

E. de Ingenierías Industriales

2013-14

Métodos Matemáticos I

Jesús Rojo

01. Introducción

01. Introducción

Parte 1. Métodos de Runge-Kutta para las E.D.O.

Introducción

Ecuaciones escalares: el método de Euler

Ecuaciones escalares: métodos de Runge-Kutta

Estimación del error y cambio de paso

Sistemas autónomos: métodos de Runge-Kutta

Sobre convergencia, consistencia y 0-estabilidad

Métodos de Runge-Kutta implícitos

01. Introducción

- 1 Ecuaciones diferenciales ordinarias: ecuaciones y sistemas de primer orden
- 2 Notación para los sistemas
- 3 Unicidad de la solución
- 4 Sistemas autónomos y no autónomos
- 5 Ecuaciones de orden superior
- 6 Red y nodos

Ecuaciones y sistemas de primer orden

Forma **normal** de una ecuación de primer orden

$$y' = f(x, y),$$

donde f es una función

$$f : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R},$$

o sea, una función escalar de dos variables. En ese caso se dice que se trata de una **ecuación escalar**.

Pero es corriente (y un poco más difícil de pensar en ello) que sea

$$f : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m,$$

en cuyo caso se trata de un **sistema de m ecuaciones diferenciales**.

Cuando f es una función vectorial, suele escribirse como

$$\mathbf{f} : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$$

o de alguna manera que 'recuerde' que f es vectorial. Si escribimos 'con bolígrafo' no es fácil señalar \mathbf{f} como negrita, y se suele reemplazar por \vec{f} o por \underline{f} , que es más fácil en ese contexto.

En ese caso, x sigue siendo escalar, pero \mathbf{y} o $\mathbf{y}(x)$ es un vector de m componentes (como \mathbf{f}).

Ejemplos:

$$y' = y$$

es una ecuación (diferencial) escalar. También lo es

$$y' = x \sin^2 y.$$

En el primer caso, la función que hemos llamado con f es

$$f(x, y) = f(y) = y,$$

mientras que en el segundo se trata de

$$f(x, y) = x \sin^2 y.$$

Cuando se busca una **solución** de la ecuación, la $y(x) = e^x$ sirve para la primera y la $y(x) = \frac{2}{1-x^2}$ para la segunda.



Ejemplos:

$$\begin{cases} y_1' = y_2 \sin^2 y_1 \\ y_2' = 1, \end{cases}$$

es un sistema con $m = 2$ ecuaciones y el mismo número de funciones incógnita. Nuestra función f es ahora $f : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ y posee dos componentes

$$f_1(x, y_1, y_2) = y_2 \sin^2 y_1, \quad y$$

$$f_2(x, y_1, y_2) = 1.$$

Ahora, la pareja $y_1(x) = \frac{2}{1-x^2}$, $y_2(x) = x$ es una solución de este sistema



Si, como ocurre en el primero de los ejemplos que hemos dado, la ecuación es de la forma

$$y' = f(y),$$

o sea, la variable independiente no interviene en la ecuación (o el sistema) se dice que la ecuación es **autónoma** (o que el sistema es autónomo).

Veremos que, en nuestro tratamiento de los sistemas de ecuaciones diferenciales, este hecho es muy ventajoso.

No debemos pensar que esto significa que la solución o soluciones no dependen tampoco de la variable independiente (o sea que son constantes). Basta recordar que $y(x) = e^x$ (que depende claramente de x) es solución.

Recordaremos (más adelante) cómo se convierte un sistema en otro autónomo con las mismas soluciones. Es un artificio sencillo pero el 'precio a pagar' es el aumento del número de cálculos a efectuar.



Un sistema como el del ejemplo anterior, en que cada ecuación (cada componente de f) depende de las distintas componentes de y y se suele llamar **acoplado**. Lo contrario, o sea que cada ecuación dependa sólo de un y'_i y del mismo y_i solamente, sería una simple yuxtaposición de ecuaciones escalares, que podrían ser tratadas una a una.



Notación para los sistemas

Problema de notación: la notación y_i que suele representar la componente número i del vector o la función vectorial $y(x)$ se encontrará pronto con el uso repetido de un nuevo índice, el correspondiente al paso n o n -ésimo, dando lugar a cosas como

$$y_{i,n}$$

que a mano e incluso en letra impresa puede resultar confuso.

Usaremos la notación

$${}^i y_n$$

para representar la aproximación en x_n del valor de la componente número i del vector o la función vectorial $y(x)$, notación bastante útil, aunque incómoda al principio.

El ejemplo que hemos citado poco antes se escribe ahora, con esta notación,

$$\begin{cases} {}^1y' = {}^2y \sin^2 {}^1y \\ {}^2y' = 1, \end{cases}$$

Corresponde a una función $f : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^2$ de componentes

$$\begin{aligned} {}^1f(x, {}^1y, {}^2y) &= {}^2y \sin^2 {}^1y, & y \\ {}^2f(x, {}^1y, {}^2y) &= 1. \end{aligned}$$

La solución que citamos se escribe de esta manera

$$\begin{aligned} {}^1y(x) &= \frac{2}{1-x^2} \\ {}^2y(x) &= x. \end{aligned}$$

Para el caso de un sistema de m ecuaciones y funciones incógnita, la función

$$\mathbf{f} : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$$

es (con nuestra notación) de componentes

$$\mathbf{f}(x, \mathbf{y}) = \begin{pmatrix} {}^1 f(x, \mathbf{y}) \\ {}^2 f(x, \mathbf{y}) \\ \dots \\ {}^m f(x, \mathbf{y}) \end{pmatrix}$$

o, si se quiere,

$$\mathbf{f}(x, {}^1 y, {}^2 y, \dots, {}^m y) = \begin{pmatrix} {}^1 f(x, {}^1 y, {}^2 y, \dots, {}^m y) \\ {}^2 f(x, {}^1 y, {}^2 y, \dots, {}^m y) \\ \dots \\ {}^m f(x, {}^1 y, {}^2 y, \dots, {}^m y) \end{pmatrix}$$

La función incógnita \mathbf{y} es, para el caso de un sistema de m ecuaciones y funciones incógnita, la función

$$\mathbf{y} = \begin{bmatrix} {}^1 y \\ {}^2 y \\ \dots \\ {}^m y \end{bmatrix} = \mathbf{y}(x) = \begin{bmatrix} {}^1 y(x) \\ {}^2 y(x) \\ \dots \\ {}^m y(x) \end{bmatrix}$$

y su derivada es también

$$\mathbf{y}' = \begin{bmatrix} {}^1 y' \\ {}^2 y' \\ \dots \\ {}^m y' \end{bmatrix} = \mathbf{y}'(x) = \begin{bmatrix} {}^1 y'(x) \\ {}^2 y'(x) \\ \dots \\ {}^m y'(x) \end{bmatrix}$$

Unicidad de la solución

Los métodos que estudiaremos sirven para aproximar soluciones de ecuaciones

$$y' = f(x, y)$$

cuando existe una única solución (Caso contrario: mezcla 'aleatoria' de soluciones).

Una ecuación de este tipo con solución única debe estar al menos acompañada de una condición del tipo

$$y(a) = \eta,$$

llamada **condición inicial**. Aunque aún esto no es suficiente.

En el caso de un sistema, la condición inicial $y(a) = \eta$ se debe entender como

$$\mathbf{y}(a) = \boldsymbol{\eta},$$

o, descrito escalarmente,

$$\begin{bmatrix} {}^1y(a) \\ {}^2y(a) \\ \dots \\ {}^my(a) \end{bmatrix} = \begin{bmatrix} {}^1\eta \\ {}^2\eta \\ \dots \\ {}^m\eta \end{bmatrix}.$$

Las condiciones suficientes para la existencia de una única solución las describe el siguiente resultado.

Teorema

Consideremos el problema de valores iniciales

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

con

$$\mathbf{f} : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$$

Se supone que \mathbf{f} está definida y es continua en la banda infinita D dada por

$$x \in [a, b], \quad {}^k y \in (-\infty, \infty), \quad k = 1, \dots, m.$$



Se supone además que existe L (hablaremos de la 'constante de LIPSCHITZ' L) tal que

$$\|\mathbf{f}(x, \mathbf{y}) - \mathbf{f}(x, \mathbf{y}^*)\| \leq L \|\mathbf{y} - \mathbf{y}^*\|$$

para $(x, \mathbf{y}), (x, \mathbf{y}^*) \in D$.

Entonces existe en $[a, b]$ una única solución $\mathbf{y}(x)$, que es continua y diferenciable en D

Es posible que referirse a una ecuación escalar y reemplazar las normas por

$$|\mathbf{f}(x, \mathbf{y}) - \mathbf{f}(x, \mathbf{y}^*)| \leq L |\mathbf{y} - \mathbf{y}^*|$$

facilite la visión del resultado, ya que el módulo de un escalar suele ser 'más visible' que la norma de un vector.



Sistemas autónomos y no autónomos

Un sistema no autónomo

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}),$$

o bien

$$\begin{bmatrix} {}^1y' \\ {}^2y' \\ \dots \\ {}^my' \end{bmatrix} = \begin{pmatrix} {}^1f(x, {}^1y, {}^2y, \dots, {}^my) \\ {}^2f(x, {}^1y, {}^2y, \dots, {}^my) \\ \dots \\ {}^mf(x, {}^1y, {}^2y, \dots, {}^my) \end{pmatrix}$$

se convierte en autónomo añadiéndole la ecuación

$${}^{m+1}y' = 1.$$

O, si se trata del problema de valores iniciales,

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

añadiendo, además, la condición inicial

$${}^{m+1}y(a) = a.$$

Es fácil de recordar si se piensa que estamos añadiendo la componente número $m + 1$ de la solución

$${}^{m+1}y(x) = x.$$

Como vemos, este procedimiento hace pasar de un sistema de tamaño m a uno de tamaño $m + 1$. Aunque esto aumenta el número de operaciones, no aumenta la complejidad de la programación, salvo cuando se intenta convertir una ecuación escalar no autónoma en un sistema, que entonces sería de tamaño 2 (y eso aumenta la complejidad de la programación, que debe pasar del caso escalar al vectorial).

Las soluciones del sistema no autónomo (de tamaño m) y las del sistema autónomo transformado (de tamaño $m + 1$) se relacionan como sigue:

- 1 Si ${}^1y(x), \dots, {}^my(x)$ son soluciones del sistema no autónomo (o del problema no autónomo de condiciones iniciales), entonces ${}^1y(x), \dots, {}^my(x), {}^{m+1}y(x) = x$ lo son del sistema transformado (o del problema transformado).
- 2 Si ${}^1y(x), \dots, {}^my(x), {}^{m+1}y(x)$ son soluciones del sistema transformado (o del problema de valores iniciales transformado), entonces ${}^1y(x), \dots, {}^my(x)$ son soluciones del sistema no autónomo (o del problema no autónomo de condiciones iniciales) y, además, ${}^{m+1}y(x) = x + c$ o ${}^{m+1}y(x) = x$ según se trate sólo del sistema o del sistema con condiciones iniciales.

Ejemplo: La ecuación escalar (o sistema de tamaño 1)

$$y' = x \sin^2 y$$

se transforma en el sistema de tamaño 2

$$\begin{cases} {}^1y' = {}^2y \sin^2 {}^1y \\ {}^2y' = 1, \end{cases}$$

La función $y(x) = \frac{2}{1-x^2}$ es una solución de la ecuación, o del problema que resulta cuando añadimos $y(0) = 2$. Lo que corresponde para el sistema transformado es la solución que ya vimos, ${}^1y(x) = \frac{2}{1-x^2}$, ${}^2y(x) = x$, que también vale para la condición inicial ${}^1y(0) = 2$, ${}^2y(0) = 0$.

Ecuaciones de orden superior

Otros casos se relacionan sencillamente con el problema fundamental que vamos a tratar. Por ejemplo, la ecuación (normal) de orden q

$$y^{(q)} = f(x, y, y', \dots, y^{(q-1)}),$$

se lleva a uno de nuestros sistemas de primer orden de tamaño q con el habitual cambio

$${}^1y = y, \quad {}^2y = y', \dots, \quad {}^qy = y^{(q-1)}.$$

Red y nodos para la aproximación numérica de la solución de $y' = f(x, y), \quad y(a) = \eta$

Supongamos que estamos ante un problema (escalar o sistema, lo que por ahora será similar, aunque es mejor pensar en un problema escalar para acompañarse de un dibujo)

$$y' = f(x, y), \quad y(a) = \eta.$$

Supongamos además que se cumplen las condiciones que hemos enunciado en un reciente teorema: f está definida y es continua en la banda dada por $x \in [a, b], \quad y \in (-\infty, \infty)$ (en el caso escalar) o por $x \in [a, b], \quad {}^k y \in (-\infty, \infty), \quad k = 1, \dots, m$ (en el caso de un sistema) y existe L tal que $|f(x, \mathbf{y}) - f(x, \mathbf{y}^*)| \leq L |\mathbf{y} - \mathbf{y}^*|$ (caso escalar) o $\|f(x, \mathbf{y}) - f(x, \mathbf{y}^*)\| \leq L \|\mathbf{y} - \mathbf{y}^*\|$ (caso de un sistema).

Un problema de esas características el lo que llamaremos de forma breve un **problema usual** o un **problema tipo**. Ya sabemos que, en este caso, en el intervalo $[a, b]$ existe una única solución $y(x)$ del problema ($y(x)$ si se trata del caso escalar), que es la solución que intentamos aproximar.

En todo lo que sigue, de este y los restantes capítulos, supondremos que las funciones ($f(x, y(x))$) y la solución $y(x)$ de la que hablábamos son tan regulares como se necesite. O sea que son indefinidamente derivables (en la forma usual o parcial, según el número de variables) y además con continuidad de las derivadas. El objetivo es que se pueda hacer uso de los desarrollos en serie que irán apareciendo. Y esto no es ninguna comprobación que debamos ir haciendo a nuestro problema, aunque conviene que esperemos al comentario que haremos enseguida sobre la naturaleza de $f(x, y(x))$ en los casos más realistas.

Dividimos el intervalo $[a, b]$ en N subintervalos tomando para ello

$$h = \frac{b - a}{N}$$

y considerando las abscisas (o **nodos**) que dividen los intervalos

$$x_0 = a, x_1 = a + h, x_2 = a + 2h, \dots, x_n = a + nh, \dots, x_N = a + Nh = b,$$

lo que proporciona una serie de nodos

$$x_0 = a < x_1 < x_2 < \dots < x_N = b,$$

que distan h entre sí.

Buscaremos aproximaciones

$$\mathbf{y}_n = [{}^1y_n, {}^2y_n, \dots, {}^my_n]$$

a los valores de $\mathbf{y}(x)$ en los nodos x_n , o sea, a

$$\mathbf{y}(x_n) = [{}^1y(x_n), {}^2y(x_n), \dots, {}^my(x_n)],$$

procurando hacer lo mejor posible el parecido

$$\mathbf{y}_n \sim \mathbf{y}(x_n),$$

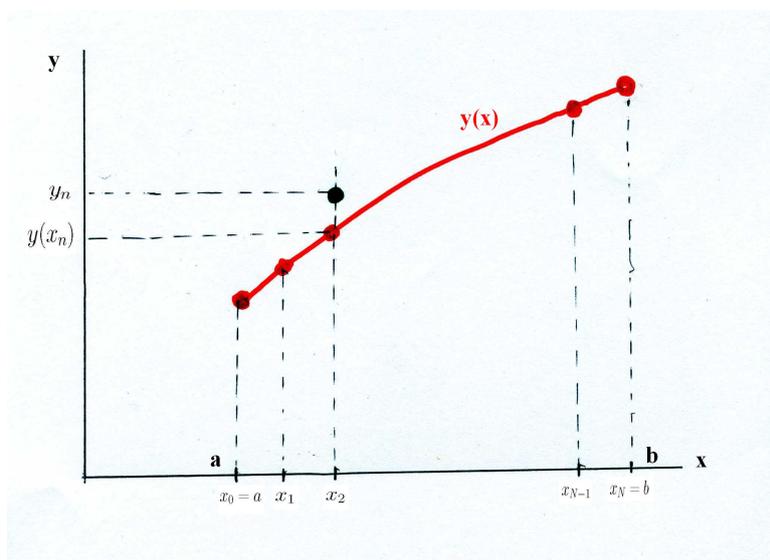
o sea,

$${}^1y_n \sim {}^1y(x_n), {}^2y_n \sim {}^2y(x_n), \dots, {}^my_n \sim {}^my(x_n), .$$

En el caso de una ecuación escalar, donde lo que buscaremos es aproximar y_n a $y(x_n)$ para cada n , o sea,

$$y_n \sim y(x_n),$$

es posible ilustrar esto mediante una gráfica sencilla:



La naturaleza de $f(x, y(x))$

Aunque el empleo de los métodos será para problemas concretos, o sea, para funciones $f(x, y(x))$ concretas, el desarrollo debe hacerse de manera general, o sea, que f será normalmente una 'simple letra'.

En cada caso concreto al que apliquemos nuestro estudio, f podrá ser una función conocida y de fórmula a veces sencilla o a veces larga o complicada. Pero de hecho, lo más usual, en casos de aplicación a la industria, es que f 'no tenga' fórmula, o sea, que no sea conocida más que en los puntos en que nos vaya interesando y ello gracias a algún experimento que nos dé como resultado el valor de $f(x, y(x))$ que se necesita en cada momento.

Por una parte, esto nos lleva a la necesidad de valorar las evaluaciones de f como lo más 'caro' o 'costoso' (en tiempo e incluso en dinero) de un proceso de cálculo.

Por otro lado, esta situación deja también claro que hipótesis como las que hemos hecho sobre $f(x, y(x))$ o $y(x)$ no son algo comprobable, sino únicamente conjeturas sobre el modelo que tengamos. O sea, que no son hechos a comprobar cuidadosamente sino a suponer en los casos en que ello pueda parecer razonable.

Ejercicios de

(1) Introducción

02. Ecuaciones escalares: el método de EULER

- 1 Métodos numéricos: métodos de un paso
- 2 El método de EULER
- 3 El error:consistencia y orden
- 4 Error local y error global

Métodos numéricos: métodos de un paso

Consideremos lo que hemos llamado un problema tipo o problema usual, consistente en una ecuación diferencial de primer orden en la forma normal, que ahora será una ecuación escalar, y una condición inicial

$$y' = f(x, y), \quad y(a) = \eta,$$

donde por lo tanto, la función f es

$$f : [a, b] \times \mathbb{R} \rightarrow \mathbb{R},$$

y en su dominio

$$D : x \in [a, b], \quad y \in (-\infty, \infty),$$

es Lipschitziana respecto de la variable y , con constante de LIPSCHITZ L .

Ya sabemos que, en estas condiciones, en el intervalo $[a, b]$ existe una única solución $y(x)$ del problema, que es la solución que intentamos aproximar.



Un **método numérico** o, simplemente **método**, es una ecuación en diferencias que hace intervenir como incógnitas valores

$$y_n \sim y(x_n)$$

que se consideran aproximaciones de la solución $y(x)$ en puntos x_0, x_1, \dots, x_N dados como

$$x_n = a + nh, \quad n = 0, \dots, N,$$

con

$$N = \frac{b-a}{h} \quad \text{o} \quad h = \frac{b-a}{N},$$

donde h es lo que se denomina **paso** del método y que en primera instancia consideraremos fijo.



Si en la ecuación en diferencias intervienen y_n e y_{n+1} , diremos que el método es de **1 paso** o de **paso simple**.

Si en la misma intervienen $y_n, y_{n+1}, y_{n+2} \dots$, diremos que el método es de **paso múltiple**.

Dentro de los métodos de 1 paso, que son los que van a centrar nuestra atención, nos interesaremos por los que tienen la forma

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

que son básicamente aquellos en los que y_{n+1} se puede despejar en función de los restantes elementos de la ecuación. De estos métodos diremos que son **explícitos**, llamándose implícitos o no explícitos los restantes.

Supondremos, además, que las ecuaciones en diferencias son las mismas para cada valor de n . De esa manera, el método se describirá dando una cualquiera de las ecuaciones. Pasando a aquellos de los que vamos a ocuparnos, el conocimiento de la expresión

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

para n genérico nos describirá el método de que se trate.

Lógicamente, un método como los anteriores no se puede resolver salvo que tengamos alguna manera de conocer el valor de y_0 , por ejemplo. Ahora bien, condiciones como la inicial

$$y(a) = \eta,$$

nos permiten añadir

$$y_0 = \eta$$

y arrancar el método en diferencias.

Digamos finalmente que un **algoritmo** es una secuencia de códigos de ordenador que permite implementar el método.
 Notemos que un método puede ser implementado por diferentes algoritmos (lo que matemáticamente puede no tener importancia); cuando se consideran los errores de redondeo es cuando un algoritmo se puede mostrar superior a otro.

El método de EULER

Hay varias maneras 'intuitivas' de introducir este método. La más rápida puede ser presentar el método como proveniente de substituir el valor de $y'(x)$ por el cociente incremental

$$y'(x) \sim \frac{y(x+h) - y(x)}{h}.$$

Entonces, de la ecuación $y'(x) = f(x, y(x))$ se obtiene la igualdad aproximada

$$f(x, y(x)) \sim \frac{y(x+h) - y(x)}{h},$$

que lleva a

$$y(x+h) \sim y(x) + hf(x, y(x));$$

para x_n (y para $x_{n+1} = x_n + h$) esto significa

$$y(x_{n+1}) \sim y(x_n) + hf(x_n, y(x_n)).$$

Pues bien, el **método de EULER** consiste en tomar y_n e y_{n+1} como los valores que verifiquen exactamente

$$y_{n+1} = y_n + h f(x_n, y_n),$$

que es la ecuación en diferencias que constituye el método. Es un método de un paso y tiene la forma ya citada

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

cuando tomamos

$$\Phi(x, y, h) = f(x, y).$$

Es claro que la ecuación del método de EULER se resuelve en cuanto se proporciona un valor para y_0 , por ejemplo. Es normal tomar

$$y_0 = y(x_0) = y(a) = \eta,$$

lo que completa el método con un arranque totalmente natural.

El error: consistencia y orden

El estudio del error debería analizar las diferencias

$$y(x_n) - y_n,$$

pero esto choca con el desconocimiento del valor de $y(x_n)$ necesario para la comparación, que sólo es posible cuando en algún ejemplo se conoce la solución exacta $y(x)$, pero que no es posible en la práctica (donde desconoceremos esta solución analítica) ni tampoco en el estudio general (en que el problema y su solución son genéricos).

Lo que vamos a poder analizar es el valor de la cantidad residual dada por la expresión

$$y(x_{n+1}) - y(x_n) - h f(x_n, y(x_n)),$$

que será pequeño cuando h lo es, teniendo en cuenta que

$$y_{n+1} - y_n - h f(x_n, y_n) = 0.$$

Vamos a denotar por T_{n+1} o $T_{n+1}(h)$ esta cantidad, y la denominaremos **Error local de truncación**, nombre que abreviaremos con **LTE**, que no son sino las iniciales inglesas de este concepto (mantenemos estas iniciales porque son las que aparecen con más frecuencia). Así pues

$$T_{n+1}(h) = y(x_{n+1}) - y(x_n) - h f(x_n, y(x_n)),$$

donde $y(x)$ es la solución de nuestro problema tipo. Los resultados que vamos a obtener no se ven influidos por el hecho del n que figura en esta expresión, sino que se obtienen trabajando simplemente con

$$T(h) = y(x+h) - y(x) - h f(x, y(x)),$$

que es lo que haremos.

Una pequeña explicación sobre el calificativo 'local' que añadimos a la palabra error. Si suponemos que $y_n = y(x_n)$, o sea si nos fijamos la solución exacta del problema

$$y' = f(x, y), \quad y(x_n) = y_n$$

(que no es la solución, claro, con la condición $y(a) = \eta$) el error vale

$$\begin{aligned} T_{n+1}(h) &= y(x_{n+1}) - y(x_n) - h f(x_n, y(x_n)) \\ &= y(x_{n+1}) - y_n - h f(x_n, y_n) \\ &= y(x_{n+1}) - y_{n+1}, \end{aligned}$$

o sea, es el que aparece cuando nos limitamos a dar entonces el paso entre y_n e y_{n+1} , sin que ningún error provenga así de los pasos precedentes.

Un método va a ser 'utilizable' para aproximar la solución $y(x)$ de un problema tipo cuando el error local de truncación converja a 0 con mayor rapidez que el paso h lo hace. O sea, cuando

$$\lim_{h \rightarrow 0} \frac{T(h)}{h} = 0.$$

¿Es de este tipo nuestro método de EULER? Para dicho método tenemos

$$\frac{T(h)}{h} = \frac{y(x+h) - y(x)}{h} - f(x, y(x)) = \frac{y(x+h) - y(x)}{h} - y'(x),$$

y esto, evidentemente, tiende a 0 ya que, por la propia definición de $y'(x)$, esta derivada es el límite cuando $h \rightarrow 0$ del cociente incremental que aparece a la izquierda.

La propiedad precedente se denomina **consistencia** del método, y el método que la posee **método consistente**. Lo que hemos visto es que el método de EULER es consistente.

También diremos que el método de EULER es de orden 1 (o al menos 1) debido a la propiedad

$$\lim_{h \rightarrow 0} \frac{T(h)}{h^1} = 0,$$

así como diremos que un método es 'al menos de orden p ' cuando

$$\lim_{h \rightarrow 0} \frac{T(h)}{h^p} = 0.$$

¿Será el método de EULER de algún orden más alto?. Para responder a ello conviene hacer algunos preparativos.

Comencemos desarrollando $T(h)$ en serie de TAYLOR de potencias de h .

$$T(h) = y(x+h) - y(x) - hf(x, y(x)),$$

y tanto $y(x) = h^0 y(x)$ como $hf(x, y(x)) = h^1 f(x, y(x))$ ya están listos. Queda hacer el desarrollo de $y(x+h)$, naturalmente de forma genérica ya que $y(x)$ no es ninguna función concreta.

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(x) + \dots$$

en desarrollo infinito de TAYLOR. Como además $y'(x) = f(x, y(x))$, resulta que

$$T(h) = \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(x) + \dots,$$

Mejor aún, haciendo el desarrollo finito de TAYLOR con resto,

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2} y''(\zeta),$$

lo que nos queda es que

$$T(h) = \frac{h^2}{2} y''(\zeta)$$

donde ζ es un punto (desconocido o no concreto) que existe y que hace cierto el desarrollo en la forma finita; de ζ sabemos únicamente que estará entre x y $x+h$, o sea en el intervalo $[x, x+h]$ si h es positivo.

Ahora podemos contestar a la pregunta antes planteada; tenemos que

$$\lim_{h \rightarrow 0} \frac{T(h)}{h^2} = \frac{1}{2} y''(\zeta),$$

y esto no es generalmente nulo. Por lo tanto, el método de EULER es exactamente de orden 1.

Perfeccionaremos este lenguaje con la introducción de la notación

$$\mathcal{O}(h^p),$$

de uso tradicional para infinitésimos.

Si $g(h)$ es un infinitésimo dependiente de la variable h que interesa cuando $h \rightarrow 0$, se escribe

$$g(h) = \mathcal{O}(h^p)$$

para significar que

$$|g(h)| \leq M h^p$$

para alguna constante M y h suficientemente pequeño.

Se trata de una notación que suele llamarse **la O de LANDAU** iniciada por el físico ruso del mismo nombre.

Conviene comprobar algunos resultados sencillos que luego usaremos, como que

$$\mathcal{O}(h^p) \cdot \mathcal{O}(h^q) = \mathcal{O}(h^{p+q}), \quad \mathcal{O}(h^p) \pm \mathcal{O}(h^q) = \mathcal{O}(h^p)$$

con $p = \min(p, q)$, por ejemplo.

Si ahora recordamos que, para el método de EULER,

$$T(h) = \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(x) + \dots$$

o mejor

$$T(h) = \frac{h^2}{2} y''(\zeta),$$

tenemos que

$$T(h) = \mathcal{O}(h^2),$$

ya que

$$|T(h)| \leq \frac{M}{2} h^2,$$

tomando como M cualquier cota superior de la función y'' en el intervalo entre x y $x + h$.

