}{\partial y}(x, y)$$

Además, para determinar completamente la solución del problema, debemos añadir la información sobre las condiciones de contorno. El contorno viene determinado por la frontera de Ω , que denotaremos por $\partial\Omega$. Hay dos tipos de condiciones de contorno que se utilizan habitualmente. La condición de contorno de tipo Dirichlet, donde se fija el valor de la deformación elástica $u(x, y)$ en la $\partial\Omega$, es decir se impone

$$u(x, y) = D(x, y) \quad \forall (x, y) \in \partial\Omega$$

y la condición de contorno de tipo Neumann donde se fija el valor de la derivada de la solución $u(x, y)$ en la dirección de la normal exterior de la $\partial\Omega$, es decir se impone

$$\frac{\partial u}{\partial \bar{n}}(x, y) = N(x, y) \quad \forall (x, y) \in \partial\Omega$$

la derivada en la dirección normal indica cual es el sentido del flujo entre el interior y el exterior del dominio Ω . Por ejemplo, en el caso de que estemos estudiando un proceso de transferencia de calor, si $\frac{\partial u}{\partial \bar{n}}(x, y) > 0$, indica que la placa está cediendo calor al exterior. Por el contrario, si $\frac{\partial u}{\partial \bar{n}}(x, y) < 0$, indica que la placa está absorbiendo calor del exterior. El caso $\frac{\partial u}{\partial \bar{n}}(x, y) = 0$, indica que no hay transferencia de calor entre el exterior y el interior, ello se produce, por ejemplo cuando se sitúa un elemento aislante en el borde de la placa.

En ocasiones, resulta de interés mezclar las dos condiciones de contorno. Ello da lugar a la denominada condición de contorno de Robin que se puede escribir como:

$$D(x, y)u(x, y) + \frac{\partial u}{\partial \bar{n}}(x, y) = N(x, y) \quad \forall (x, y) \in \partial\Omega$$

En la práctica y siguiendo las características del problema a estudiar podemos tener un tipo de condición de contorno sobre una parte de $\partial\Omega$, y otro tipo de condición

de contorno sobre otra parte de $\partial\Omega$. En este caso denotaremos por Γ_1 la parte de la frontera donde tenemos una condición Dirichlet, y Γ_2 la parte de frontera donde tenemos una condición del tipo Neumann.

Resumiendo, el problema modelo completo a través del cual vamos a desarrollar el método de elementos finitos es:

$$-\operatorname{div}(k(x,y)\nabla u) + a(x,y)u = f \quad (1)$$

$$\begin{aligned} u(x,y) &= D(x,y) \quad \forall (x,y) \in \Gamma_1 \\ \frac{\partial u}{\partial \vec{n}}(x,y) &= N(x,y) \quad \forall (x,y) \in \Gamma_2 \end{aligned} \quad (2)$$

El problema modelo (1)-(2) puede estudiarse en cualquier dimensión espacial. Nosotros estudiaremos en detalle el caso de dimensión 2, y veremos cómo el análisis es básicamente el mismo en otras dimensiones.

FORMULACION VARIACIONAL DEL PROBLEMA MODELO

Para poder abordar la resolución numérica del problema modelo (1)-(2) debemos expresar en primer lugar dicho problema bajo otra forma equivalente, mejor adaptada a nuestros propósitos. Dicha forma equivalente se denomina formulación variacional del problema. Para describir esta formulación variacional debemos introducir algunos espacios de funciones. En lo que sigue, Ω representará un dominio en dimensión cualquiera

Definición 5 El espacio de funciones $L^2(\Omega)$ es el conjunto de funciones $f : \Omega \rightarrow \mathbb{R}$ dado por

$$L^2(\Omega) = \left\{ f : \int_{\Omega} |f|^2 < \infty \right\}$$

para que este espacio esté definido correctamente desde un punto de vista matemático es necesario hacerlo a través de la integral de Lebesgue. Aquí no entraremos en esos detalles, y supondremos, para fijar ideas que las integrales son en el sentido habitual de Riemann.

Definición 6 El espacio de funciones $H^1(\Omega)$ es el conjunto de funciones $f \in L^2(\Omega)$, tal que todas sus derivadas $\frac{\partial f}{\partial x_i}$ pertenecen a $L^2(\Omega)$.

$$H^1(\Omega) = \left\{ f \in L^2(\Omega) : \int_{\Omega} \left| \frac{\partial f}{\partial x_i} \right|^2 < \infty \right\}$$

Definición 7 El espacio de funciones $H_{D,\Gamma_1}^1(\Omega)$ es el conjunto de funciones $f \in H^1(\Omega)$ tal que f coincide con la función $D(x,y)$ en la parte de la frontera dada por Γ_1

$$H_{D,\Gamma_1}^1(\Omega) = \{ f \in H^1(\Omega) : f \equiv D \text{ en } \Gamma_1 \}$$

En las definiciones que estamos introduciendo, existen, ocultos a primera vista, problemas matemáticos de gran envergadura. Por ejemplo, decir que una función f que pertenece a $H^1(\Omega)$ vale algo sobre su frontera requiere un análisis matemático bastante complejo. Nosotros no entraremos aquí en este tipo de detalles, y supondremos siempre que este tipo de definiciones son siempre correctas.

Definición 8 El espacio de funciones $H_0^1(\Omega)$ es el conjunto de funciones $f \in H^1(\Omega)$ tal que f es igual a 0 en $\partial\Omega$.

$$H_0^1(\Omega) = \{ f \in H^1(\Omega) : f \equiv 0 \text{ en } \partial\Omega \}$$

Definición 9 El espacio de funciones $H_{0,\Gamma_1}^1(\Omega)$ es el conjunto de funciones $f \in H^1(\Omega)$ tal que f es nula en la parte de la frontera dada por Γ_1

$$H_{0,\Gamma_1}^1(\Omega) = \{ f \in H^1(\Omega) : f \equiv 0 \text{ en } \Gamma_1 \}$$

Con estas definiciones ya podemos establecer la denominada formulación variacional del problema modelo (1)-(2).

Teorema 30 Si $u(x,y)$ es la solución del problema modelo (1)-(2), entonces para toda función $\phi \in H_{0,\Gamma_1}^1(\Omega)$ se verifica

$$\int_{\Omega} k(x,y)\nabla u \nabla \phi + \int_{\Omega} a(x,y)u\phi = \int_{\Omega} f\phi + \int_{\Gamma_2} k(x,y)N\phi \quad (3)$$

Demostración: Si multiplicamos la ecuación (1) por la función ϕ e integramos en Ω obtenemos:

$$\int_{\Omega} -\operatorname{div}(k(x,y)\nabla u)\phi + \int_{\Omega} a(x,y)u\phi = \int_{\Omega} f\phi$$

Ahora bien, utilizando el teorema de la divergencia obtenemos que

$$\int_{\Omega} -\operatorname{div}(k(x,y)\nabla u)\phi = - \int_{\Gamma} k(x,y)\frac{\partial u}{\partial \vec{n}}\phi + \int_{\Omega} k(x,y)\nabla u \nabla \phi$$

como $\phi \in H_{0,\Gamma_1}^1(\Omega)$, se tiene que $\phi \equiv 0$ sobre Γ_1 y como $\frac{\partial u}{\partial \vec{n}}(x,y) = N(x,y)$ sobre Γ_2 se obtiene de forma inmediata por sustitución, la formulación variacional (3)

DISCRETIZACION POR ELEMENTOS FINITOS DE ORDEN 1.

En esta sección, vamos a aplicar el método de elementos finitos a la discretización y resolución numérica del problema modelo. En primer lugar calculamos una triangulación de Ω según vimos en la sección anterior. El método de elementos finitos de orden 1, consiste en aproximar la solución $u(x, y)$ del problema modelo de la siguiente forma:

$$u(x, y) \cong u_h(x, y) = \sum_i u_i \phi_i(x, y)$$

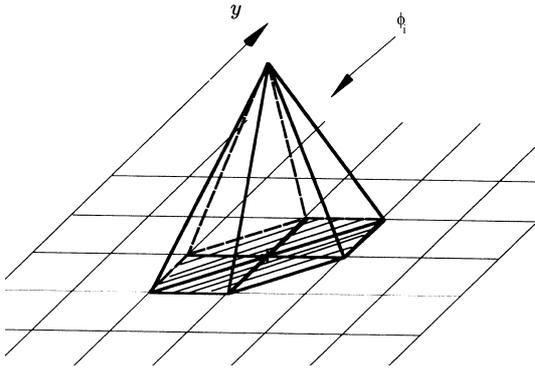
donde el índice i recorre los nodos de la triangulación, u_i representa un valor aproximado de la solución en el nodo i , es decir

$$u_i \cong u(x_i, y_i)$$

la función $\phi_i(x, y)$ es una función centrada en el nodo i que cumple las siguientes condiciones

1. $\phi_i(x_j, y_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$
2. $\phi_i(x, y) = a_n x + b_n y + c_n$ en cada triángulo K_n de la triangulación

la denominación de triángulos de orden 1, viene del hecho de que las funciones $\phi_i(x, y)$ vienen dadas por polinomios de grado 1. La función $\phi_i(x, y)$ constituye geoméricamente una pirámide centrada en el nodo i , cuyos lados son planos que van descendiendo hasta hacerse 0 en las aristas opuestas al nodo i de los triángulos que incluyen al nodo i como vértice. En la siguiente figura puede observarse una gráfica de la función $\phi_i(x, y)$.



Teorema 31 Dado un triángulo K_n de vértices $v_i = (x_i, y_i)$, $v_j = (x_j, y_j)$ y $v_m = (x_m, y_m)$, la función $\phi_i(x, y)$ centrada en el nodo i en el triángulo K_n viene dada por la expresión:

$$\phi_i(x, y) = \frac{(y - y_m)(x_j - x_m) - (x - x_m)(y_j - y_m)}{(y_i - y_m)(x_j - x_m) - (x_i - x_m)(y_j - y_m)}$$

En primer lugar, observamos que dadas las propiedades de las funciones $\phi_i(x, y)$, si aproximamos $u(x, y)$ por la combinación

$$u_h(x, y) = \sum_j u_j \phi_j(x, y)$$

tenemos que al evaluar $u_h(x, y)$ en los nodos (x_i, y_i) obtenemos $u_h(x_i, y_i) = u_i$. Ahora, las constantes u_i son las nuevas incógnitas que queremos calcular. Por otro lado en los nodos i que se encuentran en la frontera Γ_1 , que tiene condiciones de Dirichlet, conocemos ya el valor de $u(x, y)$, y por tanto para esos nodos podemos hacer directamente la asignación:

$$u_h(x_i, y_i) = D(x_i, y_i) \quad \forall (x_i, y_i) \in \Gamma_1$$

vamos ahora a reemplazar la función $u(x, y)$ por $u_h(x, y)$ en la formulación variacional del problema modelo desarrollada en el capítulo anterior, obteniendo

$$\sum_j u_j \left(\int_{\Omega} k \nabla \phi_j \nabla \phi + \int_{\Omega} a \phi_j \phi \right) = \int_{\Omega} f \phi + \int_{\Gamma_2} k(x, y) N \phi$$

para toda función $\phi \in H_{0, \Gamma_1}^1(\Omega)$. Nótese que para cualquier nodo i que no pertenezca a la frontera Γ_1 la función ϕ_i pertenece a $H_{0, \Gamma_1}^1(\Omega)$, puesto que ϕ_i y sus primeras derivadas son de cuadrado integrables, y además $\phi_i(x, y) = 0$ sobre Γ_1 , teniendo en cuenta que ahora Γ_1 viene dada por la colección de aristas de los triángulos de la triangulación que unen a puntos de Γ_1 . por tanto podemos hacer $\phi = \phi_i$ en la formulación variacional anterior, obteniendo para todos los nodos i que no pertenecen a Γ_1

$$\begin{aligned} \sum_j u_j \left(\int_{\Omega} k(x, y) \nabla \phi_j \nabla \phi_i + \int_{\Omega} a(x, y) \phi_j \phi_i \right) &= \\ &= \int_{\Omega} f \phi_i + \int_{\Gamma_2} k(x, y) N \phi_i \end{aligned}$$

La función $N(x, y)$ que determina la condición de Neumann vamos a aproximarla también por una función $N_h(x, y)$ de la siguiente forma

$$N_h(x, y) = \sum_l n_l \phi_l(x, y)$$

donde $n_l = N(x_l, y_l)$. Por tanto, sustituyendo $N(x, y)$ por $N_h(x, y)$ en la formulación variacional anterior, obtenemos

$$\begin{aligned} \sum_j u_j \left(\int_{\Omega} k(x, y) \nabla \phi_j \nabla \phi_i + \int_{\Omega} a(x, y) \phi_j \phi_i \right) &= \\ &= \int_{\Omega} f \phi_i + \sum_l n_l \int_{\Gamma_2} k(x, y) \phi_l \phi_i \end{aligned}$$

por lo tanto, hemos llegado a un sistema de ecuaciones donde las incógnitas son u_j , la matriz $A = (a_{ij})$ y el vector de términos independientes $b = (b_i)$ que determinan el sistema son:

1. Si el nodo i está en Γ_1

$$\begin{aligned} a_{ii} &= 1 \\ a_{ij} &= 0 \text{ si } i \neq j \\ b_i &= D(x_i, y_i) \end{aligned}$$

2. Si el nodo i no está en Γ_1

$$a_{ij} = \int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j$$

$$b_i = \int_{\Omega} f \phi_i + \sum_l n_l \int_{\Gamma_2} k(x,y) \phi_l \phi_i$$

por tanto hemos llegado a un sistema de ecuaciones de dimensión el número de nodos, y cuya solución determinará la solución del problema aproximado. Sin embargo, una dificultad importante es que el sistema que sale no es simétrico, debido a que si un nodo i pertenece a Γ_1 , y un nodo j no pertenece a Γ_1 , entonces $a_{ij} = 0$, pero a_{ji} puede ser distinto de cero. Para evitar esto, y poder aplicar el método del gradiente conjugado, lo que haremos, será trabajar con un sistema aproximado que sí es simétrico, definido de la siguiente forma:

1. Para cualquier par de nodos distintos i, j

$$a_{ij} = \int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j$$

2. Si el nodo i está en Γ_1

$$a_{ii} = M$$

$$b_i = D(x_i, y_i) M$$

3. Si el nodo i no está en Γ_1

$$a_{ii} = \int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_i + \int_{\Omega} a(x,y) \phi_i \phi_i$$

$$b_i = \int_{\Omega} f \phi_i + \sum_l n_l \int_{\Gamma_2} k(x,y) \phi_l \phi_i$$

donde M es una constante positiva que será muy grande respecto a los valores de a_{ij} , Este nuevo sistema es simétrico, y cuando $M \rightarrow \infty$, el sistema resultante es equivalente al inicial.

Nota: Otra forma alternativa de convertir en simétrico el sistema es eliminar las incógnitas que corresponden a los nodos de Γ_1 de la siguiente manera: Se sustituye u_i por $D(x_i, y_i)$ en los nodos de Γ_1 y se despeja en la formulación variacional del problema obteniendo

$$\sum_{j \notin \Gamma_1} u_j \left(\int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j \right) =$$

$$\int_{\Omega} f \phi_i + \sum_l n_l \int_{\Gamma_2} \phi_l \phi_i -$$

$$- \sum_{j \in \Gamma_1} D(x_j, y_j) \left(\int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j \right)$$

de tal forma que obtenemos el siguiente sistema de ecuaciones simétrico:

Si un nodo $i \in \Gamma_1$ entonces

$$a_{ii} = 1$$

$$b_i = D(x_i, y_i)$$

Si un nodo $i \notin \Gamma_1$ entonces

$$a_{ii} = \int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_i + \int_{\Omega} a(x,y) \phi_i \phi_i$$

$$b_i = \int_{\Omega} f \phi_i + \sum_l n_l \int_{\Gamma_2} k(x,y) \phi_l \phi_i -$$

$$\sum_{j \in \Gamma_1} D(x_j, y_j) \left(\int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j \right)$$

Si dados dos nodos distintos i, j alguno está en Γ_1 entonces

$$a_{ij} = 0$$

Si $i \notin \Gamma_1$ y $j \notin \Gamma_1$ entonces:

$$a_{ij} = \int_{\Omega} k(x,y) \nabla \phi_i \nabla \phi_j + \int_{\Omega} a(x,y) \phi_i \phi_j$$

ASPECTOS NUMERICOS EN LA CONSTRUCCION DEL SISTEMA DE ECUACIONES

En primer lugar, para evaluar numéricamente las integrales sobre Ω , utilizaremos el hecho de que la integral sobre Ω es igual a la suma de las integrales sobre los triángulos en que está dividido Ω . Por tanto, para construir numéricamente el sistema de ecuaciones, tenemos que dar algún procedimiento para calcular integrales numéricas sobre triángulos. Estos procedimientos ya se estudiaron en la asignatura Análisis Numérico. La técnica que utilizaremos aquí es una fórmula de integración numérica sobre triángulos que utiliza 3 puntos. Dicha fórmula viene dada por:

$$\int_K F(x,y) \cong AREA(K) \sum_{k=1}^3 w_k F(\tilde{x}_k, \tilde{y}_k)$$

donde

$$w_1 = w_2 = w_3 = \frac{1}{3}$$

$$\tilde{x}_1 = \frac{x_0 + x_1}{2} \quad \tilde{y}_1 = \frac{y_0 + y_1}{2}$$

$$\tilde{x}_2 = \frac{x_0 + x_2}{2} \quad \tilde{y}_2 = \frac{y_0 + y_2}{2}$$

$$\tilde{x}_3 = \frac{x_2 + x_1}{2} \quad \tilde{y}_3 = \frac{y_2 + y_1}{2}$$

Otra integral que debemos calcular para evaluar b_i es

$$\int_{\Gamma_2} \phi_i \phi_j$$

para calcular esta integral utilizaremos una extensión de la fórmula de Simpson de integración sobre un segmento. Sea $v_n = (x_n, y_n)$, y $v_l = (x_l, y_l)$ los vértices de un segmento y $v_m = \left(\frac{x_n + x_l}{2}, \frac{y_n + y_l}{2} \right)$ su punto medio, entonces, aproximamos la integral sobre un segmento de la siguiente forma:

$$\int_{v_n}^{v_l} F(v) \simeq \frac{F(v_n) + F(v_l) + 4f(v_m)}{6} \|v_n - v_l\|$$

donde $\|v_n - v_l\|$ representa la distancia euclídea entre v_n y v_l . Por tanto podemos aplicar esta fórmula al cálculo de la integral anterior teniendo en cuenta que

$$\int_{\Gamma_2} \phi_i \phi_j = \sum_i \int_{v_i}^{v_{i+1}} \phi_i(v) \phi_j(v)$$

donde el índice i recorre los nodos de Γ_2 , y los segmentos de vértices $[v_i, v_{i+1}]$ son las aristas de los triángulos de la triangulación de Ω que van recorriendo Γ_2 .

Por otro lado, a partir de la expresión de la función $\phi_i(x, y)$ estudiada en la sección anterior, que recordamos es :

$$\phi_i(x, y) = \frac{(y - y_j)(x_j - x_j) - (x - x_j)(y_j - y_j)}{(y_i - y_j)(x_j - x_j) - (x_i - x_j)(y_j - y_j)}$$

se obtiene fácilmente que $\nabla \phi_i(x, y)$ es un vector constante dado por

$$\nabla \phi_i(x, y) = \left(\frac{\frac{-(y_j - y_m)}{(y - y_m)(x_j - x_m) - (x - x_m)(y_j - y_m)}}{\frac{(x_j - x_m)}{(y - y_m)(x_j - x_m) - (x - x_m)(y_j - y_m)}} \right)$$

El flat de frontera de cada nodo de la triangulación nos indica si es interior ($fr = 0$), si el nodo está en Γ_1 ($fr = 1$), o si el nodo está en Γ_2 ($fr = 2$). Estamos ya pues en condiciones de dar un esquema general para el algoritmo que calcula la matriz $A = (a_{ij})$ y el vector $b = (b_i)$ del sistema de ecuaciones.

Algoritmo para el cálculo de A y b .

Se inicializa $a_{ij} = 0, b_i = 0 \forall i, j$

Para cada triángulo $K_n = (v_i, v_j, v_n)$

Para cada nodo $l = i, j, n$ del triángulo K_n

Si el nodo l está en Γ_1

$$a_{ll} = M$$

$$b_l = D(v_l)M$$

Si no

$$a_{ll+} = \int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_l + \int_{K_n} a(x, y) \phi_l \phi_l$$

$$b_{l+} = \int_{K_n} f \phi_l$$

Fin Si

Fin Para

Para cada par de nodos distintos $l, m = i, j, n$ del triángulo K_n

$$a_{lm+} = \int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_m + \int_{K_n} a(x, y) \phi_l \phi_m$$

Si los nodos l y m están en Γ_2

$$b_{l+} = N(v_m) \int_{v_l}^{v_m} k(x, y) \phi_m \phi_l + N(v_l) \int_{v_l}^{v_m} k(x, y) \phi_l \phi_l$$

$$b_{m+} = N(v_l) \int_{v_l}^{v_m} k(x, y) \phi_l \phi_m + N(v_m) \int_{v_l}^{v_m} k(x, y) \phi_m \phi_m$$

Fin Si

Fin Para

Fin Para

Si utilizamos el esquema alternativo dado en la sección anterior el algoritmo quedaría

Algoritmo para el cálculo de A y b .

Se inicializa $a_{ij} = 0, b_i = 0 \forall i, j$

Para cada triángulo $K_n = (v_i, v_j, v_n)$

Para cada nodo $l = i, j, n$ del triángulo K_n

Si el nodo l está en Γ_1

$$a_{ll} = 1$$

$$b_l = D(v_l)$$

Si no

$$a_{ll+} = \int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_l + \int_{K_n} a(x, y) \phi_l \phi_l$$

$$b_{l+} = \int_{K_n} f \phi_l$$

Fin Si

Fin Para

Para cada par de nodos distintos $l, m = i, j, n$ del triángulo K_n

Si $l \in \Gamma_1$ y $m \notin \Gamma_1$ entonces

$$b_{m-} = D(v_l) \left(\int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_m + \int_{K_n} a(x, y) \phi_l \phi_m \right)$$

Si $l \notin \Gamma_1$ y $m \in \Gamma_1$ entonces

$$b_{l-} = D(v_m) \left(\int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_m + \int_{K_n} a(x, y) \phi_l \phi_m \right)$$

Si $l \notin \Gamma_1$ y $m \notin \Gamma_1$ entonces

$$a_{lm+} = \int_{K_n} k(x, y) \nabla \phi_l \nabla \phi_m + \int_{K_n} a(x, y) \phi_l \phi_m$$

Si los nodos l y m están en Γ_2

$$b_{l+} = N(v_m) \int_{v_l}^{v_m} k(x, y) \phi_m \phi_l + N(v_l) \int_{v_l}^{v_m} k(x, y) \phi_l \phi_l$$

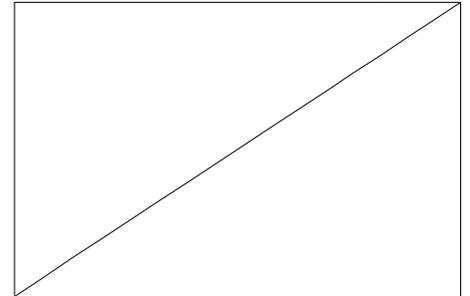
$$b_{m+} = N(v_l) \int_{v_l}^{v_m} k(x, y) \phi_l \phi_m + N(v_m) \int_{v_l}^{v_m} k(x, y) \phi_m \phi_m$$

Fin Si

Fin Para

Fin Para

Ejemplo 10 Vamos a desarrollar un ejemplo completo de cálculo de sistema. Consideremos el dominio Ω dado por la siguiente figura



compuesta de 2 triángulos $K_0 = (v_0, v_1, v_2)$, y $K_1 = (v_0, v_2, v_3)$ de vértices $v_0 = (0, 0)$, $v_1 = (1, 0)$, $v_2 = (1, 1)$ y $v_3 = (0, 1)$. Consideremos la ecuación diferencial

$$-\text{div}(\nabla u) + u = 1$$

con las condiciones de contorno $\frac{\partial u}{\partial n}(v_0) = -1$, $\frac{\partial u}{\partial n}(v_1) = -2$, $u(v_2) = 3$, $u(v_3) = 4$. Vamos a realizar una traza del algoritmo. En primer lugar, tomamos el triángulo K_0 y calculamos las funciones ϕ_i centradas sobre los nodos v_0, v_1, v_2

$$\begin{aligned}\phi_0(x, y) &= 1 - x \\ \phi_1(x, y) &= x - y \\ \phi_2(x, y) &= y\end{aligned}$$

Si siguiendo el algoritmo, en el primer bucle calculamos

$$\begin{aligned}a_{00} &= \int_0^1 \int_0^x 1 dy dx + \int_0^1 \int_0^x (1-x)^2 dy dx = \frac{7}{12} \\ b_0 &= \int_0^1 \int_0^x (1-x) dy dx = \frac{1}{6} \\ a_{11} &= \int_0^1 \int_0^x 2 dy dx + \int_0^1 \int_0^x (x-y)^2 dy dx = \frac{13}{12} \\ b_1 &= \int_0^1 \int_0^x (x-y) dy dx = \frac{1}{6} \\ a_{22} &= M \\ b_2 &= 3M\end{aligned}$$

En el siguiente bucle calculamos

$$\begin{aligned}a_{01} = a_{10} &= \int_0^1 \int_0^x -1 dy dx + \int_0^1 \int_0^x (1-x)(x-y) dy dx = -\frac{11}{24} \\ b_0 &= \frac{1}{6} - 2 \int_0^1 (1-x)x dx - 1 \int_0^1 x^2 dx = -\frac{1}{2} \\ b_1 &= \frac{1}{6} - 1 \int_0^1 (1-x)x dx - 2 \int_0^1 (1-x)^2 dx = -\frac{2}{3} \\ a_{02} = a_{20} &= \int_0^1 \int_0^x 0 dy dx + \int_0^1 \int_0^x (1-x)y dy dx = \frac{1}{24} \\ a_{12} = a_{21} &= \int_0^1 \int_0^x -1 dy dx + \int_0^1 \int_0^x y(x-y) dy dx = -\frac{11}{24}\end{aligned}$$

pasamos ahora al segundo triángulo $K_1 = (v_0, v_2, v_3)$. Las funciones ϕ_i sobre K_1 vienen dadas por

$$\begin{aligned}\phi_0(x, y) &= 1 - y \\ \phi_2(x, y) &= x \\ \phi_3(x, y) &= y - x\end{aligned}$$

Si siguiendo la traza del algoritmo obtenemos:

$$\begin{aligned}a_{00} &= \frac{7}{12} + \int_0^1 \int_x^1 1 dy dx + \int_0^1 \int_x^1 (1-y)^2 dy dx = \frac{7}{6} \\ b_0 &= -\frac{1}{2} + \int_0^1 \int_x^1 (1-y) dy dx = -\frac{1}{3} \\ a_{22} &= M \\ b_2 &= 3M \\ a_{33} &= M \\ b_3 &= 4M \\ a_{02} = a_{20} &= \frac{1}{24} + \int_0^1 \int_x^1 0 dy dx + \int_0^1 \int_x^1 (1-y)x dy dx = \frac{1}{12} \\ a_{03} = a_{30} &= \int_0^1 \int_x^1 -1 dy dx + \int_0^1 \int_x^1 (1-y)(y-x) dy dx = -\frac{11}{24} \\ a_{23} = a_{32} &= \int_0^1 \int_x^1 -1 dy dx + \int_0^1 \int_x^1 x(y-x) dy dx = -\frac{11}{24}\end{aligned}$$

Por lo tanto el sistema que hay que resolver es:

$$\begin{pmatrix} \frac{7}{6} & -\frac{11}{24} & \frac{1}{12} & -\frac{11}{24} \\ -\frac{11}{24} & \frac{13}{12} & -\frac{11}{24} & 0 \\ \frac{1}{12} & -\frac{11}{24} & M & -\frac{11}{24} \\ -\frac{11}{24} & 0 & -\frac{11}{24} & M \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} -\frac{1}{3} \\ -\frac{2}{3} \\ 3M \\ 4M \end{pmatrix}$$

cuya solución para $M = 10000$ viene dada por

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 1.5932 \\ 1.328 \\ 3.0002 \\ 4.0002 \end{pmatrix}$$

Si seguimos el algoritmo dado por la construcción del sistema alternativo llegamos al sistema:

$$\begin{pmatrix} \frac{7}{6} & -\frac{11}{24} & 0 & -0 \\ -\frac{11}{24} & \frac{13}{12} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} -\frac{1}{3} - 3\frac{1}{12} + 4\frac{11}{24} \\ -\frac{2}{3} + 3\frac{11}{24} \\ 3 \\ 4 \end{pmatrix}$$

cuya solución es

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} 1.5931 \\ 1.3278 \\ 3.0 \\ 4.0 \end{pmatrix}$$

que como puede apreciarse está muy próxima de la solución anterior.