Como anticipo de lo que luego veremos, digamos que todos los métodos consistentes van a tener un error local $T(h)$ cuyo desarrollo en potencias de h va a comenzar por el término en h^2 o posteriores, o, de forma equivalente, el cociente $T(h)/h$ va a comenzar su desarrollo por el término $h^1 = h$ (sin h^0) o posteriores. Cuando el desarrollo de $T(h)$ comience por el término h^{p+1} o, lo que es lo mismo, el de $T(h)/h$ comience por h^p , diremos que el método es de **orden p** . Esto equivale a decir que $T(h) = \mathcal{O}(h^{p+1})$ y también que

$$\lim_{h \rightarrow 0} \frac{T(h)}{h^p} = 0 \quad \text{y} \quad \lim_{h \rightarrow 0} \frac{T(h)}{h^{p+1}} \neq 0.$$

Por consiguiente, el método de EULER es, como ya habíamos dicho, de orden 1 exactamente (el menor posible entre los métodos consistentes)

El primer término del desarrollo en potencias de h de $T(h)$ se suele llamar **término principal de error**. El coeficiente numérico de este término es la **constante del error** del método.

Para nuestro método de EULER el término principal del error es, como hemos visto,

$$h^2 \frac{1}{2} y''(x).$$

y la constante del error es

$$\frac{1}{2}.$$

El término principal del error va a indicar, claro, el orden de un método. La constante del error va a permitir la comparación entre métodos que tengan el mismo orden.

Error local y error global

Regresemos a la noción de error local de truncación para el método de EULER

$$T_{n+1}(h) = y(x_{n+1}) - y(x_n) - h f(x_n, y(x_n)),$$

que se podía interpretar como el error cometido en el paso de y_n a y_{n+1} exclusivamente.

El **error global (GTE)** o **error global de truncación** será la diferencia

$$E_n = E_n(h) = y(x_n) - y_n$$

en el mismo instante que T_n pero sin suposición ninguna sobre los anteriores pasos. Es de hecho el error que finalmente interesa, pero no es directamente calculable por el desconocimiento habitual del valor de $y(x_n)$, que supondría un conocimiento de la solución exacta del problema.

Sin embargo, se puede dar una apreciación de la magnitud del error global a partir del conocimiento del error local. Para el método de EULER se puede ver que el conocido hecho

$$T_n(h) = \mathcal{O}(h^2)$$

tiene como consecuencia el que, para el error global,

$$E_n(h) = \mathcal{O}(h).$$

De nuevo, este hecho de la disminución en una unidad en la potencia de h al pasar del error local al global es generalizable a otros métodos. Es un hecho general que, para los métodos de orden p , para los que $T_n(h) = \mathcal{O}(h^{p+1})$ el error global tiene el comportamiento $E_n(h) = \mathcal{O}(h^p)$.

No es difícil probar la afirmación que, para el método de EULER, acabamos de hacer. Aunque no lo razonamos aquí para no perder en exceso el hilo de los resultados.

Ejercicios de

(2) Ecuaciones escalares: el método de EULER

Ejercicios de

(2) Ecuaciones escalares: el método de EULER

Ejercicio 02-01

Realizando el desarrollo de TAYLOR de la solución exacta, $y(x)$, del problema

tipo $y'=f(x,y)$, $y(a)=\eta$, obténgase el método de EULER.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(2) Ecuaciones escalares: el método de EULER

Ejercicio 02-02

Siendo $y(x)$ la solución exacta del problema tipo $y'=f(x,y)$, $y(a)=\eta$,

se recuerda que se verifica por la Regla de BARROW

$$y(x+h) - y(x) = \int_x^{x+h} y'(t) dt = \int_x^{x+h} f(t, y(t)) dt$$

Dígase qué aproximación hay que tomar en la integral para obtener el método de EULER. Dígase también qué método se crea cuando se aproxima la integral empleando el 'método del trapecio'

```
[ > restart:  
[ > interface(labeling=false):
```

03. Ecuaciones escalares: métodos de RUNGE-KUTTA

- 1 Métodos de Taylor
- 2 Métodos de un paso: generalidades
- 3 Métodos explícitos de RUNGE-KUTTA
- 4 Métodos de RUNGE-KUTTA explícitos de 2 etapas
- 5 Notaciones y organización
- 6 Métodos de RUNGE-KUTTA explícitos de 2 etapas
- 7 Métodos de RUNGE-KUTTA explícitos de 3 etapas
- 8 Métodos de RUNGE-KUTTA de 4 y más etapas
- 9 El caso escalar autónomo

Métodos de Taylor

Consideremos lo que hemos llamado un problema tipo o problema usual, consistente en una ecuación diferencial de primer orden en la forma normal, que ahora será una ecuación escalar, y una condición inicial

$$y' = f(x, y), \quad y(a) = \eta,$$

donde por lo tanto, la función f es

$$f : [a, b] \times \mathbb{R} \rightarrow \mathbb{R},$$

y en su dominio

$$D : x \in [a, b], \quad y \in (-\infty, \infty),$$

es Lipschitziana respecto de la variable y , con constante de Lipschitz L .

Ya sabemos que, en estas condiciones, en el intervalo $[a, b]$ existe una única solución $y(x)$ del problema, que es la solución que intentamos aproximar.



Comenzaremos construyendo métodos para aproximar la solución $y(x)$ de un tipo llamado de los **métodos de TAYLOR**. No vamos a ocuparnos apenas de estos métodos (no son muy útiles en la práctica, como explicaremos luego), pero nos serán de utilidad como referente de los de RUNGE-KUTTA, que serán los que centren nuestra atención. Su nombre hace referencia al desarrollo de TAYLOR de la solución $y(x)$.

Suponemos que nos movemos en la red usual de nodos

$$x_n = a + n h, \quad n = 0, \dots, N,$$

que dividen el intervalo $[a, b]$ en N subintervalos iguales y de longitud h . Representamos, como ya hemos explicado, por y_n la aproximación

$$y_n \sim y(x_n)$$

al valor $y(x_n)$ de la solución en x_n .



El desarrollo de TAYLOR de $y(x + h)$, con resto tras el segundo término, es

$$\begin{aligned} y(x + h) &= y(x) + h y'(x) + \frac{h^2}{2} y''(\zeta) \\ &= y(x) + h f(x, y(x)) + \frac{h^2}{2} y''(\zeta), \end{aligned}$$

y, la eliminación del término más pequeño, nos llevaba a

$$y(x + h) \sim y(x) + h f(x, y(x)),$$

aproximación que sugería

$$y_{n+1} = y_n + h T^{(1)}(x_n, y_n, h) = y_n + h f(x_n, y_n),$$

que era el método de EULER. Hemos escrito $T^{(1)}(x_n, y_n, h)$ para recordar con $T^{(1)}$ la serie de TAYLOR.



Ahora hacemos lo mismo pero extendiendo la serie de TAYLOR a un término más

$$y(x + h) = y(x) + h y'(x) + \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(\zeta).$$

Como antes, $y'(x) = f(x, y(x))$, pero ahora

$$\begin{aligned} y''(x) &= \frac{d}{dx} f(x, y(x)) \\ &= f_x(x, y(x)) + f_y(x, y(x)) y'(x) \\ &= f_x(x, y(x)) + f(x, y(x)) f_y(x, y(x)). \end{aligned}$$

O sea,

$$y(x + h) = y(x) + h T^{(2)}(x, y(x), h) + \frac{h^3}{6} y'''(\zeta),$$



donde hemos puesto

$$T^{(2)}(x, y, h) = f(x, y) + \frac{h}{2}(f_x(x, y) + f(x, y) f_y(x, y)).$$

Por el mismo argumento que antes

$$y(x+h) \sim y(x) + h T^{(2)}(x, y(x), h),$$

lo que lleva al método numérico

$$y_{n+1} = y_n + h T^{(2)}(x_n, y_n, h),$$

que es el llamado **método de TAYLOR de orden 2**.

Como la expresión de $T^{(2)}(x, y, h)$ es un poco larga, escribimos resumidamente

$$T^{(2)} = f + \frac{h}{2}(f_x + f f_y);$$

esta notación abreviada y otras similares que emplearemos hay que entenderlas siempre de esta manera. O sea, lo anterior significa

$$T^{(2)}(x, y, h) = f(x, y) + \frac{h}{2}(f_x(x, y) + f(x, y) f_y(x, y))$$

$$T^{(2)}(x, y(x), h) = f(x, y(x)) + \frac{h}{2}(f_x(x, y(x)) + f(x, y(x)) f_y(x, y(x)))$$

$$T^{(2)}(x_n, y_n, h) = f(x_n, y_n) + \frac{h}{2}(f_x(x_n, y_n) + f(x_n, y_n) f_y(x_n, y_n))$$

etc.

El error local cometido en este último método es

$$T(h) = \frac{h^3}{6} y'''(\zeta) = \mathcal{O}(h^3)$$

y se trata de un método de orden 2, como su propio nombre recuerda.

El método de EULER, que volvíamos a escribir aquí como

$$y_{n+1} = y_n + h T^{(1)}(x_n, y_n, h)$$

cometía un error de

$$T(h) = \frac{h^2}{2} y''(\zeta) = \mathcal{O}(h^2)$$

y era de orden 1.

Siguiendo la misma línea, tenemos el **método de TAYLOR de orden 3**

$$y_{n+1} = y_n + h T^{(3)}(x_n, y_n, h),$$

o, más generalmente, el **método de TAYLOR de orden n**

$$y_{n+1} = y_n + h T^{(n)}(x_n, y_n, h),$$

donde ya las expresiones de $T^{(3)}$ o $T^{(n)}$ son más costosas de obtener. Pero, por su importancia, tendremos que conocer su fórmula, y lo haremos de forma sistemática un poco más adelante.

El principal inconveniente práctico de estos **métodos de TAYLOR** es que precisan del empleo de las derivadas f_x, f_y y otras sucesivas de mayor orden. Estas derivadas no se pueden obtener (sin costo importante) cuando la expresión de $f(x, y)$ es complicada o, peor aún, cuando $f(x, y)$ sólo es conocida en forma numérica, lo que es bastante usual como ya advertimos.

La principal variante que resuelve este problema es la de los llamados métodos de RUNGE-KUTTA que centrarán nuestra atención en lo que sigue.

Métodos de un paso: generalidades

El método de EULER y, más generalmente los métodos de TAYLOR

$$y_{n+1} = y_n + h T^{(1)}(x_n, y_n, h) = y_n + h f(x_n, y_n),$$

$$y_{n+1} = y_n + h T^{(n)}(x_n, y_n, h),$$

son ejemplos de la fórmula general

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

de los que conocemos como **métodos explícitos de 1 paso**.

En el primer caso es

$$\Phi(x_n, y_n, h) = T^{(1)}(x_n, y_n, h) = f(x_n, y_n),$$

y en el segundo,

$$\Phi(x_n, y_n, h) = T^{(2)}(x_n, y_n, h) = f + \frac{h}{2} (f_x + f_y),$$

particularizado para el caso de segundo orden.

Para un método explícito de un paso $y_{n+1} = y_n + h \Phi(x_n, y_n, h)$ se define el **Error local de truncación (LTE)** como

$$T_{n+1}(h) = y(x_{n+1}) - y(x_n) - h \Phi(x_n, y(x_n), h),$$

o mejor, sin los subíndices n , como

$$T(h) = y(x+h) - y(x) - h \Phi(x, y(x), h),$$

para la solución exacta $y(x)$ del problema.

Al desarrollar $T(h)/h$ en potencias de h

$$\frac{T(h)}{h} = \frac{y(x+h) - y(x)}{h} - \Phi(x, y(x), h),$$

interviene el término que vamos a denotar por $\Delta(x, y(x), h)$

$$\Delta(x, y(x), h) = \frac{y(x+h) - y(x)}{h}$$

que es un desarrollo abstracto para una función no concreta $y(x)$ con primeros términos

$$\Delta(x, y(x), h) = y'(x) + \frac{h}{2} y''(x) + \frac{h^2}{6} y'''(x) + \dots$$

El término general de esta serie es

$$\frac{h^{k-1}}{k!} y^{(k)}(x).$$

Aparece también $\Phi(x, y(x), h)$ que es la parte que depende del método. Nosotros escribiremos $\Delta(h)$ y $\Phi(h)$ ya que nuestro interés está en ver su funcionamiento al variar h .

Con esa notación

$$\frac{T(h)}{h} = \Delta(x, y(x), h) - \Phi(x, y(x), h) = \Delta(h) - \Phi(h).$$

Recordemos que la **consistencia** de un método es, o que un método es **consistente** cuando

$$\lim_{h \rightarrow 0} \frac{T(h)}{h} = 0$$

Como ya vimos, eso quiere decir que, para los métodos consistentes, el desarrollo de $T(h)$ en potencias de h comienza en h^2 , y el de $T(h)/h$ en $h^1 = h$. Es decir que, para un método consistente,

$$\frac{T(h)}{h} = \Delta(h) - \Phi(h) = \mathcal{O}(h)$$

al menos, o sea, que al menos tiene orden 1.

Generalizando esto, si el desarrollo de $T(h)/h$ en potencias de h comienza en h^p (o $T(h)$ en h^{p+1} , que es lo mismo) diremos que el método tiene **orden p** .

Esto quiere decir que

$$\frac{T(h)}{h} = \Delta(h) - \Phi(h) = \mathcal{O}(h^p)$$

(sin que sea cierto para valores mayores que p) o que

$$T(h) = \mathcal{O}(h^{p+1}).$$

También equivale a decir que

$$\lim_{h \rightarrow 0} \frac{T(h)}{h^p} = 0 \quad \text{y} \quad \lim_{h \rightarrow 0} \frac{T(h)}{h^{p+1}} \neq 0.$$

El método de EULER hacía

$$\Phi(x, y, h) = f(x, y)$$

es decir, la diferencia

$$\frac{T(h)}{h} = \Delta(h) - \Phi(h)$$

clausuraba el término $y'(x) = f(x, y(x))$ de orden h^0 de $\Delta(h)$, hacía que $T(h)/h$ comenzase por $\frac{h}{2} y''(x)$ o sea que $T(h)/h = \mathcal{O}(h)$ y así resultaba ser de orden 1.

Lo que sigue es, por ejemplo, intentar que $\Phi(h)$ reproduzca ahora

$$f + \frac{h}{2} y''(x) = f + \frac{h}{2} (f_x + f f_y),$$

que son los términos en h^0 y h^1 de $\Delta(h)$.

Hemos visto que esto es posible hacerlo tomando

$\Phi(x, y(x), h) = T^{(2)}(x, y(x), h)$, o sea tomando el método de TAYLOR de orden 2.



Pero no nos interesa hacerlo de esta manera (tomando derivadas de f), sino usando valores de la propia f . Y es lo que sigue a continuación, los llamados 'métodos de RUNGE-KUTTA'.

Empezaremos estudiando los métodos explícitos de esta familia, que son los más fáciles de usar, dejando para más adelante métodos similares que no son explícitos.



Métodos explícitos de RUNGE-KUTTA

Cada método de RUNGE-KUTTA posee un determinado número de 'etapas', que denotaremos de momento por q , y que coincide con el número de evaluaciones de la función f necesarias para pasar de y_n a y_{n+1} . El más elemental y primero de estos métodos es el de EULER, que pasa de y_n a y_{n+1} haciendo

$$y_{n+1} = y_n + h f(x_n, y_n),$$

con una simple evaluación de f . Emparentando este método con los que presentamos a continuación, escribiremos

$$k_1 = f(x_n, y_n),$$

y

$$y_{n+1} = y_n + h k_1.$$

Pero de manera más general, un **método de RUNGE-KUTTA explícito de q etapas** comienza en cada paso (de x_n a x_{n+1}) realizando q evaluaciones de f que describiremos como

$$k_1 = f(x_n, y_n),$$

$$k_2 = f(x_n + c_2 h, y_n + a_{2,1} h k_1),$$

$$k_3 = f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2),$$

$$\vdots$$

$$k_q = f(x_n + c_q h, y_n + a_{q,1} h k_1 + a_{q,2} h k_2 + \cdots + a_{q,q-1} h k_{q-1}),$$

en puntos cuyas abscisas son creadas por coeficientes o parámetros c_2, c_3, \dots, c_q desplazando la abscisa inicial x_n y cuyas ordenadas son creadas por coeficientes

$$\begin{array}{cccc} a_{2,1} & & & \\ a_{3,1} & a_{3,2} & & \\ \vdots & \vdots & \ddots & \\ a_{q,1} & a_{q,2} & \cdots & a_{q,q-1} \end{array}$$

que multiplican a h y a los anteriores valores de los k_i calculados.

Finalmente, el valor de y_{n+1} se obtiene como

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + \cdots + b_q k_q).$$

El valor de $f(x_n, y_n)$ que aparecía en el método de EULER se substituye entonces por una media

$$b_1 k_1 + b_2 k_2 + \cdots + b_q k_q$$

de las q evaluaciones k_1, k_2, \dots, k_q obtenidas previamente.

Resumiendo el proceso,

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1), \\ k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2), \\ &\vdots \\ k_q &= f(x_n + c_q h, y_n + a_{q,1} h k_1 + a_{q,2} h k_2 + \cdots + a_{q,q-1} h k_{q-1}), \end{aligned}$$

y

$$y_{n+1} = y_n + h (b_1 k_1 + b_2 k_2 + \cdots + b_q k_q).$$

Los métodos descritos son de la forma

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h),$$

donde ahora la función $\Phi(h) = \Phi(x_n, y_n, h)$ viene dada por

$$\Phi(h) = b_1 k_1 + b_2 k_2 + \cdots + b_q k_q$$

o bien

$$\Phi(h) = b_1 k_1(h) + b_2 k_2(h) + \cdots + b_q k_q(h),$$

con los

$$k_i(h) = f(x_n + c_i h, y_n + a_{i,1} h k_1 + a_{i,2} h k_2 + \cdots + a_{i,i-1} h k_{i-1})$$

Cuando se necesite escribir $\Phi(x, y, h)$ en vez de $\Phi(h) = \Phi(x_n, y_n, h)$ escribiremos

$$k_i = f(x + c_i h, y + a_{i,1} h k_1 + a_{i,2} h k_2 + \cdots + a_{i,i-1} h k_{i-1}),$$

Ya hemos explicado por qué las evaluaciones k_i deben ser consideradas las causantes de los cálculos más costosos para el método y, por lo tanto, se debe considerar el número de evaluaciones como la penalización que lleva el método para obtener su objetivo, que será el de conseguir orden de consistencia mayor de 1.

Ahora que tenemos métodos que usan únicamente valores de f (y no de sus derivadas), nuestro objetivo es hacer que $\Phi(h)$ reproduzca los primeros términos en $\Delta(h)$ hasta lograr que

$$\frac{T(h)}{h} = \Delta(h) - \Phi(h) = \mathcal{O}(h^p)$$

para el entero p más grande posible. La relación es, como veremos, algo irregular. La cuestión se puede plantear de dos maneras:

- 1 dado un número q de etapas, ¿qué orden puede alcanzarse?; y
- 2 dado un orden p , ¿que número mínimo de etapas permite alcanzarlo?

Vamos a ir respondiendo (aunque lentamente) a ambas cuestiones.

Métodos de RUNGE-KUTTA explícitos de 2 etapas

Escrito en nuestra forma actual de presentar las etapas, el método de EULER o sus análogos de 1 etapa se escriben

$$k_1 = f(x_n, y_n),$$

$$y_{n+1} = y_n + h b_1 k_1.$$

y

Ahora

$$\Phi(h) = b_1 k_1(h).$$

Es claro que la única opción para conseguir orden 1 es hacer $b_1 = 1$, o sea,

$$\Phi(h) = k_1(h) = f(x, y).$$

Observamos que no hacemos consideraciones sobre f y es que las propiedades que obtengamos deben ser ciertas cuando f es 'abstracta' o 'cualquiera', con la sola condición de proporcionar un problema de los que llamamos problema tipo.



Podemos también intentar alcanzar orden 2. Teniendo en cuenta que

$$\Delta(x, y(x), h) = f + \frac{h}{2} (f_x + f f_y) + \mathcal{O}(h^2),$$

necesitaríamos con $\Phi(h) = k_1(h) = f(x, y)$ reproducir esos términos, salvo una diferencia de $\mathcal{O}(h^2)$, lo que es claramente imposible. Mejor resultado es el que vamos a obtener permitiendo emplear una etapa más en cada paso.

Escribimos el método de RUNGE-KUTTA explícito general de 2 etapas:

$$k_1 = f(x_n, y_n),$$

$$k_2 = f(x_n + c_2 h, y_n + a_{2,1} h k_1)$$

$$y_{n+1} = y_n + h (b_1 k_1 + b_2 k_2),$$

que depende de los parámetros c_2 , $a_{2,1}$, b_1 y b_2 (o sea, son infinitos métodos diferentes con esta descripción).



Ahora,

$$\Phi(h) = b_1 k_1 + b_2 k_2 \quad \text{o bien} \quad \Phi(h) = b_1 k_1(h) + b_2 k_2(h).$$

Como

$$\Delta(h) = f + \frac{h}{2} (f_x + f f_y) + \mathcal{O}(h^2),$$

necesitaríamos que

$$\Phi(h) = f + \frac{h}{2} (f_x + f f_y) + \mathcal{O}(h^2).$$

Procedemos al cálculo de $\Phi(h)$ y lo hacemos con el mayor orden posible, ya que las expresiones empiezan a ser algo largas.

Para $\Phi(h)$ es cierto el desarrollo

$$\Phi(h) = \Phi(0) + h \Phi'(0) + \mathcal{O}(h^2).$$

$\Phi(0)$ es

$$\Phi(0) = b_1 k_1(0) + b_2 k_2(0)$$

y $\Phi'(0)$ es

$$\Phi'(0) = b_1 k_1'(0) + b_2 k_2'(0),$$

lo que nos lleva a preocuparnos por obtener, por este orden, $k_1(h)$, $k_2(h)$, $k_1(0)$, $k_2(0)$, $k_1'(h)$, $k_2'(h)$, $k_1'(0)$ y $k_2'(0)$.

Como ya hemos dicho, f y sus derivadas se entenderán en (x_n, y_n) cuando no se indique nada en contrario; ahora bien, pondremos $f(x_n + c_2 h, y_n + a_{2,1} h k_1)$ o cosas similares cuando se trate de evaluarlas en otros puntos.

$$k_1(h) = f$$

$$k_2(h) = f(x_n + c_2 h, y_n + a_{2,1} h k_1);$$

para $h = 0$ el punto $(x_n + c_2 h, y_n + a_{2,1} h k_1)$ se transforma en (x_n, y_n) , luego

$$k_1(0) = f$$

$$k_2(0) = f;$$

$\Phi(0)$ vale entonces

$$\Phi(0) = b_1 k_1(0) + b_2 k_2(0) = (b_1 + b_2) f.$$

Prosiguiendo de forma análoga

$$k_1'(h) = 0$$

$$k_2'(h) = c_2 f_x(x_n + c_2 h, y_n + a_{2,1} h k_1) + a_{2,1} (k_1 + h k_1') f_y(x_n + c_2 h, y_n + a_{2,1} h k_1);$$

(empleamos repetidamente la regla de la cadena

$$\frac{d g(x, y(x))}{dx} = g_x(x, y(x)) \cdot 1 + g_y(x, y(x)) \frac{d y(x)}{dx}$$

teniendo en cuenta que $dx/dx = 1$);

para $h = 0$ sucede ahora

$$k_1'(0) = 0$$

$$k_2'(0) = c_2 f_x + a_{2,1} f f_y;$$

y $\Phi'(0)$ vale

$$\Phi'(0) = b_1 k_1'(0) + b_2 k_2'(0) = b_2 c_2 f_x + b_2 a_{2,1} f f_y.$$

Finalmente, hemos obtenido

$$\Phi(h) = (b_1 + b_2) f + h b_2 c_2 f_x + h b_2 a_{2,1} f f_y + \mathcal{O}(h^2).$$

Para lograr que

$$\Delta(h) - \Phi(h) = \mathcal{O}(h^2)$$

para f genérica, lo que hay que hacer es que

$$\begin{aligned} b_1 + b_2 &= 1 \\ b_2 c_2 &= 1/2 \\ b_2 a_{2,1} &= 1/2. \end{aligned}$$

Existen infinitas soluciones posibles; para describirlas podemos tomar $b_2 = \beta \neq 0$ (necesario para que puedan cumplirse las dos últimas igualdades) y hacer

$$\begin{aligned} b_1 &= 1 - \beta \\ b_2 &= \beta \\ c_2 &= \frac{1}{2\beta} \\ a_{2,1} &= \frac{1}{2\beta}. \end{aligned}$$

O sea, los métodos de RUNGE-KUTTA explícitos de 2 etapas y orden 2 son

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f\left(x_n + \frac{1}{2\beta} h, y_n + a_{2,1} \frac{1}{2\beta} k_1\right) \end{aligned}$$

$$y_{n+1} = y_n + h((1 - \beta) k_1 + \beta k_2),$$

(infinitos métodos diferentes descritos por un único parámetro $\beta \neq 0$).

Si se aspira a lograr el orden 3 con estos método de 2 etapas, hay que tomar ahora

$$\Delta(h) = y'(x) + \frac{h}{2} y''(x) + \frac{h^2}{6} y'''(x) + \mathcal{O}(h^3).$$

Pero ¿qué es $y'''(x)$? Naturalmente, podemos hacer lo mismo que hicimos con $y''(x)$, derivando ahora

$y''(x) = f_x(x, y(x)) + f(x, y(x)) f_y(x, y(x))$ respecto de x (como siempre, regla de la cadena, ...). Acaba siendo tedioso y difícil de enmarcar en una hoja de papel.

El desarrollo de $\Delta(x, y(x), h)$ en función de f y de sus derivadas parciales es ya medianamente grande y aumentará de forma explosiva en cuanto aumentemos el orden que requerimos. Hacemos un alto para dotarnos de una notación y algún procedimiento que permitan mantener las cosas en tamaños adecuados para su lectura.

La notación $f^{(k)}(x, y)$ para $y^{(k+1)}(x)$

Ya hemos dicho que

$$y'(x) = f(x, y(x))$$

y que

$$y''(x) = f_x(x, y(x)) + f(x, y(x)) f_y(x, y(x));$$

Por el mismo procedimiento de derivar f y sus derivada respecto de x , podríamos conseguir

$$y'''(x) = f_{xx}(x, y(x)) + 2 f(x, y(x)) f_{xy}(x, y(x)) + \dots,$$

pero representa ya un trabajo considerable.

Resulta más práctico usar una nueva notación $f^{(k)}$, tanto en la forma $f^{(k)}(x, y)$ como en la $f^{(k)}(x, y(x))$, consiguiendo que

$$y^{(k+1)}(x) = f^{(k)}(x, y(x)), \quad k = 0, 1, 2, \dots$$

Para ello definimos por recurrencia

$$\begin{aligned} f^{(0)}(x, y) &= f(x, y), \\ f^{(k+1)}(x, y) &= f_x^{(k)}(x, y) + f(x, y) f_y^{(k)}(x, y). \end{aligned} \quad (0)$$

Nótese que, de hecho, $f^{(k+1)}(x, y)$ se calcula poniendo

$$f^{(k+1)}(x, y(x)) = \frac{d f^{(k)}(x, y(x))}{dx}$$

y reemplazando finalmente $y(x)$ por y en la expresión final. Con esto que acabamos de decir, no es difícil probar por recurrencia que

$$y^{(k+1)}(x) = f^{(k)}(x, y(x)), \quad k = 0, 1, 2, \dots$$

con lo que (no escribimos (x, y) para hacerlo más corto)

$$\Delta(h) = \Delta(x, y, h) = f^{(0)} + \frac{h}{2} f^{(1)} + \frac{h^2}{6} f^{(2)} + \dots + \frac{h^p}{p+1!} f^{(p)} + \mathcal{O}(h^{p+1}).$$



Por ejemplo, ya sabemos que

$$f^{(0)}(x, y) = f(x, y).$$

De este obtenemos

$$\begin{aligned} f^{(1)}(x, y) &= f_x^{(0)}(x, y) + f(x, y) f_y^{(0)}(x, y) \\ &= f_x(x, y) + f(x, y) f_y(x, y), \end{aligned}$$

o sea, abreviadamente,

$$f^{(1)} = f_x + f f_y.$$



La notación $f^{(k)}$ que hace recordar a la derivada número k de f no es en absoluto desproporcionada. De hecho se trata de las derivadas ordinaria sucesivas de $f(x, y)$, pero considerada como $f(x, y(x))$ y derivando entonces respecto de la única variable x .