PRACTICA 4. METODO DE ELEMENTOS FINITOS

En esta práctica se implementará la construcción de la matriz A y el vector b por el método de elementos finitos utilizando el algoritmo descrito en la sección anterior. Las funciones $k(x, y)$, $a(x, y)$, $f(x, y)$, $D(x, y)$ y $N(x, y)$ estarán definidas dentro del cuerpo del programa, simplemente utilizando por ejemplo la instrucción `#define`. El parámetro de entrada será la triangulación T . Para construir la matriz morsa no se podrá utilizar un almacenamiento intermedio en forma de matriz plena. Se realizarán los test de comprobación siguientes:

1. Comprobar que para el ejemplo desarrollado en la sección anterior se llega al mismo resultado de matriz A y vector b .
2. Tomar la triangulación $T1.tri$ del directorio de la asignatura. Tomar como funciones $k(x, y) = 1$, $a(x, y) = 0$, $f(x, y) = 1$, $D(x, y) = 0$ y $N(x, y) = 0$. Comprobar que los valores no-nulos de la matriz son $a_{ii} = 4$ y $a_{ij} = -1$ ó $a_{ij} = 0.5$, y los valores no-nulos del vector b son: $b_i = 0.000977$

- Tomar la triangulación $T1.tri$ del directorio de la asignatura. Tomar como funciones $k(x,y) = 1$, $a(x,y) = 1$, $f(x,y) = 1$, $D(x,y) = 0$ y $N(x,y) = 0$. Comprobar que los valores no-nulos de la matriz son $a_{ii} = 4.000488$, $a_{ij} = -0.9999191$ ó $a_{ij} = 0.0000811$, y los valores no-nulos del vector b son: $b_i = 0.000977$
- Tomar una triangulación del objeto de la figura 15, tomando la circunferencia interior con condiciones de Dirichlet, y la circunferencia exterior con condiciones de Neuman. Tomar como funciones $k(x,y) = 1$, $a(x,y) = 1$, $f(x,y) = (-4+x^2+y^2)$, $D(x,y) = x^2+y^2$ y $N(x,y) = -2\sqrt{x^2+y^2}$. Construir el sistema de ecuaciones asociado, resolverlo y calcular el error máximo que se produce entre la solución real dada por la función $u(x,y) = x^2 + y^2$ y la solución suministrada al resolver el sistema. Realizar esta operación con mallas con diferente número de puntos y comprobar que cuantos más puntos se utilizan, el error máximo es menor.

TRIANGULO DE REFERENCIA Y FUNCIONES DE FORMA

Una estrategia que se utiliza con frecuencia para realizar todos los cálculos sobre los triángulos de la malla, es hacer un cambio de variable, para que los cálculos se hagan siempre sobre un mismo triángulo \hat{K} denominado triángulo de referencia. dicho triángulo \hat{K} tiene por vértices $\hat{v}_0 = (0,0)$, $\hat{v}_1 = (1,0)$ y $\hat{v}_2 = (0,1)$. Si tenemos un triángulo cualquiera K de vértices $v_i = (x_i, y_i)$, $v_j = (x_j, y_j)$ y $v_k = (x_k, y_k)$, entonces, la transformación $F : \hat{K} \rightarrow K$ dada por

$$F(\hat{x}, \hat{y}) = \begin{pmatrix} x_i + \hat{x}(x_j - x_i) + \hat{y}(x_k - x_i) \\ y_i + \hat{x}(y_j - y_i) + \hat{y}(y_k - y_i) \end{pmatrix}$$

transforma el triángulo \hat{K} en el triángulo K y además lleva el vértice \hat{v}_0 en v_i , \hat{v}_1 en v_j y \hat{v}_2 en v_k . Vamos a definir sobre el triángulo de referencia \hat{K} , 3 funciones $\hat{P}_0(\hat{x}, \hat{y})$, $\hat{P}_1(\hat{x}, \hat{y})$ y $\hat{P}_2(\hat{x}, \hat{y})$ que llamaremos funciones de forma, de tal manera que sean funciones lineales (es decir polinomios de grado 1 en \hat{x} e \hat{y}) y que verifiquen $\hat{P}_m(\hat{v}_n) = 0$ si $m \neq n$, y $\hat{P}_m(\hat{v}_m) = 1$ para todo $0 \leq m, n \leq 2$. Dichas funciones son

$$\begin{aligned} \hat{P}_0(\hat{x}, \hat{y}) &= 1 - \hat{x} - \hat{y} \\ \hat{P}_1(\hat{x}, \hat{y}) &= \hat{x} \\ \hat{P}_2(\hat{x}, \hat{y}) &= \hat{y} \end{aligned}$$

Estas funciones, via la transformación F , determinan cómo son las funciones base ϕ_i sobre los triángulos, de tal manera que sobre un triángulo dado K de vértices v_i , v_j , y v_k se tiene:

$$\begin{aligned} \phi_i(F(\hat{x}, \hat{y})) &= \hat{P}_0(\hat{x}, \hat{y}) \\ \phi_j(F(\hat{x}, \hat{y})) &= \hat{P}_1(\hat{x}, \hat{y}) \\ \phi_k(F(\hat{x}, \hat{y})) &= \hat{P}_2(\hat{x}, \hat{y}) \end{aligned}$$

es decir, que la forma que tienen las funciones ϕ_i vienen determinadas por la forma de las \hat{P}_m y la transformación F .

Otro elemento importante que debemos utilizar para realizar todos los cálculos en el triángulo de referencia es el jacobiano $Jac(F)$ de la función F . Como $F(\hat{x}, \hat{y})$ es un vector de 2 componentes, si denotamos por $F^1(\hat{x}, \hat{y})$ a la primera componente, y por $F^2(\hat{x}, \hat{y})$ a la segunda componente. El jacobiano, viene determinado por el determinante de la matriz:

$$Jac(F)(\hat{x}, \hat{y}) = \begin{vmatrix} \frac{\partial F^1}{\partial \hat{x}}(\hat{x}, \hat{y}) & \frac{\partial F^1}{\partial \hat{y}}(\hat{x}, \hat{y}) \\ \frac{\partial F^2}{\partial \hat{x}}(\hat{x}, \hat{y}) & \frac{\partial F^2}{\partial \hat{y}}(\hat{x}, \hat{y}) \end{vmatrix}$$

en el caso de la transformación $F(\hat{x}, \hat{y})$ el jacobiano es constante, y viene dado por:

$$Jac(F)(\hat{x}, \hat{y}) = (x_j - x_i)(y_k - y_i) - (y_j - y_i)(x_k - x_i)$$

nótese que en este caso el jacobiano es nulo si sólo si los vértices v_i , v_j y v_k del triángulo están alineados.

Para llevar todos los cálculos de integración sobre triángulos al triángulo de referencia aplicamos el teorema de cambio de variables en una integral múltiple, es decir, dada una función $G(x,y)$ definida sobre K , se tiene que:

$$\int_K G(x,y) dx dy = \int_{\hat{K}} G(F(\hat{x}, \hat{y})) |Jac(F)(\hat{x}, \hat{y})| d\hat{x} d\hat{y}$$

utilizando este resultado y las funciones de forma, podemos decir que dado un triángulo K cualquiera, la aportación de ese triángulo al vector b del sistema de ecuaciones al que se llega con el método de elementos finitos es

$$\begin{aligned} b_{i+} &= \int_K f(x,y) \phi_i(x,y) = \int_{\hat{K}} f(F(\hat{x}, \hat{y})) \hat{P}_0(\hat{x}, \hat{y}) |Jac(F)| \\ b_{j+} &= \int_K f(x,y) \phi_j(x,y) = \int_{\hat{K}} f(F(\hat{x}, \hat{y})) \hat{P}_1(\hat{x}, \hat{y}) |Jac(F)| \\ b_{k+} &= \int_K f(x,y) \phi_k(x,y) = \int_{\hat{K}} f(F(\hat{x}, \hat{y})) \hat{P}_2(\hat{x}, \hat{y}) |Jac(F)| \end{aligned}$$

por tanto, hemos conseguido centralizar los cálculos en un sólo triángulo \hat{K} . Nótese además que ya no se utilizan las funciones ϕ_i , sino las funciones de forma \hat{P}_m .

Con respecto a la aportación del triángulo K a los coeficientes de la matriz A que resulta del método de elementos finitos se procede del mismo modo, con respecto a la segunda parte de la aportación se aplica que

$$\int_K a(x,y) \phi_i \phi_j = \int_{\hat{K}} a(F(\hat{x}, \hat{y})) \hat{P}_0(\hat{x}, \hat{y}) \hat{P}_1(\hat{x}, \hat{y}) |Jac(F)| dx dy$$

y así sucesivamente para las otras combinaciones de los nodos v_i, v_j y v_k .

Con respecto a la primera parte de la aportación de K a la matriz A , que involucra a las derivadas de la función ϕ_i el cálculo es un poco más complicado pues debemos escribir dichas derivadas en función de las derivadas de las funciones de forma \hat{P}_m . Para ello necesitamos utilizar la inversa de la función $F(\hat{x}, \hat{y})$, es decir, la transformación

que lleva K en \widehat{K} . Dicha transformación que denotaremos por $G(x, y)$ viene dada por:

$$G(x, y) = \left(\begin{array}{c} \frac{(x-x_i)(y_k-y_i)-(y-y_i)(x_k-x_i)}{(x_j-x_i)(y_k-y_i)-(x_k-x_i)(y_j-y_i)} \\ \frac{(y-y_i)(x_j-x_i)-(x-x_i)(y_j-y_i)}{(x_j-x_i)(y_k-y_i)-(x_k-x_i)(y_j-y_i)} \end{array} \right)$$

utilizando esta transformación podemos escribir

$$\phi_i(x, y) = \widehat{P}_0(G(x, y))$$

y lo mismo para los otros índices del triángulo. Ahora ya podemos expresar las derivadas de ϕ_i respecto a las derivadas de las funciones de forma, y utilizando la regla de la cadena encontramos:

$$\frac{\partial \phi_i}{\partial x}(x, y) = \frac{\partial \widehat{P}_0}{\partial x}(G(x, y)) \frac{\partial G^1}{\partial x}(x, y) + \frac{\partial \widehat{P}_0}{\partial y}(G(x, y)) \frac{\partial G^2}{\partial x}(x, y)$$

$$\frac{\partial \phi_i}{\partial y}(x, y) = \frac{\partial \widehat{P}_0}{\partial x}(G(x, y)) \frac{\partial G^1}{\partial y}(x, y) + \frac{\partial \widehat{P}_0}{\partial y}(G(x, y)) \frac{\partial G^2}{\partial y}(x, y)$$

y lo mismo para los otros nodos del triángulo K .

Finalmente utilizando estas expresiones podemos calcular las integrales:

$$\int_K k(x, y) \nabla \phi_i \nabla \phi_j = \int_{\widehat{K}} k(F(x, y)) \nabla \phi_i \nabla \phi_j |Jac(F)|$$

donde $\nabla \phi_i \nabla \phi_j$ viene expresado utilizando las fórmulas anteriores, de tal forma que hemos centralizado todos los cálculos en el triángulo de referencia \widehat{K} .

ELEMENTOS FINITOS CURVOS

Una limitación importante al utilizar sólo triángulos para describir el dominio sobre el que trabajamos, es que cuando dicho dominio tiene fronteras con zonas curvas, la aproximación por triángulos nunca es perfecta debido a que los lados de los triángulos son rectos. Una técnica para evitar esto es utilizar como elementos de base, triángulos con lados que puedan estar deformados como el que muestra la figura siguiente

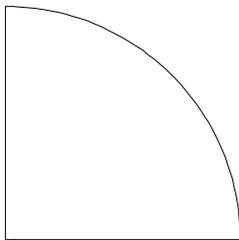


Figura 32:

Estos triángulos de lados deformados se definen utilizando 6 puntos, es decir v_i, v_j y v_k que determinan la posición de los vértices del triángulo, y v_l, v_m y v_n que

determinan la posición de los puntos medios de los lados deformados del triángulo. Por ejemplo, el triángulo de la figura anterior viene determinado por $v_i = (0, 0)$, $v_j = (1, 0)$ y $v_k = (0, 1)$ que determinan los vértices del triángulo, y $v_l = (0.5, 0)$, $v_m = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ y $v_n = (0, 0.5)$. Nótese que el punto que determina la deformación es v_m . Intuitivamente es como si cogiésemos los puntos medios de los triángulos y los estirásemos para ajustarnos mejor a la frontera.

Para trabajar computacionalmente con estos triángulos deformados, lo que se hace es transformarlos en el triángulo de referencia. Para ello, hay que buscar una transformación que lleve el triángulo de referencia \widehat{K} en el triángulo deformado que denotaremos por K_D . Dicha transformación, que denotaremos por $F(\widehat{x}, \widehat{y})$ debe cumplir

$$\begin{aligned} F(\widehat{v}_0) &= F(0, 0) = v_i = (x_i, y_i) \\ F(\widehat{v}_1) &= F(1, 0) = v_j = (x_j, y_j) \\ F(\widehat{v}_2) &= F(0, 1) = v_k = (x_k, y_k) \\ F(\widehat{v}_3) &= F(0.5, 0) = v_l = (x_l, y_l) \\ F(\widehat{v}_4) &= F(0.5, 0.5) = v_m = (x_m, y_m) \\ F(\widehat{v}_5) &= F(0, 0.5) = v_n = (x_n, y_n) \end{aligned}$$

La función $F(\widehat{x}, \widehat{y})$ más sencilla que permite ajustar estos valores es un polinomio de grado 2 en \widehat{x} e \widehat{y} . Para ello nos apoyaremos en unas funciones de forma que llamaremos $\widehat{P}_s(\widehat{x}, \widehat{y})$ con $0 \leq s \leq 5$ que cumplen que $\widehat{P}_s(\widehat{v}_r) = 0$ si $s \neq r$ y $\widehat{P}_s(\widehat{v}_s) = 1$. Imponiendo que estas funciones sean polinomios de grado 2 se obtiene fácilmente

$$\begin{aligned} \widehat{P}_0(\widehat{x}, \widehat{y}) &= (1 - \widehat{x} - \widehat{y})(1 - 2\widehat{x} - 2\widehat{y}) \\ \widehat{P}_1(\widehat{x}, \widehat{y}) &= \widehat{x}(2\widehat{x} - 1) \\ \widehat{P}_2(\widehat{x}, \widehat{y}) &= \widehat{y}(2\widehat{y} - 1) \\ \widehat{P}_3(\widehat{x}, \widehat{y}) &= 4(1 - \widehat{x} - \widehat{y})\widehat{x} \\ \widehat{P}_4(\widehat{x}, \widehat{y}) &= 4\widehat{x}\widehat{y} \\ \widehat{P}_5(\widehat{x}, \widehat{y}) &= 4(1 - \widehat{x} - \widehat{y})\widehat{y} \end{aligned}$$

de tal forma que podemos expresar la transformación $F(\widehat{x}, \widehat{y})$ como

$$F(\widehat{x}, \widehat{y}) = \sum_{t=0}^5 v_{i(t)} \widehat{P}_t(\widehat{x}, \widehat{y})$$

donde $v_{i(t)}$ son los puntos del triángulo deformado $v_i, v_j, v_k, v_l, v_m, v_n$.

Ahora bien, no es posible deformar los triángulos de cualquier manera. Los triángulos deformados admisibles son aquellos para los que la transformación $F(\widehat{x}, \widehat{y})$ es biyectiva entre \widehat{K} y el triángulo deformado K_D . Para que ello suceda, la condición es que el jacobiano de $F(\widehat{x}, \widehat{y})$ mantenga el mismo signo en todos los puntos de \widehat{K} . Veamos esto con un poco más de detalle en un ejemplo: Si consideramos que el triángulo deformado K_D es idéntico al triángulo \widehat{K} salvo que el punto $(0.5, 0.5)$ se estira (o traslada) a un punto genérico (x_0, y_0) vamos a ver cuáles

son los posibles valores de (x_0, y_0) tal que la transformación $F(\hat{x}, \hat{y})$ sea biyectiva. En este caso podemos calcular $F(\hat{x}, \hat{y})$ de forma directa obteniendo:

$$F(\hat{x}, \hat{y}) = \begin{pmatrix} \hat{x} + 4\hat{y}(x_0 - \frac{1}{2}) \\ \hat{y} + 4\hat{x}(y_0 - \frac{1}{2}) \end{pmatrix}$$

el jacobiano de esta transformación es

$$\begin{aligned} Jac(F)(\hat{x}, \hat{y}) &= \begin{vmatrix} 1 + 4\hat{y}(x_0 - \frac{1}{2}) & 4\hat{x}(x_0 - \frac{1}{2}) \\ 4\hat{y}(y_0 - \frac{1}{2}) & 1 + 4\hat{x}(y_0 - \frac{1}{2}) \end{vmatrix} \\ &= 1 + 4\hat{y}(x_0 - \frac{1}{2}) + 4\hat{x}(y_0 - \frac{1}{2}) \end{aligned}$$

como en este caso $Jac(F)(\hat{x}, \hat{y})$ es lineal, para que conserve el signo en \hat{K} es suficiente que tenga el mismo signo en los tres vértices de \hat{K} . Ahora bien

$$\begin{aligned} Jac(F)(0, 0) &= 1 \\ Jac(F)(1, 0) &= 1 + 4(y_0 - \frac{1}{2}) \\ Jac(F)(0, 1) &= 1 + 4(x_0 - \frac{1}{2}) \end{aligned}$$

por tanto las condiciones para que estos 3 valores no cambien de signo son:

$$\begin{aligned} y_0 &> \frac{1}{4} \\ x_0 &> \frac{1}{4} \end{aligned}$$

como vemos existe bastante margen para estirar el lado del triángulo.

A nivel computacional la diferencia fundamental es que ahora los triángulos deformados vienen determinados por 6 nodos y no 3. El resto sigue el análisis previamente realizado, teniendo en cuenta que todos los cálculos hay que hacerlos pasando por el triángulo de referencia.

ELEMENTOS FINITOS EN DIMENSIONES DISTINTAS DE 2.

El método de elementos finitos que hemos desarrollado puede extenderse fácilmente a dimensiones distinta de 2. En dimensión 1, el método es muy sencillo de aplicar, en este caso el problema modelo se escribe como

$$\frac{\partial}{\partial x} \left(k(x) \frac{\partial u}{\partial x} \right) (x) + a(x)u(x) = f(x) \quad \text{en } [a, b]$$

las condiciones de frontera son en este caso que la función $u(x)$ o su derivada valgan un valor determinado en a y b . En este caso los elementos finitos no son triángulos sino segmentos con dos vértices, para aplicar el método se divide el segmento $[a, b]$ en un conjunto de subsegmentos dados por un conjunto de nodos $a = x_0 < x_1 < \dots < x_N = b$ y se definen las funciones base como unas funciones $\phi_i(x)$, lineales sobre cada segmento $[x_k, x_{k+1}]$ y tal que $\phi_i(x_j) = 0$

si $i \neq j$ y $\phi_i(x_i) = 1$. A partir de aquí, el desarrollo es análogo al caso de 2 dimensiones, pero evidentemente mucho más simple.

En el caso de problemas en dimensión 3, la diferencia fundamental es que los elementos finitos en este caso son tetraedros. El desarrollo es análogo a dimensión 2. En este caso, al realizar la formulación variacional del problema aparecen integrales de superficie sobre la frontera en lugar de integrales de línea.

APLICACION DEL METODO ELEMENTOS FINITOS A PROBLEMAS DE EVOLUCION

Hemos desarrollado el método de elementos finitos para un problema modelo. Por supuesto la técnica desarrollada es de aplicación en otros muchos contextos y ecuaciones. Dos casos especialmente sencillos de extensión de lo que hemos visto a otros modelos físicos son la ecuación del calor y la ecuación de ondas, donde la solución $u(x, y)$ no es estática sino que depende del tiempo, es decir tenemos $u(t, x, y)$ la dependencia del tiempo indica que es un fenómeno que varía con el tiempo. Por ejemplo en el caso de la ecuación del calor dada por la ecuación diferencial

$$\frac{\partial u}{\partial t} = \text{div} (k(x, y) \nabla u)$$

$u(t, x, y)$ indica la evolución de la temperatura de un objeto plano a lo largo del tiempo partiendo de una distribución inicial $u_0(x, y)$.

En el caso de la ecuación de ondas dada por la ecuación diferencial

$$\frac{\partial^2 u}{\partial t^2} = \text{div} (k(x, y) \nabla u)$$

$u(t, x, y)$ indica la evolución de la amplitud de una onda sobre un dominio plano (por ejemplo al tirar una piedra en un estanque) partiendo de una amplitud inicial de la onda $u_0(x, y)$, y de una velocidad inicial $u'_0(x, y) = \frac{\partial u}{\partial t}(0, x, y)$

En ambas ecuaciones si discretizamos la variable temporal t de tal manera que buscamos una secuencia de funciones $u_n(x, y)$ de tal forma que $u_n(x, y)$ es una aproximación de $u(n\Delta t, x, y)$, y discretizamos por diferencias finitas las derivadas temporales obtenemos los siguientes esquemas: Para la ecuación del calor

$$\frac{u_n(x, y) - u_{n-1}(x, y)}{\Delta t} = \text{div} (k(x, y) \nabla u_n) (x, y)$$

que determina la forma de pasar de la etapa $u_{n-1}(x, y)$ a la etapa $u_n(x, y)$, nótese que en este caso $u_{n-1}(x, y)$ es conocido. Despejando convenientemente en esta expresión obtenemos:

$$-\text{div} (k(x, y) \nabla u_n) (x, y) + \frac{1}{\Delta t} u_n(x, y) = \frac{u_{n-1}(x, y)}{\Delta t}$$

ahora bien esta ecuación es un caso particular del problema modelo donde $a(x, y) = \frac{1}{\Delta t}$ y $f(x, y) = \frac{u_{n-1}(x, y)}{\Delta t}$

En el caso de la ecuación de ondas el esquema al que se llega utilizando diferencias finitas para las derivadas temporales es:

$$\frac{u_n(x,y) - 2u_{n-1}(x,y) + u_{n-2}(x,y)}{\Delta t} = \text{div}(k(x,y)\nabla u_n)(x,y)$$

donde ahora los datos son u_{n-1} y u_{n-2} y se pretende calcular u_n . Despejando como antes obtenemos:

$$-\text{div}(k\nabla u_n)(x,y) + \frac{1}{\Delta t}u_n(x,y) = \frac{2u_{n-1}(x,y) - u_{n-2}(x,y)}{\Delta t}$$

que es un caso particular del problema modelo estudiado tomando $a(x,y) = \frac{1}{\Delta t}$ y $f(x,y) = \frac{2u_{n-1}(x,y) - u_{n-2}(x,y)}{\Delta t}$

SISTEMAS DE ECUACIONES DIFERENCIALES LINEALES

INTRODUCCION

Un sistema de ecuaciones diferenciales es aquel en que las ecuaciones vienen expresadas en función de las variables y sus derivadas.

$$\begin{aligned} f_1(t, x_1, \dots, x_n, x'_1, \dots, x'_n, \dots, x_1^{(n)}, \dots, x_n^{(n)}) &= 0 \\ f_2(t, x_1, \dots, x_n, x'_1, \dots, x'_n, \dots, x_1^{(n)}, \dots, x_n^{(n)}) &= 0 \\ \vdots & \\ f_n(t, x_1, \dots, x_n, x'_1, \dots, x'_n, \dots, x_1^{(n)}, \dots, x_n^{(n)}) &= 0 \end{aligned}$$

Ejemplo 11 Consideremos el sistema predador-presa. Denotamos por $x(t)$ e $y(t)$ las poblaciones, en el instante t , de dos especies, una de las cuales (y) devora a la otra (x). Sin la presa, los predadores disminuirán en número y sin los predadores, los que sirven de presa aumentarán. Alrededor de 1925, Lotka y Volterra propusieron un modelo matemático que muestra la forma en que puede mantenerse en equilibrio ecológico, cuando ambos están presentes.

$$\begin{cases} x(t) = \text{Población de animales presas} \\ y(t) = \text{Población de predadores} \end{cases}$$

El sistema de ecuaciones diferenciales es de la siguiente manera:

$$\begin{cases} x'(t) = x(a - y) \\ y'(t) = y(x - b) \end{cases}$$

donde a representa el nivel de saturación de los predadores y b el nivel de presas crítico para que los predadores empiecen a crecer. Cuando el número de depredadores es mayor que a , entonces el número de presas decrece. De igual manera, cuando el número de presas es menor que b , entonces el número de depredadores decrece.

Dado un estado inicial $x(t_0)$ e $y(t_0)$ se trata de predecir el comportamiento futuro del sistema.