Un segundo aspecto se encuentra en organizar la cancelación de términos entre $\Delta(h)$ y $\Phi(h)$. Como

$$\begin{aligned}\Delta(h) &= f^{(0)} + \frac{h}{2} f^{(1)} + \frac{h^2}{6} f^{(2)} + \dots \\ \Phi(h) &= \Phi(0) + h \Phi'(0) + \frac{h^2}{2} \Phi''(0) + \dots,\end{aligned}$$

obtenemos de inmediato el siguiente resultado que mejora considerablemente la búsqueda del orden alcanzable por estos métodos:

Los métodos que estamos estudiando son de orden al menos 1 cuando

$$\Phi(0) = f^{(0)},$$

de orden 2 cuando son de orden 1 y, además,

$$\Phi'(0) = \frac{1}{2} f^{(1)},$$

y, más generalmente, de orden p cuando lo son de orden $p - 1$ y, además,

$$\Phi^{(p-1)}(0) = \frac{1}{p} f^{(p-1)}.$$

Por lo tanto, para organizarse bien en el estudio del orden, basta ir calculando los $\Phi^k(0)$ usando los valores calculados de $k_1^k(0)$, $k_2^k(0)$, etc. y, además, tener una lista de los valores de los f^k . Ahora bien, estos últimos no dependen del método, son genéricos para f , x e y abstractos, y se pueden tener calculados de una vez para siempre, con independencia del método con el cual se usen luego.

De hecho ya tenemos calculados de antes $f^{(0)}(x, y)$ y $f^{(1)}(x, y)$. En cuanto los necesitemos por primera vez, dejaremos calculados algunos de los siguientes, que van siendo más complejos.

En la actividad de diseño de los métodos del tipo que estamos estudiando, aparecen evaluaciones de f de la forma

$$k_1 = f(x_n, y_n),$$

$$k_2 = f(x_n + c_2 h, y_n + a_{2,1} h k_1),$$

$$k_3 = f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2),$$

etc. Los puntos de evaluación, (x_n, y_n) , $(x_n + c_2 h, y_n + a_{2,1} h k_1)$, $(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2)$, etc. acaban siendo encumbrantes en las fórmulas. Vamos a reemplazarlos por los breves símbolos

$$(1) \quad \text{para} \quad (x_n, y_n),$$

$$(2) \quad \text{para} \quad (x_n + c_2 h, y_n + a_{2,1} h k_1),$$

$$(3) \quad \text{para} \quad (x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2),$$

etc. y, además, el punto (1) = $f(x_n, y_n)$ lo omitiremos generalmente.

Únicamente habrá que recordar siempre que la notación resumida (2), (3), etc. representa puntos que dependen de la variable h , que es la que interviene decisivamente a la hora de derivar, por ejemplo.

Por el momento, el último aspecto organizativo que trataremos es el llamado **Tablero de BUTCHER** para los métodos de RUNGE-KUTTA. Ya hemos visto que, al describir uno de estos métodos, lo único 'novedoso' son los parámetros (abstractos o concretos) que intervienen. Entonces, describiremos un método poniendo simplemente los parámetros en una matriz o tablero en la forma siguiente

$$\begin{array}{c|c} [c_i] & [a_{ij}] \\ \hline & [b_j] \end{array}$$

$[c_i]$ es un vector 'vertical' y $[b_j]$ es 'horizontal'.

Por ejemplo, para el método genérico de RUNGE-KUTTA explícito de dos etapas, tendríamos

$$\begin{array}{c|cc} 0 & & \\ c_2 & a_{21} & \\ \hline & b_1 & b_2 \end{array}$$

donde los 'posibles' elementos a_{11} , a_{12} , a_{22} no se ponen porque no existen, o son ceros, como queramos interpretarlo, y donde el 0 puesto en la posición c_1 a veces se omite también.

Para ver la eficacia de esta organización, vamos a repetir brevemente el cálculo de aquellos métodos de 2 etapas que proporcionaban orden 2. En principio teníamos los métodos

$$\begin{aligned}k_1 &= f(x_n, y_n), \\k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1)\end{aligned}$$

$$y_{n+1} = y_n + h (b_1 k_1 + b_2 k_2),$$

correspondientes al tablero de BUTCHER que acabamos de escribir. Recordemos que k_1 no depende de h y que

$$\Phi(h) = b_1 k_1 + b_2 k_2(h),$$

lo que nos lleva a

$$\Phi(h) = b_1 k_1 + b_2 k_2(h), \quad \Phi(0) = b_1 k_1 + b_2 k_2(0),$$

$$\Phi'(h) = b_2 k_2'(h), \quad \Phi'(0) = b_2 k_2'(0),$$

y fórmulas similares que, por el momento no necesitaremos.

$$\begin{aligned}k_1(h) &= \text{cte} = f \\k_2(h) &= f(2); \end{aligned}$$

y

$$k_2(0) = f,$$

(recuérdese que (2) se transforma en (1), que no escribimos, cuando $h = 0$) por lo que

$$\Phi(0) = b_1 k_1 + b_2 k_2(0) = (b_1 + b_2) f.$$

Como $f^{(0)} = f$, la única posibilidad (para f genérica) de que $\Phi(0) = f^{(0)}$ es que

$$(1a) \quad b_1 + b_2 = 1$$

Además, $k_1'(h) = 0$, mientras que

$$k_2'(h) = c_2 f_x(2) + a_{2,1} (k_1 + h k_1') f_y(2) = c_2 f_x(2) + a_{2,1} k_1 f_y(2),$$

con lo que (de nuevo (2) se transforma en (1), que no escribimos, cuando $h = 0$)

$$k_2'(0) = c_2 f_x + a_{2,1} f f_y$$

y

$$\Phi'(0) = b_2 k_2'(0) = b_2 c_2 f_x + b_2 a_{2,1} f f_y.$$

Por su parte,

$$\frac{1}{2} f^{(1)} = \frac{1}{2} f_x + \frac{1}{2} f f_y.$$



Luego la única posibilidad (para f genérica) de que $\Phi'(0) = \frac{1}{2} f^{(1)}$ es que se cumplan las dos condiciones

$$(2a) \quad b_2 c_2 = 1/2$$

$$(2b) \quad b_2 a_{2,1} = 1/2.$$

que, para orden 2 tenemos que añadir a la que ya teníamos del orden 1. Y eso nos lleva a la descripción que ya vimos para estos métodos

$$\begin{aligned} b_1 &= 1 - \beta \\ b_2 &= \beta \\ c_2 &= \frac{1}{2\beta} \\ a_{2,1} &= \frac{1}{2\beta}, \end{aligned}$$

con $\beta \neq 0$ arbitrario.



Métodos de RUNGE-KUTTA explícitos de 2 etapas

Entre los infinitos métodos de 2 etapas y orden 2 que acabamos de describir, tal vez encontremos alguno de orden 3. Anticipamos ya que la respuesta va a ser negativa, pero veamos como se llega a ella. Nuestro objetivo ahora es encontrar si existen valores de los parámetros c_2 , $a_{2,1}$, b_1 y b_2 para los que $\Phi''(0) = \frac{1}{2} f^{(2)}$. Comencemos viendo cuál es la expresión genérica de $f^{(2)}$. Es ahora muy asequible con la regla inductiva que hemos proporcionado:

$$\begin{aligned} f^{(2)} &= f_x^{(1)} + f f_y^{(1)} \\ &= (f_x + f f_y)_x + f (f_x + f f_y)_y \\ &= f_{xx} + f_x f_y + f f_{xy} + f (f_{xy} + (f_y)^2) + f f_{yy} \\ &= f_{xx} + 2 f f_{xy} + (f)^2 f_{yy} + f_x f_y + f (f_y)^2. \end{aligned}$$

En definitiva,

$$f^{(2)} = f_{xx} + 2 f f_{xy} + (f)^2 f_{yy} + f_x f_y + f (f_y)^2,$$

con evaluación en (x, y) , $(x, y(x))$ o (x_n, y_n) .

(Obsérvese que hemos puesto en práctica nuestra hipótesis de que la regularidad de f es la que necesitamos para nuestros manejos, lo que nos ha llevado a aceptar que $f_{yx}(x, y) = f_{xy}(x, y)$ lo que precisa de la continuidad de las derivadas parciales que intervienen.)

Esta expresión no depende de los métodos, sino únicamente de la estructura del problema general que estamos resolviendo. Por lo tanto, no va a variar, y, una vez calculada, la almacenaremos (con las ya conocidas de $f^{(0)} = f$ y $f^{(1)} = f_x + f f_y$ para usarlas cuantas veces sean necesarias.

Los términos f , f_x , $f f_y$, f_{xx} , $f f_{xy}$, $(f)^2 f_{yy}$, $f_x f_y$, $f (f_y)^2$ y otros posteriores con derivadas parciales de f de orden más alto que irán apareciendo, suelen recibir el nombre de 'diferenciales elementales'. Obsérvese que son expresiones abstractas que no serán concretas en el uso que hagamos de ellas, que será el desarrollo de métodos. Lo siguiente es calcular lo que vale para nuestro método o métodos $\Phi''(0)$, como antes a partir del valor de $k_2''(0)$, teniendo en cuenta que

$$\Phi''(0) = b_2 k_2''(0).$$

Como

$$k_2'(h) = c_2 f_x(2) + a_{2,1} k_1 f_y(2),$$

resulta

$$k_2''(h) = (c_2)^2 f_{xx}(2) + 2 c_2 a_{2,1} k_1 f_{xy}(2) + (a_{2,1})^2 (k_1)^2 f_{yy}(2),$$

Ahora,

$$k_2''(0) = (c_2)^2 f_{xx} + 2 c_2 a_{2,1} f f_{xy} + (a_{2,1})^2 (f)^2 f_{yy},$$

y

$$\Phi''(0) = b_2 (c_2)^2 f_{xx} + 2 b_2 c_2 a_{2,1} f f_{xy} + b_2 (a_{2,1})^2 (f)^2 f_{yy}.$$

Para los métodos de los que nos ocupamos, que son los que ya poseen orden 2, se tiene

$$\Phi''(0) = \frac{1}{4\beta} f_{xx} + \frac{2}{4\beta} f f_{xy} + \frac{1}{4\beta} (f)^2 f_{yy},$$

y la igualdad con

$$\frac{1}{3} f^{(2)} = \frac{1}{3} f_{xx} + \frac{2}{3} f f_{xy} + \frac{1}{3} (f)^2 f_{yy} + \frac{1}{3} f_x f_y + \frac{1}{3} f (f_y)^2$$

es obviamente imposible. Por lo tanto:

2 es el mayor orden que se puede obtener con estos métodos de RUNGE-KUTTA de dos etapas.

Pero el estudio no tiene por qué quedar ahí. Es posible analizar el comportamiento del error de estos (infinitos) métodos de dos etapas y orden 2. Para ello, compararemos el término principal del error en esos métodos. Su valor es el del primer término no nulo en el desarrollo de $T(h)/h = \Delta(h) - \Phi(h)$, que es

$$\frac{h^2}{6} f^{(2)} - \frac{h^2}{2} \Phi''(0) = \frac{h^2}{2} \left(\frac{1}{3} f^{(2)} - \Phi''(0) \right)$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↻ 🔍

pudiendo limitarnos a la comparación de $(1/3) f^{(2)} - \Phi''(0)$, ya que la parte $h^2/2$ es un factor positivo común a todos los métodos.

Restando las expresiones que ya hemos calculado

$$\begin{aligned} \frac{1}{3} f^{(2)} - \Phi''(0) &= \\ &= \left(\frac{1}{3} - \frac{1}{4\beta} \right) f_{xx} + 2 \left(\frac{1}{3} - \frac{1}{4\beta} \right) f f_{xy} + \\ &\quad + \left(\frac{1}{3} - \frac{1}{4\beta} \right) (f)^2 f_{yy} + \frac{1}{3} f_x f_y + \frac{1}{3} f (f_y)^2, \end{aligned}$$

o de manera más clara

$$\left[\left(\frac{1}{3} - \frac{1}{4\beta} \right), 2 \left(\frac{1}{3} - \frac{1}{4\beta} \right), \left(\frac{1}{3} - \frac{1}{4\beta} \right), \frac{1}{3}, \frac{1}{3} \right] \begin{bmatrix} f_{xx} \\ f f_{xy} \\ (f)^2 f_{yy} \\ f_x f_y \\ f (f_y)^2 \end{bmatrix}.$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↻ 🔍

Aunque en general lo que tendremos que hacer en casos como este es minimizar los vectores como

$$\left[\left(\frac{1}{3} - \frac{1}{4\beta} \right), 2 \left(\frac{1}{3} - \frac{1}{4\beta} \right), \left(\frac{1}{3} - \frac{1}{4\beta} \right), \frac{1}{3}, \frac{1}{3} \right]$$

respecto de alguna norma (frecuentemente para la norma-2 o euclídea), en este caso la cosa es bastante fácil. El valor

$$\beta = 3/4$$

convierte el vector en

$$\left[0, 0, 0, \frac{1}{3}, \frac{1}{3} \right]$$

que es lo más pequeño que puede hacerse, independientemente de la norma con que se mida. Daremos enseguida un nombre a este método, que es el que 'minimiza el error' de entre los que alcanzan orden 2 con 2 etapas.

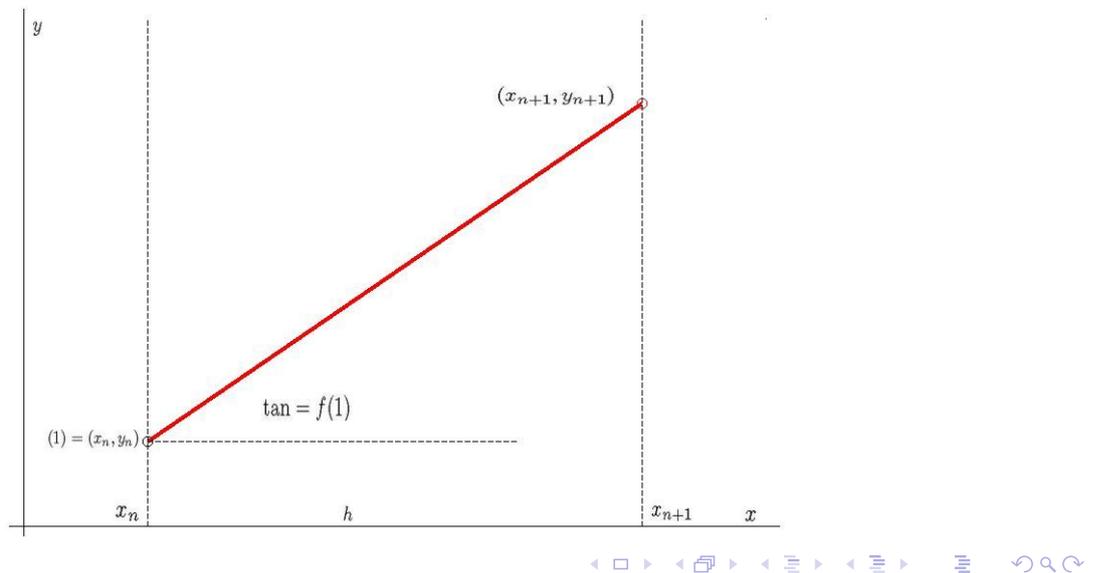
De entre la infinidad de métodos de 2 etapas y orden 2 que acabamos de ver, algunos tienen o han tenido una aceptación especial. Por supuesto este último que minimiza el error, pero también otros debido a la simplicidad de sus parámetros. Así, un parámetro igual a 0 o a 1 evita las correspondientes multiplicaciones. Además, parámetros como 1/2 facilitaban las operaciones en los momentos iniciales de esta teoría, cuando no se soñaba siquiera con el uso de ordenadores como es norma en la etapa actual.

Estos métodos que presentamos brevemente y que poseen un 'nombre propio' tradicional nos van a permitir, además, una interpretación intuitiva o gráfica del método, interpretación que es de hecho común para todos los métodos de RUNGE-KUTTA y, de una forma más sencilla, para los de 1 y 2 etapas.

El método de EULER

$$y_{n+1} = y_n + h f(x_n, y_n)$$

proporciona el valor de y_{n+1} como la ordenada del corte de la recta $x = x_{n+1}$ con la recta que pasa por (x_n, y_n) y tiene por pendiente $f(x_n, y_n)$. Es lo que trata de explicar la figura siguiente



El 'método modificado de EULER'

El 'método modificado de EULER' es el que se obtiene de la anterior familia de orden 2 cuando se toma $\beta = 1/2$. Posee el tablero

0		
1	1	
	1/2	1/2

No es difícil describirlo en una sola línea como

$$y_{n+1} = y_n + \frac{h}{2} (f(x_n, y_n) + f(x_{n+1}, y_n + h f(x_n, y_n))) ,$$

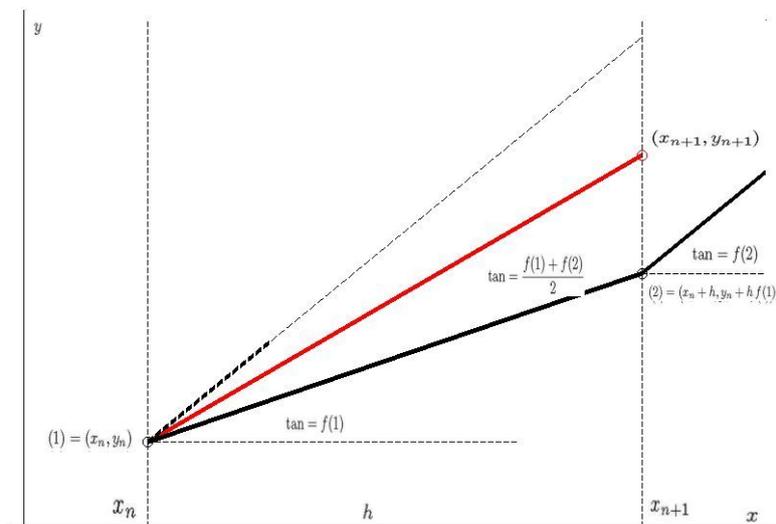
si bien resulta bastante inútil, siendo mejor la forma clásica

$$\begin{aligned}
 k_1 &= f(x_n, y_n), \\
 k_2 &= f(x_n + h, y_n + h k_1) \\
 y_{n+1} &= y_n + \frac{h}{2}(k_1 + k_2),
 \end{aligned}$$

que lo presenta tal y como conviene programarlo.

Este método proporciona el valor de y_{n+1} como la ordenada del corte de la recta $x = x_{n+1}$ con la recta que pasa por (x_n, y_n) y tiene por pendiente una media de dos valores de f : los que en su momento llamamos con $f(1)$ y $f(2)$ ($f(1) = f(x_n, y_n)$ y $f(2) = f(x_{n+1}, y_n + h k_1)$).

El punto (1) es el (x_n, y_n) (el de EULER). El (2) es el que el método de EULER tomaría como y_{n+1} , que ahora se usa sólo para buscar una segunda evaluación de f . Es lo que trata de explicar la figura siguiente



El 'método del punto medio'

El 'método del punto medio' es el que se obtiene de la anterior familia de orden 2 cuando se toma $\beta = 1$. Posee el tablero

0	
1/2	1/2
	0 1

O sea, es

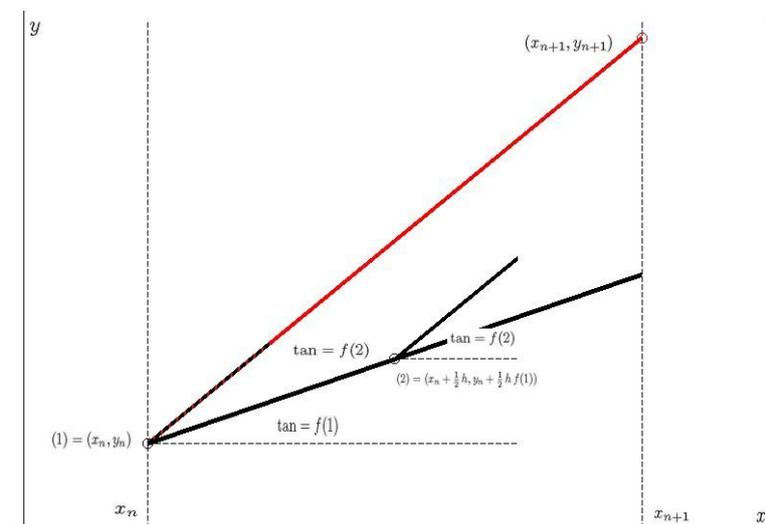
$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right) \\ y_{n+1} &= y_n + h k_2, \end{aligned}$$

o, en una sola línea,

$$y_{n+1} = y_n + h f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} f(x_n, y_n)\right),$$



Naturalmente, el significado intuitivo gráfico es similar al explicado en el caso precedente. Este método proporciona el valor de y_{n+1} como la ordenada del corte de la recta $x = x_{n+1}$ con la recta que pasa por (x_n, y_n) y tiene por pendiente el valor de $f(2)$. Este punto $(2) = f(x_n + \frac{1}{2} h, y_n + \frac{1}{2} h f(1))$ es el punto medio entre $(1) = (x_n, y_n)$ y el punto que el método de EULER tomaría como y_{n+1} , lo que se deja ver en la figura:



El 'método de HEUN'

El 'método de HEUN' toma $\beta = 3/4$. Posee el tablero

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

y se escribe

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f\left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}h k_1\right) \\ y_{n+1} &= y_n + \frac{h}{4}(k_1 + 3k_2). \end{aligned}$$

Es, de entre los métodos de 2 etapas y orden 2 el que minimiza el error, como vimos.

También se interpreta gráficamente en forma parecida a los anteriores casos, pero no insistiremos otra vez en lo mismo.

Métodos de RUNGE-KUTTA explícitos de 3 etapas

Vamos a estudiar ahora los métodos de RUNGE-KUTTA explícitos de 3 etapas:

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1) \\ k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2) \\ y_{n+1} &= y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3), \end{aligned}$$

cuyo tablero de BUTCHER es

0			
c_2	a_{21}		
c_3	a_{31}	a_{32}	
		b_1	b_2 b_3

y dependen de los parámetros c_2 , c_3 , $a_{2,1}$, $a_{3,1}$, $a_{3,2}$, b_1 , b_2 y b_3 . Siempre es posible alcanzar orden 2; bastaría tomar $b_3 = 0$ y las relaciones ya estudiadas para c_2 , $a_{2,1}$, b_1 y b_2 . Estudiaremos si se pueden alcanzar órdenes mayores.

El valor de $\Phi(h)$ es ahora

$$\Phi(h) = b_1 k_1 + b_2 k_2(h) + b_3 k_3(h)$$

(k_1 no depende de h). Ahora, hay que desarrollar al menos

$$\Delta(h) = f^{(0)} + \frac{h}{2} f^{(1)} + \frac{h^2}{6} f^{(2)} + \mathcal{O}(h^3)$$

y

$$\Phi(h) = \Phi(0) + h \Phi'(0) + \frac{h^2}{2} \Phi''(0) + \mathcal{O}(h^3).$$

Las notaciones abreviadas comienzan ahora por poner

- (1) para (x_n, y_n) ,
- (2) para $(x_n + c_2 h, y_n + a_{2,1} h k_1)$,
- (3) para $(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2)$,

omitiendo normalmente el punto (1) = (x_n, y_n) en las numerosas veces que se presente en la fórmulas.

$$\begin{aligned} k_1(h) &= \text{cte} = f \\ k_2(h) &= f(2) \\ k_3(h) &= f(3); \end{aligned}$$

y

$$\begin{aligned} k_2(0) &= f \\ k_3(0) &= f; \end{aligned}$$

(recuérdese que (2)y(3) se transforman en (1), que no escribimos, cuando $h = 0$) por lo que

$$\Phi(0) = b_1 k_1 + b_2 k_2(0) + b_3 k_3(0) = (b_1 + b_2 + b_3) f.$$

Como $f^{(0)} = f$, la única posibilidad (para f genérica) de que $\Phi(0) = f^{(0)}$ es que

$$(1a) \quad b_1 + b_2 + b_3 = 1$$

Además, $k_1'(h) = 0$, mientras que

$$\begin{aligned} k_2'(h) &= c_2 f_x(2) + a_{2,1} k_1 f_y(2) \\ k_3'(h) &= c_3 f_x(3) + [a_{3,1} k_1 + a_{3,2} k_2 + a_{3,2} h k_2'] f_y(3), \end{aligned}$$

y nos será rentable denotar por $[\]$ la cantidad

$[\] = [a_{3,1} k_1 + a_{3,2} k_2 + a_{3,2} h k_2']$ para futuras operaciones. Lo anterior hace que

$$k_2'(0) = c_2 f_x + a_{2,1} f f_y$$

y, teniendo en cuenta que

$$[\]_{h=0} = (a_{3,1} + a_{3,2}) f,$$

a su vez

$$k_3'(0) = c_3 f_x + (a_{3,1} + a_{3,2}) f f_y.$$



Por lo tanto,

$$\begin{aligned} \Phi'(0) &= b_2 k_2'(0) + b_3 k_3'(0) \\ &= (b_2 c_2 + b_3 c_3) f_x + (b_2 a_{2,1} + b_3 (a_{3,1} + a_{3,2})) f f_y \end{aligned}$$

a comparar para el orden 2 con

$$\frac{1}{2} f^{(1)} = \frac{1}{2} f_x + \frac{1}{2} f f_y.$$

La igualdad, para f genérica se produce cuando

$$(2a) \quad b_2 c_2 + b_3 c_3 = 1/2$$

$$(2b) \quad b_2 a_{2,1} + b_3 (a_{3,1} + a_{3,2}) = 1/2.$$

3 ecuaciones para 8 parámetros nos llevan a pensar que es posible ir más adelante. Buscaremos ir al orden 3, lo que representa añadir

$$\Phi''(0) = (1/3) f^{(2)}.$$



Ahora $k_1''(h) = 0$ mientras que

$$k_2''(h) = (c_2)^2 f_{xx}(2) + 2 c_2 a_{2,1} k_1 f_{xy}(2) + (a_{2,1})^2 (k_1)^2 f_{yy}(2)$$

y, teniendo en cuenta que

$$\frac{d[]}{dh} = 2 a_{3,2} k_2' + a_{3,2} h k_2'',$$

se sigue que

$$k_3''(h) = (c_3)^2 f_{xx}(3) + 2 c_3 [] f_{xy}(3) + []^2 f_{yy}(3) + (2 a_{3,2} k_2' + a_{3,2} h k_2'') f_y(3).$$

Para $h = 0$ estas igualdades se transforman en

$$\begin{aligned} k_2''(0) &= (c_2)^2 f_{xx} + 2 c_2 a_{2,1} f f_{xy} + (a_{2,1})^2 (f)^2 f_{yy} \\ k_3''(0) &= (c_3)^2 f_{xx} + 2 c_3 (a_{3,1} + a_{3,2}) f f_{xy} + (a_{3,1} + a_{3,2})^2 (f)^2 f_{yy} \\ &\quad + 2 c_2 a_{3,2} f_x f_y + 2 a_{2,1} a_{3,2} f_x (f_y)^2, \end{aligned}$$

teniendo en cuenta que los puntos (2) y (3) se transforman en (1), que $[]_{h=0} = (a_{3,1} + a_{3,2}) f$ y que $k_2'(0) = c_2 f_x + a_{2,1} f f_y$. Por eso,

$$\begin{aligned} \Phi''(0) &= (b_2 (c_2)^2 + b_3 (c_3)^2) f_{xx} \\ &\quad + 2 (b_2 c_2 a_{2,1} + b_3 c_3 (a_{3,1} + a_{3,2})) f f_{xy} \\ &\quad + (b_2 (a_{2,1})^2 + b_3 (a_{3,1} + a_{3,2})^2) (f)^2 f_{yy} \\ &\quad + 2 b_3 c_2 a_{3,2} f_x f_y + 2 b_3 a_{2,1} a_{3,2} f (f_y)^2. \end{aligned}$$

La condición para el orden 3 es $\Phi''(0) = (1/3) f^{(2)}$, y el término $f^{(2)}$ ya había sido calculado. Comparando ahora los coeficientes de las diferenciales elementales f_{xx} , f_{xy} , $(f_y)^2$, $f_x f_y$, $f (f_y)^2$ e igualando estos, se obtienen las condiciones

$$\begin{aligned} (3a) \quad & b_2 (c_2)^2 + b_3 (c_3)^2 = 1/3 \\ (3b) \quad & b_2 c_2 a_{2,1} + b_3 c_3 (a_{3,1} + a_{3,2}) = 1/3 \\ (3c) \quad & b_2 (a_{2,1})^2 + b_3 (a_{3,1} + a_{3,2})^2 = 1/3 \\ (3d) \quad & b_3 c_2 a_{3,2} = 1/6 \\ (3e) \quad & b_3 a_{2,1} a_{3,2} = 1/6. \end{aligned}$$

El orden 3 se obtiene cuando se verifican las $1 + 2 + 5 = 8$ condiciones (1a), (2a), (2b), (3a), (3b), (3c), (3d), (3e), lo que no es sencillo de obtener. El empleo de un manipulador simbólico (yo he empleado 'Maple') facilita la consecución ordenada de estas ecuaciones y de su solución. Así, el programa **Ruk3e.mws** se encarga de analizar los métodos de 3 etapas para ecuaciones escalares. Incluyo aquí alguno de sus resultados.

Las soluciones para orden 3 resultan un poco delicadas en cuanto a su descripción. La 'mayor parte', se obtiene de la familia

$$\begin{aligned} (\text{fam1}) \quad c_2 &= \frac{2}{3} \frac{-1 + 3 b_3 c_3^2}{-1 + 2 b_3 c_3} & a_{2,1} &= \frac{2}{3} \frac{-1 + 3 b_3 c_3^2}{-1 + 2 b_3 c_3} \\ a_{3,1} &= \frac{1}{4} \frac{-6 b_3 c_3 + 12 b_3^2 c_3^3 + 1}{b_3 (-1 + 3 b_3 c_3^2)} & a_{3,2} &= \frac{1}{4} \frac{-1 + 2 b_3 c_3}{b_3 (-1 + 3 b_3 c_3^2)} \\ b_1 &= \frac{1}{4} \frac{-1 + 12 b_3 c_3^2 - 12 b_3 c_3 + 4 b_3}{-1 + 3 b_3 c_3^2} & b_2 &= \frac{-3}{4} \frac{(-1 + 2 b_3 c_3)^2}{-1 + 3 b_3 c_3^2}, \end{aligned}$$

está descrita mediante 2 parámetros libres, b_3 y c_3 , que pueden tomar pares de valores arbitrarios, excepto los situados en tres curvas del plano $O_{b_3 c_3}$,

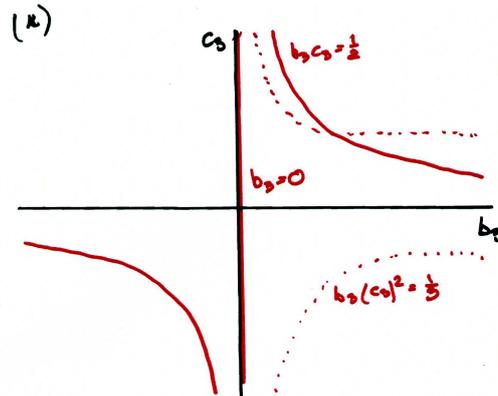
las curvas de ecuaciones

$$(c1) \quad b_3 = 0$$

$$(c2) \quad b_3 c_3 = 1/2$$

$$(c3) \quad b_3 (c_3)^2 = 1/3.$$

que se pintan en la figura que acompaña.



Las restantes soluciones se obtienen forzando a que los parámetros b_3 y c_3 se encuentren sobre alguna de las tres curvas citadas. Para $b_3 = 0$ no se obtiene solución ninguna, y es lógico, ya que eso significaría que los métodos tendrían sólo 2 etapas y ya sabemos que, entonces, el orden 3 no es alcanzable.

Haciendo que b_3 y c_3 se encuentren sobre la curva $b_3 c_3 = 1/2$, se obtienen las soluciones

$$(fam2) \quad a_{2,1} = c_2, \quad c_3 = \frac{2}{3}, \quad a_{3,1} = \frac{2}{9} \frac{3c_2 - 1}{c_2}, \quad a_{3,2} = \frac{2}{9} \frac{1}{c_2},$$

$$b_1 = \frac{1}{4}, \quad b_2 = 0, \quad b_3 = \frac{3}{4},$$

descritas por el parámetro $c_2 \neq 0$.



Finalmente, haciendo que b_3 y c_3 se encuentren sobre la curva $b_3 (c_3)^2 = 1/3$, se obtienen las soluciones

$$\text{(fam3)} \quad c_2 = \frac{2}{9} \frac{1}{a_{3,2}}, \quad a_{21} = \frac{2}{9} \frac{1}{a_{3,2}}, \quad c_3 = \frac{2}{3}, \quad a_{3,1} = \frac{2}{3} - a_{3,2},$$

$$b_1 = \frac{1}{4}, \quad b_2 = 0, \quad b_3 = \frac{3}{4}.$$

descritas por el parámetro $a_{32} \neq 0$.

A continuación veremos que 3 es el máximo orden alcanzable con estos métodos de 3 etapas y, enseguida, analizaremos el comportamiento del error en los casos de orden 3 máximo. Pero lo tendremos que analizar separadamente para cada una de las dos familias propuestas, estudiando en particular qué método de cada familia que hace que se obtenga el mínimo coeficiente del error.

Una consideración que aparece ahora. Todos los métodos de 3 etapas y orden 3, que veremos que es el máximo alcanzable para dicho número de etapas, verifican las propiedades, llamada 'de fila'

$$c_2 = a_{2,1}$$

$$c_3 = a_{3,1} + a_{3,2}.$$

El nombre proviene de la disposición en que aparecen estos elementos en el tablero de BUTCHER. Como el hecho aparecía también en los métodos de dos etapas y orden máximo 2 (allí sólo era $c_2 = a_{2,1}$), parece que esta o estas propiedades de fila aparecen en todos los casos de orden máximo para un número de etapas dado. Y así ocurre, aunque no lo probaremos. Sin embargo, muchos métodos de menor orden no las verifican, como se ve en numerosos ejercicios de manera sencilla.

¿Es 3 el mayor orden que se alcanza con estos métodos de 3 etapas, como hemos afirmado un poco antes? Vamos a ver que es así, y al hacerlo, avanzaremos en el proceso de los desarrollos en serie de $\Delta(h)$ y de $\Phi(h)$. Ahora conviene ir hasta

$$\Delta(h) = f^{(0)} + \frac{h}{2} f^{(1)} + \frac{h^2}{6} f^{(2)} + \frac{h^3}{24} f^{(3)} + \mathcal{O}(h^4)$$

y también hasta

$$\Phi(h) = \Phi(0) + h \Phi'(0) + \frac{h^2}{2} \Phi''(0) + \frac{h^3}{6} \Phi'''(0) + \mathcal{O}(h^4).$$

Y entonces trataremos de ver si

$$\Phi'''(0) = (1/4) f^{(3)}.$$

Pero, para empezar, ¿cuánto vale $f^{(3)}$? Lo haremos como antes, empleando el hecho de que

$$\begin{aligned} f^{(3)} &= f_x^{(2)} + f f_y^{(2)} \\ &= (f_{xx} + 2 f f_{xy} + (f)^2 f_{yy} + f_x f_y + f (f_y)^2)_x \\ &\quad + f (f_{xx} + 2 f f_{xy} + (f)^2 f_{yy} + f_x f_y + f (f_y)^2)_y \\ &= f_{xxx} + 2 f_x f_{xy} + 2 f f_{xxy} + 2 f f_x f_{yy} + (f)^2 f_{xyy} \\ &\quad + f_{xx} f_y + f_x f_{xy} + f_x (f_y)^2 + 2 f f_y f_{xy} \\ &\quad + f (f_{xxy} + 2 f_y f_{xy} + 2 f f_{xyy} + 2 f f_y f_{yy} + (f)^2 f_{yyy} \\ &\quad + f_{xy} f_y + f_x f_{yy} + (f_y)^3 + 2 f f_y f_{yy}) \\ &= f_{xxx} + 3 f f_{xxy} + 3 (f)^2 f_{xyy} + 3 f_x f_{xy} + 5 f f_y f_{xy} \\ &\quad + (f)^3 f_{yyy} + 3 f f_x f_{yy} + 4 (f)^2 f_y f_{yy} + f_{xx} f_y \\ &\quad + f_x (f_y)^2 + f (f_y)^3, \end{aligned}$$

y como se ve, posee un total de 11 términos.

Lo siguiente es buscar los valores de $k_2'''(0)$ y $k_3'''(0)$, necesarios para calcular $\Phi'''(0)$ mediante $\Phi'''(0) = b_2 k_2'''(0) + b_3 k_3'''(0)$. Dejando que el citado programa Ruk33.mws de Maple haga por nosotros estos cálculos bastante laboriosos, nos limitamos a recoger las ecuaciones que proporciona la igualdad

$$\Phi'''(0) = (1/4) f^{(3)}.$$

Se obtiene

$$(4a) \quad b_2 c_2^3 + b_3 c_3^3 = 1/4 \quad \text{para } f_{xxx}$$

$$(4b) \quad b_2 c_2^2 a_{2,1} + b_3 c_3^2 a_{3,1} + b_3 c_3^2 a_{3,2} = 1/4 \quad \text{para } f f_{xyy}$$

$$(4c) \quad b_2 c_2 a_{2,1}^2 + b_3 c_3 a_{3,1}^2 + b_3 c_3 a_{3,2}^2 + 2 b_3 c_3 a_{3,1} a_{3,2} = 1/4$$

para $(f)^2 f_{xyy}$

$$(4d) \quad b_3 c_3 a_{3,2} c_2 = 1/8 \quad \text{para } f_x f_{xy}$$

$$(4e) \quad b_3 c_3 a_{3,2} a_{2,1} + b_3 a_{3,2} c_2 a_{2,1} = 5/24 \quad \text{para } f f_y f_{xy}$$

$$(4f) \quad b_2 a_{2,1}^3 + b_3 a_{3,2}^3 + 3 b_3 a_{3,1}^2 a_{3,2} + 3 b_3 a_{3,1} a_{3,2}^2 + b_3 a_{3,1}^3 = 1/4$$

para $(f)^3 f_{yyy}$

$$(4g) \quad b_3 a_{3,2} c_2 a_{3,1} + b_3 a_{3,2}^2 c_2 = 1/8 \quad \text{para } f f_x f_{yy}$$

$$(4h) \quad 2 b_3 a_{3,2} a_{2,1} a_{3,1} + 2 b_3 a_{3,2}^2 a_{2,1} + b_3 a_{3,2} a_{2,1}^2 = 1/3$$

para $(f)^2 f_y f_{yy}$

$$(4i) \quad b_3 a_{3,2} c_2^2 = 1/12 \quad \text{para } f_{xx} f_y$$

$$(4j) \quad 0 = 1/4 \quad \text{para } f_x (f_y)^2$$

$$(4k) \quad 0 = 1/4 \quad \text{para } f (f_y)^3$$

Sin entrar a examinar las ecuaciones es claro que son imposibles las dos últimas, debido a que $\Phi'''(0)$ no aporta términos en $f_x (f_y)^2$ y $f (f_y)^3$. En otras palabras, estos métodos de 3 etapas no pueden alcanzar el orden 4.

Como ocurre esto, el último paso es el estudio del error entre los métodos de 3 etapas que alcanzan el orden 3. Para ello nos vienen muy bien los cálculos efectuados, ya que, prescindiendo ahora (como antes) del factor $h^3/6$, el término principal del error es

$$\frac{1}{4} f^{(3)} - \Phi'''(0) = \begin{bmatrix} \tau_{4a} & \tau_{4b} & \tau_{4c} & \tau_{4d} & \tau_{4e} & \tau_{4f} & \tau_{4g} & \tau_{4h} & \tau_{4i} & \frac{1}{4} & \frac{1}{4} \end{bmatrix} \begin{bmatrix} f_{xxx} \\ f f_{xxy} \\ (f)^2 f_{xyy} \\ f_x f_{xy} \\ f f_y f_{xy} \\ (f)^3 f_{yyy} \\ f f_x f_{yy} \\ (f)^2 f_y f_{yy} \\ f_{xx} f_y \\ f_x (f_y)^2 \\ f (f_y)^3 \end{bmatrix},$$

donde las τ_{4l} son básicamente las igualdades de antes, convertidas en diferencias, o sea

$$(\tau_{4a}) \quad \frac{1}{4} - b_2 c_2^3 - b_3 c_3^3$$

$$(\tau_{4b}) \quad \frac{3}{4} - 3 b_2 c_2^2 a_{2,1} - 3 b_3 c_3^2 a_{3,1} - 3 b_3 c_3^2 a_{3,2},$$

y así sucesivamente.

Vamos a minimizar una norma del vector de coeficientes del error

$$\begin{aligned} \|E\|_2 &= \left\| \begin{bmatrix} \tau_{4a} & \tau_{4b} & \tau_{4c} & \tau_{4d} & \tau_{4e} & \tau_{4f} & \tau_{4g} & \tau_{4h} & \tau_{4i} & 1/4 & 1/4 \end{bmatrix} \right\|_2 \\ &= \sqrt{(\tau_{4a})^2 + (\tau_{4b})^2 + \cdots + (\tau_{4i})^2 + 1/16 + 1/16} \end{aligned}$$

aunque es más sencillo y equivalente minimizar dicha norma al cuadrado, $\|E\|_2^2$, ya que así prescindimos de la raíz

De nuevo nos ayudamos en las operaciones con el programa ya citado. Además, no tenemos más remedio que ir calculando en principio el mínimo para cada una de las tres familias. Para la primera, la que llamamos (**fam1**), el programa proporciona (sólo en forma numérica) el valor

$$\min(\|E\|_2^2) \sim 0,13234712654,$$

para determinados valores numéricos de b_3 y de c_3 .

Y es que se enfrenta a una expresión lo suficientemente compleja como para no poder realizar la minimización con aritmética exacta (es decir, manejando fracciones y no números en la forma de punto flotante). En efecto, para la familia (**fam1**), en que los parámetros quedaban en función de b_3 y de c_3 , la expresión a minimizar es



$$\begin{aligned} \|E\|_2^2 &= (\tau_{4a})^2 + (\tau_{4b})^2 + \dots + (\tau_{4i})^2 + 1/16 + 1/16 \\ &= -1/2 b_3 a_{3,1}^3 - 1/2 b_2 c_2^3 + b_3^2 a_{3,1}^6 - 1/2 b_2 a_{2,1}^3 + 9 b_2^2 c_2^2 a_{2,1}^4 - \\ &\quad - 9/2 b_3 c_3^2 a_{3,2} - 9/2 b_3 c_3^2 a_{3,1} - 9/2 b_2 c_2^2 a_{2,1} - 1/2 b_3 c_3^3 - 9/2 b_2 c_2 a_{2,1}^2 - \\ &\quad - 9/2 b_3 c_3 a_{3,1}^2 - 9/2 b_3 c_3 a_{3,2}^2 + 6 b_2 a_{2,1}^3 b_3 a_{3,1}^2 a_{3,2} + b_3^2 a_{3,2}^6 - \\ &\quad - 1/2 b_3 a_{3,2}^3 - 3/2 b_3 a_{3,1}^2 a_{3,2} - 3/2 b_3 a_{3,1} a_{3,2}^2 - 3/2 b_3 a_{3,2} c_2^2 - \\ &\quad - 9 b_3 c_3 a_{3,1} a_{3,2} - 9 b_3 a_{3,2}^2 c_2 - 9 b_3 c_3 a_{3,2} c_2 - 12 b_3 a_{3,2}^2 a_{2,1} - 6 b_3 a_{3,2} a_{2,1}^2 - \\ &\quad - 15 b_3 c_3 a_{3,2} a_{2,1} - 15 b_3 a_{3,2} c_2 a_{2,1} - 9 b_3 a_{3,2} c_2 a_{3,1} - 12 b_3 a_{3,2} a_{2,1} a_{3,1} + \\ &\quad + 18 b_2 c_2^2 a_{2,1} b_3 c_3^2 a_{3,1} + 9 b_3^2 a_{3,2}^2 a_{2,1}^4 + 9 b_3^2 a_{3,2}^2 c_2^4 + \\ &\quad + 36 b_3^2 a_{3,2}^4 c_2^2 + 36 b_3^2 a_{3,2}^4 a_{2,1}^2 + 36 b_3^2 a_{3,2}^3 a_{2,1}^3 + 6 b_3^2 a_{3,2}^5 a_{3,1} + \\ &\quad + 15 b_3^2 a_{3,2}^4 a_{3,1}^2 + 20 b_3^2 a_{3,2}^3 a_{3,1}^3 + 6 b_3^2 a_{3,1}^5 a_{3,2} + 15 b_3^2 a_{3,1}^4 a_{3,2}^2 + \end{aligned}$$



$$\begin{aligned}
&+9 b_3^2 c_3^4 a_{3,2}^2 + 9 b_2^2 c_2^4 a_{2,1}^2 + 9 b_3^2 c_3^4 a_{3,1}^2 + 9 b_3^2 c_3^2 a_{3,1}^4 + 9 b_3^2 c_3^2 a_{3,2}^4 + \\
&+ 2 b_2 c_2^3 b_3 c_3^3 + 18 b_3^2 c_3^4 a_{3,1} a_{3,2} + 54 b_3^2 c_3^2 a_{3,1}^2 a_{3,2}^2 + 36 b_3^2 c_3^2 a_{3,1}^3 a_{3,2} + \\
&\quad + 36 b_3^2 c_3^2 a_{3,2}^3 a_{3,1} + 36 b_3^2 c_3^2 a_{3,2}^2 c_2^2 + 36 b_3^2 c_3^2 a_{3,2}^2 a_{2,1}^2 + \\
&\quad + 36 b_3^2 a_{3,2}^2 c_2^2 a_{2,1}^2 + 2 b_2 a_{2,1}^3 b_3 a_{3,2}^3 + 36 b_3^2 a_{3,2}^2 c_2^2 a_{3,1}^2 + \\
&+ 72 b_3^2 a_{3,2}^3 c_2^2 a_{3,1} + 36 b_3^2 a_{3,2}^2 a_{2,1}^2 a_{3,1}^2 + 72 b_3^2 a_{3,2}^3 a_{2,1}^2 a_{3,1} + \\
&\quad + 36 b_3^2 a_{3,2}^2 a_{2,1}^3 a_{3,1} + b_3^2 c_3^6 + b_2^2 c_2^6 + 72 b_3^2 c_3 a_{3,2}^2 a_{2,1}^2 c_2 + \\
&+ 18 b_2 c_2^2 a_{2,1} b_3 c_3^2 a_{3,2} + 6 b_2 a_{2,1}^3 b_3 a_{3,1} a_{3,2}^2 + 2 b_2 a_{2,1}^3 b_3 a_{3,1}^3 + \\
&\quad + 18 b_2 c_2 a_{2,1}^2 b_3 c_3 a_{3,1}^2 + 18 b_2 c_2 a_{2,1}^2 b_3 c_3 a_{3,2}^2 + \\
&\quad + 36 b_2 c_2 a_{2,1}^2 b_3 c_3 a_{3,1} a_{3,2} + b_2^2 a_{2,1}^6 + \frac{41}{8}
\end{aligned}$$

y al escribir la expresión en función de b_3 y de c_3 , queda

$$\begin{aligned}
\|E\|_2^2 &= (\tau_{4a})^2 + (\tau_{4b})^2 + \dots + (\tau_{4i})^2 + 1/16 + 1/16 \\
&= \frac{1}{648} \frac{12960 b_3^2 c_3^6 - 34560 c_3^5 b_3^2 + 58032 b_3^2 c_3^4 - 46440 b_3^2 c_3^3 -}{-14976 b_3 c_3^3 + 13284 b_3^2 c_3^2 + 20076 b_3 c_3^2 + 2592 c_3^2 - 3564 c_3 -} \\
&\quad \frac{-7164 b_3 c_3 + 1333}{(-1 + 2 b_3 c_3)^2}
\end{aligned}$$

expresión mucho más sencilla pero que supera aún la posibilidad de minimización en forma exacta.

Por razones que más tarde se entenderán al comparar nuestro mínimo con el que proporciona una de nuestras referencias, vamos a limitarnos a calcular ese mínimo para la subfamilia caracterizada por el valor concreto

$$c_3 = 3/4,$$

lo que deja la función $\|E\|_2^2$ dependiente de una sola variable b_3

$$\begin{aligned} \|E\|_2^2 &= (\tau_{4a})^2 + (\tau_{4b})^2 + \cdots + (\tau_{4i})^2 + 1/16 + 1/16 \\ &= \frac{1}{648} \frac{\left(\frac{44469}{128} b_3^2 - \frac{1593}{4} b_3 + 118\right)}{(-1 + 3/2 b_3)^2} \end{aligned}$$

Ahora ese mínimo vale en forma exacta

$$\min(\|E\|_2^2) = \frac{59}{442} \sim 0,1334841629,$$

y toma ese valor para

$$b_3 = \frac{944}{2079} \quad (\text{y } c_3 = \frac{3}{4}, \text{ claro}).$$



Hacemos lo mismo con la familia (**fam2**), con resultados

$$\min(\|E\|_2^2) = \frac{409}{2592} \sim 0,1577932099,$$

y

$$c_2 = \frac{7}{12},$$

y con la familia (**fam3**), con idéntico resultado

$$\min(\|E\|_2^2) = \frac{409}{2592} \sim 0,1577932099,$$

para

$$a_{32} = \frac{8}{21}.$$

El mínimo se alcanza entonces para el citado método de la familia (**fam1**).



Un último apunte. El excelente libro J.R. DORMAND: 'Numerical Methods for Differential Equations', que citamos entre las referencias utilizadas afirma que el mínimo del error para estos métodos de 3 etapas y orden 3 se produce para el método

0			
1/2	1/2		
3/4	0	3/4	
	2/9	1/3	4/9

Sin embargo, este método produce para nuestra norma al cuadrado el valor

$$\|E\|_2^2 = \frac{77}{576} \sim 0,1336805556,$$

ligeramente superior al que antes hemos obtenido, incluso para el obtenido para los métodos de **(fam1)** con $c_3 = 3/4$, entre los que está el de DORMAND.

No hablaremos aquí de los métodos de RUNGE-KUTTA con cuatro o más etapas. El desarrollo detallado de estos métodos con los procedimientos que venimos empleando sobrepasa nuestra capacidad. Además, salvo una excepción que citaremos de inmediato, no son métodos muy usados, sobre todo por el importante coste en evaluaciones de f que conllevan.

No obstante tendremos ocasión de encontrarnos con alguno de ellos cuando hablemos de 'pares encajados' de los que se emplean en casos difíciles o cuando es muy grande la precisión requerida.

Por ahora, nos limitamos sólo a presentar un extraordinario método de RUNGE-KUTTA que llama la atención por su simplicidad combinada con un orden 4 relativamente alto.

El método de RUNGE-KUTTA clásico

El siguiente método explícito de RUNGE-KUTTA y de orden 4, con 4 etapas, recibe el nombre de **método de RUNGE-KUTTA clásico**. Tiene por tablero

0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
	1/6	1/3	1/3	1/6

y se describe detalladamente como

$$\begin{aligned}
 k_1 &= f(x_n, y_n), \\
 k_2 &= f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1\right) \\
 k_3 &= f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2\right) \\
 k_4 &= f(x_n + h, y_n + hk_3) \\
 y_{n+1} &= y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4).
 \end{aligned}$$

A notar las pocas operaciones que presenta (multiplicaciones por 1 y por 0 no son tales, y en la programación se evitan, claro). Y las operaciones que quedan son extremadamente sencillas, sobre todo pensando en el momento de su descubrimiento (por KUTTA hacia 1901) en que no existía la posibilidad de usar ordenadores para los cálculos, como sucede ahora.

Orden alcanzable

Otras técnicas de trabajo (y estoy hablando de la conocida como 'teoría de árboles' de BUTCHER) permiten abordar de manera muy eficaz el estudio de los órdenes que pueden alcanzarse con los distintos números de etapas. Gracias a ellas se ha puesto en claro la respuesta a la doble pregunta sobre etapas y orden que parece lógico hacerse: ¿con q etapas, qué orden se puede alcanzar? o, para alcanzar orden p ¿cuál es el número mínimo de etapas a emplear?

El siguiente cuadro aclara estas cosas para las primeras etapas:

etapas	1	2	3	4	5	6	7	8	9	10	11
orden alcanzable	1	2	3	4	4	5	6	6	7	7	8

El caso escalar autónomo

Aunque es poco importante en el caso escalar, merece un pequeño estudio el caso en el que se trata de una ecuación escalar autónoma, o sea, un problema tipo de la forma

$$y' = f(y), \quad y(a) = \eta,$$

con la variable independiente no presente en la ecuación. En este caso, de las diferenciales elementales a las que nos vamos acostumbrando

$$\begin{aligned}
 & f \\
 & f_x, f_y \\
 & f_{xx}, f_{xy}, (f)^2 f_{yy}, f_x f_y, f (f_y)^2 \\
 & f_{xxx}, f_{xxy}, (f)^2 f_{xyy}, f_x f_{xy}, f f_y f_{xy}, (f)^3 f_{yyy}, \\
 & f f_x f_{yy}, (f)^2 f_y f_{yy}, f_{xx} f_y, f_x (f_y)^2, f (f_y)^3 \\
 & \dots
 \end{aligned}$$

desaparecen (son nulas) todas aquellas en las que interviene como subíndice la variable independiente x , quedando sólo

$$\begin{aligned}
 & f \\
 & f f_y \\
 & (f)^2 f_{yy}, f (f_y)^2 \\
 & (f)^3 f_{yyy}, (f)^2 f_y f_{yy}, f (f_y)^3 \\
 & \dots
 \end{aligned}$$

Cuando se van buscando las condiciones para el orden, resulta que en esta caso hay menos condiciones, lo que, de manera intuitiva, hace suponer que una o las dos afirmaciones siguientes podría ser cierta: existen más métodos para un mismo orden (que el el caso completo o no autónomo), o incluso, se puede alcanzar un orden mayor. Vamos a ver si esto es cierto.

Cuando examinamos el caso de una etapa,
 $k_1 = f(x_n, y_n)$, $y_{n+1} = y_n + b_1 k_1$ (con el único parámetro b_1),
 es claro que, para el orden 1, en ambos casos aparece la única condición

$$b_1 = 1,$$

y que no se puede obtener el orden 2 puesto que

$$\Phi'(0) = b_1 k_1'(0) = b_1 0 = 0$$

no hace intervenir $f f_y$ que, en ambos casos, está en $f^{(1)}$. En definitiva, las dos afirmaciones son en ese caso falsas, obteniéndose el mismo orden máximo, 1, y con el mismo método (el de EULER).