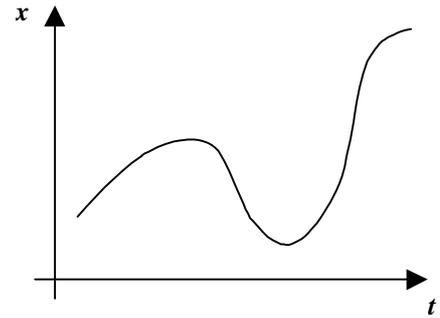


Figura 33: Variación de x en el tiempo

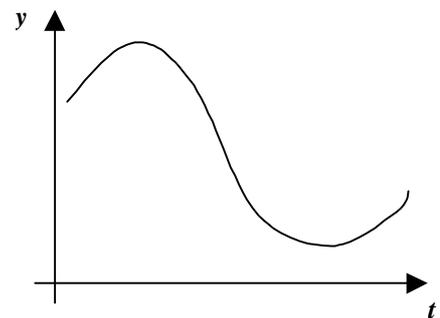
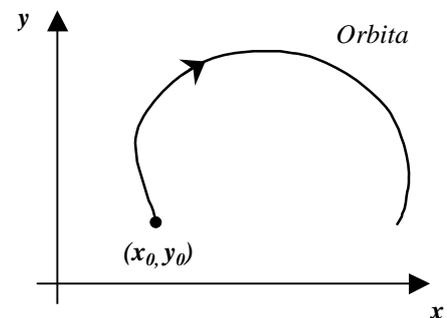


Figura 34: Variación de y en el tiempo

Representación gráfica: En estos sistemas de ecuaciones, tenemos una representación de x e y en forma paramétrica. Su representación gráfica es de la siguiente manera:

La representación paramétrica de x e y se puede interpretar como un punto que traza una curva en el plano xy . El plano xy se conoce como **plano de fases**. Una curva descrita de forma paramétrica, se conoce como trayectoria, órbita o camino.



Órbita $x - y$.

Órbita: Trayectoria geométrica que siguen las variables relacionadas entre sí. Es periódica si existen repeticiones de la misma trayectoria a intervalos regulares de tiempo. Cuando se pasa de la representación en el tiempo al plano de fases se pierde información de los instantes de tiempo.

De forma general, un sistema de ecuaciones diferenciales de primer orden se puede expresar de la siguiente manera:

$$\begin{aligned} x_1' &= f_1(t, x_1, x_2, \dots, x_n) \\ x_2' &= f_2(t, x_1, x_2, \dots, x_n) \\ &\vdots \\ x_n' &= f_n(t, x_1, x_2, \dots, x_n) \end{aligned} \quad (4)$$

$$\text{con } \begin{pmatrix} x_1(t_0) \\ x_2(t_0) \\ \vdots \\ x_n(t_0) \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \text{ condiciones iniciales}$$

SISTEMAS LINEALES

Si cada una de las funciones f_1, f_2, \dots, f_n en el sistema de ecuaciones (4) es una función lineal de las variables x_1, x_2, \dots, x_n , entonces se dice que el sistema de ecuaciones es *lineal*. El sistema de n ecuaciones diferenciales lineales de primer orden tiene la forma:

$$\begin{aligned} x_1' &= p_{11}(t)x_1 + \dots + p_{1n}(t)x_n + g_1(t) \\ x_2' &= p_{21}(t)x_1 + \dots + p_{2n}(t)x_n + g_2(t) \\ &\vdots \\ x_n' &= p_{n1}(t)x_1 + \dots + p_{nn}(t)x_n + g_n(t) \end{aligned} \quad (5)$$

Si las funciones $g_i(t)$ son nulas, entonces se dice que el sistema (5) es *homogéneo*.

Si expresamos el anterior sistema de ecuaciones en forma matricial, tenemos:

$$\bar{x}' = \mathbf{P}(t)\bar{x} + \mathbf{g}(t)$$

En el caso en que tengamos un sistema homogéneo y las funciones $p_{ij}(t)$ sean constantes ($p_{ij}(t) = a_{ij}$), entonces el sistema (5) se puede expresar de la siguiente forma:

$$\begin{pmatrix} x_1' \\ \vdots \\ x_n' \end{pmatrix} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad (6)$$

$$\bar{x}'(t) = A\bar{x}(t)$$

Proposición 1 Sea una ecuación diferencial de la forma $x' = ax$ con $x(t_0) = x_0$, entonces su solución viene dada por

$$x(t) = x_0 e^{a(t-t_0)}$$

Demostración: A partir de la ecuación despejamos del segundo miembro x ,

$$\frac{x'}{x} = a,$$

esto es una ecuación diferencial de variables separables,

$$\frac{dx}{x} = a dt$$

integramos,

$$\int_{t_0}^t \frac{dx}{x} = \int_{t_0}^t a dt$$

y resolviendo

$$\ln(x(t)) - \ln(x(t_0)) = a(t - t_0)$$

$$\ln(x(t)) = \ln(x(t_0)) + a(t - t_0)$$

aplicando en ambos términos la exponencial,

$$e^{\ln(x(t))} = e^{\ln(x(t_0)) + a(t-t_0)}$$

obtenemos la solución,

$$x(t) = x(t_0)e^{a(t-t_0)}$$

En el caso general ($n > 1$), la solución del sistema de ecuaciones (6) tiene la siguiente expresión:

$$\boxed{\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0} \quad (7)$$

donde el valor de la exponencial se calcula como se explica en la siguiente definición.

Definición 10 Vamos a definir e^B , con B matriz cualquiera (cuadrada), a través del desarrollo de Taylor como:

$$\boxed{e^B = Id + B + \frac{B^2}{2} + \dots + \frac{B^n}{n!} + \dots}$$

a partir del desarrollo de Taylor de e^t .

$$e^t = 1 + t + \frac{t^2}{2} + \dots + \frac{t^n}{n!}$$

Para ver que este desarrollo es consistente con la solución del sistema (7), vamos a comprobar que efectivamente $\bar{x}'(t) = A\bar{x}$ para este cálculo de la exponencial. Partimos de la expresión:

$$\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$$

aplicamos el desarrollo de la exponencial dada en la definición anterior,

$$\bar{x}(t) = Id\bar{x}_0 + A(t-t_0)\bar{x}_0 + A^2 \frac{(t-t_0)^2}{2} \bar{x}_0 + \dots + A^n \frac{(t-t_0)^n}{n!} \bar{x}_0$$

derivamos esta expresión,

$$\bar{x}'(t) = A\bar{x}_0 + A^2(t-t_0)\bar{x}_0 + \dots + A^n \frac{(t-t_0)^{n-1}}{(n-1)!} \bar{x}_0$$

multiplicamos $\bar{x}(t)$ por la matriz A y obtenemos,

$$A\bar{x}(t) = A\bar{x}_0 + A^2(t-t_0)\bar{x}_0 + \dots + A^{n+1} \frac{(t-t_0)^n}{n!} \bar{x}_0 = \bar{x}'(t),$$

luego $\bar{x}'(t) = A\bar{x}$ tiene por solución $\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$

Ejemplo 12 Calcular la solución del sistema homogéneo de coeficientes constantes dado por la siguiente matriz

$$A = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

a partir de esta matriz se obtiene el sistema lineal siguiente:

$$\begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Por la proposición y la definición anterior sabemos que la solución tiene la forma $\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$. Calculamos la exponencial según la definición y obtenemos

$$e^{A(t-t_0)} = Id$$

Todas las potencias $A^n (n \geq 1)$ son nulas, luego $\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0 = \bar{x}_0$.

Este resultado significa que la solución del sistema es constante, es decir, que no varía con el paso del tiempo. Si observamos el sistema, nos damos cuenta que las derivadas $(x_1', x_2') = (0, 0)$ son nulas, lo que quiere decir que el sistema no progresa.

Ejemplo 13 Calcular la solución del sistema dado por la siguiente matriz

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

El sistema que se obtiene de esta matriz es de la forma:

$$\begin{pmatrix} x_1' \\ x_2' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

La solución viene dada por $\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$, donde el valor de la exponencial es

$$\begin{aligned} e^{A(t-t_0)} &= \\ &= \left(Id + A(t-t_0) + A^2 \frac{(t-t_0)^2}{2} \dots + A^n \frac{(t-t_0)^n}{n!} \right) \\ &= Id \left(1 + (t-t_0) + \frac{(t-t_0)^2}{2} + \dots + \frac{(t-t_0)^n}{n!} \right) = \\ &= Id \cdot e^{(t-t_0)} = \begin{pmatrix} e^{(t-t_0)} & 0 \\ 0 & e^{(t-t_0)} \end{pmatrix}, \end{aligned}$$

luego la solución es:

$$\begin{aligned} \bar{x}(t) &= e^{A(t-t_0)}\bar{x}_0 = \begin{pmatrix} e^{(t-t_0)} & 0 \\ 0 & e^{(t-t_0)} \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \\ \bar{x}(t) &= \begin{pmatrix} e^{(t-t_0)}x_0 \\ e^{(t-t_0)}y_0 \end{pmatrix} \end{aligned}$$

Las variables son independientes entre sí.

Ejemplo 14 Calcular la solución del sistema dado por la matriz

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Como ya sabemos, la solución del sistema es de la forma $\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$, con lo que tenemos que calcular $e^{A(t-t_0)}$ y luego multiplicarlo por las condiciones iniciales:

$$\begin{aligned} e^{A(t-t_0)} &= Id + A(t-t_0) + A^2 \frac{(t-t_0)^2}{2} + \\ &+ \dots + A^n \frac{(t-t_0)^n}{n!} \end{aligned}$$

Calculamos las distintas potencias de la matriz:

$$\begin{aligned} A^2 &= \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}^2 = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} = -Id \\ A^3 &= A^2 \cdot \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} = -A \\ A^4 &= A^3 \cdot \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = Id \\ A^5 &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = Id \cdot A = A \end{aligned}$$

Como se puede ver, se repite el mismo ciclo $\{A, -Id, -A, Id\}$, con lo que, teniendo en cuenta la posición que toman los distintos factores en las matrices, obtenemos la siguiente expresión:

$$\begin{aligned} e^{A(t-t_0)} &= \begin{pmatrix} 1 - \frac{(t-t_0)^2}{2} + \frac{(t-t_0)^4}{4!} - \dots \\ -(t-t_0) + \frac{(t-t_0)^3}{3!} - \dots \\ (t-t_0) - \frac{(t-t_0)^3}{3!} + \dots \\ 1 - \frac{(t-t_0)^2}{2} + \frac{(t-t_0)^4}{4!} - \dots \end{pmatrix} \end{aligned}$$

Simplificando los términos de esta matriz por sus relaciones trigonométricas tenemos que

$$e^{A(t-t_0)} = \begin{pmatrix} \cos(t-t_0) & \sin(t-t_0) \\ -\sin(t-t_0) & \cos(t-t_0) \end{pmatrix},$$

con lo que la solución final del sistema es

$$\bar{x}(t) = e^{A(t-t_0)}\bar{x}_0$$

$$\bar{x}(t) = \begin{pmatrix} \cos(t-t_0)x_0 + \sin(t-t_0)y_0 \\ -\sin(t-t_0)x_0 + \cos(t-t_0)y_0 \end{pmatrix}$$

Por último, vamos a dibujar las órbitas que se obtienen para este sistema. Para esto, tenemos que intentar pasar de una representación paramétrica dada por la matriz anterior, a una representación en el plano de fases.

Si elevamos ambos términos al cuadrado y los sumamos, obtenemos la siguiente relación:

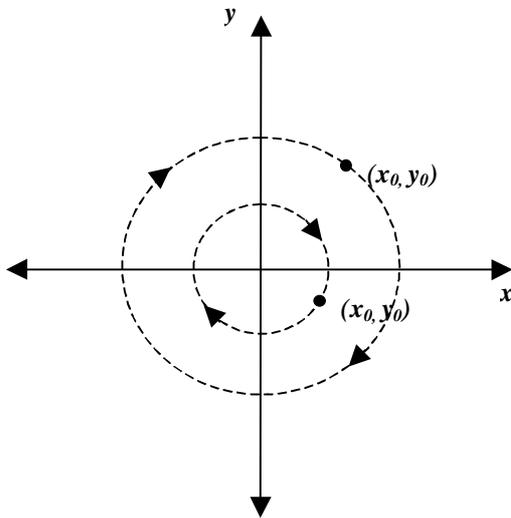
$$x^2(t) + y^2(t) = \cos^2(t-t_0)x_0^2 +$$

$$\begin{aligned}
&+2 \cos(t-t_0) \sin(t-t_0) x_0 y_0 + \sin^2(t-t_0) y_0^2 + \\
&+ \sin^2(t-t_0) x_0^2 - 2 \cos(t-t_0) \sin(t-t_0) x_0 y_0 + \\
&+ \cos^2(t-t_0) y_0^2 = x_0^2 + y_0^2
\end{aligned}$$

Con lo que tenemos una ecuación de la forma:

$$x^2 + y^2 = x_0^2 + y_0^2$$

Esta es la fórmula de una circunferencia de radio $\sqrt{x_0^2 + y_0^2}$ y de centro el origen, con lo que la órbita tendrá forma de circunferencia con un radio que depende de las condiciones iniciales. Para obtener el sentido del desplazamiento en la circunferencia, basta con coger un punto en uno de los cuadrantes y sustituirlo en el sistema $\bar{x}' = A\bar{x}$. El signo que toman x' e y' nos indica el sentido.



Órbitas para $A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$

Ejemplo 15 Calcular la solución del sistema dado por la siguiente matriz

$$A = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

Las potencias de la matriz de orden 2 y superior son nulas.

$$A^2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

El resultado de la exponencial es,

$$e^{A(t-t_0)} = Id + A(t-t_0)$$

$$e^{A(t-t_0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ t-t_0 & 0 \end{pmatrix}$$

$$e^{A(t-t_0)} = \begin{pmatrix} 1 & 0 \\ t-t_0 & 1 \end{pmatrix}$$

La solución del sistema,

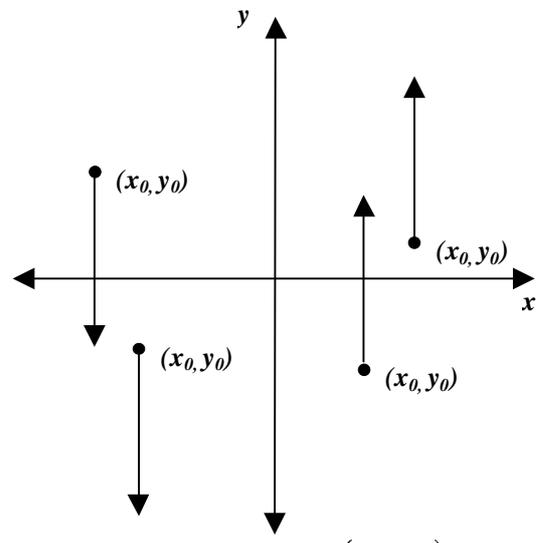
$$\bar{x}(t) = \begin{pmatrix} 1 & 0 \\ t-t_0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$$

$$\bar{x}(t) = \begin{pmatrix} x_0 \\ (t-t_0)x_0 + y_0 \end{pmatrix}$$

Si observamos este resultado, podemos ver que x siempre vale x_0 mientras que y varía dependiendo del signo de x_0 . Si este es positivo y tenderá a ∞ , y si es negativo, a $-\infty$.

Dibujamos la órbita teniendo en cuenta que:

$$\begin{aligned}
x_0 > 0 &\implies y \text{ tiende a } \infty \\
x_0 < 0 &\implies y \text{ tiende a } -\infty
\end{aligned}$$



Órbitas para $A = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$

Ejemplo 16 Calcular la solución del sistema dado por la matriz

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Las potencias de la matriz:

$$A^2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = Id$$

$$A^3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = Id \cdot A = A$$

Se repite el mismo ciclo $\{A, Id\}$. Calculamos la exponencial:

$$\begin{aligned}
e^{A(t-t_0)} &= Id + A(t-t_0) + A^2 \frac{(t-t_0)^2}{2} + A^3 \frac{(t-t_0)^3}{3!} + \\
&+ A^4 \frac{(t-t_0)^4}{4!} \dots + A^n \frac{(t-t_0)^n}{n!}
\end{aligned}$$

$$e^{A(t-t_0)} = \begin{pmatrix} 1 + \frac{(t-t_0)^2}{2} + \frac{(t-t_0)^4}{4!} + \dots \\ (t-t_0) + \frac{(t-t_0)^3}{3!} + \dots \end{pmatrix}$$

Estos desarrollos en serie son equivalentes a las siguientes expresiones:

$$e^{A(t-t_0)} = \begin{pmatrix} \frac{e^{(t-t_0)} + e^{-(t-t_0)}}{2} & \frac{e^{(t-t_0)} - e^{-(t-t_0)}}{2} \\ \frac{e^{(t-t_0)} - e^{-(t-t_0)}}{2} & \frac{e^{(t-t_0)} + e^{-(t-t_0)}}{2} \end{pmatrix}$$

La solución del sistema:

$$\bar{x}(t) = e^{A(t-t_0)} \bar{x}_0$$

$$\bar{x}(t) = \begin{pmatrix} \frac{1}{2} \left[(e^{t-t_0} + e^{-(t-t_0)}) x_0 + (e^{t-t_0} - e^{-(t-t_0)}) y_0 \right] \\ \frac{1}{2} \left[(e^{t-t_0} - e^{-(t-t_0)}) x_0 + (e^{t-t_0} + e^{-(t-t_0)}) y_0 \right] \end{pmatrix}$$

Vamos a intentar dibujar la órbita. Cuando $t \rightarrow \infty$ se anulan las exponenciales negativas y la solución nos queda

$$\bar{x}(t) \simeq \begin{pmatrix} \frac{1}{2} e^{t-t_0} (x_0 + y_0) \\ \frac{1}{2} e^{t-t_0} (x_0 + y_0) \end{pmatrix}$$

Esto quiere decir que el sistema se aproxima a una recta que pasa por el origen y con una pendiente de 45° ó -45° , cuando t tiende a infinito. Dependiendo de los signos de x_0 e y_0 , tenemos dos rectas de la forma que se puede ver en la gráfica 35.

El sentido de desplazamiento se obtiene igual que se explicó en el ejemplo anterior, sustituyendo x_0 e y_0 en el sistema y observando los signos que toman x' e y' .

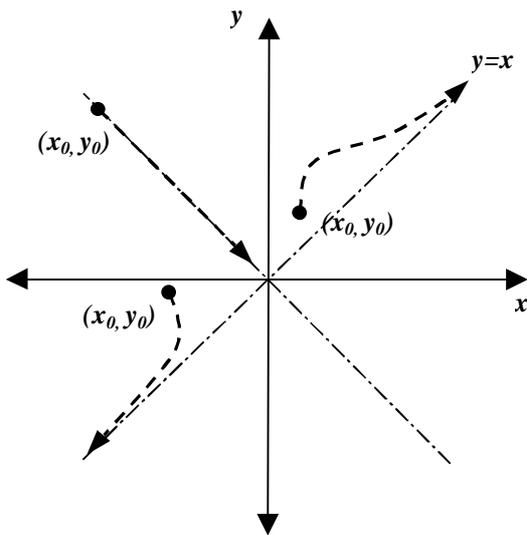


Figura 35: Órbita para $A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.

A continuación vamos a ver cómo se puede calcular, de forma sencilla, la exponencial $e^{A(t-t_0)}$ factorizando la matriz A . Para calcular

$$e^{A(t-t_0)} = Id + A(t-t_0) + A^2 \frac{(t-t_0)^2}{2} + \dots + A^n \frac{(t-t_0)^n}{n!}$$

podemos escribir la matriz A como

$$A = P^{-1} \cdot D \cdot P$$

Sustituyendo A por la expresión anterior, tenemos

$$\begin{aligned} e^{A(t-t_0)} &= e^{P^{-1}DP(t-t_0)} = \\ &= Id + P^{-1}DP(t-t_0) + \\ &+ P^{-1}D \underbrace{PP^{-1}}_I DP \frac{(t-t_0)^2}{2} + \\ &+ P^{-1}D \underbrace{PP^{-1}}_I D \underbrace{PP^{-1}}_I DP \frac{(t-t_0)^3}{3!} \dots + \\ &+ P^{-1}D^n P \frac{(t-t_0)^n}{n!} \end{aligned}$$

Se cancelan los productos de la matriz P con su inversa, con lo que nos queda la siguiente expresión:

$$\begin{aligned} e^{P^{-1}DP(t-t_0)} &= Id + P^{-1}DP(t-t_0) + \\ &+ P^{-1}D^2P \frac{(t-t_0)^2}{2} + \\ &+ P^{-1}D^3P \frac{(t-t_0)^3}{3!} \dots + \\ &+ P^{-1}D^n P \frac{(t-t_0)^n}{n!} \end{aligned}$$

Todos los términos de esta serie están multiplicados por P y por P^{-1} , con lo que si extraemos estos dos factores obtenemos lo siguiente:

$$\begin{aligned} e^{P^{-1}DP(t-t_0)} &= P^{-1} (Id + D(t-t_0) + \\ &+ D^2 \frac{(t-t_0)^2}{2} \dots + D^n \frac{(t-t_0)^n}{n!}) P = \\ &= P^{-1} e^{D(t-t_0)} P, \end{aligned}$$

entonces si $A = P^{-1} \cdot D \cdot P$, se cumple que

$$e^{A(t-t_0)} = P^{-1} e^{D(t-t_0)} P \quad (8)$$

Cuando D es una matriz diagonal, este cálculo es bastante sencillo como se puede ver a continuación:

Una matriz diagonal tiene todos los elementos fuera de la diagonal iguales a cero.

$$D = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{pmatrix}$$

Se puede comprobar fácilmente que las potencias de D son de la forma,

$$D^n = \begin{pmatrix} \lambda_1^n & 0 & \cdots & 0 \\ 0 & \lambda_2^n & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n^n \end{pmatrix}$$

Vamos a calcular la exponencial de la parte derecha de la ecuación (8):

$$e^{D(t-t_0)} = Id + D(t-t_0) + D^2 \frac{(t-t_0)^2}{2} + \dots + D^n \frac{(t-t_0)^n}{n!}$$

Si desarrollamos esta expresión en forma matricial, obtenemos:

$$e^{D(t-t_0)} = \begin{pmatrix} 1 + \lambda_1(t-t_0) + \lambda_1^2 \frac{(t-t_0)^2}{2} \dots + \lambda_1^n \frac{(t-t_0)^n}{n!} & 0 & \cdots & 0 \\ 0 & 1 + \lambda_2(t-t_0) + \lambda_2^2 \frac{(t-t_0)^2}{2} \dots + \lambda_2^n \frac{(t-t_0)^n}{n!} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & 1 + \lambda_n(t-t_0) + \lambda_n^2 \frac{(t-t_0)^2}{2} \dots + \lambda_n^n \frac{(t-t_0)^n}{n!} \end{pmatrix}$$

Cada término de la diagonal es el desarrollo de Taylor de la función $e^{\lambda_i(t-t_0)}$, con lo que $e^{D(t-t_0)}$ queda, en forma matricial, como

$$e^{D(t-t_0)} = \begin{pmatrix} e^{\lambda_1(t-t_0)} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2(t-t_0)} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & e^{\lambda_n(t-t_0)} \end{pmatrix}$$

Lo que se consigue evitar con esta factorización, es calcular las distintas potencias de A y trabajar con desarrollos en serie grandes. La dificultad reside, ahora, en encontrar P tal que cumpla $A = P^{-1} \cdot D \cdot P$, es decir, que A sea *diagonalizable*.

DIAGONALIZACION DE A

Sea A una matriz con una base de autovectores y $\bar{X} = (\bar{x}_1 \ \bar{x}_2 \ \cdots \ \bar{x}_n)$ la matriz que tiene por columnas los autovectores de A . Un cálculo directo demuestra que las columnas de la matriz $A\bar{X}$ son precisamente los vectores $A\bar{x}_1, A\bar{x}_2, \dots, A\bar{x}_n$,

$$A \cdot \bar{X} = (A \cdot \bar{x}_1 \ A \cdot \bar{x}_2 \ \cdots \ A \cdot \bar{x}_n)$$

como $A\bar{x}_i = \lambda_i \bar{x}_i$, se tiene que

$$A \cdot \bar{X} = (\lambda_1 \bar{x}_1 \ \lambda_2 \bar{x}_2 \ \cdots \ \lambda_n \bar{x}_n)$$

Se demuestra fácilmente que la parte derecha de la expresión anterior se puede escribir como

$$(\lambda_1 \bar{x}_1 \ \cdots \ \lambda_n \bar{x}_n) = \bar{X} \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{pmatrix} = \bar{X} \cdot D,$$

donde D es una matriz diagonal que contiene los autovalores de A .

Multiplicando por la inversa de la matriz de autovectores,

$$A \cdot \bar{X} \cdot \bar{X}^{-1} = \bar{X} \cdot D \cdot \bar{X}^{-1}$$

se obtiene que $A = \bar{X} \cdot D \cdot \bar{X}^{-1}$, luego, para $P = \bar{X}^{-1}$ tenemos

$$A = P^{-1} \cdot D \cdot P \quad (9)$$

Este proceso se conoce como *transformación de semejanza* y la ecuación anterior se expresa diciendo que A es *semejante* a la matriz diagonal D . De igual forma, podemos decir que A es *diagonalizable*.

Si los autovectores de A no son linealmente independientes, entonces la matriz X es singular, con lo que no se puede calcular su inversa y, por lo tanto, la matriz A no es diagonalizable.

En el caso en que sus autovectores sean ortonormales, el cálculo de la inversa es inmediato:

$$\bar{X}^{-1} = \bar{X}^T = (\bar{x}_1 \ \bar{x}_2 \ \cdots \ \bar{x}_n)^T = \begin{pmatrix} \bar{x}_1^T \\ \bar{x}_2^T \\ \vdots \\ \bar{x}_n^T \end{pmatrix}$$

Proposición 2 Toda matriz simétrica tiene una base ortonormal de autovectores.