El caso de dos etapas es de igual resultado. Tenemos

$$\begin{aligned} k_1 &= f(x_n, y_n), & k_1 &= f(y_n), \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1) & k_2 &= f(y_n + a_{2,1} h k_1) \\ y_{n+1} &= y_n + h(b_1 k_1 + b_2 k_2), & y_{n+1} &= y_n + h(b_1 k_1 + b_2 k_2), \end{aligned}$$

según el caso. El primero lleva para el orden 2 a las condiciones

$$\begin{aligned} (1a) \quad & b_1 + b_2 = 1 \\ (2a) \quad & b_2 c_2 = 1/2 \\ (2b) \quad & b_2 a_{2,1} = 1/2. \end{aligned}$$

mientras que el caso autónomo lleva a

$$\begin{aligned} (1a) \quad & b_1 + b_2 = 1 \\ (2b) \quad & b_2 a_{2,1} = 1/2. \end{aligned}$$

Pero a (1a), (2a) y (2b) se añade, como sabemos

$$(f2) \quad c_2 = a_{21},$$

y las cuatro equivalen a (1a), (2b) y (f2), que son las del caso autónomo y una condición sobre c_2 que ahora no interviene.

Por otro lado, con dos etapas no se alcanza tampoco en el caso autónomo el orden 3, ya que $\Phi''(0) = b_2 k_2'(0)$ no hace intervenir $f(f_y)^2$, que en ambos casos está en $f^{(2)}$.

Por lo tanto, en el caso de dos etapas, continúan idénticos resultados para el caso autónomo y el caso completo.

Lo mismo sucede también en el caso de 3 etapas. Dejando de lado lo que ocurre con la relación entre **(1a)**, **(2a)** y **(2b)** más la **(f2)**, que acaban siendo lo mismo que las **(1a)** y **(2b)** del caso autónomo, comprobamos lo que ocurre con

$$(3a) \quad b_2 (c_2)^2 + b_3 (c_3)^2 = 1/3$$

$$(3b) \quad b_2 c_2 a_{2,1} + b_3 c_3 (a_{3,1} + a_{3,2}) = 1/3$$

$$(3c) \quad b_2 (a_{2,1})^2 + b_3 (a_{3,1} + a_{3,2})^2 = 1/3$$

$$(3d) \quad b_3 c_2 a_{3,2} = 1/6$$

$$(3e) \quad b_3 a_{2,1} a_{3,2} = 1/6.$$

en relación con las

$$(3c) \quad b_2 (a_{2,1})^2 + b_3 (a_{3,1} + a_{3,2})^2 = 1/3$$

$$(3e) \quad b_3 a_{2,1} a_{3,2} = 1/6.$$

que aparecen en el caso autónomo.

Ahora, las **(3a)**, **(3b)**, **(3c)**, **(3d)** y **(3e)**, al juntarse con la **(f2)** y la

$$(f3) \quad c_3 = a_{31} + a_{32},$$

se puede reemplazar por el conjunto de las dos **(3c)** y **(3e)** y las dos de fila **(f2)** y **(f3)** que no dan más que valores para los parámetros c_2 y c_3 que no intervienen en el caso autónomo.

Y tampoco el caso autónomo alcanza orden 4 con tres etapas, aunque no nos entretendremos en esta vía.

Explicaremos un poco lo que ocurre en el caso de 4 etapas, que es donde los resultados del caso autónomo y del no autónomo empiezan a ser diferentes.

Otra vez nos desentendemos de las condiciones **(1a)** (orden 1), **(2a)**, **(2b)** (orden 2), **(3a)**, **(3b)**, **(3c)**, **(3d)** y **(3e)** (orden 3), junto con las de fila **(f2)** y **(f3)** que acaban siendo equivalentes a las **(1a)** (orden 1), **(2b)** (orden 2), **(3c)** y **(3e)** (orden 3) del caso autónomo.

Al pasar a las del orden 4, bastante trabajosas, nos ayudamos de los cálculos que 'Maple' hace por nosotros en el programa

Ruk44.mws que se encarga de analizar los métodos de 4 etapas para ecuaciones autónomas y no autónomas y para sistemas.

Los métodos explícitos de 4 etapas responden al tablero

0				
c_2	a_{21}			
c_3	a_{31}	a_{32}		
c_4	a_{41}	a_{42}	a_{43}	
	b_1	b_2	b_3	b_4

dependen de los parámetros c_2 , c_3 , c_4 , $a_{2,1}$, $a_{3,1}$, $a_{3,2}$, $a_{4,1}$, $a_{4,2}$, $a_{4,3}$, b_1 , b_2 , b_3 y b_4 , y se expresan detalladamente como

$$\begin{aligned}
 k_1 &= f(x_n, y_n), \\
 k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1) \\
 k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2) \\
 k_4 &= f(x_n + c_4 h, y_n + a_{4,1} h k_1 + a_{4,2} h k_2 + a_{4,3} h k_3)
 \end{aligned}$$

$$y_{n+1} = y_n + h (b_1 k_1 + b_2 k_2 + b_3 k_3 + b_4 k_4),$$

Para la ecuación no autónoma resultan las condiciones que no evocaremos hasta el orden 3 y las

$$\begin{aligned}
 (4a) \quad & b_2 c_2^3 + b_3 c_3^3 + b_4 c_4^3 = 1/4 && \text{para } f_{xxx} \\
 (4b) \quad & b_2 c_2^2 a_{2,1} + b_3 c_3^2 a_{3,1} + b_3 c_3^2 a_{3,2} + b_4 c_4^2 a_{4,1} \\
 & + b_4 c_4^2 a_{4,2} + b_4 c_4^2 a_{4,3} = 1/4 && \text{para } f f_{xy} \\
 (4c) \quad & 2 b_4 c_4 a_{4,1} a_{4,2} + b_2 c_2 a_{2,1}^2 + b_3 c_3 a_{3,1}^2 + b_3 c_3 a_{3,2}^2 + 2 b_3 c_3 a_{3,1} a_{3,2} \\
 & + b_4 c_4 a_{4,1}^2 + b_4 c_4 a_{4,2}^2 + b_4 c_4 a_{4,3}^2 + 2 b_4 c_4 a_{4,1} a_{4,3} \\
 & + 2 b_4 c_4 a_{4,2} a_{4,3} = 1/4 && \text{para } (f)^2 f_{xy} \\
 (4d) \quad & b_3 c_3 a_{3,2} c_2 + b_4 c_4 a_{4,2} c_2 + b_4 c_4 a_{4,3} c_3 = 1/8 && \text{para } f_x f_{xy} \\
 (4e) \quad & b_3 c_3 a_{3,2} a_{2,1} + b_3 a_{3,2} c_2 a_{2,1} + b_4 c_4 a_{4,2} a_{2,1} + b_4 c_4 a_{4,3} a_{3,1} \\
 & + b_4 c_4 a_{4,3} a_{3,2} + b_4 a_{4,2} c_2 a_{2,1} + b_4 a_{4,3} c_3 a_{3,1} \\
 & + b_4 a_{4,3} c_3 a_{3,2} = 5/24 && \text{para } f f_y f_{xy} \\
 (4f) \quad & 3 b_4 a_{4,1} a_{4,2}^2 + b_2 a_{2,1}^3 + b_3 a_{3,1}^3 + b_3 a_{3,2}^3 + b_4 a_{4,3}^3 \\
 & + b_4 a_{4,1}^3 + b_4 a_{4,2}^3 + 3 b_3 a_{3,1}^2 a_{3,2} + 3 b_3 a_{3,1} a_{3,2}^2 + 3 b_4 a_{4,1}^2 a_{4,2} \\
 & + 3 b_4 a_{4,1}^2 a_{4,3} + 3 b_4 a_{4,1} a_{4,3}^2 + 3 b_4 a_{4,2}^2 a_{4,3} + 6 b_4 a_{4,1} a_{4,2} a_{4,3} \\
 & + 3 b_4 a_{4,2} a_{4,3}^2 = 1/4 && \text{para } (f)^3 f_{yyy}
 \end{aligned}$$



$$\begin{aligned}
 (4g) \quad & b_4 a_{4,2}^2 c_2 + b_3 a_{3,2} c_2 a_{3,1} + b_3 a_{3,2}^2 c_2 \\
 & + b_4 a_{4,2} c_2 a_{4,1} + b_4 a_{4,2} c_2 a_{4,3} + b_4 a_{4,3} c_3 a_{4,1} \\
 & + b_4 a_{4,3} c_3 a_{4,2} + b_4 a_{4,3}^2 c_3 = 1/8 && \text{para } f f_x f_{yy} \\
 (4h) \quad & b_4 a_{4,2} a_{2,1}^2 + 2 b_3 a_{3,2} a_{2,1} a_{3,1} + 2 b_3 a_{3,2}^2 a_{2,1} + b_3 a_{3,2} a_{2,1}^2 \\
 & + 2 b_4 a_{4,2} a_{2,1} a_{4,1} + 2 b_4 a_{4,2}^2 a_{2,1} + 2 b_4 a_{4,2} a_{2,1} a_{4,3} \\
 & + 2 b_4 a_{4,3} a_{3,1} a_{4,1} + 2 b_4 a_{4,3} a_{3,1} a_{4,2} + 2 b_4 a_{4,3}^2 a_{3,1} \\
 & + 2 b_4 a_{4,3} a_{3,2} a_{4,1} + 2 b_4 a_{4,3} a_{3,2} a_{4,2} + 2 b_4 a_{4,3}^2 a_{3,2} \\
 & + b_4 a_{4,3} a_{3,1}^2 + 2 b_4 a_{4,3} a_{3,1} a_{3,2} + b_4 a_{4,3} a_{3,2}^2 = 1/3 && \text{para } (f)^2 f_y f_{yy} \\
 (4i) \quad & b_3 a_{3,2} c_2^2 + b_4 a_{4,2} c_2^2 + b_4 a_{4,3} c_3^2 = 1/12 && \text{para } f_{xx} f_y \\
 (4j) \quad & b_4 a_{4,3} a_{3,2} c_2 = 1/24 && \text{para } f_x (f_y)^2 \\
 (4k) \quad & b_4 a_{4,3} a_{3,2} a_{2,1} = 1/24 && \text{para } f (f_y)^3
 \end{aligned}$$

Reuniremos estas condiciones con las ya conocidas de fila, que se cumplen en este caso.



El mismo programa citado nos muestra las condiciones para el orden 4 en el caso autónomo. Son sólo las tres siguientes

$$\begin{aligned}
 (4f) \quad & 3 b_4 a_{4,1} a_{4,2}^2 + b_2 a_{2,1}^3 + b_3 a_{3,1}^3 + b_3 a_{3,2}^3 + b_4 a_{4,3}^3 \\
 & + b_4 a_{4,1}^3 + b_4 a_{4,2}^3 + 3 b_3 a_{3,1}^2 a_{3,2} + 3 b_3 a_{3,1} a_{3,2}^2 + 3 b_4 a_{4,1}^2 a_{4,2} \\
 & + 3 b_4 a_{4,1}^2 a_{4,3} + 3 b_4 a_{4,1} a_{4,3}^2 + 3 b_4 a_{4,2}^2 a_{4,3} + 6 b_4 a_{4,1} a_{4,2} a_{4,3} \\
 & + 3 b_4 a_{4,2} a_{4,3}^2 = 1/4 \quad \text{para } (f)^3 f_{yyy} \\
 (4h) \quad & b_4 a_{4,2} a_{2,1}^2 + 2 b_3 a_{3,2} a_{2,1} a_{3,1} + 2 b_3 a_{3,2}^2 a_{2,1} + b_3 a_{3,2} a_{2,1}^2 \\
 & + 2 b_4 a_{4,2} a_{2,1} a_{4,1} + 2 b_4 a_{4,2}^2 a_{2,1} + 2 b_4 a_{4,2} a_{2,1} a_{4,3} \\
 & + 2 b_4 a_{4,3} a_{3,1} a_{4,1} + 2 b_4 a_{4,3} a_{3,1} a_{4,2} + 2 b_4 a_{4,3}^2 a_{3,1} \\
 & + 2 b_4 a_{4,3} a_{3,2} a_{4,1} + 2 b_4 a_{4,3} a_{3,2} a_{4,2} + 2 b_4 a_{4,3}^2 a_{3,2} \\
 & + b_4 a_{4,3} a_{3,1}^2 + 2 b_4 a_{4,3} a_{3,1} a_{3,2} + b_4 a_{4,3} a_{3,2}^2 = 1/3 \\
 & \text{para } (f)^2 f_y f_{yy} \\
 (4k) \quad & b_4 a_{4,3} a_{3,2} a_{2,1} = 1/24 \quad \text{para } f (f_y)^3
 \end{aligned}$$

Pero ahora la casuística es la siguiente. Cuando a las (4a) a (4k) añadimos las (f2), (f3) y

$$(f4) \quad c_4 = a_{41} + a_{42} + a_{43} ,$$

ciertas para los métodos de orden 4, el conjunto equivale a las (4f), (4h) y (4k) del caso autónomo más las (f2), (f3) y (f4) como antes pero a las que hay que sumar también ahora las (4d), (4e), (4g) y (4i). No obstante no todas son independientes y es sencillo comprobar que un conjunto de condiciones independientes para el caso no autónomo es

$$(4f), (4h), (4i) \text{ y } (4k) \text{ (con (f2), (f3) y (f4), claro)}$$

ya que (4e)=(4h)+(4i) y (4d)=(4g)=(4h)-(4i) con

$$\begin{aligned}
 (4i) \quad & b_3 a_{3,2} a_{2,1}^2 + b_4 a_{4,2} a_{2,1}^2 + b_4 a_{4,3} (a_{3,1} + a_{3,2})^2 = 1/12 \\
 & \text{para } f_{xx} f_y
 \end{aligned}$$

Pues bien, aunque no se puede lograr orden 5 ni en el caso autónomo, existen para el caso autónomo más métodos de orden 4 que para el caso completo. Basta para probar esto encontrar un método que sea de orden 4 en un caso y no lo sea en el otro. El método de tablero

0				
1/2	1/2			
-1	1/2	-3/2		
1	0	4/3	-1/3	
	1/6	2/3	0	1/6

propuesto en el libro J.D. LAMBERT: 'Numerical Methods for Ordinary Differential Systems: The Initial Value Problem', es de orden 4 para el caso escalar autónomo y sólo de orden 3 para el no autónomo; en particular, cumple (4f), (4h) y (4k), pero da $0 = 1/12$ al substituir sus valores en nuestra (4i).



Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-01

a) Cuál es el mayor orden posible que puede alcanzar un método de Runge-Kutta con el siguiente tablero

0		
1/(2 α)		1/(2 α)

		1 - α α

(desde aquí es Ejercicio del capítulo 8)

b) Para α arbitrario, calcúlese la función de estabilidad de estos métodos

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/(2*alpha)*h,y0+1/(2*alpha)*h*k1_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand((1-alpha)*k1_0+alpha*k2_0);
[                                     Fi_0 := f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                     cond1a := 1 = 1
[ condiciones para el orden 2
[ > Fi10:=expand((1-alpha)*k110+alpha*k210);
[                                     Fi10 := 1/2 D1(f)(x0, y0) + 1/2 D2(f)(x0, y0) f(x0, y0)
```

```

> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
cond2a :=  $\frac{1}{2} = \frac{1}{2}$ 
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
cond2b :=  $\frac{1}{2} = \frac{1}{2}$ 

```

condiciones para el orden 3

```

> Fi20:=expand((1-alpha)*k120+alpha*k220);
Fi20 :=  $\frac{1}{4} \frac{D_{1,1}(f)(x0,y0)}{\alpha} + \frac{1}{2} \frac{D_{1,2}(f)(x0,y0) f(x0,y0)}{\alpha} + \frac{1}{4} \frac{f(x0,y0)^2 D_{2,2}(f)(x0,y0)}{\alpha}$ 
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
cond3a :=  $\frac{1}{6} = \frac{1}{8\alpha}$ 
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
cond3b :=  $\frac{1}{3} = \frac{1}{4\alpha}$ 
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
cond3c :=  $\frac{1}{6} = \frac{1}{8\alpha}$ 
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
cond3d :=  $\frac{1}{6} = 0$ 
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
cond3e :=  $\frac{1}{6} = 0$ 

```

b)

función de estabilidad

```

> restart;
> interface(labeling=false);
> f:=(x,y)->lambda*y;
f := (x, y) → λ y
> k1_:=f(x0,y0);
k1_ := λ y0
> k2_:=f(x0+1/(2*alpha)*h,y0+1/(2*alpha)*h*k1_);
k2_ :=  $\lambda \left( y0 + \frac{h \lambda y0}{2 \alpha} \right)$ 
> y1:=expand(y0+h*((1-alpha)*k1_+alpha*k2_));
y1 :=  $y0 + h \lambda y0 + \frac{1}{2} \lambda^2 h^2 y0$ 
> y1:=collect(subs(lambda=z/h,y1),y0);
y1 :=  $\left( 1 + z + \frac{1}{2} z^2 \right) y0$ 
> r:=unapply(collect(simplify(y1/y0),z),z);
r :=  $z \rightarrow 1 + z + \frac{1}{2} z^2$ 
>

```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-02

Considérese la familia general de métodos de Runge-Kutta de 2 etapas

$$\begin{array}{c|cc} 0 & & \\ c_2 & | & a_{2,1} \\ \hline & | & b_1 \quad b_2 \end{array}$$

Búsqense los métodos de orden 2 y de ellos el que minimiza la norma-2

de los coeficientes de las diferenciales elementales del término en h^3

```
[ > restart:
[ > interface(labeling=false):
[ desarrollo de la verdadera solución
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ desarrollo del método
[ > k10:=f(x,y):
[ > k20:=f(x+c[2]*h,y+a[2,1]*h*k10):
[ > k100:=subs(h=0,k10):
[ > k200:=subs(h=0,k20):
[ > Fi0:=b[1]*k100+b[2]*k200:Fi0:=subs({x=x0,y=y0},Fi0):Fi0:=expand(Fi0):
[ > k11:=diff(k10,h):
[ > k21:=diff(k20,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > Fi1:=b[1]*k110+b[2]*k210:Fi1:=subs({x=x0,y=y0},Fi1):Fi1:=expand(Fi1):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > Fi2:=b[1]*k120+b[2]*k220:Fi2:=subs({x=x0,y=y0},Fi2):Fi2:=expand(Fi2):
[ condiciones para el orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
[ 
$$cond1a := b_1 + b_2 = 1$$

[ condiciones para el orden 2 y su solución
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
[ 
$$cond2a := b_2 c_2 = \frac{1}{2}$$

[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)),f(x0,y0));
```

$$\text{cond2b} := b_2 a_{2,1} = \frac{1}{2}$$

```
> sol_orden2:=solve({cond1a,cond2a,cond2b},{a[2,1],b[1],b[2]});
```

$$\text{sol_orden2} := \left\{ b_1 = \frac{1}{2} \frac{-1 + 2 c_2}{c_2}, b_2 = \frac{1}{2} \frac{1}{c_2}, a_{2,1} = c_2 \right\}$$

```
> a[2,1]:=subs(sol_orden2,a[2,1]);b[1]:=subs(sol_orden2,b[1]);b[2]:=subs(sol_orden2,b[2]);
```

$$a_{2,1} := c_2$$

$$b_1 := \frac{1}{2} \frac{-1 + 2 c_2}{c_2}$$

$$b_2 := \frac{1}{2} \frac{1}{c_2}$$

coeficientes del término de error

```
# coeficiente del término de error
```

```
> term:=expand((1/2!)*f12-(1/3!)*f2):
```

```
> erro3a:=coeff(term,D[1,1](f)(x0,y0));
```

$$\text{erro3a} := \frac{1}{4} c_2 - \frac{1}{6}$$

```
> erro3b:=coeff(coeff(term,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$\text{erro3b} := \frac{1}{2} c_2 - \frac{1}{3}$$

```
> erro3c:=coeff(coeff(term,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$\text{erro3c} := \frac{1}{4} c_2 - \frac{1}{6}$$

```
> erro3d:=coeff(coeff(term,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$\text{erro3d} := -\frac{1}{6}$$

```
> erro3e:=coeff(coeff(term,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{erro3e} := -\frac{1}{6}$$

minimizar la norma-2 de estos coeficientes del error

```
> norma2cuadrado_error:=expand(erro3a^2+erro3b^2+erro3c^2+erro3d^2+erro3e^2);
```

$$\text{norma2cuadrado_error} := \frac{3}{8} c_2^2 - \frac{1}{2} c_2 + \frac{2}{9}$$

```
> der_norma2cuadrado_error:=diff(norma2cuadrado_error,c[2]);
```

$$\text{der_norma2cuadrado_error} := \frac{3}{4} c_2 - \frac{1}{2}$$

```
> der2_norma2cuadrado_error:=diff(der_norma2cuadrado_error,c[2]);
```

$$\text{der2_norma2cuadrado_error} := \frac{3}{4}$$

```
> c2_minimo:=subs(solve({der_norma2cuadrado_error},{c[2]}),c[2]);
```

$$c2_minimo := \frac{2}{3}$$

```
> en_minimo:=subs(c[2]=c2_minimo,norma2cuadrado_error);en_minimo:=evalf(en_minimo);
```

$$\text{en_minimo} := \frac{1}{18}$$

$$\text{en_minimo} := 0.0555555555556$$

```
> subs(c[2]=c2_minimo,['c[2]`=c[2]',`a[2,1]`=a[2,1]',`b[1]`=b[1]',`b[2]`=b[2]]);
```

$$\left[c[2] = \frac{2}{3}, a[2,1] = \frac{2}{3}, b[1] = \frac{1}{4}, b[2] = \frac{3}{4} \right]$$

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-03 (del EXAMEN ORDINARIO de 02FEB01)

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0				
3/2		β		
0		1	-1	

		1/3	α	1/3

- a) Para qué valores de los parámetros α y β los métodos son de orden 2 ?
b) Existe alguna elección de α y β que proporcione un método de orden 3 ?

(desde aquí es Ejercicio del capítulo 8)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α y β .

d) En el supuesto de que $|\alpha| < \frac{2}{3}$ y de que $\beta = 0$, represéntese la región de estabilidad del método.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+3/2*h,y0+beta*h*k1_):
[ > k3_:=f(x0,y0+h*k1_-h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
```

```
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(1/3*k1_0+alpha*k2_0+1/3*k3_0);
```

$$Fi_0 := \frac{2}{3} f(x0, y0) + \alpha f(x0, y0)$$

```
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

$$cond1a := 1 = \frac{2}{3} + \alpha$$

```
[ > sol1:=solve(cond1a,{alpha});
```

$$sol1 := \{ \alpha = \frac{1}{3} \}$$

```
[ > alpha:=subs(sol1,alpha);
```

$$\alpha := \frac{1}{3}$$

[condiciones para el orden 2

```
[ > Fi10:=expand(1/3*k110+alpha*k210+1/3*k310);
```

$$Fi10 := \frac{1}{2} D_1(f)(x0, y0) + \frac{1}{3} D_2(f)(x0, y0) \beta f(x0, y0)$$

```
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
```

$$cond2a := \frac{1}{2} = \frac{1}{2}$$

```
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = \frac{\beta}{3}$$

```
[ > sol2:=solve({cond2a,cond2b},{beta});
```

$$sol2 := \{ \beta = \frac{3}{2} \}$$

```
[ > beta:=subs(sol2,beta);
```

$$\beta := \frac{3}{2}$$

[b)

[condiciones para el orden 3

```
[ > Fi20:=expand(1/3*k120+alpha*k220+1/3*k320);
```

$$Fi20 := \frac{3}{4} D_{1,1}(f)(x0, y0) + \frac{3}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{3}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2 - D_2(f)(x0, y0) D_1(f)(x0, y0) - D_2(f)(x0, y0)^2 f(x0, y0)$$

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{3}{8}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{3}{4}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{3}{8}$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[
```

```
2](f)(x0,y0),D[1](f)(x0,y0));
```

$$\text{cond3d} := \frac{1}{6} = \frac{-1}{2}$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \frac{-1}{2}$$

c)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false):
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+3/2*h,y0+beta*h*k1_);
```

$$k2_ := \lambda (y0 + \beta h \lambda y0)$$

```
> k3_:=f(x0,y0+h*k1_-h*k2_);
```

$$k3_ := \lambda (y0 + h \lambda y0 - h \lambda (y0 + \beta h \lambda y0))$$

```
> y1:=expand(y0+h*(1/3*k1_+alpha*k2_+1/3*k3_));
```

$$y1 := y0 + \frac{2 h \lambda y0}{3} + h \alpha \lambda y0 + \alpha \lambda^2 \beta h^2 y0 - \frac{h^3 \lambda^3 \beta y0}{3}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \frac{2}{3} z + \alpha z + \alpha z^2 \beta - \frac{1}{3} z^3 \beta \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 - \frac{1}{3} z^3 \beta + \alpha z^2 \beta + \left(\frac{2}{3} + \alpha \right) z$$

d)

```
> beta:=0;
```

$$\beta := 0$$

```
> r(z);
```

$$1 + \left(\frac{2}{3} + \alpha \right) z$$

```
> region:=abs(r(z))<1;
```

$$\text{region} := \left| 1 + \left(\frac{2}{3} + \alpha \right) z \right| < 1$$

```
> div:=2/3+alpha;
```

$$\text{div} := \frac{2}{3} + \alpha$$

```
> region:=abs(collect(r(z)/div,z))<1/div;
```

$$\text{region} := \left| z + \frac{1}{\frac{2}{3} + \alpha} \right| < \frac{1}{\frac{2}{3} + \alpha}$$

```
> radio:=1/div;
```

$$\text{radio} := \frac{1}{\frac{2}{3} + \alpha}$$

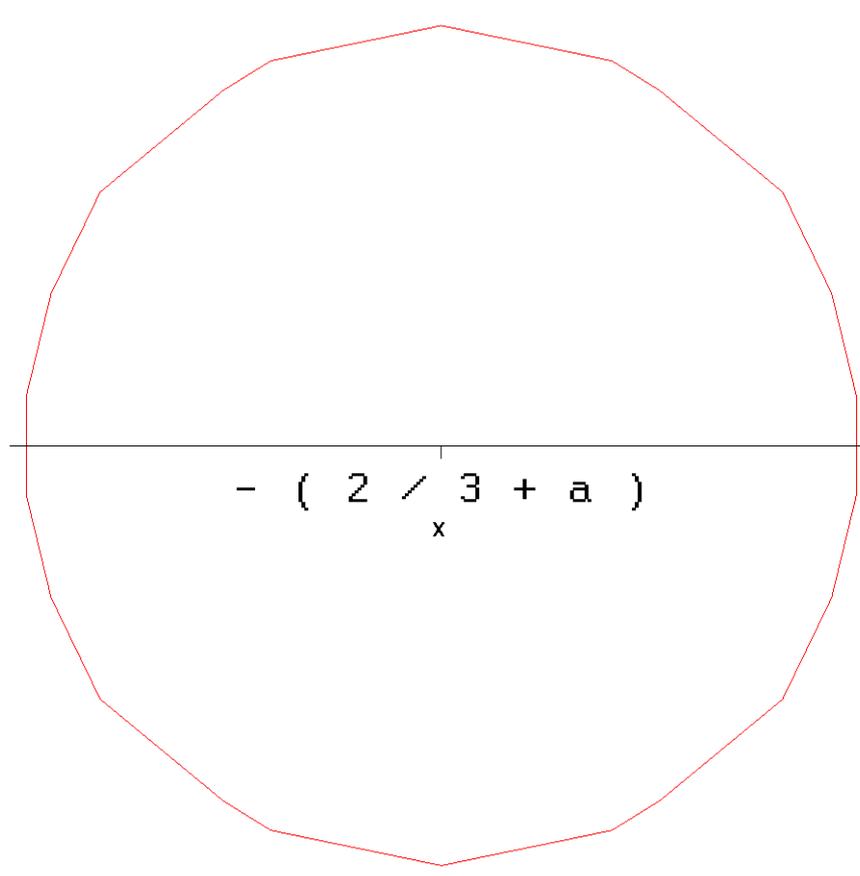
```
> with(plots):alpha:=1/3;
```

```
implicitplot((x+radio)^2+y^2=radio^2,x=-5..5,y=-3..3
```

```
,xtickmarks=[-1=`- ( 2 / 3 + a
```

```
)`],ytickmarks=0,scaling=constrained,axesfont=[COURIER,BOLD,18]);
```

$$\alpha := \frac{1}{3}$$



Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-04 (del EXAMEN EXTRAORDINARIO de 07JUL01)

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0			
1/2		1/2	
1		1- α	α

		1/6	b_2 b_3

- a) Para qué valores de los parámetros α , b_2 y b_3 los métodos son de orden 2 ?
- b) Existen valores de los parámetros α , b_2 y b_3 para los que los métodos sean de orden 3 ?