Ejemplo 17 Calcular la solución del sistema

$$\bar{x}(t) = e^{A(t-t_0)} \bar{x}_0,$$

donde

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Al ser una matriz simétrica, sabemos, por la proposición anterior, que posee una base de autovectores ortogonales. Calculamos sus autovalores:

$$|A - \lambda Id| = \begin{vmatrix} -\lambda & 1 \\ 1 & -\lambda \end{vmatrix}$$

$$\lambda^2 - 1 = 0$$

$$\lambda = \pm 1$$

Calculamos los autovectores a partir de sus autovalores. El autovector asociado al autovalor $\lambda = 1$ viene dado por el sistema,

$$\begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

despejando se obtiene la siguiente relación entre las variables,

$$\begin{cases} -x + y = 0 \\ x - y = 0 \end{cases} \quad \begin{cases} x = y, \end{cases}$$

que nos da como autovector $\bar{x}_1 = \begin{pmatrix} x \\ x \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$

Para el autovalor $\lambda = -1$ encontramos el siguiente autovector,

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\begin{cases} x + y = 0 \\ x + y = 0 \end{cases} \quad \begin{cases} x = -y \end{cases}$$

$$\bar{x}_2 = \begin{pmatrix} x \\ -x \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}$$

Elegimos por comodidad autovectores de norma 1, con el fin de que la matriz posea autovectores ortonormales y, poder calcular así, de forma inmediata, su inversa.

$$\bar{X} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Como los autovectores son ortonormales, la inversa de la matriz \bar{X} es:

$$\bar{X}^{-1} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}^T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

que coincide con la matriz original. Sustituyendo estos valores en la ecuación (9), se tiene:

$$A = P^{-1} \cdot D \cdot P = \bar{X} \cdot D \cdot \bar{X}^{-1}$$

$$A = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

El cálculo de la exponencial se simplifica sustancialmente si tenemos en cuenta la ecuación (8):

$$e^{A(t-t_0)} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} e^{D(t-t_0)} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Desarrollando la exponencial, y sabiendo que D es diagonal, obtenemos,

$$e^{A(t-t_0)} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} e^{(t-t_0)} & 0 \\ 0 & e^{-(t-t_0)} \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

realizando las multiplicaciones entre las matrices, tenemos que

$$e^{A(t-t_0)} = \begin{pmatrix} \frac{1}{2}e^{t-t_0} + \frac{1}{2}e^{-t+t_0} & \frac{1}{2}e^{t-t_0} - \frac{1}{2}e^{-t+t_0} \\ \frac{1}{2}e^{t-t_0} - \frac{1}{2}e^{-t+t_0} & \frac{1}{2}e^{t-t_0} + \frac{1}{2}e^{-t+t_0} \end{pmatrix}$$

Ejemplo 18 Calcular la solución del sistema

$$\bar{x}(t) = e^{A(t-t_0)}x_0,$$

donde

$$A = \begin{pmatrix} 2 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{pmatrix}$$

Lo primero que tenemos que hacer es determinar los autovalores de la matriz:

$$|A - \lambda Id| = \begin{vmatrix} 2 - \lambda & 0 & 1 \\ 0 & 1 - \lambda & 0 \\ 1 & 0 & 2 - \lambda \end{vmatrix}$$

$$3 - 7\lambda + 5\lambda^2 - \lambda^3 = 0,$$

Resolviendo esta ecuación por Ruffini obtenemos que los autovalores son $\lambda_1 = 3, \lambda_2 = 1, \lambda_3 = 1$

Calculamos sus autovectores. Para $\lambda_1 = 3$ obtenemos el siguiente autovector:

$$\begin{pmatrix} -1 & 0 & 1 \\ 0 & -2 & 0 \\ 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{cases} -x + z = 0 \\ -2y = 0 \\ x - z = 0 \end{cases} \quad \begin{cases} y = 0 \\ x = z \end{cases}$$

Podemos elegir el primer autovector de la forma:

$$\bar{x}_1 = \begin{pmatrix} x \\ 0 \\ x \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

Los otros dos autovectores corresponden a autovalores repetidos $\lambda_2, \lambda_3 = 1$. Procedemos de la misma manera que antes para calcular los autovectores:

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{cases} x + z = 0 \\ y \text{ libre} \end{cases} \quad \begin{cases} x = -z \end{cases}$$

De esta igualdad, obtenemos que los autovectores 2 y 3 son de la forma:

$$\bar{x}_2, \bar{x}_3 = \begin{pmatrix} x \\ y \\ -x \end{pmatrix}$$

Si damos valores distintos a x y a y , podemos obtener dos autovectores linealmente independientes de la siguiente manera:

$$\bar{x}_2 = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \quad \bar{x}_3 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Ya hemos calculado los autovalores y autovectores de la matriz. Si nos damos cuenta, hemos elegido los autovectores de forma que su norma sea igual a la unidad. La matriz de autovectores viene dada por:

$$\bar{X} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \end{pmatrix}$$

Al ser una matriz de vectores ortonormales, su inversa coincide con su traspuesta:

$$\bar{X}^{-1} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \end{pmatrix}^T = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} \\ 0 & 1 & 0 \end{pmatrix}$$

La diagonalización de A queda de la siguiente manera,

$$A = P^{-1} \cdot D \cdot P = \bar{X}^{-1} \cdot D \cdot \bar{X},$$

donde D es la matriz diagonal de autovalores.

Por último resolvemos la exponencial y multiplicamos las matrices:

$$\begin{aligned} e^{A(t-t_0)} &= P^{-1} \cdot e^{D(t-t_0)} \cdot P = \\ &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \end{pmatrix} \cdot \\ &\cdot \begin{pmatrix} e^{3(t-t_0)} & 0 & 0 \\ 0 & e^{(t-t_0)} & 0 \\ 0 & 0 & e^{(t-t_0)} \end{pmatrix} \cdot \\ &\cdot \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} \\ 0 & 1 & 0 \end{pmatrix} \end{aligned}$$

La solución final es,

$$e^{A(t-t_0)} = \begin{pmatrix} \frac{1}{2}e^{3t-3t_0} + \frac{1}{2}e^{t-t_0} & 0 & \frac{1}{2}e^{3t-3t_0} - \frac{1}{2}e^{t-t_0} \\ 0 & e^{t-t_0} & 0 \\ \frac{1}{2}e^{3t-3t_0} - \frac{1}{2}e^{t-t_0} & 0 & \frac{1}{2}e^{3t-3t_0} + \frac{1}{2}e^{t-t_0} \end{pmatrix}$$

Notas: Esta forma de diagonalización presenta algunas limitaciones como se explica en los siguientes puntos:

- Si la matriz a diagonalizar posee unos autovalores complejos, entonces sus autovectores son también complejos, con lo que la diagonalización tiene otra forma. No vamos a tener en cuenta este caso ya que queda fuera del ámbito de estos apuntes.

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}; \lambda^2 + 1 = 0$$

Tiene raíces complejas, con lo que la diagonalización se realiza de otra manera.

- Si la matriz A no posee una base de autovectores.

- Si A no es simétrica el cálculo de la inversa de la matriz de autovectores no es inmediato.

$$A = \begin{pmatrix} 2 & -1 \\ 0 & 1 \end{pmatrix}; (2-\lambda)(1-\lambda) = 0$$

$$\lambda_1 = 1$$

$$\lambda_2 = 2$$

Autovectores:

$$\underline{\lambda_1 = 1} \quad \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

$$\underline{\lambda_2 = 2} \quad \begin{pmatrix} 0 & -1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

La matriz de autovectores es:

$$X = \begin{pmatrix} \frac{1}{\sqrt{2}} & 1 \\ \frac{1}{\sqrt{2}} & 0 \end{pmatrix},$$

y su inversa:

$$X^{-1} = \begin{pmatrix} 0 & \sqrt{2} \\ 1 & -1 \end{pmatrix}$$

Los autovectores no son ortonormales, con lo que el cálculo de la matriz inversa, X^{-1} , no es inmediato.

A continuación vamos a mostrar un método que permite simplificar el cálculo de e^{At} en ciertos casos. .

Teorema 32 Sean B y C dos matrices, entonces

$$e^{(B+C)t} = e^{Bt} \cdot e^{Ct} \iff B \cdot C = C \cdot B$$

Demostración: El desarrollo en serie de la primera exponencial es de la forma,

$$e^{(B+C)t} = Id + (B+C)t + (B+C)^2 \frac{t^2}{2} \dots + (B+C)^n \frac{t^n}{n!}$$

Demostación 1 El cuadrado de la suma de dos matrices es igual a

$$(B + C)^2 = B^2 + BC + CB + C^2$$

Desarrollamos la expresión de la exponencial $e^{(B+C)t}$ como el producto de dos exponenciales:

$$\begin{aligned} e^{Bt} \cdot e^{Ct} &= \left(Id + Bt + B^2 \frac{t^2}{2} \dots \right) \left(Id + Ct + C^2 \frac{t^2}{2} \dots \right) = \\ &= Id + Ct + C^2 \frac{t^2}{2} \dots + Bt + BCt^2 + BC^2 \frac{t^3}{2} \dots + \\ &+ B^2 \frac{t^2}{2} + B^2 C \frac{t^3}{2} + B^2 C^2 \frac{t^4}{4} \dots = Id + (B+C)t + \\ &+ (C^2 + B^2 + 2BC) \frac{t^2}{2} \dots \end{aligned}$$

Si tomamos el coeficiente del término de segundo orden e igualamos con la expresión dada para $(B + C)^2$ tenemos,

$$\begin{aligned} (C^2 + B^2 + 2BC) &= B^2 + BC + CB + C^2 \\ BC + CB &= 2BC \\ BC &= CB \end{aligned}$$

Para términos de orden superior ocurre lo mismo —se simplifica el coeficiente del término n -ésimo por $(B + C)^n$ — con lo que queda demostrado el teorema.

Cuando la matriz A se pueda expresar como $A = B + C$, cumpliendo que $BC = CB$, entonces el cálculo de e^{At} se simplifica.

Sea $A = B + C$ y $BC = CB$, entonces

$$e^{A(t-t_0)} = e^{(B+C)t} = e^{B(t-t_0)} e^{C(t-t_0)}$$

Ejemplo 19 Supongamos una matriz $A = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$ para la que queremos calcular su exponencial.

Esta matriz se puede separar en la suma de otras dos matrices dadas por

$$B = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \text{ y } C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Para estas dos matrices se cumple que $BC = CB$. Teniendo en cuenta el teorema anterior el cálculo de la exponencial de la matriz A se puede realizar de la siguiente manera:

$$e^{A(t-t_0)} = e^{B(t-t_0)} e^{C(t-t_0)}$$

De problemas anteriores conocemos el resultado de las exponenciales de la parte derecha de la expresión anterior, con lo que la solución viene dada por,

$$e^{A(t-t_0)} = \begin{pmatrix} \cos(t-t_0) & \sin(t-t_0) \\ -\sin(t-t_0) & \cos(t-t_0) \end{pmatrix} \begin{pmatrix} e^{(t-t_0)} & 0 \\ 0 & e^{(t-t_0)} \end{pmatrix}$$

Lo que se persigue a la hora de separar la matriz A en dos sumandos, es encontrar dos matrices B y C cuyas exponenciales ya conozcamos o sean fáciles de calcular.

RESOLUCION DEL CASO MAS GENERAL EN DIMENSION 2

A continuación vamos a ver cómo a partir de una matriz cualquiera $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ se puede proceder, de forma general, para calcular su exponencial.

- Si la matriz A es diagonalizable entonces ya sabemos resolver el sistema.

- Si A no es diagonalizable se puede realizar algún tipo de factorización como la vista en el teorema anterior, para simplificar los cálculos. En los siguientes pasos vamos a ver cómo se puede manipular A para realizar esta factorización:

1. Buscamos una matriz equivalente a A y de la forma $A' = \begin{pmatrix} a' & b' \\ c' & a' \end{pmatrix}$ con los elementos de la diagonal iguales. Si conseguimos una matriz de esta forma, A' se puede dividir en dos matrices en las que una de ellas es igual a la matriz identidad por una constante (a'). Al ser una de estas matrices proporcional a la identidad, se cumple que $BC = CB$, con lo que se puede utilizar el teorema anterior para obtener el resultado de la exponencial.

Para calcular A' probamos factorizaciones del tipo $A' = P^{-1}AP$ de tal forma que, al igual que vimos para la diagonalización de una matriz, se cumple que $e^{A'(t-t_0)} = P^{-1}e^{A(t-t_0)}P$. La exponencial de A se obtiene como $e^{A(t-t_0)} = Pe^{A'(t-t_0)}P^{-1}$. Vamos a suponer que P es una matriz de rotación y vamos a ver cuál es la relación entre las matrices A y A' .

$$\begin{aligned} A' &= \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \cdot A \cdot \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} = \\ &= \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} a \cos \theta - b \sin \theta & a \sin \theta + b \cos \theta \\ c \cos \theta - d \sin \theta & c \sin \theta + d \cos \theta \end{pmatrix} \\ &= \begin{pmatrix} a \cos^2 \theta - b \cos \theta \sin \theta - c \sin \theta \cos \theta + d \sin^2 \theta & \\ a \cos \theta \sin \theta - b \sin^2 \theta + c \cos^2 \theta - d \sin \theta \cos \theta & \\ a \cos \theta \sin \theta + b \cos^2 \theta - c \sin^2 \theta - d \sin \theta \cos \theta & \\ a \sin^2 \theta + b \cos \theta \sin \theta + c \sin \theta \cos \theta + d \cos^2 \theta & \end{pmatrix} \end{aligned}$$

Tal y como se había comentado anteriormente, buscamos que los elementos de la diagonal sean iguales:

$$\begin{aligned} a \cos^2 \theta - b \cos \theta \sin \theta - c \sin \theta \cos \theta + d \sin^2 \theta &= \\ = a \sin^2 \theta + b \cos \theta \sin \theta + c \sin \theta \cos \theta + d \cos^2 \theta & \\ a (\cos^2 \theta - \sin^2 \theta) + d (\sin^2 \theta - \cos^2 \theta) &= \end{aligned}$$

$$= 2(b+c) \sin \theta \cos \theta$$

Sustituyendo en la ecuación anterior $\cos(2\theta) = \cos^2 \theta - \sin^2 \theta$ y $\sin(2\theta) = 2 \sin \theta \cos \theta$, obtenemos

$$a \cos(2\theta) - d \cos(2\theta) = (b+c) \sin(2\theta)$$

Dividiendo por el $\cos(2\theta)$ y agrupando términos,

$$\tan(2\theta) = \frac{a-d}{b+c}$$

Finalmente, el valor del ángulo de rotación viene dado por la siguiente expresión:

$$\theta = \frac{1}{2} \arctan \left(\frac{a-d}{b+c} \right)$$

Una vez calculado el ángulo de rotación, sustituimos su valor en la matriz anterior para obtener la matriz A' . Esta matriz tiene los elementos de la diagonal iguales, con lo que manipulando su exponencial nos queda lo siguiente:

$$\begin{aligned} e^{\begin{pmatrix} a' & b' \\ c' & a' \end{pmatrix}(t-t_0)} &= \\ &= e^{\begin{pmatrix} a' & 0 \\ 0 & a' \end{pmatrix}(t-t_0)} \cdot e^{\begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix}(t-t_0)} \end{aligned}$$

La primera exponencial de la parte derecha de la expresión anterior se calcula directamente. Ahora nos queda calcular la segunda exponencial:

2. Intentamos convertir $C = \begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix}$ en $C' = \begin{pmatrix} 0 & d' \\ d' & 0 \end{pmatrix}$, para que la segunda exponencial nos quede de la forma:

$$e^{\begin{pmatrix} 0 & d' \\ d' & 0 \end{pmatrix}(t-t_0)} = e^{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}(d'(t-t_0))},$$

de la que ya conocemos su solución.

Realizamos la misma operación que en el caso anterior, $C' = Q^{-1}CQ$, de forma que

$$\begin{aligned} e^{C'(t-t_0)} &= Q^{-1}e^{C(t-t_0)}Q \\ e^{C(t-t_0)} &= Qe^{C'(t-t_0)}Q^{-1} \end{aligned}$$

En este caso elegimos, por comodidad, la matriz de la forma $Q^{-1} = \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix}$. La matriz C' tiene la siguiente expresión:

$$C' = Q^{-1}CQ =$$

$$\begin{aligned} &= \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix} \begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{m} & 0 \\ 0 & \frac{1}{n} \end{pmatrix} = \\ &= \begin{pmatrix} 0 & m\frac{b'}{n} \\ n\frac{c'}{m} & 0 \end{pmatrix} \end{aligned}$$

Para conseguir una matriz C' de la forma que estamos buscando, se tiene que cumplir que:

$$m\frac{b'}{n} = n\frac{c'}{m} \iff \frac{b'}{n^2} = \frac{c'}{m^2}$$

Las soluciones posibles que se pueden obtener atendiendo al signo de las variables b' y c' son:

$$\begin{aligned} n = \sqrt{b'}, m = \sqrt{c'} & \quad \text{si } b', c' > 0 \\ n = \sqrt{-b'}, m = \sqrt{-c'} & \quad \text{si } b', c' < 0 \\ n = \sqrt{b'}, m = \sqrt{-c'} & \quad \text{si } b' > 0 \text{ y } c' < 0 \\ n = \sqrt{-b'}, m = \sqrt{c'} & \quad \text{si } b' < 0 \text{ y } c' > 0 \end{aligned}$$

Una vez obtenidos los valores de los elementos de la matriz C' , se obtiene el valor de la exponencial como:

$$e^{C'(t-t_0)} = Qe^{\begin{pmatrix} 0 & m\frac{b'}{n} \\ n\frac{c'}{m} & 0 \end{pmatrix}(t-t_0)}Q^{-1},$$

en donde los elementos de esta matriz son iguales.

Resumen del proceso A continuación se muestra de forma esquematizada el proceso de transformación que se realiza sobre la matriz A para una factorización de la misma.

En el primer paso, se busca una matriz en la que los elementos de la diagonal sean iguales para poder separarla en dos matrices en las que una de ellas sea la matriz identidad multiplicada por una constante:

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \rightarrow A' = \begin{pmatrix} a' & b' \\ c' & a' \end{pmatrix} \rightarrow \left\langle \begin{matrix} B = \begin{pmatrix} a' & 0 \\ 0 & a' \end{pmatrix} \\ C = \begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix} \end{matrix} \right\rangle$$

En el siguiente paso, se intenta transformar la otra matriz para que también tenga valores iguales en la diagonal inversa. Hay dos formas posibles para esta transformación que vienen dadas por:

$$C = \begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix} \rightarrow \left\langle \begin{matrix} \text{Si } b'c' > 0 \quad C' = \begin{pmatrix} 0 & d' \\ d' & 0 \end{pmatrix} \\ \text{Si } b'c' < 0 \quad C' = \begin{pmatrix} 0 & d' \\ -d' & 0 \end{pmatrix} \end{matrix} \right\rangle$$

Al final de este proceso, obtenemos dos matrices de las que conocemos la solución a su exponencial. La solución final es de la forma:

$$\begin{aligned}
e \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{(t-t_0)} &= P e \begin{pmatrix} a' & b' \\ c' & a' \end{pmatrix}^{(t-t_0)} P^{-1} = \\
&= P e^{(B+C)(t-t_0)} P^{-1} = \\
&= P e \begin{pmatrix} a' & 0 \\ 0 & a' \end{pmatrix}^{(t-t_0)} e \begin{pmatrix} 0 & b' \\ c' & 0 \end{pmatrix}^{(t-t_0)} P^{-1} = \\
&= P e \begin{pmatrix} a' & 0 \\ 0 & a' \end{pmatrix}^{(t-t_0)} Q e \begin{pmatrix} 0 & d' \\ d' & 0 \end{pmatrix}^{(t-t_0)} Q^{-1} P^{-1}
\end{aligned}$$

Este proceso es válido para sistemas de orden 2, pero para sistemas de orden 3 o superior el análisis es más complejo. Un caso particular, de orden 3, que sí se podría resolver por los métodos anteriores sería, por ejemplo

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & -1 & 3 \end{pmatrix},$$

en donde en realidad tenemos dos matrices desacopladas que se pueden resolver de forma separada:

$$e^{A(t-t_0)} = \bar{M} \begin{pmatrix} e^{(t-t_0)} & 0 & 0 \\ 0 & e \begin{pmatrix} 1 & 2 \\ -1 & 3 \end{pmatrix}^{(t-t_0)} \\ 0 & & \end{pmatrix}$$

Ejemplo 20 Calcular la solución de

$$\bar{x}' = A\bar{x}$$

con

$$\bar{x}(1) = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \text{ y } A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{pmatrix}$$

Si observamos la matriz A , podemos ver que hay dos matrices desacopladas, que denotaremos por A_1 y A_2 , la primera formada por las tres primeras filas y columnas, y la segunda, formada por las dos últimas. Para resolver este problema, consideramos ambas matrices por separado y trabajamos con cada una como si fuera un sistema independiente.

Empezamos con la primera submatriz:

$$A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} = Id + \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

Vamos a aplicar el método que hemos estudiado anteriormente sabiendo que se cumple $BC = CB$. La exponencial que tenemos que calcular se nos queda entonces como

$$e^{A_1(t-1)} = e \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{(t-1)} e \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}^{(t-1)}$$

La primera exponencial es inmediata. Ahora vamos a ver cómo se puede calcular la segunda.

Calculamos las distintas potencias de la matriz para obtener el desarrollo en serie de la exponencial:

$$M = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, M^2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, M^3 = \bar{0}$$

La exponencial nos da el siguiente resultado:

$$\begin{aligned}
e^{M(t-1)} &= Id + M(t-1) + \frac{M^2}{2!}(t-1)^2 = \\
&= \begin{pmatrix} 1 & 0 & 0 \\ (t-1) & 1 & 0 \\ (t-1)^2 & (t-1) & 1 \end{pmatrix}
\end{aligned}$$

La exponencial de la primera submatriz nos queda entonces como sigue:

$$\begin{aligned}
e^{A_1(t-1)} &= \begin{pmatrix} e^{(t-1)} & 0 & 0 \\ 0 & e^{(t-1)} & 0 \\ 0 & 0 & e^{(t-1)} \end{pmatrix} \cdot \\
&\cdot \begin{pmatrix} 1 & 0 & 0 \\ (t-1) & 1 & 0 \\ (t-1)^2 & (t-1) & 1 \end{pmatrix}
\end{aligned}$$

Ahora vamos a resolver el sistema para la segunda submatriz.

$$A_2 = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} = 2Id + \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}$$

Al igual que para la primera submatriz dividimos esta en dos sumandos simplificando el cálculo de la exponencial

$$e^{A_2(t-1)} = e \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}^{(t-1)} e \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}^{(t-1)}$$

La primera exponencial es inmediata. La segunda se puede reducir a una de las que ya conocemos.

$$e \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}^{(t-1)} = e \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^{(-t+1)}$$

Su solución

$$e \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^{(-t+1)} = \begin{pmatrix} \frac{(e^{t-1} + e^{-(t-1)})}{2} & \frac{(e^{-(t-1)} - e^{t-1})}{2} \\ \frac{(e^{-(t-1)} - e^{t-1})}{2} & \frac{(e^{t-1} + e^{-(t-1)})}{2} \end{pmatrix}$$

para esta segunda submatriz la solución es entonces,

$$\begin{aligned}
e^{A_2(t-1)} &= \begin{pmatrix} e^{2(t-1)} & 0 \\ 0 & e^{2(t-1)} \end{pmatrix} \cdot \\
&\cdot \begin{pmatrix} \frac{1}{2}(e^{t-1} + e^{-(t-1)}) & \frac{1}{2}(e^{-(t-1)} - e^{t-1}) \\ \frac{1}{2}(e^{-(t-1)} - e^{t-1}) & \frac{1}{2}(e^{t-1} + e^{-(t-1)}) \end{pmatrix}
\end{aligned}$$

Por tanto, la solución final es

$$e^{A(t-1)} = \begin{pmatrix} & 0 & 0 \\ e^{A_1(t-1)} & 0 & 0 \\ & 0 & 0 \\ 0 & 0 & 0 & e^{A_2(t-1)} \\ 0 & 0 & 0 & \end{pmatrix}$$

CLASIFICACION DE LAS ORBITAS

En esta sección vamos a explicar cómo se pueden obtener, de forma general, las órbitas asociadas a un sistema de ecuaciones diferenciales partiendo de los autovalores y autovectores de la matriz asociada.

Sea el siguiente sistema lineal de ecuaciones diferenciales

$$\bar{x}'(t) = A\bar{x}(t)$$

en donde $\bar{x}'(t)$ es el vector de primeras derivadas de las variables y A es la matriz del sistema que relaciona las derivadas con sus variables. Por el siguiente teorema se tiene que las posibles soluciones del sistema vienen dadas por:

Teorema 33 Si \bar{x}_i es un autovector de A para λ_i autovalor, entonces $\bar{x}(t) = \bar{x}_i e^{\lambda_i(t-t_0)}$ es una solución del sistema.

Demostración: Tenemos el sistema $\bar{x}'(t) = A\bar{x}(t)$. La solución dada por el teorema es $\bar{x}(t) = \bar{x}_i e^{\lambda_i(t-t_0)}$ y su derivada $\bar{x}'(t) = \lambda_i \bar{x}_i e^{\lambda_i(t-t_0)}$.

Demostración 2 Sustituimos la solución y su derivada en el sistema para comprobar que efectivamente se cumple el teorema:

$$\begin{aligned} \bar{x}'(t) &= A\bar{x}(t) \\ \lambda_i \bar{x}_i e^{\lambda_i(t-t_0)} &= A\bar{x}_i e^{\lambda_i(t-t_0)}, \end{aligned}$$

como \bar{x}_i es un autovector asociado a $\lambda_i \implies \lambda_i \bar{x}_i = A\bar{x}_i$ resulta $\bar{x}'(t) = A\bar{x}_i e^{\lambda_i(t-t_0)} = A\bar{x}(t)$, c.q.d.

Supongamos que tenemos dos autovalores y sus autovectores, entonces tenemos dos soluciones de la forma,

$$\begin{aligned} \lambda_1 &\longrightarrow \bar{x}_1(t) = \bar{x}_1 e^{\lambda_1(t-t_0)} \\ \lambda_2 &\longrightarrow \bar{x}_2(t) = \bar{x}_2 e^{\lambda_2(t-t_0)} \end{aligned}$$

Los puntos de salida en $t = t_0$ son \bar{x}_1 y \bar{x}_2 (autovectores) respectivamente.

La solución general del sistema $\bar{x}(t)$, para $\bar{x}(t_0) = \bar{x}_0$, se puede obtener como una combinación lineal entre las soluciones particulares obtenidas, $\bar{x}(t) = \eta_1 \bar{x}_1(t) + \eta_2 \bar{x}_2(t)$, donde η_1, η_2 se calculan forzando que $\bar{x}(t_0)$ sea igual a \bar{x}_0 .

En este caso tenemos $\bar{x}_0 = \eta_1 \bar{x}_1 + \eta_2 \bar{x}_2$, lo que nos da un sistema de ecuaciones con dos incógnitas.

Ejemplo 21 Supongamos que tenemos los autovectores de la matriz del sistema dados por

$$\bar{x}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}; \quad \bar{x}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

y una condición inicial sobre el sistema de la forma

$$\bar{x}_0 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

Podemos calcular los coeficientes de la combinación lineal a partir del sistema de ecuaciones siguiente:

$$\begin{aligned} 2 &= \eta_1 + \eta_2 \\ 1 &= \eta_1 - \eta_2 \end{aligned}$$

Uno de los η_i será cero si \bar{x}_0 coincide con la dirección de uno de los autovectores.

A continuación vamos a hacer un estudio de los posibles casos que se puedan dar dependiendo del valor de los autovalores. En sistemas de 2 coordenadas, nos encontramos con estas posibles situaciones:

1. $\lambda_1, \lambda_2 > 0$
2. $\lambda_1 \cdot \lambda_2 < 0$
3. $\lambda_1, \lambda_2 < 0$
4. $\lambda_1 = 0, \lambda_2 = 0$
5. $\lambda_1 = 0, \lambda_2 > 0$
6. $\lambda_1 = 0, \lambda_2 < 0$
7. λ_1, λ_2 complejos

Vamos a hacer un estudio detallado para cada caso mostrando cómo serían las órbitas para cada uno.

Caso 1. Autovalores reales positivos $\lambda_1, \lambda_2 > 0$.

En este caso, la solución del sistema será de la forma:

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

en donde λ_1, λ_2 son positivos. Al ser estos positivos, las exponenciales tenderán hacia infinito a medida que t aumenta.

El origen, punto $(0,0)$, funciona como *repulsor* (ver figura 36). Si partimos del origen, el sistema permanece en él.

El sistema $A\bar{x}$ está centrado en el origen. Si el sistema fuera $A(\bar{x} - \bar{x}_o)$, estaría centrado en \bar{x}_o .

Un caso especial viene dado si $\lambda_1 = \lambda_2$. En el siguiente ejemplo se estudia este caso.

Ejemplo 22 Supongamos que tenemos la matriz del sistema siguiente

$$A = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Los autovalores de esta son iguales y su valor es $\lambda = 1$ (doble).

Calculamos los autovectores:

$$\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$x = 0$$

Los autovectores son de la forma $\bar{x} = (0, a)$

Calculando la solución de forma explícita:

$$\begin{aligned} e^{\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}(t-t_0)} &= e^{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}(t-t_0)} e^{\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}(t-t_0)} = \\ &= \begin{pmatrix} e^{(t-t_0)} & 0 \\ 0 & e^{(t-t_0)} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ (t-t_0) & 1 \end{pmatrix} = \\ &= \begin{pmatrix} e^{(t-t_0)} & 0 \\ e^{(t-t_0)}(t-t_0) & e^{(t-t_0)} \end{pmatrix} \end{aligned}$$

Multiplicando por $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$ tenemos la solución. Las exponenciales siguen siendo positivas y el origen sigue funcionando como repulsor.

Caso 2. Autovalores reales de signo contrario $\lambda_1 > 0, \lambda_2 < 0$

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

El primer término tiende a ∞ , mientras que el segundo tiende a 0. Las órbitas tendrán forma de punto de silla como se puede ver en la figura 37.

Si partimos de un punto sobre la recta \bar{x}_1 el sistema tiende hacia ∞ . Si partimos de un punto sobre la recta \bar{x}_2 el sistema tiende hacia 0.

Caso 3. Autovalores reales negativos $\lambda_1, \lambda_2 < 0$

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

Ambos términos tienden a 0 pero sin alcanzarlo. Las órbitas son de tipo atractor (ver figura 38).

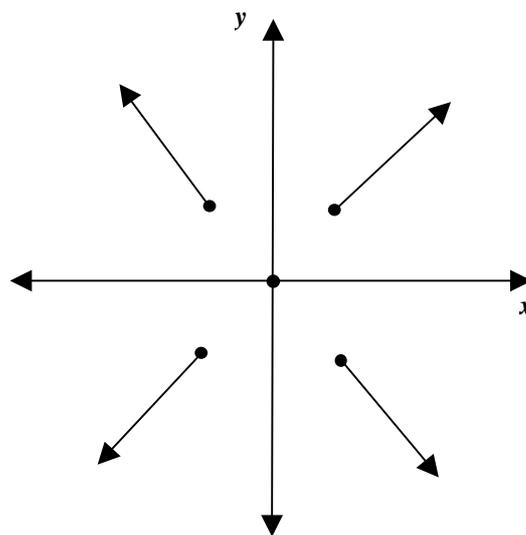


Figura 36: Repulsor

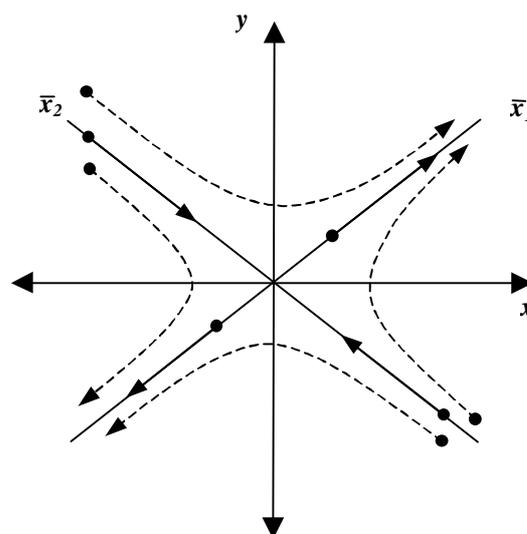


Figura 37: Punto de silla

Caso 4. Autovalores nulos $\lambda_1 = \lambda_2 = 0$

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

Supongamos que $A = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, entonces $e^{A(t-t_0)} = Id$, con lo que la solución del sistema es siempre $\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$, y no hay movimiento.

Si $A = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$, entonces tenemos que $e^{A(t-t_0)} = \begin{pmatrix} 1 & 0 \\ t & 1 \end{pmatrix}$, a pesar de que los autovalores siguen siendo 0. En este caso, la posición en x permanecería constante mientras que la variación en y sería negativa si \bar{x}_0 es negativa o positiva si esta es positiva. En la figura 39 se puede ver la forma de las órbitas para este caso.

La diferencia está en que en el primer ejemplo se pueden encontrar dos autovectores, mientras que en el segundo sólo se puede encontrar 1.

Caso 5. Autovalores $\lambda_1 = 0, \lambda_2 > 0$

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

En este caso, tenemos una mezcla entre los casos 1 y 4, en los que por un lado teníamos una órbita de tipo repulsor y por otro, no teníamos movimiento. Lo que sucede ahora es que en vez de tener un punto que actúa como repulsor, tenemos una recta dada por el autovector asociado al autovalor λ_1 . En la gráfica 40 vemos la forma que adoptarían las órbitas con respecto a esta recta.

Caso 6. Autovalores $\lambda_1 = 0, \lambda_2 < 0$

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

Este caso es muy similar al anterior excepto que la recta en vez de comportarse como repulsora, se comporta como atractora (ver figura 41).

Caso 7. Raíces complejas conjugadas λ_1, λ_2 .

$$\bar{x}(t) = \eta_1 \bar{x}_1 e^{\lambda_1(t-t_0)} + \eta_2 \bar{x}_2 e^{\lambda_2(t-t_0)}$$

La forma genérica de una matriz con autovalores complejos es la siguiente:

$$\begin{pmatrix} a & b \\ -b & a \end{pmatrix},$$

ya que si calculamos sus autovalores obtenemos

$$\begin{vmatrix} a - \lambda & b \\ -b & a - \lambda \end{vmatrix} = (a - \lambda)^2 + b^2$$

$$\boxed{\lambda = a \pm bi}$$

que es la forma general de un número complejo.

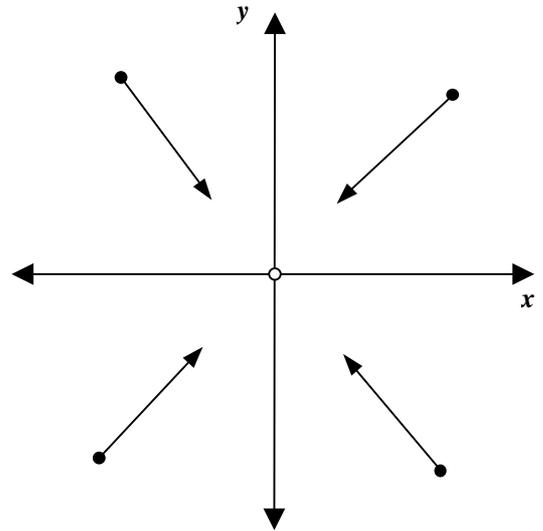


Figura 38: Atractor

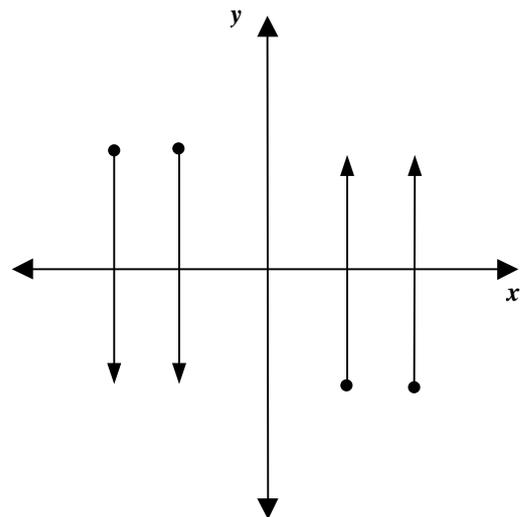


Figura 39: Autovalores nulos $\lambda_1 = \lambda_2 = 0$.

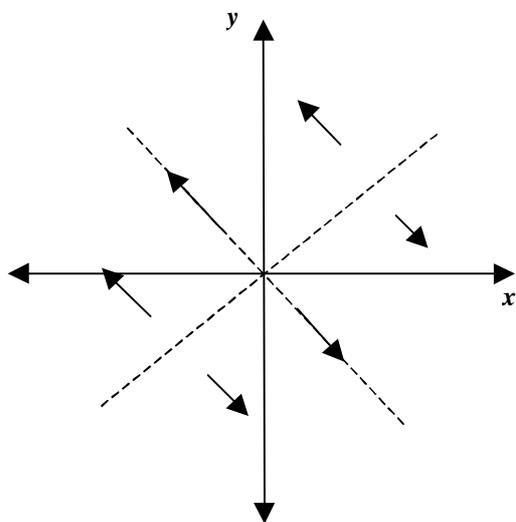


Figura 40: Autovalores $\lambda_1 = 0, \lambda_2 > 0$.

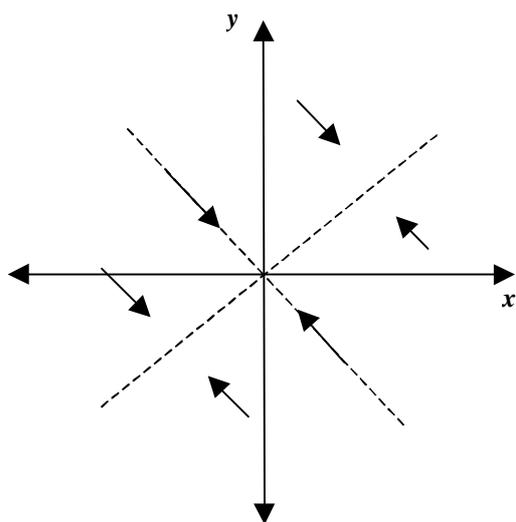


Figura 41: Autovalores $\lambda_1 = 0, \lambda_2 < 0$.

Si descomponemos A de la siguiente manera:

$$\begin{pmatrix} a & b \\ -b & a \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & a \end{pmatrix} + \begin{pmatrix} 0 & b \\ -b & 0 \end{pmatrix}$$

y luego calculamos la exponencial,

$$\begin{aligned} e^{\begin{pmatrix} a & b \\ -b & a \end{pmatrix} t} &= e^{\begin{pmatrix} a & 0 \\ 0 & a \end{pmatrix} t} e^{\begin{pmatrix} 0 & b \\ -b & 0 \end{pmatrix} t} = \\ &= \begin{pmatrix} e^{a(t-t_0)} & 0 \\ 0 & e^{a(t-t_0)} \end{pmatrix} \cdot \\ &\cdot \begin{pmatrix} \cos(b(t-t_0)) & \sin(b(t-t_0)) \\ -\sin(b(t-t_0)) & \cos(b(t-t_0)) \end{pmatrix} = \\ &= e^{a(t-t_0)} \begin{pmatrix} \cos(b(t-t_0)) & \sin(b(t-t_0)) \\ -\sin(b(t-t_0)) & \cos(b(t-t_0)) \end{pmatrix} \end{aligned}$$

En esta ecuación tenemos dos términos diferentes. Por una parte, tenemos una exponencial que depende del valor de a y que actúa a modo de amplitud sobre el segundo término; y por otro lado, tenemos una matriz de rotación que hace que el sistema progrese de forma angular dependiendo del valor de b . Si atendemos al valor de a se pueden distinguir varios casos:

1. Cuando $a = 0$, tenemos

$$e^{\begin{pmatrix} a & b \\ -b & a \end{pmatrix} t} = \begin{pmatrix} \cos(b(t-t_0)) & \sin(b(t-t_0)) \\ -\sin(b(t-t_0)) & \cos(b(t-t_0)) \end{pmatrix}$$

En esta ecuación se selecciona el segundo término, con lo que el sistema sólo tendrá un movimiento circular. Las órbitas son circunferencias que se recorren en un sentido u otro dependiendo del signo de b . En la gráfica 42 vemos la forma de la órbita para los posibles valores de b .

2. Para $a > 0$, la solución tendrá la siguiente forma:

$$e^{a(t-t_0)} \begin{pmatrix} \cos(b(t-t_0)) & \sin(b(t-t_0)) \\ -\sin(b(t-t_0)) & \cos(b(t-t_0)) \end{pmatrix}$$

en donde la exponencial está elevada a un valor positivo, con lo que su valor tenderá a infinito a medida que aumenta t .

El efecto que se produce es el de un aumento en la magnitud del radio de la circunferencia, resultando una órbita en forma de espiral.

En las figuras 43, 44 se pueden ver la órbitas dependiendo del valor de b .

3. Para $a < 0$, el resultado es parecido al del apartado anterior excepto que el valor de la exponencial tiende a 0. Esto produce un efecto de reducción del radio con lo que las espirales serán hacia el interior. En las gráficas 45 y 46 se pueden ver las órbitas dependiendo del valor de b .

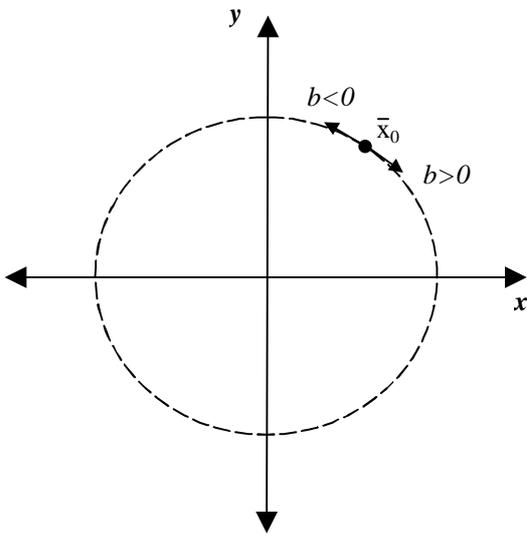


Figura 42: $a = 0$

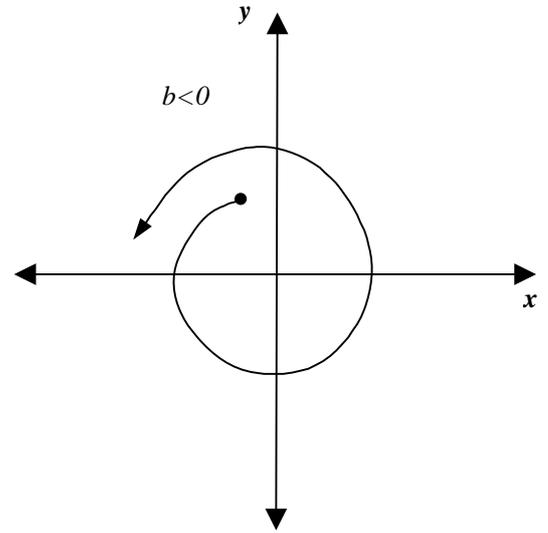


Figura 44: $a > 0, b < 0$

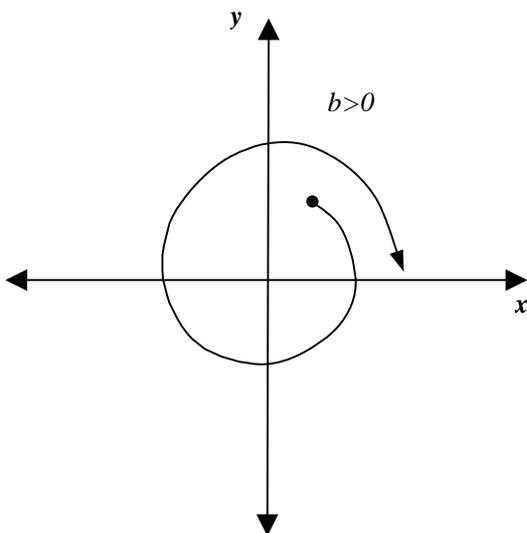


Figura 43: $a > 0, b > 0$

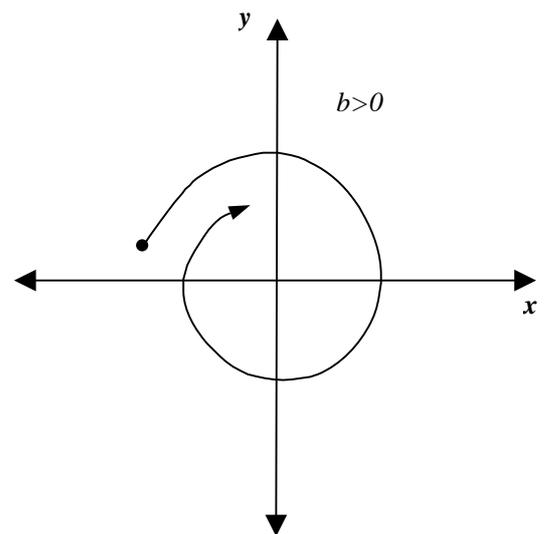


Figura 45: $a < 0, b > 0$

SISTEMAS DE ECUACIONES DIFERENCIALES NO LINEALES

En los sistemas no lineales no podemos llegar tan fácilmente a una expresión matemática de la solución. Por lo general es muy difícil, si no imposible, encontrar una solución a una ecuación diferencial no lineal; por lo tanto, es importante considerar que puede obtenerse información cualitativa acerca de las soluciones sin tener que resolverlas. En muchas aplicaciones una pregunta importante es si pequeños cambios en las condiciones iniciales (entradas) conducen a pequeños cambios (estabilidad) o a grandes cambios (inestabilidad) en la solución (salida). Una segunda pregunta surge cuando una ecuación no lineal se aproxima por medio de una lineal más sencilla. Puede resultar que este tipo de aproximación sea conveniente para algunos casos, mientras que para otros, la linealización no dé buenos resultados.

PUNTOS CRITICOS

Los puntos críticos del sistema son aquellos en los que todas las derivadas se anulan simultáneamente.

$$\left. \begin{array}{l} f_1(\bar{x}_c) = 0 \\ \vdots \\ f_n(\bar{x}_c) = 0 \end{array} \right\}, \bar{x}_c \text{ punto crítico.}$$

Estos puntos son interesantes para el análisis ya que cuando el sistema se encuentra en un punto crítico, éste no tiene movimiento y permanece estable.

Teorema 34 *El comportamiento de las órbitas de un sistema de ecuaciones diferenciales no lineal cerca de un punto crítico es similar al del sistema lineal correspondiente excepto en dos casos particulares donde no podemos asegurar nada. Dichos casos se producen cuando los autovalores del sistema lineal son imaginarios puros o cuando son reales e iguales.*

No vamos a demostrar este teorema, pero vamos a ver de forma intuitiva por qué se dan estas dos excepciones:

- En el caso en que los autovalores son imaginarios puros, las órbitas del sistema lineal son periódicas. Sin embargo, al considerar el sistema no-lineal completo, los términos no lineales del desarrollo pueden hacer que la órbita se desvie de su trayectoria perfecta y se transformen en órbitas en forma de espiral.

- Cuando los autovalores son reales e iguales significa que el discriminante de $\lambda = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ es igual a cero. Esto corresponde al caso límite que separa el hecho de tener autovalores complejos (que dan un comportamiento en forma de espiral) del hecho de tener autovalores reales distintos (No forman espirales). Por ello, en este caso, los términos de orden superior del desarrollo de Taylor pueden

modificar la trayectoria y sacarla del tipo de trayectorias del sistema lineal. Un caso extremo de esto es cuando la matriz del sistema lineal es nula. En cuyo caso, el sistema lineal no nos dice nada sobre el comportamiento del no-lineal cerca del punto crítico.

A partir del siguiente ejemplo vamos a ver cómo se puede estudiar un sistema no lineal de forma sistemática.

Ejemplo 23 *Sea el siguiente sistema de ecuaciones diferenciales no lineales*

$$\begin{aligned} x'(t) &= x(a - y) \\ y'(t) &= y(x - b) \end{aligned}$$

Lo primero que vamos a hacer para estudiar este sistema, es determinar los *puntos críticos*. En nuestro ejemplo, los puntos críticos se calculan de la siguiente manera:

$$\begin{aligned} 0 &= x(a - y) \\ 0 &= y(x - b) \end{aligned}$$

La primera ecuación se anula para valores de $x = 0$ y para valores de $y = a$. La segunda ecuación se anula para $x = b$ e $y = 0$. A partir de estos valores, tenemos que los puntos críticos son $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ y $\begin{pmatrix} b \\ a \end{pmatrix}$, que son los puntos para los que ambas ecuaciones se anulan simultáneamente.

LINEALIZACION

Una vez hemos determinado los puntos críticos del sistema, hay que linealizar el sistema en torno a estos puntos para estudiar la forma que van a tener las órbitas. De forma general, las órbitas serán,

1. Periódicas
2. Tipo Repulsor
3. Tipo Atractor

Para realizar la linealización utilizamos el desarrollo en serie de Taylor:

$$f(x) = f(x_c) + f'(x_c)(x - x_c) + \frac{f''(x_c)}{2!}(x - x_c)^2 + \dots \quad (10)$$

eliminando los términos no lineales, de orden 2 y superior.

Para sistemas de dimensión 2 ó superior, en vez de derivadas de la función f , tenemos el gradiente de f , ∇f .

$$\bar{x}' = \nabla f(\bar{x}_c)(\bar{x} - \bar{x}_c),$$

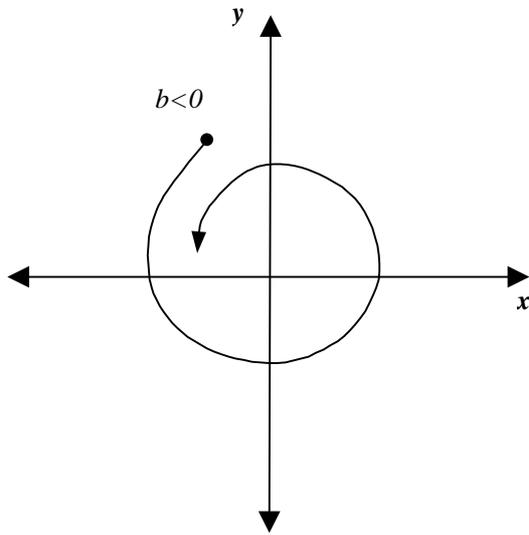


Figura 46: $a < 0, b < 0$

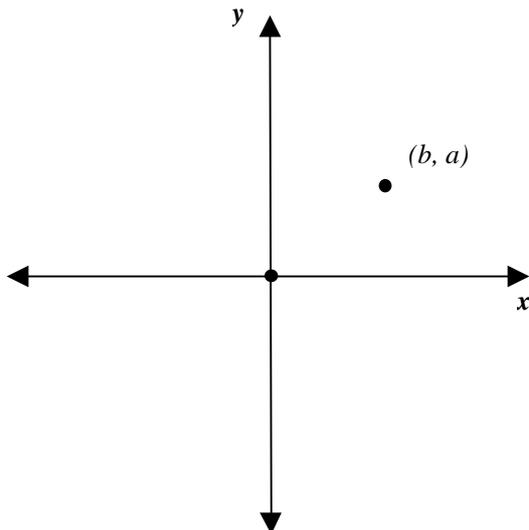


Figura 47: Puntos críticos

siendo

$$\nabla f = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

Siguiendo con el ejemplo anterior, calculamos el gradiente de la función obteniendo

$$\nabla f = \begin{pmatrix} a - y & -x \\ y & x - b \end{pmatrix}$$

Una vez calculado el gradiente, ya podemos linealizar el sistema en torno a cada punto crítico. La linealización para el ejemplo anterior es de la siguiente manera:

Para el punto crítico $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$, sustituimos este valor en la ecuación (10). $f(x_c)$ es igual a cero ya que el sistema se anula para el punto crítico.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \nabla f \begin{pmatrix} 0 \\ 0 \end{pmatrix} \left(\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \end{pmatrix} \right)$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & -b \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Para el otro punto $\begin{pmatrix} b \\ a \end{pmatrix}$, sustituyendo en (10).

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \nabla f \begin{pmatrix} b \\ a \end{pmatrix} \left(\begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} b \\ a \end{pmatrix} \right)$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 0 & -b \\ a & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Llegados a este punto tenemos dos sistemas lineales y podemos realizar el análisis como si de sistemas lineales se tratara. Analizamos cada punto crítico por separado:

1. Punto crítico $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$

Tal y como vimos en los sistemas lineales, tenemos que calcular los autovalores de la matriz $\begin{pmatrix} a & 0 \\ 0 & -b \end{pmatrix}$ y sus autovectores para determinar la forma de la órbita. Esta matriz tiene dos autovalores, $\lambda_1 = a$, $\lambda_2 = -b$, con $a > 0$ y $-b < 0$. Ahora vamos a calcular los autovectores:

Para $\lambda = a$:

$$\begin{pmatrix} 0 & 0 \\ 0 & -b - a \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Los autovectores son de la forma:

$$\bar{x}_1 = \begin{pmatrix} v \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Para $\lambda = b$:

$$\begin{pmatrix} a+b & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Los autovectores son de la forma:

$$\bar{x}_2 = \begin{pmatrix} 0 \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

En la gráfica 48, se puede ver la forma de la órbita alrededor de este punto crítico. Tal y como vimos para los sistemas lineales, cuando la matriz tiene dos autovalores reales de signos distintos, la órbita es en forma de punto de silla.

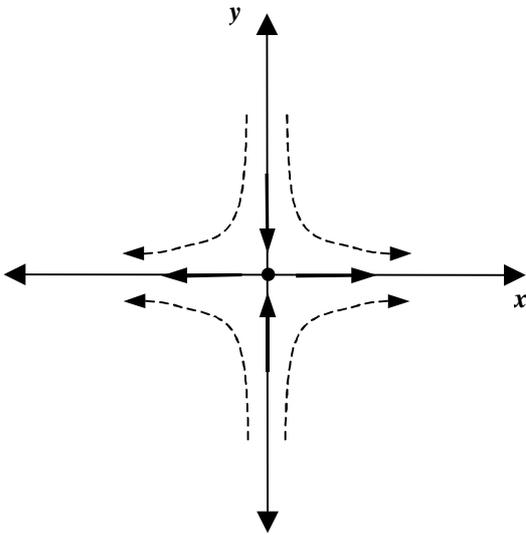


Figura 48: Órbita cerca del punto crítico $(0,0)$.

En la dirección del primer autovector, $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, nos alejamos del punto crítico ($a > 0$). En la dirección del segundo, $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$, nos acercamos al punto crítico ($-b < 0$).