(desde aquí es Ejercicio del capítulo 8)

- c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α , b_2 y b_3 . Calcúlese esta misma función para los métodos de orden 2 y orden 3 que, en su caso, se hayan podido obtener.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/2*h,y0+1/2*h*k1_):
[ > k3_:=f(x0+h,y0+(1-alpha)*h*k1_+alpha*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
```

```
[ > k220:=subs(h=0,k22):
```

```
[ > k320:=subs(h=0,k32):
```

```
[ a)
```

```
[ condiciones para el orden 1
```

```
[ > Fi_0:=expand(1/6*k1_0+b[2]*k2_0+b[3]*k3_0);
```

$$Fi_0 := \frac{1}{6}f(x0, y0) + b_2 f(x0, y0) + b_3 f(x0, y0)$$

```
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

$$cond1a := 1 = \frac{1}{6} + b_2 + b_3$$

```
[ condiciones para el orden 2
```

```
[ > Fi10:=expand(1/6*k110+b[2]*k210+b[3]*k310);
```

$$Fi10 := \frac{1}{2}b_2 D_1(f)(x0, y0) + \frac{1}{2}b_2 D_2(f)(x0, y0) f(x0, y0) + b_3 D_1(f)(x0, y0) + b_3 D_2(f)(x0, y0) f(x0, y0)$$

```
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
```

$$cond2a := \frac{1}{2} = \frac{1}{2}b_2 + b_3$$

```
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = \frac{1}{2}b_2 + b_3$$

```
[ > sol1:=solve({cond1a,cond2a,cond2b},{alpha,b[2],b[3]});
```

$$sol1 := \{ \alpha = \alpha, b_2 = \frac{2}{3}, b_3 = \frac{1}{6} \}$$

```
[ > b[2]:=subs(sol1,b[2]);
```

$$b_2 := \frac{2}{3}$$

```
[ > b[3]:=subs(sol1,b[3]);
```

$$b_3 := \frac{1}{6}$$

```
[ b)
```

```
[ condiciones para el orden 3
```

```
[ > Fi20:=expand(1/6*k120+b[2]*k220+b[3]*k320);
```

$$Fi20 := \frac{1}{3}D_{1,1}(f)(x0, y0) + \frac{2}{3}D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{3}D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{1}{6}D_2(f)(x0, y0) \alpha D_1(f)(x0, y0) + \frac{1}{6}D_2(f)(x0, y0)^2 \alpha f(x0, y0)$$

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{1}{6}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{1}{3}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{6}$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = \frac{\alpha}{12}$$

```
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \frac{\alpha}{12}$$

```
> sol2:=solve({cond3a,cond3b,cond3c,cond3d,cond3e},{alpha});
```

$$\text{sol2} := \{\alpha = 2\}$$

```
> alpha:=subs(sol2,alpha);
```

$$\alpha := 2$$

c)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false):
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+1/2*h,y0+1/2*h*k1_);
```

$$k2_ := \lambda \left(y0 + \frac{1}{2} h \lambda y0 \right)$$

```
> k3_:=f(x0+h,y0+(1-alpha)*h*k1_+alpha*h*k2_);
```

$$k3_ := \lambda \left(y0 + (1 - \alpha) h \lambda y0 + \alpha h \lambda \left(y0 + \frac{1}{2} h \lambda y0 \right) \right)$$

```
> y1:=expand(y0+h*(1/6*k1_+b[2]*k2_+b[3]*k3_));
```

$$y1 := y0 + \frac{h \lambda y0}{6} + h b_2 \lambda y0 + \frac{1}{2} b_2 \lambda^2 h^2 y0 + h b_3 \lambda y0 + b_3 \lambda^2 h^2 y0 + \frac{1}{2} b_3 \lambda^3 \alpha h^3 y0$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \frac{z}{6} + b_2 z + \frac{1}{2} b_2 z^2 + b_3 z + b_3 z^2 + \frac{1}{2} b_3 z^3 \alpha \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + \frac{1}{2} b_3 z^3 \alpha + \left(\frac{1}{2} b_2 + b_3 \right) z^2 + \left(\frac{1}{6} + b_2 + b_3 \right) z$$

función de estabilidad para los metodos de orden 2

```
> r2:=unapply(subs({b[2]=2/3,b[3]=1/6},r(z)),z);
```

$$r2 := z \rightarrow 1 + \frac{1}{12} z^3 \alpha + \frac{1}{2} z^2 + z$$

función de estabilidad para los metodos de orden 3

```
> r3:=unapply(subs({alpha=2},r2(z)),z);
```

$$r3 := z \rightarrow 1 + \frac{1}{6} z^3 + \frac{1}{2} z^2 + z$$

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-05

Considérese el método genérico explícito de Runge-Kutta dado por el siguiente tablero

0				
c_2		c_2		
c_3		0	c_3	

		0	0	1

a) Cuáles deben ser los coeficientes para alcanzar al menos el orden 2 ?

b) Es posible alcanzar el orden 3 ?

(desde aquí es Ejercicio del capítulo 8)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+c[2]*h,y0+c[2]*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+c[3]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(k3_0);
```

```

[                                      $Fi_0 := f(x_0, y_0)$ 
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                      $cond1a := 1 = 1$ 
[ condiciones para el orden 2
[ > Fi10:=expand(k310);
[                                      $Fi10 := D_1(f)(x_0, y_0) c_3 + D_2(f)(x_0, y_0) c_3 f(x_0, y_0)$ 
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                      $cond2a := \frac{1}{2} = c_3$ 
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                                      $cond2b := \frac{1}{2} = c_3$ 
[ > sol1:=solve({cond2a,cond2b},{c[3]});
[                                      $sol1 := \{c_3 = \frac{1}{2}\}$ 
[ > c[3]:=subs(sol1,c[3]);
[                                      $c_3 := \frac{1}{2}$ 

```

b)

condiciones para el orden 3

```

[ > Fi20:=expand(k320);
[  $Fi20 := \frac{1}{4} D_{1,1}(f)(x_0, y_0) + \frac{1}{2} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{4} D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2 + D_2(f)(x_0, y_0) D_1(f)(x_0, y_0) c_2$ 
[  $+ D_2(f)(x_0, y_0)^2 c_2 f(x_0, y_0)$ 
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
[                                      $cond3a := \frac{1}{6} = \frac{1}{8}$ 
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
[                                      $cond3b := \frac{1}{3} = \frac{1}{4}$ 
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
[                                      $cond3c := \frac{1}{6} = \frac{1}{8}$ 
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
[                                      $cond3d := \frac{1}{6} = \frac{1}{2} c_2$ 
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
[                                      $cond3e := \frac{1}{6} = \frac{1}{2} c_2$ 

```

c)

función de estabilidad

```

[ > restart;
[ > interface(labeling=false);
[ > f:=(x,y)->lambda*y;
[                                      $f := (x, y) \rightarrow \lambda y$ 
[ > k1_:=f(x0,y0);
[                                      $k1_ := \lambda y_0$ 
[ > k2_:=f(x0+c[2]*h,y0+c[2]*h*k1_);
[                                      $k2_ := \lambda (y_0 + c_2 h \lambda y_0)$ 

```

```

[ > k3_:=f(x0+c[3]*h,y0+c[3]*h*k2_);
                                     k3_ := λ (y0 + c3 h λ (y0 + c2 h λ y0))
[ > y1:=expand(y0+h*(k3_));
                                     y1 := y0 + h λ y0 + h2 λ2 c3 y0 + h3 λ3 c3 c2 y0
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                     y1 := (1 + z + z2 c3 + z3 c3 c2) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
                                     r := z → 1 + z + z2 c3 + z3 c3 c2
[ función de estabilidad para los metodos de orden 2
[ > r2:=unapply(subs(c[3]=1/2,r(z)),z);
                                     r2 := z → 1 + z +  $\frac{1}{2}z^2 + \frac{1}{2}z^3 c_2$ 

```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-06

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0				
c_2		$a_{2,1}$		
c_3		$a_{3,1}$	$a_{3,2}$	

		0	0	b_3

- a) Para qué valores de los parámetros $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 los métodos son de orden 2 ?
- b) De los métodos anteriores, cuáles minimizan además el término del error ?
- c) Existen valores de los parámetros $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 para los que los métodos sean de orden 3 ?

(desde aquí es Ejercicio del capítulo 8)

- d) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 . Calcúlese esta misma función para los métodos de orden 2, los que minimizan el error y los de orden 3 que, en su caso, se hayan podido obtener.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+c[2]*h,y0+a[2,1]*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[
```

```

[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand(0*k1_0+0*k2_0+b[3]*k3_0);
[                                      $Fi_0 := b_3 f(x0, y0)$ 
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                      $cond1a := 1 = b_3$ 
[ condiciones para el orden 2
[ > Fi10:=expand(0*k110+0*k210+b[3]*k310);
[                                      $Fi10 := b_3 D_1(f)(x0, y0) c_3 + b_3 D_2(f)(x0, y0) a_{3,1} f(x0, y0) + b_3 D_2(f)(x0, y0) a_{3,2} f(x0, y0)$ 
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                      $cond2a := \frac{1}{2} = b_3 c_3$ 
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                                      $cond2b := \frac{1}{2} = b_3 a_{3,1} + b_3 a_{3,2}$ 
[ > sol1:=solve({cond1a,cond2a,cond2b},{b[3],c[3],a[3,1]});
[                                      $sol1 := \{c_3 = \frac{1}{2}, b_3 = 1, a_{3,1} = \frac{1}{2} - a_{3,2}\}$ 
[ > b[3]:=subs(sol1,b[3]);
[                                      $b_3 := 1$ 
[ > c[3]:=subs(sol1,c[3]);
[                                      $c_3 := \frac{1}{2}$ 
[ > a[3,1]:=subs(sol1,a[3,1]);
[                                      $a_{3,1} := \frac{1}{2} - a_{3,2}$ 
[ b)
[ condiciones para el orden 3 o para el error
[ > Fi20:=expand(0*k120+0*k220+b[3]*k320);
[  $Fi20 := \frac{1}{4} D_{1,1}(f)(x0, y0) + \frac{1}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2 + 2 D_2(f)(x0, y0) a_{3,2} D_1(f)(x0, y0) c_2$ 
[  $+ 2 D_2(f)(x0, y0)^2 a_{3,2} a_{2,1} f(x0, y0)$ 
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
[                                      $cond3a := \frac{1}{6} = \frac{1}{8}$ 
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
[                                      $cond3b := \frac{1}{3} = \frac{1}{4}$ 
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);

```

$$\text{cond3c} := \frac{1}{6} = \frac{1}{8}$$

```
> cond3d := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$\text{cond3d} := \frac{1}{6} = a_{3,2} c_2$$

```
> cond3e := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2,f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = a_{3,2} a_{2,1}$$

c)

minimización del error

```
> sol2:=solve({cond3d,cond3e},{a[2,1],a[3,2]});
```

$$\text{sol2} := \{ a_{2,1} = c_2, a_{3,2} = \frac{1}{6} \frac{1}{c_2} \}$$

```
> a[2,1]:=subs(sol2,a[2,1]);
```

$$a_{2,1} := c_2$$

```
> a[3,2]:=subs(sol2,a[3,2]);
```

$$a_{3,2} := \frac{1}{6} \frac{1}{c_2}$$

d)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false):
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+c[2]*h,y0+a[2,1]*h*k1_);
```

$$k2_ := \lambda (y0 + a_{2,1} h \lambda y0)$$

```
> k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_);
```

$$k3_ := \lambda (y0 + a_{3,1} h \lambda y0 + a_{3,2} h \lambda (y0 + a_{2,1} h \lambda y0))$$

```
> y1:=expand(y0+h*(0*k1_+0*k2_+b[3]*k3_));
```

$$y1 := y0 + h b_3 \lambda y0 + h^2 b_3 \lambda^2 a_{3,1} y0 + h^2 b_3 \lambda^2 a_{3,2} y0 + h^3 b_3 \lambda^3 a_{3,2} a_{2,1} y0$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := (1 + b_3 z + b_3 z^2 a_{3,1} + b_3 z^2 a_{3,2} + b_3 z^3 a_{3,2} a_{2,1}) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + b_3 z^3 a_{3,2} a_{2,1} + (b_3 a_{3,1} + b_3 a_{3,2}) z^2 + b_3 z$$

función de estabilidad para los metodos de orden 2

```
> r2:=unapply(subs({c[3]=1/2,a[3,1]=1/2-a[3,2],b[3]=1},r(z)),z);
```

$$r2 := z \rightarrow 1 + z^3 a_{3,2} a_{2,1} + \frac{1}{2} z^2 + z$$

función de estabilidad para los metodos que minimizan el error

```
> r3:=unapply(subs({a[2,1]=c[2],a[3,2]=1/6/c[2]},r2(z)),z);
```

$$r3 := z \rightarrow 1 + \frac{1}{6} z^3 + \frac{1}{2} z^2 + z$$

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-07 (del EXAMEN EXTRAORDINARIO de 09JUL03)

Considérese la familia de métodos de RUNGE-KUTTA de 3 etapas y explícitos dados por el siguiente tablero

0				
1/2		α		
1		β	2	

		1/6	2/3	1/6

- a) Para qué valores de los parámetros α y β los métodos son de orden 2 ?
b) Existe alguna elección de α y β que proporcione un método de orden 3 ?

(desde aquí es Ejercicio del capítulo 8)

- c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α y β .
d) Calcúlese ahora la función de estabilidad para los valores para los que se obtiene el orden 2. Si en este caso hacemos $\beta = 1$ o $\alpha = 0$, que es lo mismo, dibújese la correspondiente región de estabilidad.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/2*h,y0+alpha*h*k1_):
[ > k3_:=f(x0+h,y0+h*beta*k1_+h*2*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ ]
```



```
> sol3:=solve({cond3b,cond3c,cond3d,cond3e},{alpha});
```

$$\text{sol3} := \left\{ \alpha = \frac{1}{2} \right\}$$

```
> alpha:=subs(sol3,alpha);
```

$$\alpha := \frac{1}{2}$$

c)

[función de estabilidad, caso general

```
> restart;
```

```
> interface(labeling=false);
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+1/2*h,y0+alpha*h*k1_);
```

$$k2_ := \lambda (y0 + \alpha h \lambda y0)$$

```
> k3_:=f(x0+h,y0+h*beta*k1_+h*2*k2_);
```

$$k3_ := \lambda (y0 + h \beta \lambda y0 + 2 h \lambda (y0 + \alpha h \lambda y0))$$

```
> y1:=expand(y0+h*(1/6*k1_+2/3*k2_+1/6*k3_));
```

$$y1 := y0 + h \lambda y0 + \frac{2 h^2 \lambda^2 \alpha y0}{3} + \frac{h^2 \beta \lambda^2 y0}{6} + \frac{h^2 \lambda^2 y0}{3} + \frac{h^3 \lambda^3 \alpha y0}{3}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + z + \frac{2}{3} z^2 \alpha + \frac{1}{6} \beta z^2 + \frac{1}{3} z^2 + \frac{1}{3} z^3 \alpha \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + \frac{1}{3} z^3 \alpha + \left(\frac{2\alpha}{3} + \frac{\beta}{6} + \frac{1}{3} \right) z^2 + z$$

d)

[función de estabilidad, orden 2

```
> beta := 1-4*alpha;
```

$$\beta := 1 - 4 \alpha$$

```
> r(z);
```

$$1 + \frac{1}{3} z^3 \alpha + \frac{1}{2} z^2 + z$$

[región de estabilidad para $\alpha = 0$

```
> alpha:=0;
```

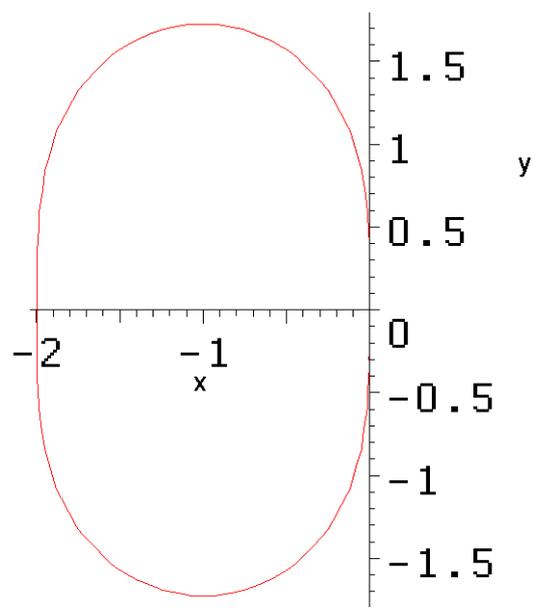
$$\alpha := 0$$

```
> r(z);
```

$$1 + \frac{1}{2} z^2 + z$$

[la región de estabilidad es la figura ya estudiada:

```
> z:=x+I*y:with(plots):implicitplot(abs(r(z))=1,x=-3..1,y=-3..3,scaling=constrained,axes  
font=[COURIER,BOLD,18]);
```



Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-08 (del EXAMEN ORDINARIO de 07FEB03)

Considérese el método genérico explícito de Runge-Kutta dado por el siguiente tablero

0				
$\frac{1}{2}$		a_{21}		
1		0	a_{32}	

		0	0	1

a) Dígase cual es el orden máximo que se puede alcanzar y cuáles deben ser los coeficientes para alcanzar dicho orden

b) De entre los métodos de orden máximo, dígase cuáles son los que minimizan el error de truncación

(desde aquí es Ejercicio del capítulo 8)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad. Calcúlese la misma función para los métodos de orden máximo y para los que minimizan el error

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+(1/2)*h,y0+a[2,1]*h*k1_):
[ > k3_:=f(x0+h,y0+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(k3_0);
```

```

[                                      $Fi_0 := f(x_0, y_0)$ 
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                      $cond1a := 1 = 1$ 
[ condiciones para el orden 2
[ > Fi10:=expand(k310);
[                                      $Fi10 := D_1(f)(x_0, y_0) + D_2(f)(x_0, y_0) a_{3,2} f(x_0, y_0)$ 
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                      $cond2a := \frac{1}{2} = 1$ 
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                                      $cond2b := \frac{1}{2} = a_{3,2}$ 

```

y no se puede conseguir el orden 2 de ninguna manera. El orden máximo es 1 y cualquiera de los métodos de la familia tiene ese orden

b)

Para minimizar el error, de entre los métodos de orden 1 (que son todos) habrá que elegir

```

[ > sol1:=solve({cond2b},{a[3,2]});
[                                      $sol1 := \{a_{3,2} = \frac{1}{2}\}$ 
[ > a[3,2]:=subs(sol1,a[3,2]);
[                                      $a_{3,2} := \frac{1}{2}$ 

```

y se obtiene $a_{32} = \frac{1}{2}$ como valor para la minimización del error, siendo aún a_{21} arbitrario

c)

función de estabilidad general y para el orden máximo 1

```

[ > restart;
[ > interface(labeling=false);
[ > f:=(x,y)->lambda*y;
[                                      $f := (x, y) \rightarrow \lambda y$ 
[ > k1_:=f(x0,y0);
[                                      $k1_ := \lambda y_0$ 
[ > k2_:=f(x0+(1/2)*h,y0+a[2,1]*h*k1_);
[                                      $k2_ := \lambda (y_0 + a_{2,1} h \lambda y_0)$ 
[ > k3_:=f(x0+h,y0+a[3,2]*h*k2_);
[                                      $k3_ := \lambda (y_0 + a_{3,2} h \lambda (y_0 + a_{2,1} h \lambda y_0))$ 
[ > y1:=expand(y0+h*(k3_));
[                                      $y1 := y_0 + h \lambda y_0 + h^2 \lambda^2 a_{3,2} y_0 + h^3 \lambda^3 a_{3,2} a_{2,1} y_0$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                      $y1 := (1 + z + z^2 a_{3,2} + z^3 a_{3,2} a_{2,1}) y_0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[                                      $r := z \rightarrow 1 + z + z^2 a_{3,2} + z^3 a_{3,2} a_{2,1}$ 

```

función de estabilidad para los metodos que minimizan el error

```

[ > r2:=unapply(subs(a[3,2]=1/2,r(z)),z);
[                                      $r2 := z \rightarrow 1 + z + \frac{1}{2} z^2 + \frac{1}{2} z^3 a_{2,1}$ 

```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-09 (del EXAMEN EXTRAORDINARIO de 22JUN11)

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos de tablero

0				
α		α		
α		0	α	

		β	θ	θ
--	--	---------	----------	----------

De entre ellos se buscaran los que proporcionen el orden mas alto.

(desde aquí es Ejercicio del capítulo 8)

Dígase cuál es la función de estabilidad de esos métodos de mayor orden.

```
[ > restart;
[ > interface(labeling=false):
[ > y1:=f(x,y(x));
[                                     y1 := f(x, y(x))
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x));
[                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x));
[ y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
[       + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
[ > Y1:=subs({y(x)=y0,x=x0},y1);
[                                     Y1 := f(x0, y0)
[ > Y2:=subs({y(x)=y0,x=x0},y2);
[                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
[ > Y3:=subs({y(x)=y0,x=x0},y3);
[ Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
[       + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
[ > k1_:=f(x0,y0);
[                                     k1_ := f(x0, y0)
[ > k2_:=f(x0+alpha*h,y0+alpha*h*k1_);
[                                     k2_ := f(x0 + alpha h, y0 + alpha h f(x0, y0))
[ > k3_:=f(x0+alpha*h,y0+alpha*h*k2_);
[                                     k3_ := f(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0)))
[ > k1_0:=subs(h=0,k1_);
[                                     k1_0 := f(x0, y0)
[ > k2_0:=subs(h=0,k2_);
[                                     k2_0 := f(x0, y0)
[ > k3_0:=subs(h=0,k3_);
[                                     k3_0 := f(x0, y0)
[ > k11:=diff(k1_,h);
```

```

[
    k11 := 0
> k21:=diff(k2_,h);
    k21 := D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)
> k31:=diff(k3_,h);
k31 := D1(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha +
    D2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))
> k110:=subs(h=0,k11);
    k110 := 0
> k210:=subs(h=0,k21);
    k210 := D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0)
> k310:=subs(h=0,k31);
    k310 := D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0)
> k12:=diff(k11,h);
    k12 := 0
> k22:=diff(k21,h);
k22 := (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha
    + (D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha f(x0, y0)
> k32:=diff(k31,h);
k32 := (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha +
    D1,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))) alpha + (
    D1,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (
    alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))) (
    alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0))) +
    D2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (
    2 alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) + alpha h (
    (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha
    + (D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha f(x0, y0)))
> k120:=subs(h=0,k12);
    k120 := 0
> k220:=subs(h=0,k22);
    k220 := (D1,1(f)(x0, y0) alpha + D1,2(f)(x0, y0) alpha f(x0, y0)) alpha + (D1,2(f)(x0, y0) alpha + D2,2(f)(x0, y0) alpha f(x0, y0)) alpha f(x0, y0)
> k320:=subs(h=0,k32);
k320 := (D1,1(f)(x0, y0) alpha + D1,2(f)(x0, y0) alpha f(x0, y0)) alpha + (D1,2(f)(x0, y0) alpha + D2,2(f)(x0, y0) alpha f(x0, y0)) alpha f(x0, y0)
    + 2 D2(f)(x0, y0) alpha (D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0))
a)
condiciones para el orden 1
> Fi_0:=expand(beta*k1_0+theta*k2_0+theta*k3_0);
    Fi_0 := beta f(x0, y0) + 2 theta f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
    cond1a := 1 = beta + 2 theta
condiciones para el orden 2
> Fi10:=expand(beta*k110+theta*k210+theta*k310);
    Fi10 := 2 theta D1(f)(x0, y0) alpha + 2 theta D2(f)(x0, y0) alpha f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
    cond2a := 1/2 = 2 theta alpha
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0))

```

```
), f(x0, y0));
```

$$\text{cond2b} := \frac{1}{2} = 2 \theta \alpha$$

```
> sol1:=solve({cond1a, cond2a, cond2b});
```

$$\text{sol1} := \{ \beta = 1 - 2 \theta, \alpha = \frac{1}{4 \theta}, \theta = \theta \}$$

b)

condiciones para el orden 3

```
> Fi20:=expand(beta*k120+theta*k220+theta*k320);
```

$$\text{Fi20} := 2 \theta D_{1,1}(f)(x_0, y_0) \alpha^2 + 4 \theta D_{1,2}(f)(x_0, y_0) \alpha^2 f(x_0, y_0) + 2 \theta \alpha^2 f(x_0, y_0)^2 D_{2,2}(f)(x_0, y_0) \\ + 2 \theta D_2(f)(x_0, y_0) \alpha^2 D_1(f)(x_0, y_0) + 2 \theta D_2(f)(x_0, y_0)^2 \alpha^2 f(x_0, y_0)$$

```
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$\text{cond3a} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$\text{cond3b} := \frac{1}{3} = 2 \theta \alpha^2$$

```
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$\text{cond3c} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$\text{cond3d} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \theta \alpha^2$$

```
> sol2:=solve({cond1a, cond2a, cond2b, cond3a, cond3b, cond3c, cond3d, cond3e}, {alpha, beta, theta});
```

$$\text{sol2} := \{ \alpha = \frac{2}{3}, \theta = \frac{3}{8}, \beta = \frac{1}{4} \}$$

metodo de mayor orden, orden 3,

0 |

2/3 | 2/3

2/3 | 0 2/3

| 1/4 3/8 3/8

que suele llamarse de NYSTRÖM de tercer orden

es sabido que los métodos de 3 etapas y orden máximo 3 tienen función de estabilidad

$$r(z) = 1 + z + \frac{1}{2} z^2 + \frac{1}{6} z^3$$

o, sea, cuatro primeros términos del desarrollo de e^z

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-10 (del EXAMEN ORDINARIO de 24ENE11)

Considérese la familia de métodos de RUNGE-KUTTA de 3 etapas y explícitos dados por el tablero

0				
3/2		β		
0		α	-1	

		1/3	1/3	1/3

a) Dígase para qué valores de α y β los métodos son de orden 2

b) De entre los anteriores métodos, búsquense los que minimizan el coeficiente del término del error.