2. Punto crítico $\begin{pmatrix} b \\ a \end{pmatrix}$

En este caso la matriz $\begin{pmatrix} 0 & -b \\ a & 0 \end{pmatrix}$ tiene dos autovalores imaginarios puros:

$$\lambda^2 + ab = 0, \lambda = \pm\sqrt{abi}.$$

Al ser los autovalores complejos puros, las órbitas son circulares y su sentido está definido por el signo del elemento superior derecho de la matriz ($-b$).

CAMPO DE DIRECCIONES

A continuación vamos a ver una técnica que sirve para completar la información que se obtiene a partir de los puntos críticos.

Esta técnica consiste en dividir el plano en regiones y dibujar en cada una de ellas la tendencia de las órbitas.

En los siguientes ejemplos vemos cómo se puede aplicar esta técnica a distintos tipos de sistemas:

Ejemplo 24 Sea el siguiente sistema de ecuaciones diferenciales

$$\begin{aligned} x' &= y \\ y' &= -x \end{aligned}$$

para el que queremos estudiar su campo de direcciones.

Se dividen las regiones de manera que x' e y' tengan signos diferentes. Considerando solamente el sistema, y dando valores distintos a x e y , vamos dibujando, para cada región, el sentido del vector dado por x' e y' en cada caso. Este vector nos da la dirección en que se mueve la órbita para esta región. Finalmente, una vez hayamos dibujado los vectores de movimiento para cada región, podemos intuir, de forma aproximada, la trayectoria global de la órbita.

En la figura 50 podemos ver las distintas regiones ($x > 0, y > 0$; $x < 0, y > 0$; $x < 0, y < 0$; $x > 0, y < 0$) y el vector de movimiento de la órbita obtenido a partir de la variación en cada componente (x', y').

Si consideramos de forma global los distintos vectores en cada región, parece que la órbita se aproxima a una elipse o circunferencia.

Efectivamente, la matriz del sistema está dada por $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$, de la que, analizando sus autovalores, $\lambda = \pm i$, vemos que estos son complejos puros y, por lo tanto, su órbita es circular.

Ejemplo 25 Vamos a analizar ahora el campo de direcciones del siguiente sistema no lineal

$$\begin{aligned} x' &= x(a - y) \\ y' &= y(x - b) \end{aligned}$$

En primer lugar vamos a determinar las distintas regiones que pueden haber. Para esto, estudiamos los cambios de signo de x' e y' :

- x' cambia de signo cuando x pasa por cero y también cambia de signo cuando y pasa por el valor a .

- y' cambia de signo cuando x pasa por b y cuando y pasa por 0.

Una vez hemos determinado las regiones dibujamos, para cada una, la dirección que toman x' e y' . En las gráficas 52 y 51 se pueden ver las distintas regiones para x' e y' y las direcciones por separado. Para determinar estas direcciones, lo que hay que hacer es sustituir valores de x e y , de cada región, en el sistema y ver los signos que toman x' e y' . Estos signos determinan la orientación de las flechas (hacia la izquierda si son negativos en x' o hacia la derecha si son positivos; hacia arriba o hacia abajo para los de y').

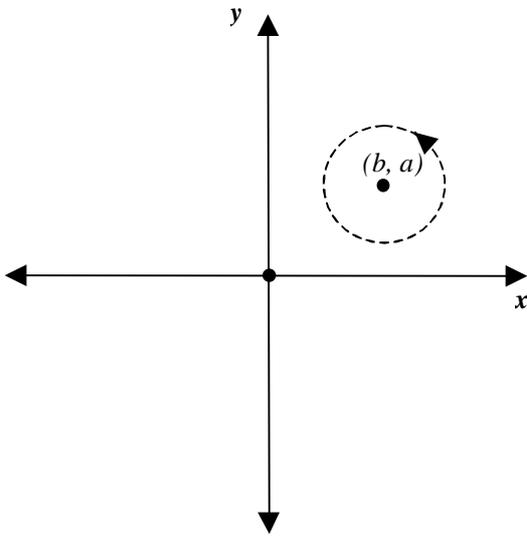


Figura 49: Orbita cerca del punto crítico (b, a) .

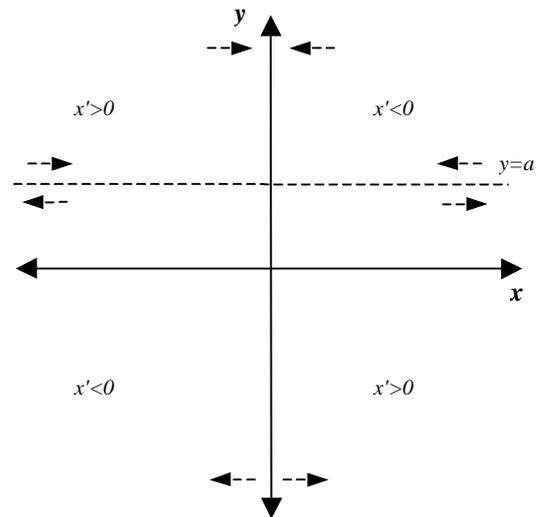


Figura 51: Regiones determinadas por x' .

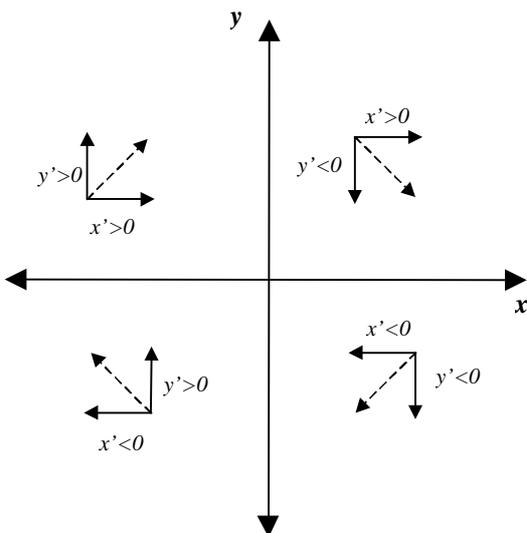


Figura 50: Campo de direcciones.

Cuando ya tenemos las direcciones en cada región para x' e y' , componemos las dos gráficas y obtenemos la figura 53.

Ejemplo 26 Estudiar el siguiente sistema y dibujar su campo de direcciones

$$\begin{aligned} x' &= y(x^2 - 3) - 2x \\ y' &= -x(1 - y) \end{aligned}$$

En primer lugar vamos a determinar los puntos críticos del sistema. Para esto, igualamos ambas ecuaciones a cero y obtenemos los valores de x e y para los que se cumple la igualdad:

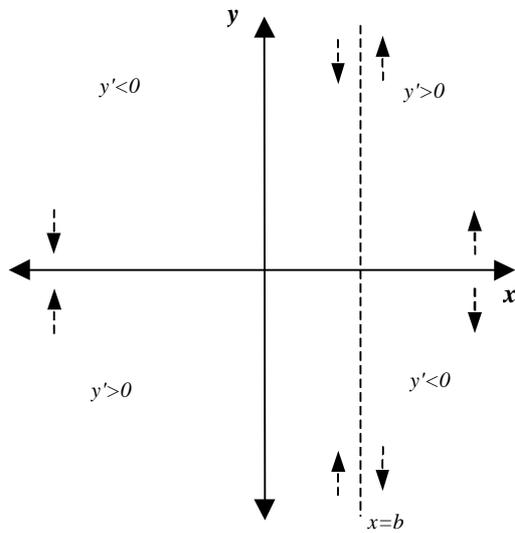


Figura 52: Regiones determinadas por y' .

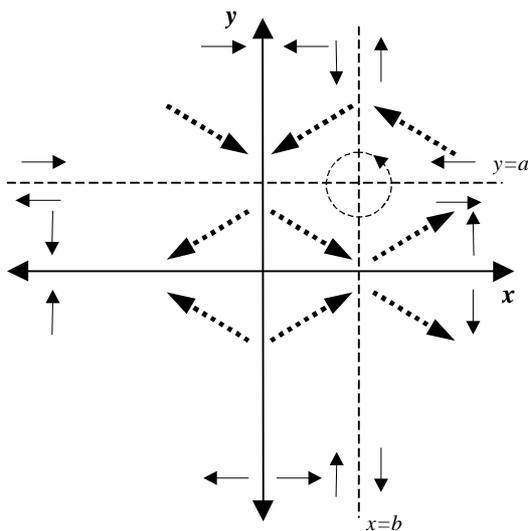


Figura 53: Campo de direcciones.

$$\begin{aligned} x' = 0 &= y(x^2 - 3) - 2x \\ y' = 0 &= -x(1 - y) \end{aligned}$$

Las ecuaciones anteriores se anulan para los siguientes valores de x e y :

$$\begin{aligned} y &= 2\frac{x}{x^2-3} \\ y &= 1 \text{ y } x = 0 \end{aligned}$$

Resolviendo $y = 2\frac{x}{x^2-3}$ para los valores de $y = 1$ y $x = 0$ obtenemos:

$$\begin{aligned} y = 0, x &= -1 \text{ y } x = 3 \\ y = 1 \text{ y } x &= 0 \end{aligned}$$

Los puntos críticos son entonces $(0,0)$, $(-1,1)$, $(3,1)$.

Después de determinar los puntos críticos, habría que linealizar el sistema no lineal con el fin de estudiar el comportamiento del sistema entorno a estos puntos. El sistema tiene el siguiente comportamiento:

$\begin{pmatrix} 0 \\ 0 \end{pmatrix}$ Punto de silla.

$\begin{pmatrix} -1 \\ 1 \end{pmatrix}$ Atractor

$\begin{pmatrix} 3 \\ 1 \end{pmatrix}$ Repulsor

El campo de direcciones se obtiene observando el signo que toman x' e y' en cada una de las regiones definidas por los valores de x e y obtenidos para los puntos críticos (ver figura 54).

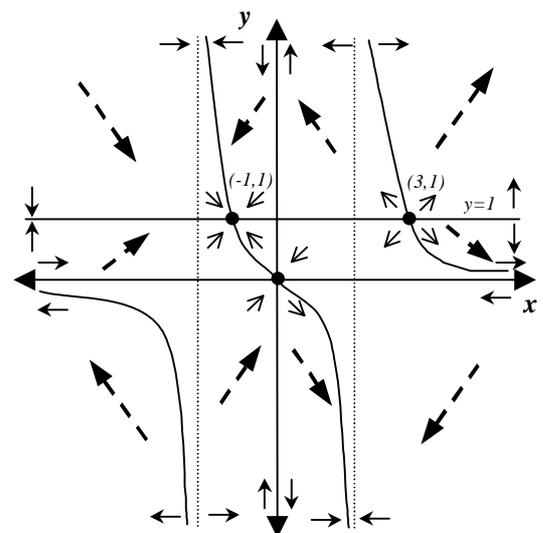


Figura 54: Campo de direcciones.

A continuación vamos a ver un ejemplo completo en el que se estudia un sistema no lineal.

Ejemplo 27 Estudiar puntos críticos, campo de direcciones y órbitas de

$$x'' + \sin(x) = 0$$

Esta ecuación diferencial se puede expresar en forma de sistema si hacemos el siguiente cambio de variable:

$$\begin{aligned} x' &= y \\ x'' &= y' = -\sin(x) \end{aligned}$$

Los puntos críticos del sistema están dados por:

$$\begin{aligned} y &= 0 \\ x &= k\pi, \quad k \in \mathbb{Z} \end{aligned}$$

El $\sin(x)$ se anula en valores múltiplos enteros de π , con lo que vamos a tener infinitos puntos críticos situados sobre el eje x ($y = 0$) y espaciados por una distancia de π .

Linealizamos el sistema para comprobar cuál es el comportamiento del sistema alrededor de los puntos críticos:

$$\nabla f(\bar{x}_c) = \begin{pmatrix} 0 & 1 \\ -\cos(x) & 0 \end{pmatrix}$$

Como los puntos críticos están situados a una distancia de π , el valor del coseno podrá ser 1 ó -1 dependiendo del valor de k :

$$\cos(k\pi) = \begin{cases} 1 & \text{si } k \text{ es par} \\ -1 & \text{si } k \text{ es impar} \end{cases}$$

Como tenemos dos casos distintos que incluyen puntos críticos distintos, estudiamos cada caso por separado:

k par

La forma de la matriz es

$$\nabla f(\bar{x}_c) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

que, como ya hemos visto, tiene autovalores imaginarios $\lambda_{1,2} = \pm i$. Por lo tanto, las órbitas son circulares y periódicas.

k impar

La matriz es:

$$\nabla f(\bar{x}_c) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

cuyos autovalores son $\lambda_{1,2} = \pm 1$. Como ya hemos visto, sus órbitas son en forma de punto de silla. Calculamos sus autovectores para obtener la inclinación de los puntos

de silla. Para $\lambda_1 = 1$ su autovector es

$$\begin{aligned} \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \bar{x}_1 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix} \end{aligned}$$

Para $\lambda_2 = -1$

$$\begin{aligned} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \bar{x}_2 &= \begin{pmatrix} 1 \\ -1 \end{pmatrix} \end{aligned}$$

En la figura 55 se pueden ver las órbitas superpuestas para los dos casos posibles dependiendo del valor de k .

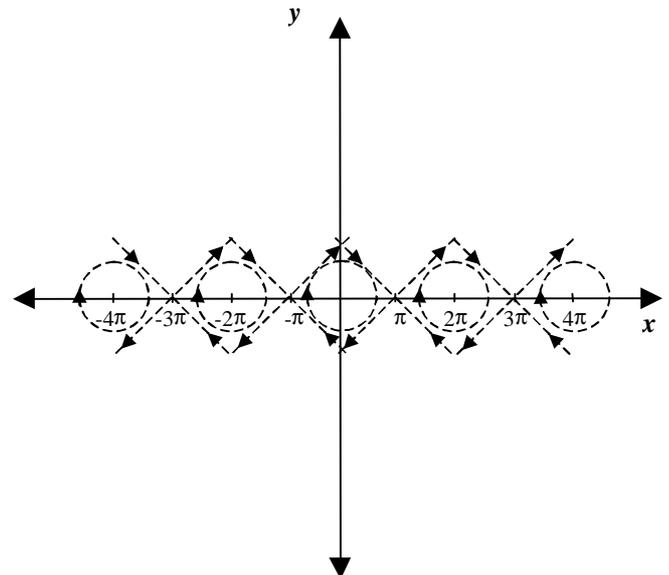
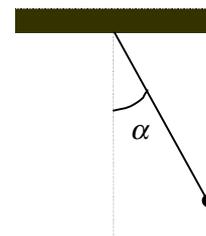


Figura 55: Órbita para $x'' + \sin(x) = 0$.

En la figura 56 se representa el campo de direcciones para este sistema.

$$\begin{aligned} x' &= y \\ y' &= -\sin(x) \end{aligned}$$

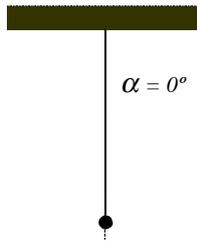
Esta ecuación modeliza la oscilación de un péndulo. $x(t)$ mide el ángulo de desviación con respecto a la vertical.



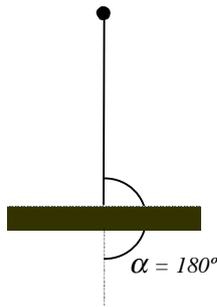
Cada una de las órbitas representa un caso de la realidad.

1. Las órbitas periódicas se producen cuando se suelta el péndulo y no existe rozamiento.
2. Las órbitas en forma de punto de silla aparecen cuando el péndulo se tira con tanta fuerza que empieza a dar vueltas sobre el eje. Las ondulaciones en y (recordemos que y es igual a x') significan cambios de velocidad en el péndulo.

En los puntos críticos de la forma $(2n\pi, 0)$, el péndulo está en reposo.



En los puntos críticos de la forma $((2n-1)\pi, 0)$, el péndulo también está en reposo.



Formalmente, existe la posibilidad de darle al péndulo la energía exacta para que se acerque infinitamente a esta posición pero sin llegar a alcanzarla (y sin que vuelva a caer hacia atrás). Esto ocurre cuando nos aproximamos a los puntos $((2n-1)\pi, 0)$ justo por la línea $y = -x$.

Ejemplo 28 Estudiar puntos críticos, campo de direcciones y órbitas de

$$\begin{aligned} x' &= (y^2 - 2|y| + 1)y \\ y' &= -x \end{aligned}$$

Puntos críticos:

Para la segunda ecuación, el punto crítico tiene valor $x = 0$. De la primera ecuación obtenemos los siguientes valores:

$$\begin{aligned} y^2 - 2|y| + 1 &= 0 \\ (|y| - 1)^2 &= 0 \\ |y| &= 1 \end{aligned}$$

$$y = \pm 1, y = 0$$

Los puntos críticos son entonces $\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ y $\begin{pmatrix} 0 \\ -1 \end{pmatrix}$

Linealizamos:

Para calcular la derivada de $(y^2 - 2|y| + 1)y$ con respecto a y , hay que tener en cuenta dos casos:

$$\frac{\partial((y^2 - 2|y| + 1)y)}{\partial y} = \begin{cases} (y-1)^2 + 2(y-1)y & \text{si } y > 0 \\ (-y-1)^2 + 2(-y-1)y & \text{si } y < 0 \end{cases}$$

El gradiente de f es,

$$\nabla f(x_c) = \begin{pmatrix} 0 & \begin{cases} y > 0 : 3y^2 - 4y + 1 \\ y < 0 : -y^2 + 1 \end{cases} \\ -1 & 0 \end{pmatrix}$$

Evaluamos el gradiente en cada punto crítico:

P.C.(0,0)

$$\nabla f(0,0) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Como ya sabemos, la órbita es circular centrada en el origen.

P.C.(0,-1)

$$\nabla f(0,-1) = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}$$

En ejemplos anteriores ya habíamos trabajado con una matriz del tipo $A = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$, que es similar a esta pero cambiada de signo, luego la solución será:

$$e^{A(t-t_0)} = \begin{pmatrix} 1 & 0 \\ -(t-t_0) & 1 \end{pmatrix}$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_0 \\ -(t-t_0)x_0 + y_0 \end{pmatrix}$$

Como ya habíamos visto, las órbitas son verticales.

P.C.(0,1)

$$\nabla f(0,1) = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}$$

El resultado es igual al caso anterior.

En la gráfica 57 se puede ver el campo de direcciones para este sistema.

En la figura 58 hemos dibujado la órbita del sistema.

Hay algunos tipos de ecuaciones diferenciales de orden 2 ó superior que se pueden manipular de forma que se obtenga un sistema de ecuaciones diferenciales en el que sus ecuaciones son de orden 1 y al que podemos aplicarle las técnicas vistas hasta ahora. Un ejemplo de este tipo de ecuaciones diferenciales es la ecuación del movimiento armónico simple:

$$x'' + kx' + w^2x = 0$$

Para convertirla en un sistema de ecuaciones diferenciales se hace un cambio de variables $y = x'$ de forma que nos quede este sistema:

$$\begin{aligned} x' &= y \\ y' &= x'' = -kx' - w^2x \end{aligned}$$

De forma análoga, la ecuación $x''' + 2x'' + 3x' + 4x = 0$ se puede transformar en un sistema de tres ecuaciones de orden 1:

$$\begin{aligned} x' &= y \\ y' &= x'' = z \\ z' &= x''' = -2z - 3y - 4x \end{aligned}$$

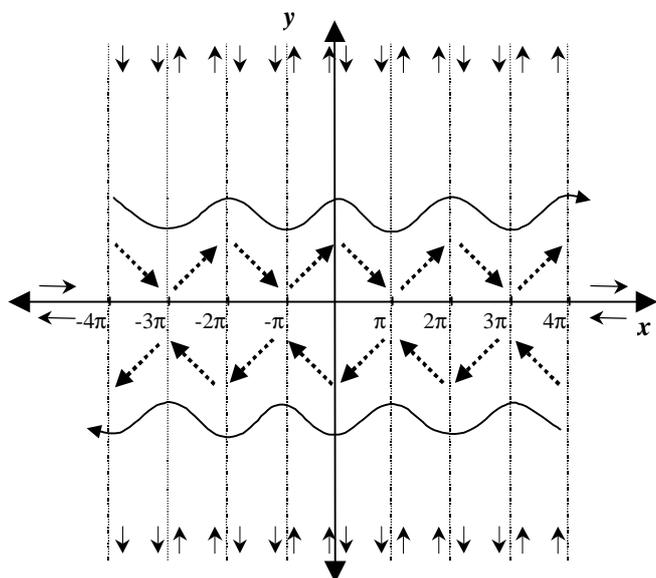


Figura 56: Campo de direcciones para $x'' + \sin(x) = 0$.

Ejemplo 29 Estudiar $x'' + kx' + w^2x = 0$ en función de k y w ($k, w > 0$).

Como vimos más arriba, podemos sustituir esta ecuación por un sistema de la forma,

$$\begin{aligned} x' &= y \\ y' &= x'' = -ky - w^2x \end{aligned}$$

Este sistema tiene un punto crítico en $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$.

Linealizamos:

$$\nabla f(x_c) = \begin{pmatrix} 0 & 1 \\ -w^2 & -k \end{pmatrix}$$

Calculamos sus autovalores:

$$\begin{vmatrix} -\lambda & 1 \\ -w^2 & -\lambda - k \end{vmatrix} = 0$$

$$\lambda^2 + \lambda k + w^2 = 0$$

$$\lambda = \frac{-k \pm \sqrt{(k^2 - 4w^2)}}{2}$$

Los autovalores dependen de los valores de k y w . Estos autovalores pueden ser reales o imaginarios dependiendo del signo del discriminante.

Los casos posibles son:

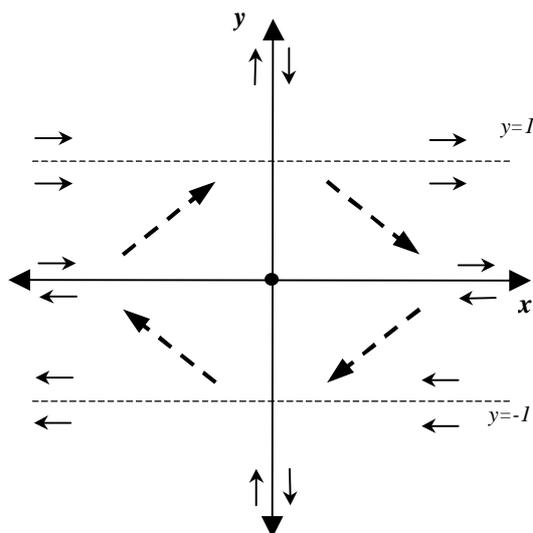


Figura 57: Campo de direcciones.

1. $(k^2 - 4w^2) > 0$. Autovalores reales. Como $k > 0$, los dos autovalores son negativos, con lo que el origen se comporta como un atractor.
2. $(k^2 - 4w^2) = 0$. Autovalor doble negativo. Sigue siendo atractor.
3. $(k^2 - 4w^2) < 0$. Autovalores complejos conjugados con parte real negativa. Es una espiral que se acerca al punto crítico.

Igualando y' a cero, obtenemos la separación entre las distintas regiones para el estudio del campo de direcciones.

$$-ky - w^2x = 0$$

$$y = -w^2 \frac{x}{k}$$

Esta ecuación es una recta de pendiente negativa. Las regiones están determinadas por esta recta y por la recta $x = 0$. La figura 59 representa el campo de direcciones:

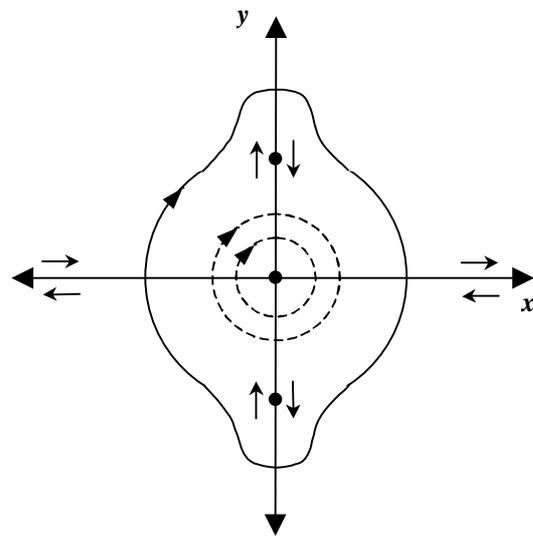


Figura 58: Orbita para el sistema.

Ejemplo 30 Estudiar el comportamiento del sistema

$$x'(t) = x(y^2 - 1)$$

$$y'(t) = y(x - 1)$$

Puntos críticos:

$$0 = x(y^2 - 1)$$

$$0 = y(x - 1)$$

Los puntos críticos son $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ y $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$

Linealizamos:

$$\nabla f = \begin{pmatrix} y^2 - 1 & 2xy \\ y & x - 1 \end{pmatrix}$$

Análisis para cada punto crítico:

P.C.(0,0)

$$\nabla f \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

Calculamos los autovalores,

$$\begin{vmatrix} -1 - \lambda & 0 \\ 0 & -1 - \lambda \end{vmatrix} = 0$$

$$\lambda_{1,2} = -1$$

Como los autovalores son negativos, el sistema tendrá un comportamiento atractor hacia el origen.

P.C.(1,1)

$$\nabla f \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & 2 \\ 1 & 0 \end{pmatrix}$$

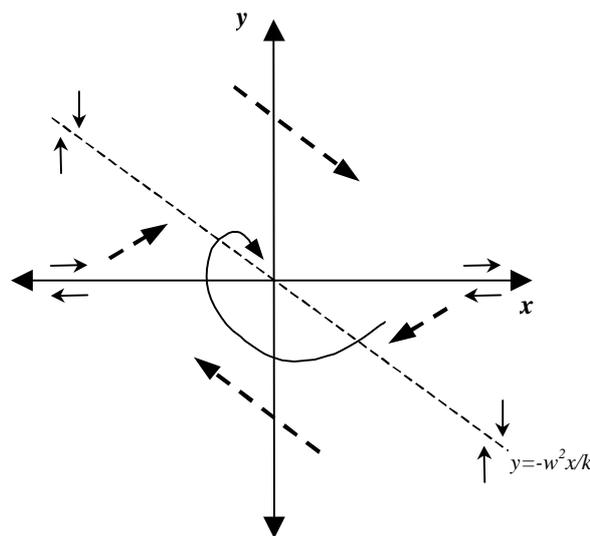


Figura 59: Campo de direcciones.

Calculamos los autovalores,

$$\begin{vmatrix} -\lambda & 2 \\ 1 & -\lambda \end{vmatrix} = 0$$

$$\lambda_{1,2} = \pm\sqrt{2}$$

Los autovalores son de distinto signo con lo que el sistema tiene una órbita en forma de punto de silla.