(desde aquí es Ejercicio del capítulo 8)

c) Calcúlese la función de estabilidad de los métodos de orden 2

```
> restart;
> interface(labeling=false):
> y1:=f(x,y(x));
                                     y1 := f(x, y(x))
> y2:=subs(diff(y(x),x)=y1,diff(y1,x));
                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
> y3:=subs(diff(y(x),x)=y1,diff(y2,x));
y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
      + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
> Y1:=subs({y(x)=y0,x=x0},y1);
                                     Y1 := f(x0, y0)
> Y2:=subs({y(x)=y0,x=x0},y2);
                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
> Y3:=subs({y(x)=y0,x=x0},y3);
Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
      + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
> k1_:=f(x0,y0);
                                     k1_ := f(x0, y0)
> k2_:=f(x0+alpha*h,y0+alpha*h*k1_);
                                     k2_ := f(x0 + alpha h, y0 + alpha h f(x0, y0))
> k3_:=f(x0+h,y0+(1/2)*h*k1_+(1/2)*h*k2_);
                                     k3_ := f(x0 + h, y0 + 1/2 h f(x0, y0) + 1/2 h f(x0 + alpha h, y0 + alpha h f(x0, y0)))
> k1_0:=subs(h=0,k1_);
                                     k1_0 := f(x0, y0)
> k2_0:=subs(h=0,k2_);
```

```

[
> k3_0:=subs(h=0,k3_);                                k2_0 := f(x0, y0)
[
> k11:=diff(k1_,h);                                    k3_0 := f(x0, y0)
[
> k21:=diff(k2_,h);                                    k11 := 0
[
> k31:=diff(k3_,h);
k21 := D1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0)
k31 := D1(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ +
D2(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ ⎛ $\frac{1}{2}$  f(x0, y0) +  $\frac{1}{2}$  f(x0 + α h, y0 + α h f(x0, y0))
+  $\frac{1}{2}$  h (D1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0))⎞
[
> k110:=subs(h=0,k11);                                k110 := 0
[
> k210:=subs(h=0,k21);
k210 := D1(f)(x0, y0) α + D2(f)(x0, y0) α f(x0, y0)
[
> k310:=subs(h=0,k31);
k310 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
[
> k12:=diff(k11,h);                                    k12 := 0
[
> k22:=diff(k21,h);
k22 := (D1,1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D1,2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0)) α
+ (D1,2(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2,2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0)) α f(x0, y0)
[
> k32:=diff(k31,h);
k32 := D1,1(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ +
D1,2(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ ⎛ $\frac{1}{2}$  f(x0, y0) +  $\frac{1}{2}$  f(x0 + α h, y0 + α h f(x0, y0))
+  $\frac{1}{2}$  h (D1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0))⎞⎞ + ⎛
D1,2(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞⎞ +
D2,2(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ ⎛ $\frac{1}{2}$  f(x0, y0) +  $\frac{1}{2}$  f(x0 + α h, y0 + α h f(x0, y0))
+  $\frac{1}{2}$  h (D1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0))⎞⎞⎞ ⎛ $\frac{1}{2}$  f(x0, y0)
+  $\frac{1}{2}$  f(x0 + α h, y0 + α h f(x0, y0))
+  $\frac{1}{2}$  h (D1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0))⎞⎞ +
D2(f)⎛x0 + h, y0 +  $\frac{1}{2}$  h f(x0, y0) +  $\frac{1}{2}$  h f(x0 + α h, y0 + α h f(x0, y0))⎞ ⎛D1(f)(x0 + α h, y0 + α h f(x0, y0)) α
+ D2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0) +  $\frac{1}{2}$  h (
(D1,1(f)(x0 + α h, y0 + α h f(x0, y0)) α + D1,2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0)) α
+ (D1,2(f)(x0 + α h, y0 + α h f(x0, y0)) α + D2,2(f)(x0 + α h, y0 + α h f(x0, y0)) α f(x0, y0)) α f(x0, y0))⎞
[
> k120:=subs(h=0,k12);                                k120 := 0
[
> k220:=subs(h=0,k22);
k220 := (D1,1(f)(x0, y0) α + D1,2(f)(x0, y0) α f(x0, y0)) α + (D1,2(f)(x0, y0) α + D2,2(f)(x0, y0) α f(x0, y0)) α f(x0, y0)

```

```

> k320:=subs(h=0,k32);
k320 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
+ D2(f)(x0, y0) (D1(f)(x0, y0) α + D2(f)(x0, y0) α f(x0, y0))
a)
condiciones para el orden 1
> Fi_0:=expand(beta*k1_0+0*k2_0+(1-beta)*k3_0);
Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
cond1a := 1 = 1
condiciones para el orden 2
> Fi10:=expand(beta*k110+0*k210+(1-beta)*k310);
Fi10 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0) - β D1(f)(x0, y0) - β D2(f)(x0, y0) f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
cond2a :=  $\frac{1}{2} = 1 - \beta$ 
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
cond2b :=  $\frac{1}{2} = 1 - \beta$ 
> sol2a:=solve({cond2a,cond2b},{beta});
sol2a := {β =  $\frac{1}{2}$ }
> beta:=subs(sol2a,beta);
β :=  $\frac{1}{2}$ 
b)
coeficientes del termino  $\frac{1}{3}f^2 - \Phi'(0)$  para el orden 3
> Fi20:=expand(beta*k120+0*k220+(1-beta)*k320);
Fi20 :=  $\frac{1}{2} D_{1,1}(f)(x0, y0) + D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{2} D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{1}{2} D_2(f)(x0, y0) D_1(f)(x0, y0) \alpha$ 
+  $\frac{1}{2} D_2(f)(x0, y0)^2 \alpha f(x0, y0)$ 
> tau3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))-coeff(Fi20,D[1,1](f)(x0,y0));
tau3a :=  $-\frac{1}{6}$ 
> tau3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))-coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
tau3b :=  $-\frac{1}{3}$ 
> tau3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)-coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
tau3c :=  $-\frac{1}{6}$ 
> tau3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
tau3d :=  $\frac{1}{3} - \frac{\alpha}{2}$ 
> tau3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
tau3e :=  $\frac{1}{3} - \frac{\alpha}{2}$ 
> sol3:=solve({tau3d,tau3e},{alpha});

```

$$\text{sol3} := \left\{ \alpha = \frac{2}{3} \right\}$$

[c)

[función de estabilidad para orden 2

[> restart:

[> interface(labeling=false):

[> beta:=1/2:

[> f:=(x,y)->lambda*y;

$$f := (x, y) \rightarrow \lambda y$$

[> k1_:=f(x0,y0);

$$k1_ := \lambda y0$$

[> k2_:=f(x0+alpha*h,y0+alpha*h*k1_);

$$k2_ := \lambda (y0 + \alpha h \lambda y0)$$

[> k3_:=f(x0+h,y0+(1/2)*h*k1_+(1/2)*h*k2_);

$$k3_ := \lambda \left(y0 + \frac{h \lambda y0}{2} + \frac{h \lambda (y0 + \alpha h \lambda y0)}{2} \right)$$

[> y1:=expand(y0+h*(beta*k1_+0*k2_+(1-beta)*k3_));

$$y1 := y0 + h \lambda y0 + \frac{h^2 \lambda^2 y0}{2} + \frac{h^3 \lambda^3 \alpha y0}{4}$$

[> y1:=collect(subs(lambda=z/h,y1),y0);

$$y1 := \left(1 + z + \frac{1}{2} z^2 + \frac{1}{4} z^3 \alpha \right) y0$$

[> r:=unapply(collect(simplify(y1/y0),z),z);

$$r := z \rightarrow 1 + z + \frac{1}{2} z^2 + \frac{1}{4} z^3 \alpha$$

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-11

Vamos a hacer un test numérico sobre la relación entre el error global (GTE)

y el error local de truncación (LTE) en un problema concreto. Consideremos el problema

$$y' = -y + x + 1, \quad y(0) = 1$$

del que conocemos la verdadera solución

$$y(x) = \frac{x e^x + 1}{e^x}$$

Para este problema utilizamos un método numérico, por ejemplo, el

método del punto medio, que es un Runge-Kutta de 2 etapas y

orden 2 dado por el tablero

$$\begin{array}{c|c} 0 & \\ \hline 1/2 & 1/2 \\ \hline & 0 \quad 1 \end{array}$$

Para el problema citado y su verdadera solución, $y(x)$, vamos a utilizar

este método en el intervalo $[a,b] = [0,1]$, integrando buscando las aproximaciones

$y_0, y_1, y_2, \dots, y_N$ de $y(x)$ sobre el retículo $x_0, x_1, x_2, \dots, x_N$ de abscisas

definidas como $x_n = a + h n$ con $h = \frac{b-a}{N}$ (de manera que $x_0 = a, x_N = b$)

Por una parte tendremos en el paso $n+1$ el GTE definido como

$$E_{n+1} = y(x_{n+1}) - y_{n+1}$$

Por otra tendremos en el paso $n+1$ el LTE, definido para nuestro método

como

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h f(x_n + \frac{1}{2}h, y(x_n) + \frac{1}{2}h f(x_n, y(x_n)))$$

o sea, como lo que resulta cuando en el método sustituimos y_{n+1} por $y(x_{n+1})$ e

y_n por $y(x_n)$, donde $y(x)$ es la verdadera solución del problema. Recuérdese que

esto es lo mismo que la diferencia entre $y(x_{n+1})$ e y_{n+1} bajo la suposición

de que $y_n = y(x_n)$.

También (aunque por ahora no lo usaremos) vamos a introducir el llamado error local (LE),
definido como

$$L_{n+1} = z(x_{n+1}) - y_{n+1}$$

donde $z(x)$ representa la solución exacta que verifica $z(x_n) = y_n$ en el nodo x_n

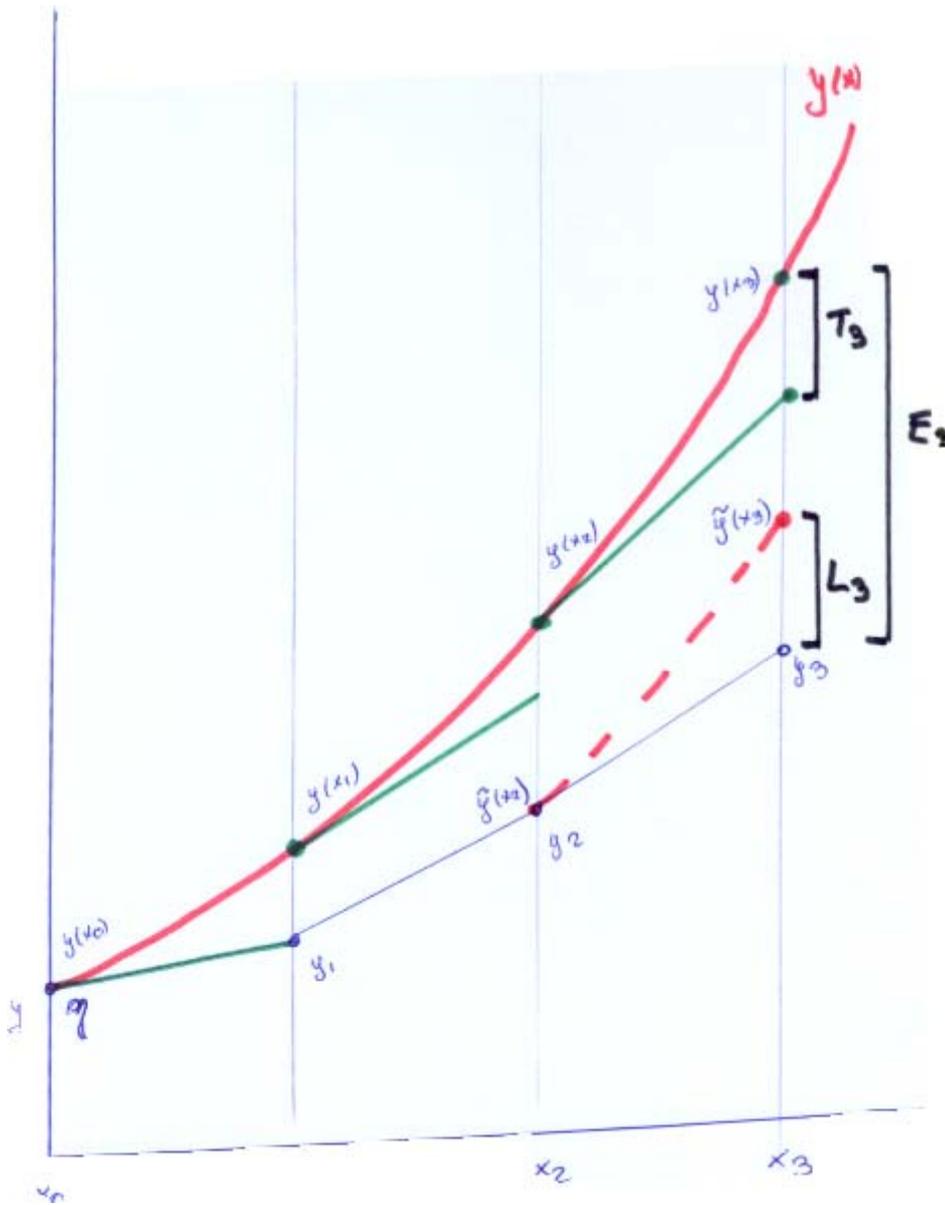
Integramos la ecuación en el intervalo citado para un número creciente de subintervalos

10, 100, 1000, ...

o sea, disminuyendo cada vez h a la décima parte (valores de h iguales a

$10^{(-1)}$, $10^{(-2)}$, $10^{(-3)}$, ...)

Se compararán los valores de E_N y de T_{N+1} obtenidos al disminuir h .



```
[ > restart:  
[ > interface(labeling=false):  
[ > Digits:=20:  
[
```


Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-12

Para integrar una ecuación escalar no autónoma se diseñan dos métodos numéricos de integración de ecuaciones diferenciales que designaremos con los nombres de M1 y M2 .

Para determinar cual es el más adecuado, se realiza la integración de un problema test entre 0 y 1 y se emplean los pasos

$$1, \frac{1}{2} = 0.5, \frac{1}{2^2} = 0.25, \frac{1}{2^3} = 0.125, \frac{1}{2^4} = 0.0625, \frac{1}{2^5} = 0.0313, \frac{1}{2^6} = 0.0156, \frac{1}{2^7} = 0.0078, \frac{1}{2^8} = 0.0039$$

Los números de pasos son, en todos los casos,

1, 2, 4, 8, 16, 32, 64, 128, 256

Los números de evaluaciones son para el M1

2, 4, 8, 16, 32, 64, 128, 256, 512

y para el M2

3, 6, 12, 24, 48, 96, 192, 384, 768

Para cada método se calcula el $\log[10]$ del error cometido en la aproximación de la solución en $x=1$. Los errores que se obtienen son,

para el M1

-0.8790, -1.643, -2.333, -2.977, -3.600, -4.213, -4.820, -5.424, -6.028

y para el M2

-1.462, -2.543, -3.534, -4.480, -5.405, -6.319, -7.228, -8.133, -9.038

Con estos datos, se deben construir las siguientes 'gráficas de eficiencia' que mezclen los resultados de M1 y M2 :

- La gráfica 'paso versus $\log[10]$ del error'.
- La gráfica 'número de evaluaciones versus $\log[10]$ del error'.
- Finalmente, se dirá qué método parece más adecuado.

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

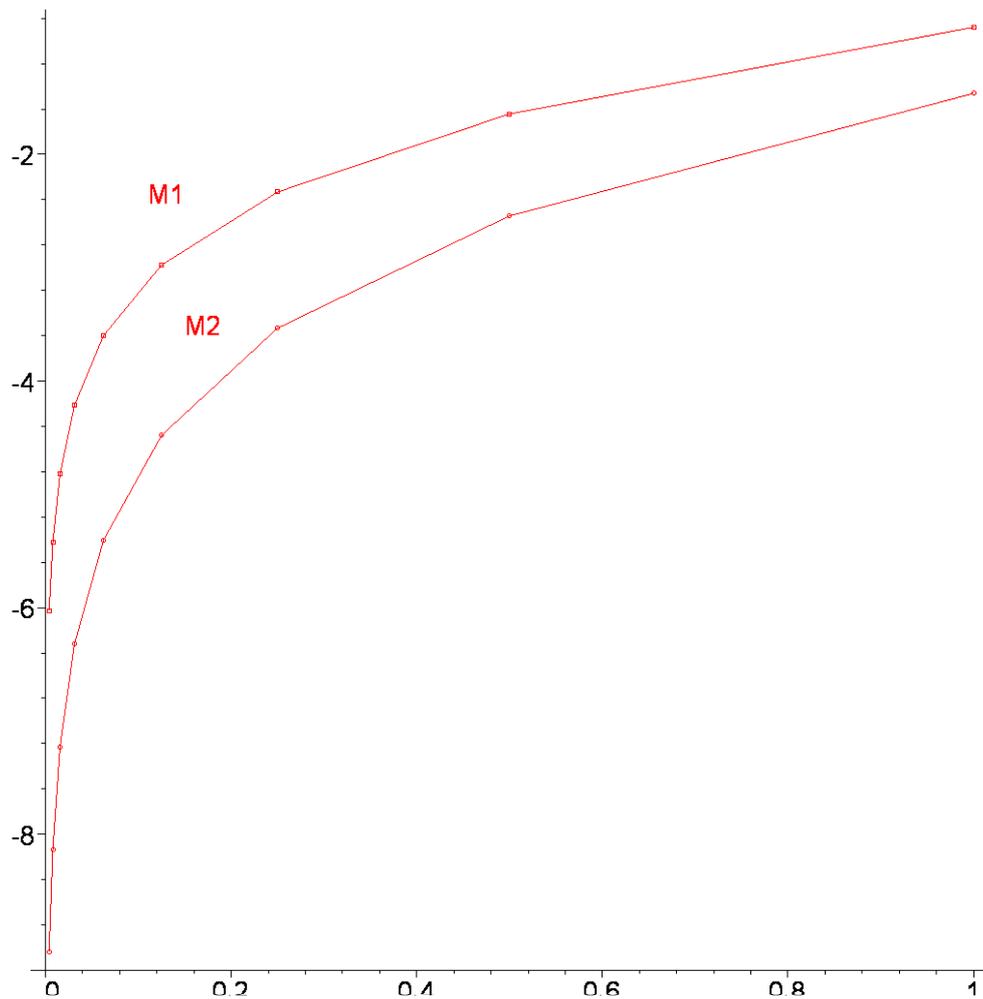
[Variables lista

```
[ > listanumpasos:=[0,0,0,0,0,0,0,0,0,0]:
[
```

```

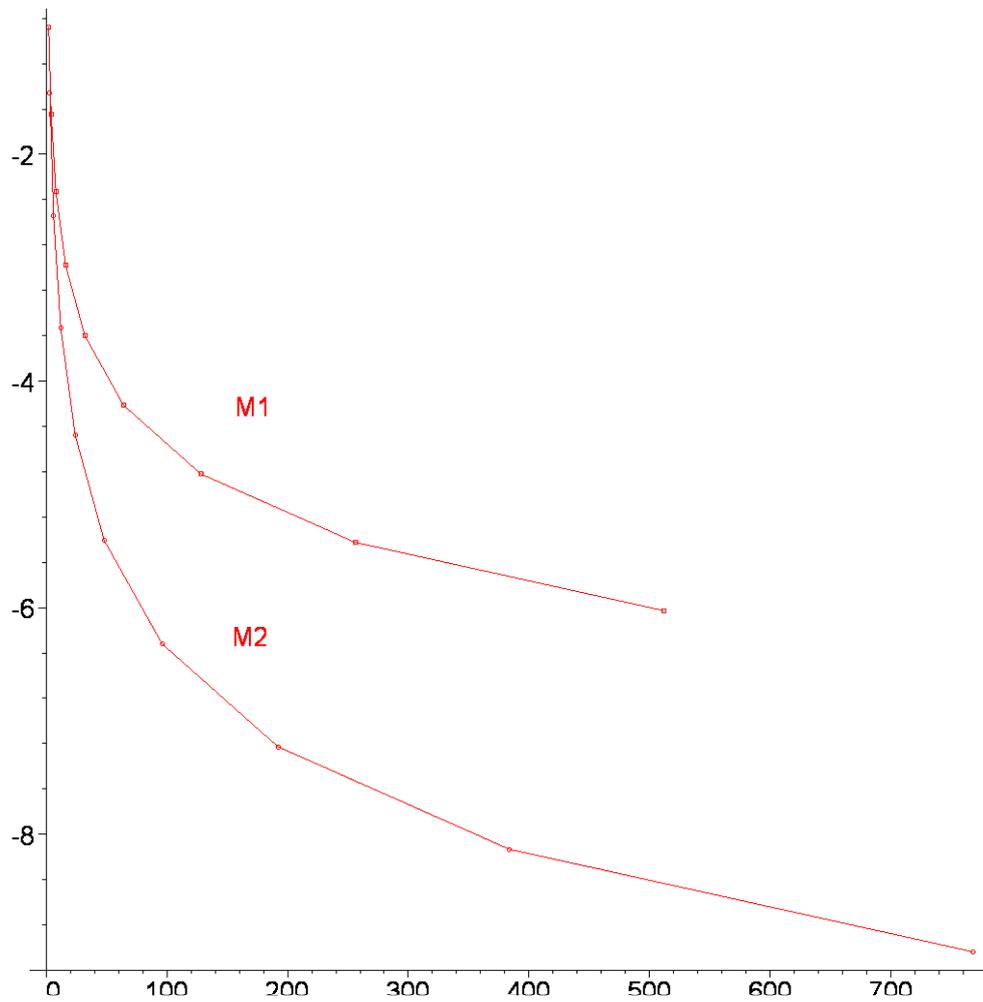
[ > listpaso:=[0,0,0,0,0,0,0,0,0]:
[ > listaeval:=[0,0,0,0,0,0,0,0,0]:
[ > lista1logerro:=[0,0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0,0,0,0]:
[ La ecuación (no aut.)
[ > listanumpasos:=[1 , 2 , 4 , 8 , 16 , 32 , 64 , 128 , 256]:
[ > for i from 1 to 9 do listapaso[i]:=1./listanumpasos[i]; od:
[ > f:=(x,y)->-y+x+1:ytotalini:=[0.,1.]:
[ > u:=x->(x*exp(x)+1)/exp(x):
[ M1 -> Runge-Kutta 2 evaluaciones y orden 2 'Heun' (e. no aut.)
[ > for i from 1 to 9 do listaeval[i]:=2*listanumpasos[i]; od:
[ > Heun2:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
  local x0,y0,k,k1,k2:
  x0:=yttotalini[1]:y0:=yttotalini[2]:
  for k from 1 to nit do
  k1:=f(x0,y0):
  k2:=f(x0+(2./3.)*h,y0+(2./3.)*h*k1):
  x0:=x0+h:y0:=y0+h*((1./4.)*k1+(3./4.)*k2):
  od:
  end:
[ > for i from 1 to 9 do y1:=Heun2(f,ytotalini,listanumpasos[i],evalf(listapaso[i])):
  lista1logerro[i]:=log[10](abs(y1-u(1.))): od:
[ > evalf(lista1logerro,4);
                                [-0.8790,-1.643,-2.333,-2.977,-3.600,-4.213,-4.820,-5.424,-6.028]
[ M2 -> Runge-Kutta 3 evaluaciones y orden 3 'Heun' (e. no aut.)
[ > for i from 1 to 9 do lista2eval[i]:=3*listanumpasos[i]; od:
[ > Heun3:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
  local x0,y0,k,k1,k2,k3:
  x0:=yttotalini[1]:y0:=yttotalini[2]:
  for k from 1 to nit do
  k1:=f(x0,y0):
  k2:=f(x0+(1./3.)*h,y0+(1./3.)*h*k1):
  k3:=f(x0+(2./3.)*h,y0+(2./3.)*h*k2):
  x0:=x0+h:y0:=y0+h*((1./4.)*k1+(3./4.)*k3):
  od:
  end:
[ > for i from 1 to 9 do y1:=Heun3(f,ytotalini,listanumpasos[i],evalf(listapaso[i])):
  lista2logerro[i]:=log[10](abs(y1-u(1.))): od:
[ > evalf(lista2logerro,4);
                                [-1.462,-2.543,-3.534,-4.480,-5.405,-6.319,-7.228,-8.133,-9.038]
[ a) Gráficas comparadas 'paso versus log[10] del error'
[ > listagraa1:=[seq([listapaso[i],lista1logerro[i]],i=1..9)]:
[ > graa11:=plot(listagraa1,style=LINE):
[ > graa12:=plot(listagraa1,style=POINT,symbol=BOX):
[ > graa13:=textplot([0.13, -2.34,`M1`]):
[ > listagraa2:=[seq([listapaso[i],lista2logerro[i]],i=1..9)]:
[ > graa21:=plot(listagraa2,style=LINE):
[ > graa22:=plot(listagraa2,style=POINT,symbol=CIRCLE):
[ > graa23:=textplot([0.1693, -3.503,`M2`]):
[ > display(graa11,graa12,graa13,graa21,graa22,graa23);

```



b) Gráficas comparadas 'número de evaluaciones versus log₁₀ del error'

```
[ > listagrab1:=seq([lista1eval[i],lista1logerro[i]],i=1..9]):
[ > grab11:=plot(listagrab1,style=LINE):
[ > grab12:=plot(listagrab1,style=POINT,symbol=BOX):
[ > grab13:=textplot([171.5, -4.22,`M1`]):
[ > listagrab2:=seq([lista2eval[i],lista2logerro[i]],i=1..9]):
[ > grab21:=plot(listagrab2,style=LINE):
[ > grab22:=plot(listagrab2,style=POINT,symbol=CIRCLE):
[ > grab23:=textplot([169.2, -6.25,`M2`]):
[ > display(grab11,grab12,grab13,grab21,grab22,grab23);
```



c) Método más adecuado

El método M2 funciona mejor en cuanto el paso se hace adecuadamente pequeño, lo que corresponde al mayor número de evaluaciones de la función.

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-13

El problema escalar $y' = -y + x + 1$, $y(0) = 1$, del que se conoce la verdadera solución

$$y(x) = \frac{x e^x + 1}{e^x}$$

se integra numéricamente en el intervalo $[0, 1]$ con los métodos RK2, RK3 y RK4, que son los siguientes métodos

de Runge-Kutta de 2, 3 y 4 evaluaciones, y órdenes 2, 3 y 4:

- el método de Heun de dos evaluaciones, con tablero

0		
2/3		2/3

		1/4 3/4

- el método de Heun de tres evaluaciones, con tablero

0			
1/3		1/3	
2/3		0	2/3

		1/4	0 3/4

- el método clásico de Runge-Kutta de cuatro evaluaciones, con tablero

0				
1/2		1/2		
1/2		0	1/2	
1		0	0	1

		1/6	1/3	1/3 1/6

Las listas de 8 datos que se consideran corresponden a los ocho pasos

$$\frac{1}{2^4} = 0.625, \frac{1}{2^5} = 0.03125, \frac{1}{2^6} = 0.01563, \frac{1}{2^7} = 0.007813, \frac{1}{2^8} = 0.003906, \frac{1}{2^9} = 0.001953, \frac{1}{2^{10}} = 0.0009766, \frac{1}{2^{11}} = 0.0004883.$$

Los números de pasos son, en todos los casos,

16 , 32 , 64 , 128 , 256 , 512 , 1024 , 2048

Los números de evaluaciones son para el RK2

32 , 64 , 128 , 256 , 512 , 1024 , 2048 , 4096

para el RK3

48 , 96 , 192 , 384 , 768 , 1536 , 3072 , 6144

y para el RK4

64 , 128 , 256 , 512 , 1024 , 2048 , 4096 , 8192

Para cada método se calcula el \log_{10} del error cometido.

Como podemos ver, para nuestros métodos, vamos obteniendo los siguientes logaritmos de los errores; para el RK2

-3.600, -4.213, -4.820, -5.424, -6.028, -6.630, -7.233, -7.835

para el RK3

-5.405, -6.319, -7.228, -8.133, -9.038, -9.942, -10.85, -11.75

y para el RK4

-7.307, -8.523, -9.733, -10.94, -12.15, -13.35, -14.55, -15.76

Con estos datos, se deben construir las siguientes 'gráficas de eficiencia' que mezcles los resultados de RK2 , RK3 y RK4 :

a) La gráfica 'paso versus \log_{10} del error'.

b) La gráfica 'número de evaluaciones versus \log_{10} del error'.

c) La gráfica ' \log_{10} del paso versus \log_{10} del error'. En ésta se puede comprobar la tendencia a dar rectas con pendiente igual en módulo al orden, usando bien la recta de regresión de los datos, o mejor la pendiente de alguno de los últimos segmentos que componen la recta.

d) La gráfica ' \log_{10} del número de evaluaciones versus \log_{10} del error'. Convendrá establecer las mismas comprobaciones que en el apartado precedente.

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(stats):with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listanumpasos:=[0,0,0,0,0,0,0,0]:
[ > listpaso:=[0,0,0,0,0,0,0,0]:
[ > listalogpaso:=[0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0]:
[ > lista2logeval:=[0,0,0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0,0,0]:
[ > lista3eval:=[0,0,0,0,0,0,0,0]:
[ > lista3logeval:=[0,0,0,0,0,0,0,0]:
```

```
[ > lista3logerro:=[0,0,0,0,0,0,0,0]:
[ > lista4eval:=[0,0,0,0,0,0,0,0]:
[ > lista4logeval:=[0,0,0,0,0,0,0,0]:
[ > lista4logerro:=[0,0,0,0,0,0,0,0]:
```

[La ecuación (no aut.)