Autovectores:

Para $\lambda = \sqrt{2}$

$$\begin{pmatrix} -\sqrt{2} & 2 \\ 1 & -\sqrt{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

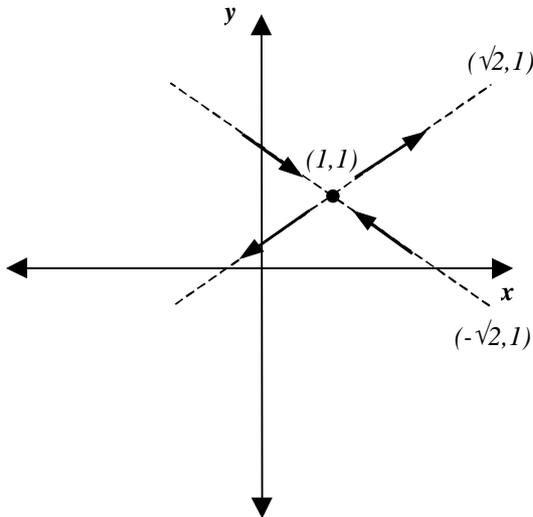
$$\bar{x}_1 = \begin{pmatrix} 2 \\ \sqrt{2} \end{pmatrix}$$

Para $\lambda = -\sqrt{2}$

$$\begin{pmatrix} \sqrt{2} & 2 \\ 1 & \sqrt{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\bar{x}_2 = \begin{pmatrix} \sqrt{2} \\ -1 \end{pmatrix}$$

Orbita en las cercanías del punto crítico:



P.C.(1, -1)

$$\nabla f \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 & -2 \\ -1 & 0 \end{pmatrix}$$

Calculamos los autovalores,

$$\begin{vmatrix} -\lambda & -2 \\ -1 & -\lambda \end{vmatrix} = 0$$

$$\lambda_{1,2} = \pm\sqrt{2}$$

Los autovalores son de distinto signo con lo que el sistema tiene una órbita en forma de punto de silla.

Autovectores:

Para $\lambda = \sqrt{2}$

$$\begin{pmatrix} -\sqrt{2} & -2 \\ -1 & -\sqrt{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

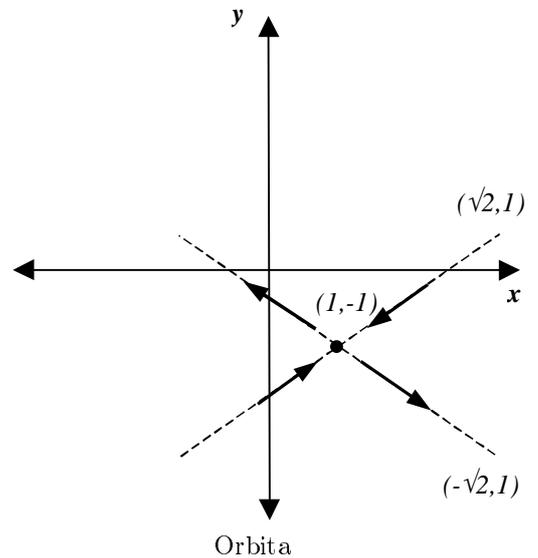
$$\bar{x}_1 = \begin{pmatrix} -\sqrt{2} \\ 1 \end{pmatrix}$$

Para $\lambda = -\sqrt{2}$

$$\begin{pmatrix} \sqrt{2} & -2 \\ -1 & \sqrt{2} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\bar{x}_2 = \begin{pmatrix} \sqrt{2} \\ 1 \end{pmatrix}$$

Orbita en las cercanías del punto crítico:



Si unimos las distintas órbitas obtenidas para cada punto crítico, tenemos una gráfica como la que se muestra en la figura 60.

ANÁLISIS NUMÉRICO DE LOS SISTEMAS

En esta sección vamos a explicar algunos métodos para resolver sistemas de ecuaciones diferenciales de forma numérica.

Partimos de la definición del sistema $\bar{x}' = f(\bar{x})$ y de una posición inicial $\bar{x}(t_0) = \bar{x}_0$.

Un método numérico para resolver un sistema con valores iniciales, es un procedimiento para obtener valores aproximados $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n$, de la solución en los instantes $t_1 < t_2 < \dots < t_n$. Estamos sustituyendo un problema que

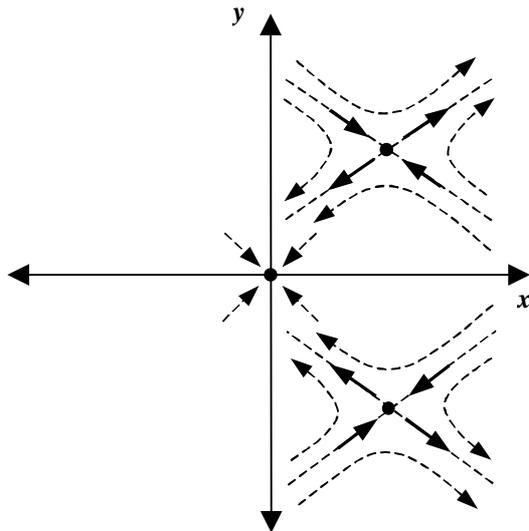


Figura 60: Orbita global.

comprende variables continuas, por otro en el que intervienen variables discretas. Discretizamos la variable temporal $t_n = t_0 + h \cdot n$

$$\begin{aligned} t_0 &= t_0 \\ t_1 &= t_0 + h \\ t_2 &= t_0 + 2h \\ &\vdots \end{aligned}$$

La solución en el instante n será de la forma:

$$\bar{x}_n \simeq \bar{x}(t_n) \simeq \bar{x}(t_0 + n \cdot h)$$

De forma gráfica, el método numérico genera distintas aproximaciones a la solución real, cometiendo un error entre ambas.

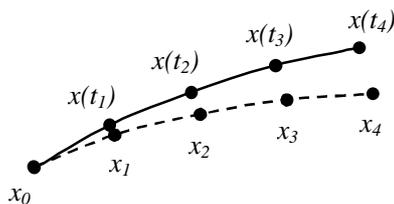


Figura 61: Error entre la solución real y la aproximación.

El error que se produce es, por un lado, la diferencia en el paso actual entre la solución real y la calculada, y por otro, el error acumulado al calcular el paso siguiente en función de un dato ya erróneo.

Existen dos tipos de métodos para el cálculo de las aproximaciones:

Los métodos que sólo requieren un conocimiento de \tilde{x}_{n-1} para determinar \tilde{x}_n se denominan *métodos de un paso* o *de arranque* y los métodos en los que se utilizan varias aproximaciones anteriores ($\tilde{x}_{n-1}, \tilde{x}_{n-2}, \dots$) para determinar \tilde{x}_n se denominan *métodos multipaso* o *prolongados*. En los métodos multipaso se tiene que utilizar un método de un paso para determinar las primeras aproximaciones \tilde{x}_i

De forma general, se cometen errores al calcular los valores $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n$. Este error tiene dos fuentes: primero, la fórmula que se usa en el método numérico sólo es aproximada, lo que conduce a un *error de fórmula*, *error de truncamiento* o *error por no continuidad*; segundo, es posible llevar sólo un número limitado de dígitos en cualquier cálculo, lo que da lugar a un *error de redondeo*.

Para cada uno de los métodos que se explican a continuación, se hace un estudio del error que se comete al aplicarlos.

MÉTODOS DE UN PASO

\tilde{x}_n se calcula en función de \tilde{x}_{n-1}

$$\tilde{x}_n = \tilde{x}(t_n)$$

$$\tilde{x}_{n-1} = \tilde{x}(t_{n-1})$$

Para aproximar este valor, desarrollamos por Taylor $\tilde{x}(t_n)$ a partir de $\tilde{x}(t_{n-1})$:

$$\begin{aligned} \tilde{x}(t_n) &= \tilde{x}(t_{n-1}) + \tilde{x}'(t_{n-1}) \underbrace{(t_n - t_{n-1})}_h + \dots + \\ &+ \tilde{x}^{(n)}(t_{n-1}) \frac{(t_n - t_{n-1})^n}{n!} \end{aligned}$$

METODO DE EULER

Uno de los métodos de un paso más sencillos para resolver numéricamente el problema con valores iniciales

$$\tilde{x}' = f(x), \quad \tilde{x}(t_0) = \tilde{x}_0$$

es el método de Euler. En este método se consideran los primeros términos del desarrollo del Taylor:

$$\tilde{x}(t_n) = \tilde{x}(t_{n-1}) + \tilde{x}'(t_{n-1}) \cdot h$$

$$\boxed{\tilde{x}_n = \tilde{x}_{n-1} + f(\tilde{x}(t_{n-1})) \cdot h}$$

En este método nos vamos moviendo por la recta tangente para calcular la siguiente aproximación. Por esta razón, este método también se llama *método de Euler* o *de la recta tangente*. En la gráfica 62 se puede ver la representación geométrica del método de Euler.

Tomamos el siguiente punto \tilde{x}_n sumando h en la dirección de la tangente a la órbita.

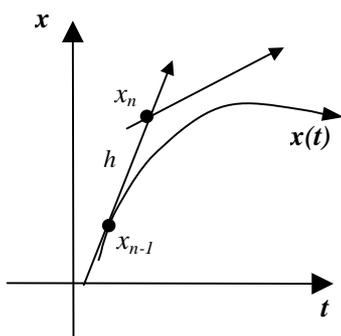


Figura 62: Método de Euler.



Podemos suponer que el error global es equivalente al error local en n pasos:

$$ErrorGlobal \leq n \cdot ErrorLocal = n \cdot O(h^2)$$

en donde n depende del intervalo h

$$n \simeq \frac{t_n - t_0}{h}$$

$$ErrorGlobal = \frac{t_n - t_0}{h} \cdot O(h^2) = O(h)$$

Por lo tanto el orden del método en n pasos es de 1. Esto quiere decir que el método converge, ya que si $h \rightarrow 0$ entonces $O(h) \rightarrow 0$.

El error decrece según decrece h . Cuanto más pequeño sea el paso, mejor será la aproximación a la solución real.

METODO DE EULER MEJORADO

Vamos a ver una mejora para el método anterior, consiguiendo que el error local de fórmula sea proporcional a h^3 . Para esto, intentamos aproximar la solución avanzando un término más en el desarrollo de Taylor:

$$\tilde{x}(t_n) = \tilde{x}(t_{n-1}) + \tilde{x}'(t_{n-1}) \cdot h + \frac{\tilde{x}''(t_{n-1}) \cdot h^2}{2} + O(h^3) \quad (11)$$

En esta ecuación tenemos derivadas primeras y segundas del vector \tilde{x} . Vamos a ver cómo podemos calcular estas derivadas por separado.

La primera derivada viene dada por la función del sistema de la manera:

$$\tilde{x}' = f(\tilde{x}(t))$$

Vamos a calcular ahora la segunda derivada. Si derivamos el vector \tilde{x}' obtenemos una expresión de la forma:

$$\tilde{x}''(t_{n-1}) = \frac{\partial^2 \tilde{x}(t_{n-1})}{\partial t^2} = \frac{\partial}{\partial t} f(\tilde{x}(t_{n-1}))$$

La derivada de f con respecto a t viene dada por:

$$\tilde{x}''(t_{n-1}) = f'(\tilde{x}(t_{n-1})) \cdot \tilde{x}'(t_{n-1})$$

Sustituyendo en esta ecuación \tilde{x}' por f obtenemos:

$$\tilde{x}''(t_{n-1}) = \nabla f(\tilde{x}(t_{n-1})) \cdot f(\tilde{x}(t_{n-1})) \quad (12)$$

De la parte derecha de esta ecuación podemos calcular $f(\tilde{x}(t_{n-1}))$ sustituyendo el valor de $\tilde{x}(t_{n-1})$ en el

ERROR La diferencia entre la solución exacta \bar{x} y la solución aproximada del problema

$$E_n = \bar{x}_n - \tilde{x}_n$$

se conoce como error de fórmula o error acumulado de fórmula. Surge por dos causas: 1) en cada paso usamos un fórmula aproximada para determinar \tilde{x}_n . 2) los datos de entrada en cada paso no concuerdan con la solución exacta, ya que, en general, \bar{x}_n no es igual a \tilde{x}_n . Si suponemos que los datos de entrada son correctos, el único error al ir de un paso a otro es el que se debe al uso de una fórmula aproximada. Este error se conoce como error local de fórmula.

Error local Es el error que se comete al calcular \tilde{x}_n a partir de \tilde{x}_{n-1} . Este error es equivalente a los términos del desarrollo de Taylor desechados. Al quedarnos con los dos primeros términos del desarrollo, el error vendrá dado por:

$$ErrorLocal = \frac{x_n''(t_{n-1}) h^2}{2!} + \dots + \frac{x_n^m(t_{n-1}) h^m}{m!} = O(h^2)$$

El orden de este método es 2, ya que se desechan los términos de orden 2 y superiores.

A continuación se explican, de forma escueta, algunas propiedades del operador $O(h^n)$.

Propiedades de $O(h^n)$

$$O(h^2) + O(h^2) = O(h^2)$$

$$O(h^2) + O(h^3) = O(h^2)$$

$$k \cdot O(h^2) = O(h^2)$$

Error Global Es el error que se comete al pasar del instante t_0 al instante t_n . El análisis para estimar el error global es más complejo que para el error local; sin embargo, conociendo el error local podemos hacer una estimación intuitiva del error global.

sistema dado por $f(\tilde{x}(t))$. Tan solo nos queda determinar cómo se puede calcular el valor del gradiente de la función ($\nabla f(\tilde{x}(t_{n-1}))$). Para esto, desarrollamos por Taylor f de la siguiente manera:

$$f(\bar{y} + h\bar{z}) = f(y) + \nabla f(y) \cdot hz + O(h^2) \quad (13)$$

Lo que nos interesa es despejar el factor que contiene al gradiente de la función, de forma que podamos calcular la segunda derivada de \tilde{x} dada por la ecuación (12). Si tomamos para la ecuación (13) los siguientes valores:

$$\begin{aligned} \bar{y} &= \tilde{x}(t_{n-1}) \\ \bar{z} &= f(\tilde{x}(t_{n-1})) \end{aligned} \quad ,$$

tenemos

$$\begin{aligned} f(\tilde{x}(t_{n-1}) + hf(\tilde{x}(t_{n-1}))) &= f(\tilde{x}(t_{n-1})) + \\ &+ h\nabla f(\tilde{x}(t_{n-1})) \cdot f(\tilde{x}(t_{n-1})) + O(h^2) \end{aligned} \quad ,$$

Si nos damos cuenta, la parte que incluye al gradiente de la función es igual que el de la ecuación (12) multiplicado por h . Despejando este factor:

$$\begin{aligned} h\nabla f(\tilde{x}(t_{n-1})) \cdot f(\tilde{x}(t_{n-1})) &= \\ &= f(\tilde{x}(t_{n-1}) + hf(\tilde{x}(t_{n-1}))) - f(\tilde{x}(t_{n-1})) + \\ &+ O(h^2) \end{aligned}$$

sustituimos esta ecuación en $\tilde{x}''(t_{n-1}) = \nabla f(\tilde{x}(t_{n-1})) \cdot f(\tilde{x}(t_{n-1}))$ y obtenemos,

$$\begin{aligned} \tilde{x}''(t_{n-1})h &= f(\tilde{x}(t_{n-1}) + hf(\tilde{x}(t_{n-1}))) - \\ &- f(\tilde{x}(t_{n-1})) + O(h^2) \end{aligned}$$

De esta forma podemos calcular la segunda derivada de \tilde{x} . Sustituimos esta expresión en el desarrollo de Taylor de $\tilde{x}(t_n)$ dada por la ecuación (11):

$$\begin{aligned} \tilde{x}(t_n) &= \tilde{x}(t_{n-1}) + h \cdot f(\tilde{x}(t_{n-1})) + \\ &+ \frac{h^2}{2} \cdot (f(\tilde{x}(t_{n-1}) + hf(\tilde{x}(t_{n-1}))) - f(\tilde{x}(t_{n-1}))) + O(h^2) \\ &+ O(h^3) \end{aligned}$$

Agrupamos términos y simplificamos:

$$\begin{aligned} \tilde{x}_n &= \tilde{x}_{n-1} + \frac{h}{2} [f(\tilde{x}_{n-1}) + f((\tilde{x}_{n-1}) + hf(\tilde{x}_{n-1}))] + \\ &+ O(h^3) \end{aligned}$$

Finalmente, la fórmula del método de Euler mejorado queda como sigue:

$$\tilde{x}_n = \tilde{x}_{n-1} + \frac{h}{2} [f(\tilde{x}_{n-1}) + f((\tilde{x}_{n-1}) + hf(\tilde{x}_{n-1}))]$$

En la figura 63 se puede ver una interpretación gráfica del método de Euler mejorado. En cada paso, calculamos la siguiente aproximación a partir de la aproximación anterior y un promedio de la función f en instantes consecutivos. Vamos a analizar un poco más este promedio:

$f(\tilde{x}_{n-1})$ representa la pendiente de la recta tangente en ese punto. Si solo estuviera este término, tendríamos exactamente el método de Euler.

$f((\tilde{x}_{n-1}) + hf(\tilde{x}_{n-1}))$ nos da la pendiente en un instante posterior. Con $\tilde{x}_{n-1} + hf(\tilde{x}_{n-1})$ nos estamos desplazando por la recta tangente en el punto \tilde{x}_{n-1} una distancia de h .

La flecha de puntos discontinuos superior es el resultado del promedio dado por estas dos rectas tangentes. Por último, nos desplazamos desde el punto \tilde{x}_{n-1} por esta recta (en la gráfica la flecha de puntos discontinuos inferior) para obtener la aproximación actual.

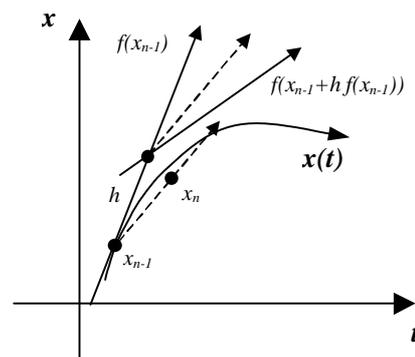


Figura 63: Método de Euler mejorado.

Ejemplo 31 Sea el siguiente sistema de ecuaciones diferenciales

$$\begin{cases} x' = y \\ y' = -x \end{cases}$$

vamos a calcular la primera aproximación a partir de la siguiente condición inicial

$$x(t_0) = x_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

y de un paso de $h = 0.1$ por el método de Euler mejorado.

La función del sistema viene dada por:

$$f(\bar{x}) = \begin{pmatrix} y \\ -x \end{pmatrix}$$

Aplicando la fórmula numérica del método de Euler mejorado obtenemos:

$$\tilde{x}_1 = \tilde{x}_0 + \frac{h}{2} \cdot [f(\tilde{x}_0) + f((\tilde{x}_0) + hf(\tilde{x}_0))] = \begin{pmatrix} 1 \\ 1 \end{pmatrix} +$$

$$\begin{aligned}
& + \frac{0.1}{2} \cdot \left[\begin{pmatrix} 1 \\ -1 \end{pmatrix} + f \left(\begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.1 \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right) \right] = \\
& = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{2} \cdot \left[\begin{pmatrix} 1 \\ -1 \end{pmatrix} + f \left(\begin{pmatrix} 1.1 \\ 0.9 \end{pmatrix} \right) \right] = \\
& = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{2} \left[\begin{pmatrix} 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 0.9 \\ -1.1 \end{pmatrix} \right] = \\
& = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{0.1}{2} \cdot \left[\begin{pmatrix} 1.9 \\ -2.1 \end{pmatrix} \right] = \begin{pmatrix} 1.105 \\ 0.905 \end{pmatrix}
\end{aligned}$$

La solución final es entonces $\tilde{x}_1 = \begin{pmatrix} 1.105 \\ 0.905 \end{pmatrix}$.

Si queremos seguir mejorando el método, debemos ir cogiendo más términos en el desarrollo de Taylor, sin embargo, esto no es tan sencillo, ya que calcular derivadas de tercer orden o superior significa manejar derivadas que dificultan bastante los cálculos. En el siguiente apartado estudiaremos el método de Runge-Kutta que también utiliza un promedio entre distintos puntos y para el que se obtienen errores locales de orden $\mathcal{O}(h^5)$.

METODO DE RUNGE-KUTTA

Las fórmulas con series de Taylor de orden superior son difíciles de manejar debido a que deben calcularse derivadas parciales de f de orden superior.

Sin embargo es posible desarrollar fórmulas equivalentes a las fórmulas con series de Taylor de tercero, cuarto, quinto o aún mayor orden en las que no aparecen derivadas parciales de f .

La fórmula clásica de Runge-Kutta es equivalente a una fórmula de cinco términos de la serie de Taylor.

$$\bar{x}_n = \bar{x}_{n-1} + h\bar{x}'_{n-1} + \frac{h^2}{2!}\bar{x}''_{n-1} + \frac{h^3}{3!}\bar{x}'''_{n-1} + \frac{h^4}{4!}\bar{x}^{iv}_{n-1}$$

La fórmula de Runge-Kutta comprende un promedio pesado de los valores de $f(x, y)$, tomado en puntos diferentes del intervalo $t_{n-1} \leq t \leq t_n$.

$$\bar{x}_{n+1} = \bar{x}_n + \frac{h}{6} (k_{n1} + 2k_{n2} + 2k_{n3} + k_{n4}), \quad (14)$$

donde

$$\begin{aligned}
k_{n1} &= f(t_n, \bar{x}_n) \\
k_{n2} &= f\left(t_n + \frac{h}{2}, \bar{x}_n + \frac{h}{2}k_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{h}{2}, \bar{x}_n + \frac{h}{2}k_{n2}\right) \\
k_{n4} &= f(t_n + h, \bar{x}_n + hk_{n3})
\end{aligned} \quad (15)$$

La suma $\frac{1}{6}(k_{n1} + 2k_{n2} + 2k_{n3} + k_{n4})$ puede interpretarse como una pendiente promedio. k_{n1} representa la pendiente en el extremo izquierdo del intervalo; k_{n2} es la pendiente en el punto medio usando la fórmula de Euler para ir de t_n a $t_n + \frac{h}{2}$, k_{n3} es una segunda aproximación de la pendiente en el punto medio y, finalmente, k_{n4} es la pendiente en $t_n + h$ usando la fórmula de Euler y la pendiente k_{n3} para ir de t_n a $t_n + h$.

METODOS MULTIPASO

En estos métodos, se calcula \bar{x}_n a partir de varias aproximaciones anteriores, $\bar{x}_{n-1}, \bar{x}_{n-2}, \dots, \bar{x}_{n-k}$. Se necesitan k valores iniciales que se pueden calcular a partir de un método de un paso de los explicados en la sección anterior. En estos métodos se puede proceder de dos maneras distintas:

1. \bar{x}_n se calcula en función de $\bar{x}_{n-1}, \bar{x}_{n-2}, \dots, \bar{x}_{n-k}$. Se tienen en cuenta solo las k aproximaciones anteriores (Predictor)
2. \bar{x}_n se calcula en función de $\bar{x}_n, \bar{x}_{n-1}, \dots, \bar{x}_{n-k+1}$ (Corrector)

Integrando $\bar{x}'(t)$ desde t_{n-1} hasta t_n obtenemos:

$$\begin{aligned}
\int_{t_c}^{t_n} \bar{x}'(t) dt &= \int_{t_c}^{t_n} f(\bar{x}(t)) dt \\
\bar{x}(t_n) - \bar{x}(t_c) &= \int_{t_c}^{t_n} f(\bar{x}(t)) dt \quad (16)
\end{aligned}$$

Aproximamos $f(\bar{x}(t))$ por un polinomio de grado $k-1$ que interpola en los puntos t_n, \dots, t_{n-k} .

Polinomios interpoladores de Lagrange

Para calcular el polinomio de interpolación vamos a utilizar los polinomios interpoladores de Lagrange. Los polinomios base de Lagrange tienen la siguiente forma:

$$P_m(t) = \frac{\prod_{l \neq m} (t - t_l)}{\prod_{l \neq m} (t_m - t_l)}, \quad m \in \{n-1, n-2, \dots, n-k\} \quad (17)$$

Estos polinomios valen 1 en uno de los puntos t_i y cero en el resto. De forma gráfica, estos polinomios se representan como se puede ver en la figura 64. En el punto en donde se toma el valor de la aproximación, el polinomio vale 1 y en el resto de los puntos, el polinomio se anula.

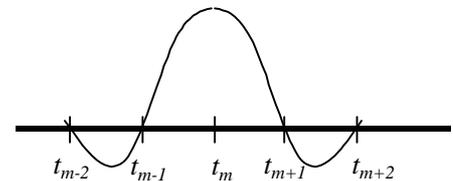


Figura 64: Polinomios base de Lagrange

A partir de los polinomios base de Lagrange, el polinomio interpolador se genera de modo que en cada t_m este valga $f(\bar{x}(t))$.

$$P(t) = \sum_{m=n-k}^n f(\bar{x}_m) P_m(t) \quad (18)$$

Si sustituimos este polinomio en la ecuación (16) obtenemos la siguiente ecuación:

$$\boxed{\bar{x}(t_n) - \bar{x}(t_c) \simeq \sum_{m=n-k}^n f(\bar{x}_m) \int_{t_c}^{t_n} P_m(t)} \quad (19)$$

En los siguientes ejemplos, se puede ver cómo se calculan las fórmulas numéricas para distintos métodos multipaso dependiendo de las aproximaciones anteriores que se deseen tomar.

Ejemplo 32 *Se desea buscar un método multipaso de la forma*

$$\bar{x}_n = \bar{x}_{n-1} + G(f(\bar{x}_{n-1}), f(\bar{x}_{n-2}))$$

en donde se tengan en cuenta las aproximaciones en los instantes t_{n-1} y t_{n-2} .

Si analizamos un poco más esta ecuación, vemos que queremos aproximar la solución actual a partir de una solución anterior y un promediado en los instantes t_{n-1} y t_{n-2} . Utilizando la fórmula dada en (19) obtenemos la siguiente ecuación:

$$\int_{t_{n-1}}^{t_n} \bar{x}'(t) dt = \int_{t_{n-1}}^{t_n} [f(\bar{x}_{n-1})P_{n-1}(t) + f(\bar{x}_{n-2})P_{n-2}(t)] dt$$

Ahora tenemos que calcular los polinomios base de Lagrange $P_{n-1}(t)$ y $P_{n-2}(t)$:

$$P_{n-1}(t) = \frac{(t-t_{n-2})}{(t_{n-1}-t_{n-2})} = \frac{(t-t_{n-2})}{h}$$

$$P_{n-2}(t) = \frac{(t-t_{n-1})}{(t_{n-2}-t_{n-1})} = \frac{-(t-t_{n-1})}{h}$$

Sustituyendo estos valores en la ecuación y resolviendo las integrales,

$$\begin{aligned} \bar{x}_n - \bar{x}_{n-1} &= \\ &= f(\bar{x}_{n-1}) \int_{t_{n-1}}^{t_n} \frac{(t-t_{n-2})}{h} + \\ &+ f(\bar{x}_{n-2}) \int_{t_{n-1}}^{t_n} \frac{-(t-t_{n-1})}{h} dt = \\ &= f(\bar{x}_{n-1}) \left[\frac{(t-t_{n-2})^2}{2h} \right]_{t_{n-1}}^{t_n} + \\ &+ f(\bar{x}_{n-2}) \left[\frac{(t-t_{n-1})^2}{-2h} \right]_{t_{n-1}}^{t_n} = \\ &= f(\bar{x}_{n-1}) \left[\frac{(2h)^2 - h^2}{2h} \right] + f(\bar{x}_{n-2}) \left[\frac{h^2 - 0}{-2h} \right] \end{aligned}$$

La fórmula numérica de nuestro método multipaso es de la forma:

$$\boxed{\bar{x}_n = \bar{x}_{n-1} + \frac{3}{2}hf(\bar{x}_{n-1}) - \frac{h}{2}f(\bar{x}_{n-2})}$$

Para comprobar que la segunda parte de la ecuación es un promedio, la suma de los coeficientes de f debe ser igual a 1. En este caso la suma de los coeficientes es $\frac{3}{2} - \frac{1}{2} = 1$, con lo que efectivamente es un promedio. En el caso en que el cálculo de la integral sea entre \bar{x}_n y \bar{x}_{n-k} , entonces la suma de los coeficientes debe ser igual a k .

Ejemplo 33 *En este ejemplo vamos a buscar una expresión del tipo*

$$\bar{x}_n = \bar{x}_{n-1} + G(f(\bar{x}_{n-1}))$$

En este caso tenemos que la ecuación a resolver debe ser de la forma

$$\int_{t_{n-1}}^{t_n} \bar{x}'(t) dt = \int_{t_{n-1}}^{t_n} f(\bar{x}_{n-1})P_{n-1}(t) dt$$

en donde solo se tiene un único polinomio base de Lagrange. Este polinomio base es igual a 1, y, resolviendo la integral, obtenemos:

$$\begin{aligned} \int_{t_{n-1}}^{t_n} \bar{x}'(t) dt &= \int_{t_{n-1}}^{t_n} f(\bar{x}_{n-1})P_{n-1}(t) dt = \\ &= f(\bar{x}_{n-1}) \int_{t_{n-1}}^{t_n} 1 dt = f(\bar{x}_{n-1}) \cdot h \end{aligned}$$

De donde se obtiene que la fórmula numérica es:

$$\boxed{\bar{x}_n = \bar{x}_{n-1} + hf(\bar{x}_{n-1})}$$

Esta fórmula coincide con la fórmula del método de Euler.

Ejemplo 34 *Buscamos ahora una expresión en la que se tengan en cuenta tres términos en el promedio:*

$$\bar{x}_n = \bar{x}_{n-2} + G(f(\bar{x}_n), f(\bar{x}_{n-1}), f(\bar{x}_{n-2}))$$

Utilizando los polinomios base de Lagrange para interpolar la función en estos puntos, la integral nos queda de la forma,

$$\int_{t_{n-2}}^{t_n} \bar{x}'(t) dt = \int_{t_{n-2}}^{t_n} [f(\bar{x}_n)P_n(t) + f(\bar{x}_{n-1})P_{n-1}(t) + f(\bar{x}_{n-2})P_{n-2}(t)] dt$$

Los polinomios base de Lagrange vienen dados por:

$$P_n(t) = \frac{(t-t_{n-1})(t-t_{n-2})}{(t_n-t_{n-1})(t_n-t_{n-2})} = \frac{(t-t_{n-1})(t-t_{n-2})}{2h^2}$$

$$P_{n-1}(t) = \frac{(t-t_n)(t-t_{n-2})}{(t_{n-1}-t_n)(t_{n-1}-t_{n-2})} = \frac{(t-t_n)(t-t_{n-2})}{-h^2}$$

$$P_{n-2}(t) = \frac{(t-t_n)(t-t_{n-1})}{(t_{n-2}-t_n)(t_{n-2}-t_{n-1})} = \frac{(t-t_n)(t-t_{n-1})}{2h^2}$$

Resolvemos la integral

$$\begin{aligned} \bar{x}_n - \bar{x}_{n-2} &= f(\bar{x}_n) \int_{t_{n-2}}^{t_n} \frac{(t-t_{n-1})(t-t_{n-2})}{2h^2} dt + \\ &+ f(\bar{x}_{n-1}) \int_{t_{n-2}}^{t_n} \frac{(t-t_n)(t-t_{n-2})}{-h^2} dt + \\ &+ f(\bar{x}_{n-2}) \int_{t_{n-2}}^{t_n} \frac{(t-t_n)(t-t_{n-1})}{2h^2} dt = \end{aligned}$$

Realizamos un cambio de variable $\alpha = t - t_{n-2}$ de forma que $t - t_{n-1} = \alpha - h$ y $t - t_n = \alpha - 2h$.

$$\begin{aligned}
\bar{x}_n - \bar{x}_{n-2} &= \frac{f(\bar{x}_n)}{2h^2} \int_0^2 \alpha (\alpha - h) d\alpha + \\
&+ \frac{f(\bar{x}_{n-1})}{-h^2} \int_0^2 \alpha (\alpha - 2h) d\alpha + \\
&+ \frac{f(\bar{x}_{n-2})}{2h^2} \int_0^2 (\alpha - 2h) (\alpha - h) d\alpha = \\
&= \frac{f(\bar{x}_n)}{2h^2} \frac{2}{3} h^3 + \frac{f(\bar{x}_{n-1})}{-h^2} \frac{-4}{3} h^3 + \frac{f(\bar{x}_{n-2})}{2h^2} \frac{2}{3} h^3 = \\
&= \frac{1}{3} h f(\bar{x}_n) + \frac{4}{3} h f(\bar{x}_{n-1}) + \frac{1}{3} h f(\bar{x}_{n-2})
\end{aligned}$$

Finalmente tenemos que la fórmula numérica es de la forma

$$\bar{x}_n = \bar{x}_{n-2} + h \left[\frac{1}{3} f(\bar{x}_n) + \frac{4}{3} f(\bar{x}_{n-1}) + \frac{1}{3} f(\bar{x}_{n-2}) \right]$$

Sumamos los pesos para comprobar que efectivamente es igual a la diferencia entre los instantes t_n y t_{n-2} : $\frac{1}{3} + \frac{4}{3} + \frac{1}{3} = 2$.

En los ejemplos que hemos visto hasta ahora, hemos obtenido soluciones del tipo:

$$\bar{x}_n = \bar{x}_{n-1} + G(f(\bar{x}_{n-1}), f(\bar{x}_{n-2}), \dots)$$

Pero si queremos obtener soluciones de la forma

$$\bar{x}_n = \frac{1}{2} \bar{x}_{n-1} + \frac{1}{2} \bar{x}_{n-2} + G(f(\bar{x}_{n-1}), f(\bar{x}_{n-2}), \dots)$$

se pueden buscar fórmulas por separado,

$$\begin{aligned}
\bar{x}_n &= \bar{x}_{n-1} + G_1(f(\bar{x}_{n-1}), f(\bar{x}_{n-2}), \dots) \\
\bar{x}_n &= \bar{x}_{n-2} + G_2(f(\bar{x}_{n-1}), f(\bar{x}_{n-2}), \dots)
\end{aligned}$$

y luego promediarlas:

$$\begin{aligned}
\frac{1}{2} \bar{x}_n + \frac{1}{2} \bar{x}_n &= \\
\frac{1}{2} \bar{x}_{n-1} + \frac{1}{2} \bar{x}_{n-2} + \underbrace{\frac{1}{2} G_1(f(\bar{x}_{n-1}), \dots) + \frac{1}{2} G_2(f(\bar{x}_{n-1}), \dots)}_G
\end{aligned}$$

ESTUDIO DEL ERROR

Al interpolar, estamos aproximando la integral de f como,

$$\int_{t_c}^{t_n} f(\bar{x}(t)) dt \simeq \int_{t_c}^{t_n} \Gamma(t) dt$$

El error de interpolación $\int_{t_c}^{t_n} Error(t) dt$ con

$$Error(t) = \frac{\prod (t - t_k)}{k!} \cdot f^{(k)}(\xi)$$

Hay que tener en cuenta que el punto intermedio ξ depende de t , por tanto al hacer la integral del error no podemos sacarlo fuera de la integra. Sin embargo existe un caso en que sí lo podemos hacer, utilizando el resultado del siguiente teorema:

Teorema 35 $\int_a^b g(t)f(t)dt = g(\xi) \int_a^b f(t)dt$ si f no cambia de signo en $[a, b]$ y g es continua en $[a, b]$.

Demostración: Al ser g una función continua en el intervalo, podemos encontrar dos constantes tal que

$$m = \min_{[a, b]} g(t), \quad M = \max_{[a, b]} g(t)$$

A partir de estas constantes, construimos la siguiente desigualdad:

$$\frac{mf(t)}{\int_a^b f(t)dt} \leq \frac{g(t)f(t)}{\int_a^b f(t)dt} \leq \frac{Mf(t)}{\int_a^b f(t)dt}$$

Teorema 36 Demostración 3 Integrando en el intervalo se tiene,

$$m \leq \frac{\int_a^b g(t)f(t)dt}{\int_a^b f(t)dt} \leq M$$

Como $g(t)$ es continua, podemos encontrar un ξ tal que

$$g(\xi) = \frac{\int_a^b g(t)f(t)dt}{\int_a^b f(t)dt}$$

Despejando de la última igualdad, llegamos al resultado

$$\int_a^b g(t)f(t)dt = g(\xi) \int_a^b f(t)dt, \text{ c. q. d.}$$

Apoyándonos en este teorema, podemos expresar la integral del error como:

$$\int_{t_c}^{t_n} Error(t)dt = \frac{f^{(k)}(\xi)}{k!} \int_{t_c}^{t_n} \prod (t - t_k) dt$$

siempre y cuando $f(t)$ sea continua en el intervalo $[t_c, t_n]$ y $\prod (t - t_k)$ no cambie de signo en el mismo.

Ejemplo 35 Para la siguiente fórmula numérica,

$$\bar{x}_n = \bar{x}_{n-1} + G(f(\bar{x}_{n-1}), f(\bar{x}_{n-2})) \quad k = 2$$

estimar el error que se produce.

La expresión del error tiene la siguiente forma:

$$Error = \frac{f''(\xi)}{2!} \int_{t_{n-1}}^{t_n} (t - t_{n-1})(t - t_{n-2}) dt$$

haciendo un cambio de variable, $\alpha = t - t_{n-1}$

$$\begin{aligned}
Error &= \frac{f^{(2)}(\xi)}{2!} \int_0^h \alpha (\alpha + h) d\alpha = \\
&= \frac{f^{(2)}(\xi)}{2!} \left[\frac{\alpha^3}{3} + \frac{\alpha^2}{2} h \right]_0^h = \frac{f^{(2)}(\xi)}{2!} \frac{5}{6} h^3
\end{aligned}$$

El error depende de una constante por h^3 , por lo tanto, es de orden $\mathcal{O}(h^3)$

Ejemplo 36 *Volvemos a calcular el error para la siguiente fórmula,*

$$\bar{x}_n = \bar{x}_{n-1} + hf(\bar{x}_{n-1}) \quad k = 1$$

La expresión del error:

$$Error = f'(\xi) \int_{t_{n-1}}^{t_n} (t - t_{n-1}) dt$$

Haciendo $\alpha = t - t_{n-1}$, obtenemos,

$$\begin{aligned} Error &= f'(\xi) \int_0^h \alpha d\alpha = f'(\xi) \left[\frac{\alpha^2}{2} \right]_0^h = \\ &= f'(\xi) \frac{h^2}{2} \end{aligned}$$

Orden del error $\mathcal{O}(h^2)$

Ejemplo 37 *Vamos a calcular el error para la siguiente fórmula numérica*

$$\bar{x}_n = \bar{x}_{n-2} + G(f(\bar{x}_n), f(\bar{x}_{n-1}), f(\bar{x}_{n-2})), k = 3$$

La expresión del error viene dada por

$$Error = \frac{f'''(\xi)}{3!} \int_{t_{n-2}}^{t_n} (t - t_n)(t - t_{n-1})(t - t_{n-2}) dt$$

Haciendo $\alpha = t - t_{n-2}$, obtenemos,

$$\begin{aligned} Error &= \frac{f'''(\xi)}{3!} \int_0^{2h} \alpha(\alpha - h)(\alpha - 2h) d\alpha = \\ &= \frac{f'''(\xi)}{3!} \int_0^{2h} (\alpha^3 - 3\alpha^2h + 2\alpha h^2) d\alpha = 0 \end{aligned}$$

Esta integral es igual a cero ya que el integrando cambia de signo y se compensan la parte positiva con la negativa. El teorema anterior no se cumple con lo que el cálculo de este error es incorrecto. Para calcularlo, podemos separar la integral en dos integrales distintas, con puntos intermedios distintos:

$$\begin{aligned} Error &= \frac{f'''(\xi)}{3!} \int_{t_{n-1}}^{t_n} (t - t_n)(t - t_{n-1})(t - t_{n-2}) dt + \\ &+ \frac{f'''(\mu)}{3!} \int_{t_{n-2}}^{t_{n-1}} (t - t_n)(t - t_{n-1})(t - t_{n-2}) dt \\ &= \frac{f'''(\xi)}{3!} \int_0^h (\alpha^3 - 3\alpha^2h + 2\alpha h^2) d\alpha + \\ &+ \frac{f'''(\mu)}{3!} \int_h^{2h} (\alpha^3 - 3\alpha^2h + 2\alpha h^2) d\alpha = \\ &= \frac{1}{24} f'''(\xi) h^4 - \frac{1}{24} f'''(\mu) h^4 \end{aligned}$$

como puede observarse, el orden del error es $\mathcal{O}(h^4)$.

Ejemplo 38 *Obtener una fórmula que tenga la expresión*

$$\bar{x}_n = \bar{x}_{n-1} + G(f(\bar{x}_{n-2}), f(\bar{x}_{n-3}))$$

y determinar el error producido.

La expresión de la fórmula numérica es de la siguiente manera:

$$\int_{t_{n-1}}^{t_n} \bar{x}'(t) dt = \int_{t_{n-1}}^{t_n} [f(\bar{x}_{n-2})P_{n-2}(t) + f(\bar{x}_{n-3})P_{n-3}(t)] dt$$

Calculamos los polinomios base de Lagrange para t_{n-2} y t_{n-3} .

$$P_{n-2}(t) = \frac{(t - t_{n-3})}{(t_{n-2} - t_{n-3})} = \frac{(t - t_{n-3})}{h}$$

$$P_{n-3}(t) = \frac{(t - t_{n-2})}{(t_{n-3} - t_{n-2})} = \frac{-(t - t_{n-2})}{h}$$

Interpolamos la función f con estos polinomios e integramos:

$$\begin{aligned} \bar{x}_n - \bar{x}_{n-1} &= f(\bar{x}_{n-2}) \int_{t_{n-1}}^{t_n} \frac{(t - t_{n-3})}{h} dt + \\ &+ f(\bar{x}_{n-3}) \int_{t_{n-1}}^{t_n} \frac{-(t - t_{n-2})}{h} dt = \\ &= f(\bar{x}_{n-2}) \left[\frac{(t - t_{n-3})^2}{2h} \right]_{t_{n-1}}^{t_n} + \\ &+ f(\bar{x}_{n-3}) \left[\frac{(t - t_{n-2})^2}{-2h} \right]_{t_{n-1}}^{t_n} = \\ &= f(\bar{x}_{n-2}) \left[\frac{(3h)^2 - 2h^2}{2h} \right] + f(\bar{x}_{n-3}) \left[\frac{(2h)^2 - h^2}{-2h} \right] \end{aligned}$$

Simplificando y despejando \bar{x}_n obtenemos la fórmula,

$$\boxed{\bar{x}_n = \bar{x}_{n-1} + \frac{1}{2}h [5f(\bar{x}_{n-2}) - 3f(\bar{x}_{n-3})]}$$

El error producido se obtiene a través de la siguiente expresión,

$$Error = \frac{f''(\xi)}{2!} \int_{t_{n-1}}^{t_n} (t - t_{n-2})(t - t_{n-3}) dt$$

haciendo un cambio de variable, $\alpha = t - t_{n-3}$

$$\begin{aligned} Error &= \frac{f''(\xi)}{2!} \int_0^h \alpha(\alpha + h) d\alpha = \\ &= \frac{f''(\xi)}{2!} \left[\frac{\alpha^3}{3} + \frac{\alpha^2}{2}h \right]_0^h = \frac{f''(\xi)}{2!} \frac{5}{6}h^3 \end{aligned}$$

Obtenemos que el error tiene un orden $\mathcal{O}(h^3)$

INTRODUCCION AL PROGRAMA PHASER

El programa phaser es un Software muy sencillo que permite simular en ordenador, las trayectorias de un sistema de ecuaciones diferenciales. El programa está desarrollado para PC y sólo requiere el MS-DOS para ser ejecutado. El programa consta de 3 menus: NUMERICS (al que se entra por defecto), UTILITIES y VISUALIDS.

Descripción de los comandos del Menu NUMERICS

Go : Calcula las órbitas.
Clear : Limpia la pantalla.
Dimension : Cambia la dimensión del sistema.
Algorithm : Cambia el tipo de algoritmo: Euler, Euler mejorado, Runge-Kutta o difference para una ecuación en diferencias.
Step size : Establece el paso de la discretización.
Time : Establece el intervalo de tiempo para calcular la trayectoria.
Xtend : Permite una vez calculada la órbita, continuar para tiempos posteriores.
Equation : Activa una ecuación diferente dentro de la biblioteca de ecuaciones. Es importante tener en cuenta que las ecuaciones están clasificadas por la dimensión y por si son diferenciales o en diferencias.
Parameter : Cambia los parámetros de la ecuación (si los hubiera).
InitCond : Establece las condiciones iniciales (pueden ser varias simultáneamente).
Window Size : Establece el tamaño de la ventana de visualización.
Jump/Plt : Establece cada cuantas iteraciones se dibuja la órbita.
UTILITIES : Cambia al menu UTILITIES
VISUALIDS : Cambia al menu VISUALIDS
Orbit : Dibuja las órbitas en la dirección inversa
Quit : Salir del Programa

Descripción de los comandos del Menu UTILITIES

Two Views : Da como salida por pantalla dos ventanas para representar diferentes cosas.
Big View : Da como salida por pantalla una única ventana.
Xi vs. T : Establece la variable x_i que será dibujada respecto al tiempo.
Flow : Elige de forma automática una malla de condiciones iniciales para calcular sus trayectorias.
DirField : Establece el campo de direcciones.
1d Stairs : Establece cada cuantas iteraciones de una ecuación en diferencias se dibuja.
Save : Salva en un archivo la configuración de una ventana. Las ventanas pueden concatenarse utilizando el mismo nombre de archivo al guardarlas.
Restore : Recupera una cadena de ventanas archivadas con la opción *Save*. El programa trae un archivo llamado demo donde se presentan una serie de pantallas que ilustran las posibilidades del programa. Se recomienda la ejecución de esta demostración.
Ad/Del : Permite añadir/borrar ecuaciones de la biblioteca de ecuaciones. Hay que tener en cuenta que hay que introducir a priori las constantes y variables de la ecuación. Así por ejemplo la ecuación $x' = (a + 3)x + 1$ se escribe en el phaser como $X1' = (a + C1)X1 + C2$, donde existen 2 constantes: $C1 = 3$ y $C2 = 1$, y a es un parámetro que puede cambiarse en la ecuación.

El menu VISUALIDS ofrece opciones para visualizar (sobre todo en 3 - D). Es interesante activar la opción *Flash - Top* que produce destellos luminosos por donde va pasando la trayectoria.

PRACTICA 5.

(1) Introducir en el phaser la ecuación:

$$\begin{cases} x' = y(y^2 - 2|y| + 1) \\ y' = -x \end{cases}$$

Mostrar la ecuación en la pantalla y salvarla utilizando la opción *Save* del menu **UTILITIES**.

(2) Seleccionar la ecuación precedente. Cambiar las dimensiones de la ventana a $[-2, 2]x[-2, 2]$, fijar el tiempo para dibujar las trayectorias de 0 a 30, y dibujar las órbitas, en una ventana grande para las condiciones iniciales (0.1, 0.1), (0.2, 0.2), (0.3, 0.3), (0.4, 0.4), y (0.5, 0.5). (Salvar la pantalla correspondiente concatenándola con la pantalla anterior).

(3) Probar los resultados cambiando el paso y el algoritmo. (Salvar las pantallas de comportamientos más relevantes, concatenándolas con las pantallas anteriores).

(4) Dibujar $x(t)$ y $y(t)$ respecto a t . (Salvar las pantallas concatenándolas con las pantallas anteriores).

PRACTICA 6.

Introducir en el phaser la ecuación diferencial asociada al modelo del movimiento armónico simple:

$$x'' + kx' + w^2x = a \sin(\beta t)$$

donde $x(t)$ representa la amplitud de la ondulación del movimiento, w un parámetro que representa la frecuencia natural de oscilación, $k \geq 0$ un parámetro que representa el rozamiento ($k = 0$ indica que no hay rozamiento) y $a \sin(\beta t)$ representa la fuerza externa. Poner en pantalla la gráfica de $x(t)$ y analizar los siguientes pasos (salvando de forma concatenada las correspondientes pantallas):

(1) $k = a = 0$ (movimiento oscilatorio puro, darle diferentes valores a w).

(2) $w = 1, k = 0$ (movimiento oscilatorio perturbado por una fuerza externa). Darle diferentes valores a a y β . Buscar valores de w y β para los cuales el acoplamiento de las dos frecuencias sea malo y el movimiento resultante no sea periódico.

(3) $w = \beta = 1$. Darle valores a a y k y analizar el resultado.

BIBLIOGRAFIA BASICA

[Al-Sa] Alvarez L. y Sánchez J. "Análisis Numérico". Ed. Universidad de Las Palmas de Gran Canaria. ISBN: 84-699-0762-X.

[Ci] Ciarlet P.G. "Introduction à l'analyse numérique matricielle et à l'optimisation", Masson, 1990.

[La-Th] Lascaux P., Théodor R. "Analyse numérique matricielle appliquée à l'art de l'ingénieur. Vol. 1 Méthodes directes y Vol. 2 Méthodes itératives", Masson, 1993.

[Lu-Pi] Lucquin B., Pironneau O. "Introduction au calcul scientifique", Masson, 1996.

[Ge-Bo] P.L. George, H.Borouchaki. "Triangulation de Delaunay et maillage", Hermes, 1997.

[Jo] C.Johnson "Numerical Solutions of Partial Differential Equations", Cambridge University Press, 1987

[Bo-Di] W.Boyce, R.DiPrima "Ecuaciones Diferenciales", Limusa, 1994

[Ra-Th] P.A.Raviart, J.M.Thomas "Introduction à l'analyse numérique des équations aux dérivées partielles", Masson, 1983.

[Hu] P.Hulquitz "Numerical Methods", Benjamin/Cummings Publishing Company, 1988.

APENDICE A: Resumen de los comandos de UNIX

En este breve resumen seguiremos el siguiente esquema. En primer lugar aparece el comando UNIX, a continuación entre paréntesis su equivalente en MS-DOS (si existe), y finalmente un comentario y un ejemplo.

cd (cd) cambia el directorio activo

>cd /users/p701/fortran77

more (type) visualiza el contenido de un fichero

>more /users/p701/fortran77/programas/prog1.f

ls (dir) Visualiza contenido de un directorio

>ls /users/p701/fortran77

cp (copy) Copia un fichero en otro.

>cp /users/p701/fortran77/programas/prog1.f .

rm (del) borra un fichero

>del prog1.f

man (help) suministra ayuda sobre un comando

> man ls

logout Se termina la sesión y se sale del sistema

>logout

ps Visualiza los números de procesos que están abiertos que corresponden al usuario *alumno*

>ps -u *alumno*

kill Interrumpe la ejecución de un proceso de numero *Nproceso*

>kill -9 *Nproceso*

mkdir (mkdir) Crea un directorio

>mkdir practical

rmdir (rmdir) Borra un directorio

>rmdir practical

mv (move) Cambia de nombre o ubicación un archivo.

>mv prog1.f practical.f

chmod Cambia los permisos de lectura, escritura y ejecución de un fichero. Este comando es de utilidad para salvaguardar la información de directorios y ficheros de miradas ajenas.

Hacer > *man* chmod para mirar las opciones.

chown Cambia el propietario de un fichero

Hacer > *man* chown para mirar las opciones.

du (tree) Visualiza la cadena de directorios

>du /users/p701

find Busca un archivo de nombre *file* en el directorio *dir*

>find *dir* -name *file* -print

grep Busca los ficheros que contenga la cadena de caracteres *string*

>grep *string* *

APENDICE B: Resumen del procesador de texto *vi*.

El procesador de texto *vi*, tiene la ventaja de estar presente en cualquier máquina que trabaje sobre UNIX, y no requiere ningún entorno gráfico. Puede ejecutarse en dos modos. El modo comando (el que está por defecto al entrar en *vi*). Donde se ejecutan comandos como copy-paste, ect.. y el modo edición que es donde se escribe normalmente el texto.

Intercambio entre modo comando y modo edición

ESC; Pasa de modo edición a modo comando

i; Pasa de modo comando a modo edición

A; Pasa a modo edición y pone el cursor al final de la línea

O; inserta una nueva línea, pasa a modo edición y pone el cursor al principio de la nueva línea.

Manejo de Ficheros (En modo comando)

:w; escribe en disco el fichero

:wq; escribe en disco el fichero y sale del *vi*

:e fichero.name; edita el fichero *fichero.name*

:q!; sale del *vi* sin guardar cambios.

:w fichero.name; escribe el fichero actual en el fichero *fichero.name* en disco

!comando; ejecuta el comando UNIX *comando*

:set nu; presenta los números de línea en pantalla

Comandos para desplazarse por el texto (En modo comando)

Crtl F; Página Adelante

Crtl B; Página Atrás

\$; Pone el cursor en el final de la línea

0; Pone el cursor en el principio de línea

/string; Busca hacia adelante el string *string*

?string; Busca hacia atrás el string *string*

n; Repite la última búsqueda

G; Va al final del texto

3 G; Va a la línea número 3.

Comandos para borrar líneas o caracteres (En modo comando)

- x**; borra el caracter donde se encuentra el cursor
- r** *character*; reemplaza el caracter donde se encuentra el cursor por el caracter *character*.
- dd**; borra la línea donde se encuentra el cursor.
- 3 dd**; borra 3 líneas desde donde se encuentra el cursor hacia abajo.
- dw**; borra la palabra donde se encuentra el cursor.

Comandos para copiar y desplazar bloques (En modo comando)

- yy**; copia en el buffer la linea donde se encuentra el cursor
- 3yy**; copia en el buffer 3 líneas hacia abajo desde el cursor
- dd**; copia (y borra) al buffer la linea donde se encuentra el cursor
- 3dd**; copia (y borra) al buffer 3 líneas hacia abajo desde el cursor
- p**; copia el contenido del buffer en el texto.