```
[ > listanumpasos:=[16,32,64,128,256,512,1024,2048]:
[ > for i from 1 to 8 do listapaso[i]:=1./listanumpasos[i]; od:
[ > for i from 1 to 8 do listalogpaso[i]:=evalf(log[10](listapaso[i])); od:
[ > f:=(x,y)->-y+x+1:ytotalini:=[0.,1.]:
[ > u:=x->(x*exp(x)+1)/exp(x):
```

[RK2 -> Runge-Kutta 2 evaluaciones y orden 2 'Heun' (e. no aut.)

```
[ > for i from 1 to 8 do lista2eval[i]:=2*listanumpasos[i]; od:
[ > for i from 1 to 8 do lista2logeval[i]:=evalf(log[10](lista2eval[i])); od:
[ > Heun2:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
  local x0,y0,k,k1,k2:
  x0:=ytotalini[1]:y0:=ytotalini[2]:
  for k from 1 to nit do
    k1:=f(x0,y0):
    k2:=f(x0+(2./3.)*h,y0+(2./3.)*h*k1):
    x0:=x0+h:y0:=y0+h*((1./4.)*k1+(3./4.)*k2):
  od:
end:
[ > for i from 1 to 8 do y1:=Heun2(f,ytotalini,listanumpasos[i],evalf(listapaso[i])):
  lista2logerro[i]:=log[10](abs(y1-u(1.))): od:
[ > evalf(lista2logerro,4);
[-3.600, -4.213, -4.820, -5.424, -6.028, -6.630, -7.233, -7.835]
```

[RK3 -> Runge-Kutta 3 evaluaciones y orden 3 'Heun' (e. no aut.)

```
[ > for i from 1 to 8 do lista3eval[i]:=3*listanumpasos[i]; od:
[ > for i from 1 to 8 do lista3logeval[i]:=evalf(log[10](lista3eval[i])); od:
[ > Heun3:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
  local x0,y0,k,k1,k2,k3:
  x0:=ytotalini[1]:y0:=ytotalini[2]:
  for k from 1 to nit do
    k1:=f(x0,y0):
    k2:=f(x0+(1./3.)*h,y0+(1./3.)*h*k1):
    k3:=f(x0+(2./3.)*h,y0+(2./3.)*h*k2):
    x0:=x0+h:y0:=y0+h*((1./4.)*k1+(3./4.)*k3):
  od:
end:
[ > for i from 1 to 8 do y1:=Heun3(f,ytotalini,listanumpasos[i],evalf(listapaso[i])):
  lista3logerro[i]:=log[10](abs(y1-u(1.))): od:
[ > evalf(lista3logerro,4);
[-5.405, -6.319, -7.228, -8.133, -9.038, -9.942, -10.85, -11.75]
```

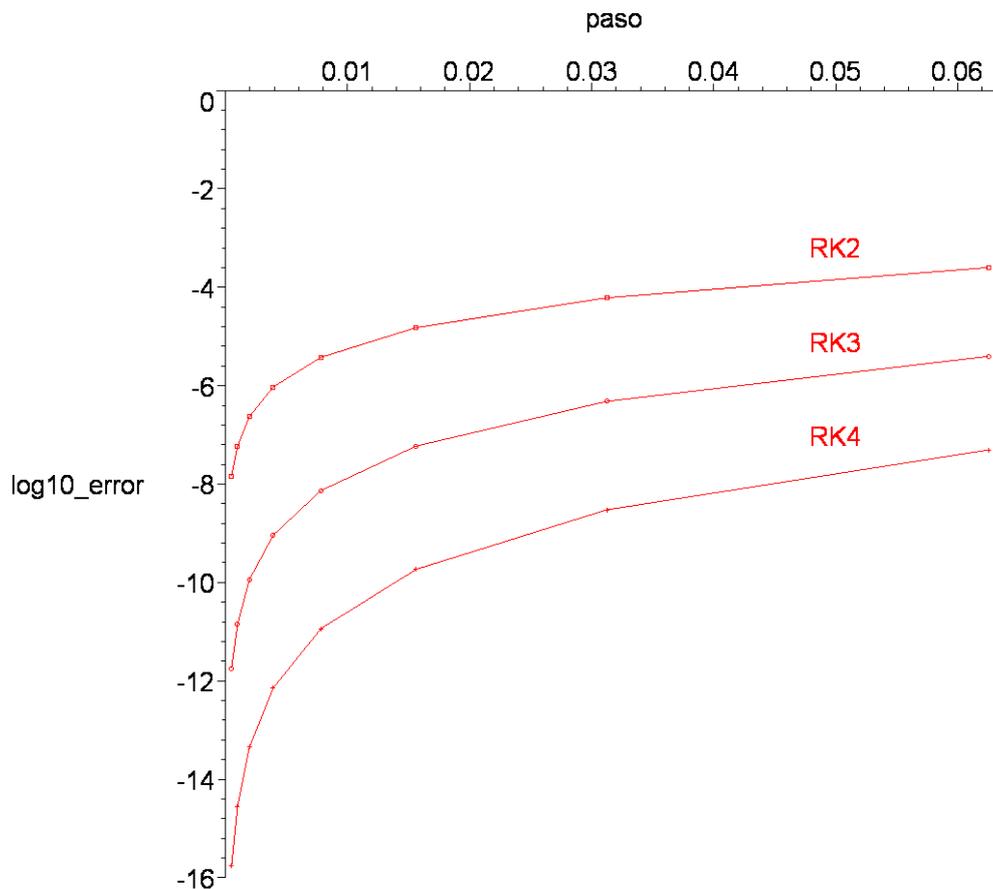
[RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. aut.)

```
[ > for i from 1 to 8 do lista4eval[i]:=4*listanumpasos[i]; od:
[ > for i from 1 to 8 do lista4logeval[i]:=evalf(log[10](lista4eval[i])); od:
[ > Rkut4:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
  local x0,y0,k,k1,k2,k3,k4:
  x0:=ytotalini[1]:y0:=ytotalini[2]:
  for k from 1 to nit do
    k1:=f(x0,y0):
    k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1):
    k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2):
    k4:=f(x0+h,y0+h*k3):
    x0:=x0+h:y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
  od:
end:
[ > for i from 1 to 8 do y1:=Rkut4(f,ytotalini,listanumpasos[i],evalf(listapaso[i])):
  lista4logerro[i]:=log[10](abs(y1-u(1.))): od:
[ > evalf(lista4logerro,4);
```

[-7.307, -8.523, -9.733, -10.94, -12.15, -13.35, -14.55, -15.76]

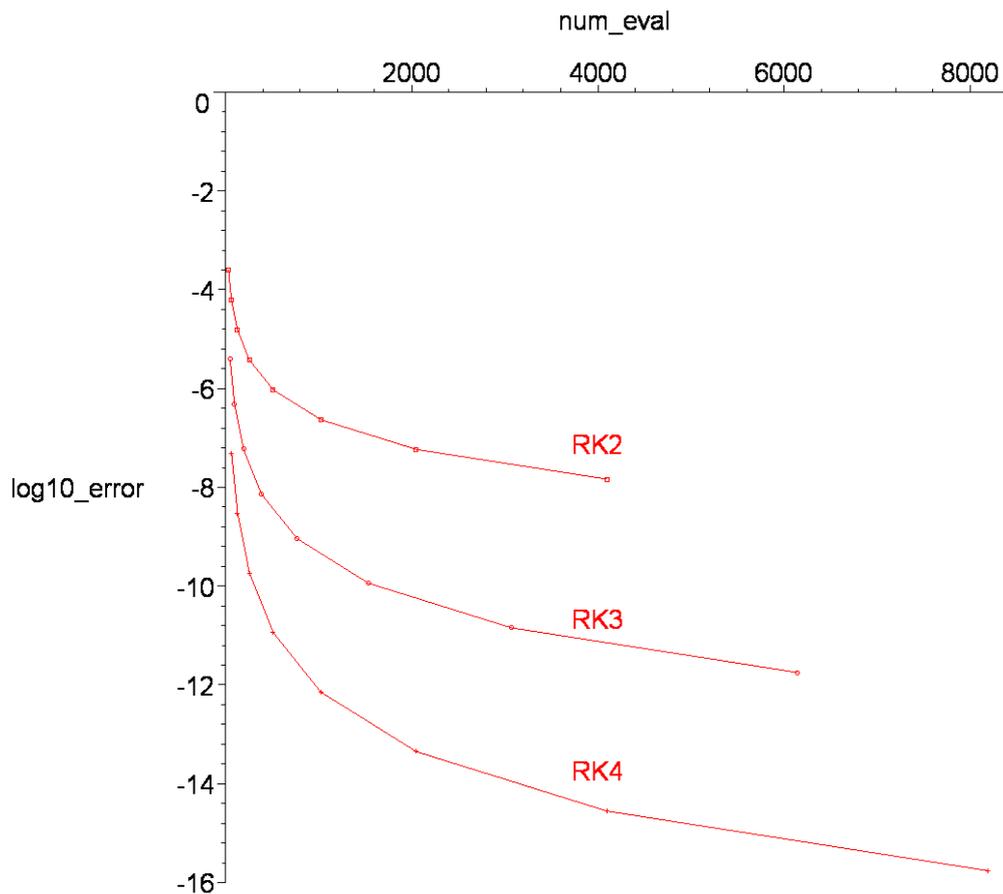
[a) Gráficas comparadas 'paso versus \log_{10} del error'

```
[ > listagraa2:= [seq([listapaso[i], lista2logerro[i]], i=1..8)]:  
[ > graa21:=plot(listagraa2, paso, log10_error=0..-16, style=LINE):  
[ > graa22:=plot(listagraa2, style=POINT, symbol=BOX):  
[ > graa23:=textplot([0.05, -3.17, `RK2`]):  
[ > listagraa3:= [seq([listapaso[i], lista3logerro[i]], i=1..8)]:  
[ > graa31:=plot(listagraa3, style=LINE):  
[ > graa32:=plot(listagraa3, style=POINT, symbol=CIRCLE):  
[ > graa33:=textplot([0.05, -5.11, `RK3`]):  
[ > listagraa4:= [seq([listapaso[i], lista4logerro[i]], i=1..8)]:  
[ > graa41:=plot(listagraa4, style=LINE):  
[ > graa42:=plot(listagraa4, style=POINT, symbol=CROSS):  
[ > graa43:=textplot([0.05, -7.01, `RK4`]):  
[ > display(graa21, graa22, graa23, graa31, graa32, graa33, graa41, graa42, graa43);
```



[b) Gráficas comparadas 'número de evaluaciones versus \log_{10} del error'

```
[ > listagrab2:= [seq([lista2eval[i], lista2logerro[i]], i=1..8)]:  
[ > grab21:=plot(listagrab2, num_eval, log10_error=0..-16, style=LINE):  
[ > grab22:=plot(listagrab2, style=POINT, symbol=BOX):  
[ > grab23:=textplot([4000, -7.11, `RK2`]):  
[ > listagrab3:= [seq([lista3eval[i], lista3logerro[i]], i=1..8)]:  
[ > grab31:=plot(listagrab3, style=LINE):  
[ > grab32:=plot(listagrab3, style=POINT, symbol=CIRCLE):  
[ > grab33:=textplot([4000, -10.65, `RK3`]):  
[ > listagrab4:= [seq([lista4eval[i], lista4logerro[i]], i=1..8)]:  
[ > grab41:=plot(listagrab4, style=LINE):  
[ > grab42:=plot(listagrab4, style=POINT, symbol=CROSS):  
[ > grab43:=textplot([4000, -13.72, `RK4`]):  
[ > display(grab21, grab22, grab23, grab31, grab32, grab33, grab41, grab42, grab43);
```



c) Gráficas comparadas 'log[10] del paso versus log[10] del error'

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los más representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del número de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta

es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

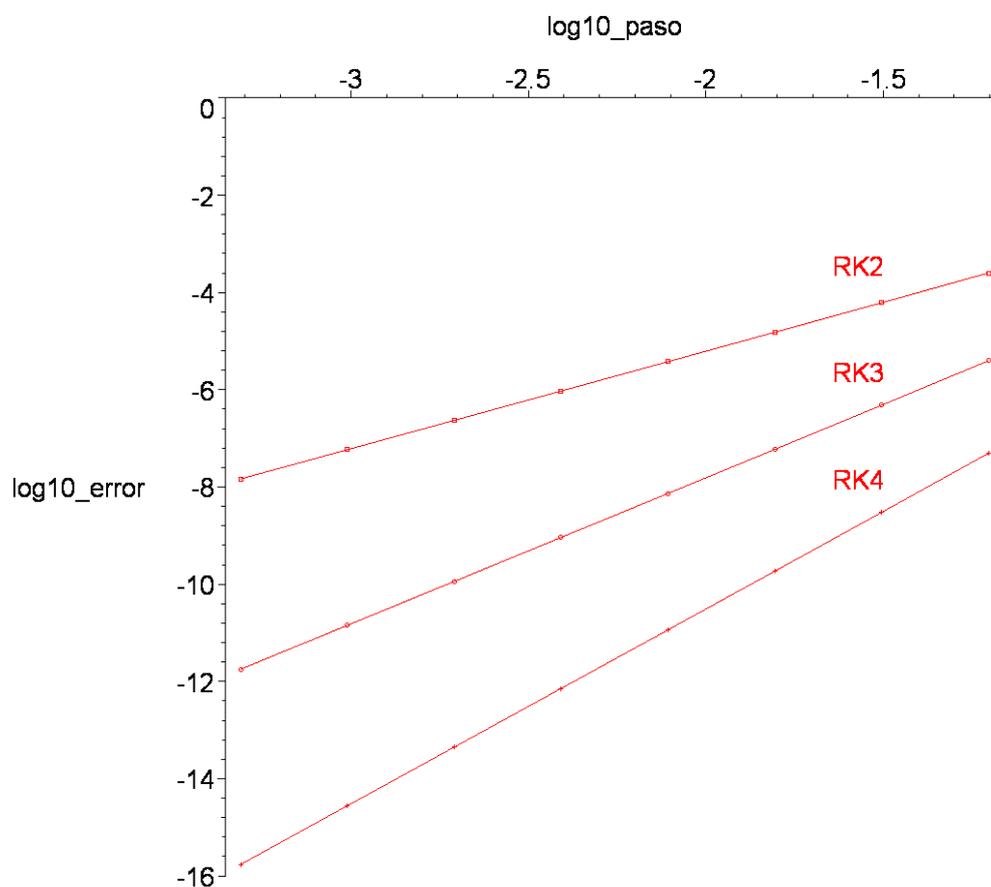
$$\log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) = \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) =$$

$$= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2)$$

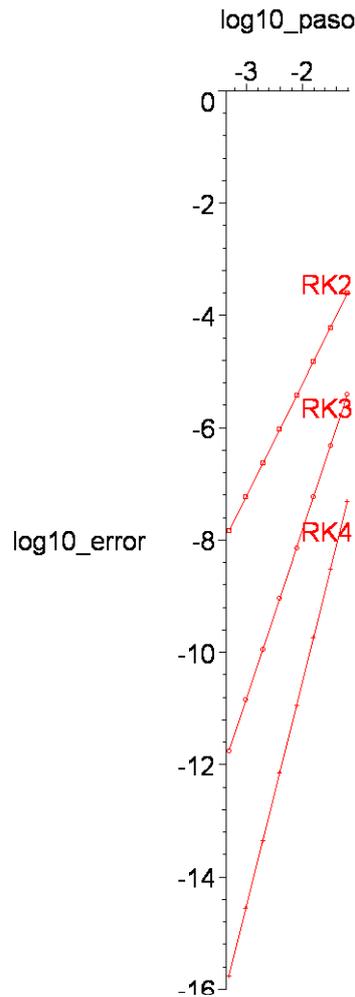
en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

```
[ > listagrac2=[seq([listalogpaso[i],lista2logerro[i]],i=1..8)]:
[ > grac21:=plot(listagrac2,log10_paso,log10_error=0..-16,style=LINE):
[ > grac22:=plot(listagrac2,style=POINT,symbol=BOX):
[ > grac23:=textplot([-1.57, -3.44,`RK2`]):
[ > listagrac3=[seq([listalogpaso[i],lista3logerro[i]],i=1..8)]:
[ > grac31:=plot(listagrac3,style=LINE):
[ > grac32:=plot(listagrac3,style=POINT,symbol=CIRCLE):
[ > grac33:=textplot([-1.57, -5.63,`RK3`]):
[ > listagrac4=[seq([listalogpaso[i],lista4logerro[i]],i=1..8)]:
[ > grac41:=plot(listagrac4,style=LINE):
[ > grac42:=plot(listagrac4,style=POINT,symbol=CROSS):
[ > grac43:=textplot([-1.57, -7.83,`RK4`]):
[ > display(grac21,grac22,grac23,grac31,grac32,grac33,grac41,grac42,grac43);
```



```
> display(grac21,grac22,grac23,grac31,grac32,grac33,grac41,grac42,grac43,scaling='CONSTRAINED');
```



Rectas de regresión

```
> fit[leastsquare[[x,y]]]([listalogpaso,lista2logerro]);
      y = -1.1888863124563252540 + 2.0081769984334828545 x
> fit[leastsquare[[x,y]]]([listalogpaso,lista3logerro]);
      y = -1.7895244832654788703 + 3.0086681635418645661 x
> fit[leastsquare[[x,y]]]([listalogpaso,lista4logerro]);
      y = -2.4855049398570456188 + 4.0101112987766774101 x
```

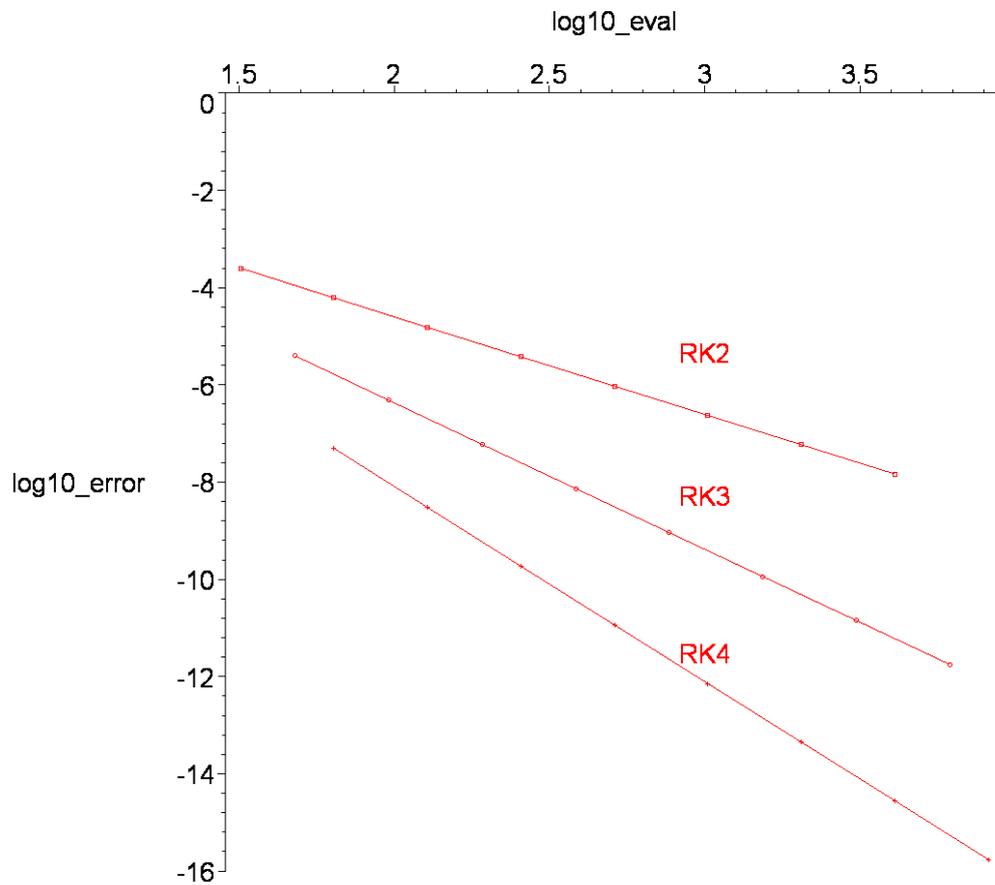
Pendiente del último segmento

```
> (lista2logerro[8]-lista2logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      2.000528435955558128
> (lista3logerro[8]-lista3logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      3.0005639795203120915
> (lista4logerro[8]-lista4logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      4.0129865767341455195
```

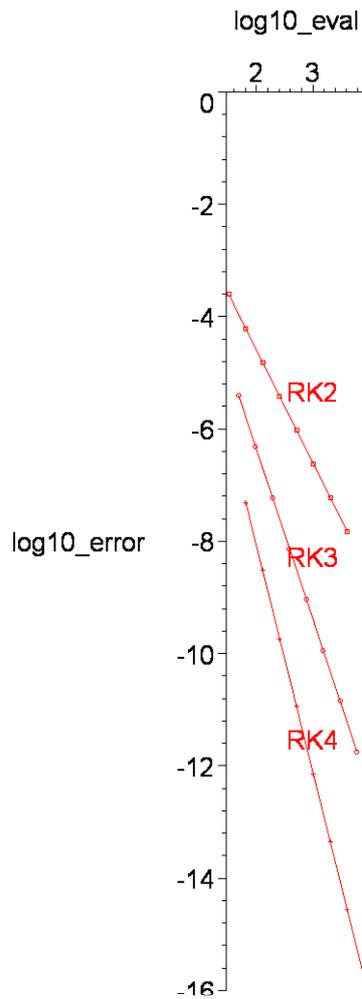
d) Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

```
> listagrad2:= [seq([lista2logeval[i],lista2logerro[i]],i=1..8)]:
> grad21:=plot(listagrad2,log10_eval,log10_error=0..-16,style=LINE):
> grad22:=plot(listagrad2,style=POINT,symbol=BOX):
> grad23:=textplot([3., -5.32, `RK2`]):
> listagrad3:= [seq([lista3logeval[i],lista3logerro[i]],i=1..8)]:
> grad31:=plot(listagrad3,style=LINE):
> grad32:=plot(listagrad3,style=POINT,symbol=CIRCLE):
> grad33:=textplot([3., -8.27, `RK3`]):
> listagrad4:= [seq([lista4logeval[i],lista4logerro[i]],i=1..8)]:
> grad41:=plot(listagrad4,style=LINE):
```

```
[ > grad42:=plot(listagrad4,style=POINT,symbol=CROSS):  
[ > grad43:=textplot([3., -11.51,`RK4`]):  
> display(grad21,grad22,grad23,grad31,grad32,grad33,grad41,grad42,grad43);
```



```
> display(grad21,grad22,grad23,grad31,grad32,grad33,grad41,grad42,grad43,scaling='CONSTRAINED');
```



[Rectas de regresión

```
[ > fit[leastsquare[[x,y]]]([lista2logeval,lista2logerro]);
      y = -0.58436479932538714443 - 2.0081769984334828523 x
[ > fit[leastsquare[[x,y]]]([lista3logeval,lista3logerro]);
      y = -0.35402495404128191086 - 3.0086681635418645634 x
[ > fit[leastsquare[[x,y]]]([lista4logeval,lista4logerro]);
      y = -0.071177366091395218034 - 4.0101112987766774104 x
```

[Pendiente del último segmento

```
[ > (lista2logerro[8]-lista2logerro[7])/(lista2logeval[8]-lista2logeval[7]);
      -2.0005284359555558135
[ > (lista3logerro[8]-lista3logerro[7])/(lista3logeval[8]-lista3logeval[7]);
      -3.0005639795203120925
[ > (lista4logerro[8]-lista4logerro[7])/(lista4logeval[8]-lista4logeval[7]);
      -4.0129865767341455208
```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-14

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos para los que, además, $c_2 = c_3$ y $b_2 = b_3$ y $a_{3,1} = 0$. De entre ellos se buscaran los que proporcionen el orden mas alto.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > c[3]:=c[2]:b[3]:=b[2]:a[3,1]:=0:
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+c[2]*h,y0+a[2,1]*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0+b[3]*k3_0);
[                               Fi_0 := b_1 f(x0, y0) + 2 b_2 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                               cond1a := 1 = b_1 + 2 b_2
[ condiciones para el orden 2
[ > Fi10:=expand(b[1]*k110+b[2]*k210+b[3]*k310);
[                               Fi10 := 2 b_2 D_1(f)(x0, y0) c_2 + b_2 D_2(f)(x0, y0) a_{2,1} f(x0, y0) + b_2 D_2(f)(x0, y0) a_{3,2} f(x0, y0)
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                               cond2a := 1/2 = 2 b_2 c_2
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                               cond2b := 1/2 = b_2 a_{2,1} + b_2 a_{3,2}
```

```
> sol1:=solve({cond1a,cond2a,cond2b});
```

$$sol1 := \{ a_{3,2} = -\frac{1}{2} \frac{-1 + 2 b_2 a_{2,1}}{b_2}, b_1 = 1 - 2 b_2, c_2 = \frac{1}{4} \frac{1}{b_2}, b_2 = b_2, a_{2,1} = a_{2,1} \}$$

b)

condiciones para el orden 3

```
> Fi20:=expand(b[1]*k120+b[2]*k220+b[3]*k320);
```

$$Fi20 := 2 b_2 D_{1,1}(f)(x0, y0) c_2^2 + 2 b_2 c_2 D_{1,2}(f)(x0, y0) a_{2,1} f(x0, y0) + b_2 a_{2,1}^2 f(x0, y0)^2 D_{2,2}(f)(x0, y0) \\ + 2 b_2 c_2 D_{1,2}(f)(x0, y0) a_{3,2} f(x0, y0) + b_2 a_{3,2}^2 f(x0, y0)^2 D_{2,2}(f)(x0, y0) + 2 b_2 D_2(f)(x0, y0) a_{3,2} D_1(f)(x0, y0) c_2 \\ + 2 b_2 D_2(f)(x0, y0)^2 a_{3,2} a_{2,1} f(x0, y0)$$

```
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = b_2 c_2^2$$

```
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = b_2 c_2 a_{2,1} + b_2 c_2 a_{3,2}$$

```
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{2} b_2 a_{2,1}^2 + \frac{1}{2} b_2 a_{3,2}^2$$

```
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = b_2 c_2 a_{3,2}$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{6} = b_2 a_{3,2} a_{2,1}$$

```
> sol2:=solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
```

$$sol2 := \{ c_2 = \frac{2}{3}, b_2 = \frac{3}{8}, b_1 = \frac{1}{4}, a_{2,1} = \frac{2}{3}, a_{3,2} = \frac{2}{3} \}$$

metodo

0 |

2/3 | 2/3

2/3 | 0 2/3

| 1/4 3/8 3/8

que suele llamarse de NYSTRÖM de tercer orden

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-15 (del EXAMEN EXTRAORDINARIO de 07JUL12)

Se considera el método de Runge-Kutta de 3 etapas dado

por el tablero

1/2		0		
1/2		1/2		
1		0	1	

		1/3	1/3	1/3

para el que debe notarse que $f(x_n + \frac{1}{2}h, y_n)$

Búsqese el orden de consistencia que tiene este método.

Búsqese también el orden del método para el caso escalar autónomo,

o sea, el caso en el que $f(x,y)=f(y)$.

Compruébese que, para el problema 'test' $y' = \lambda y$, el método posee exactamente orden 3.

[a) ECUACION COMPLETA

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0+1/2*h,y0):
[ > k2_:=f(x0+1/2*h,y0+1/2*h*k1_):
[ > k3_:=f(x0+h,y0+h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[condiciones para el orden 1

```
> Fi_0:=expand(1/3*k1_0+1/3*k2_0+1/3*k3_0);  
  
Fi_0 := f(x0, y0)  
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));  
  
cond1a := 1 = 1
```

[condiciones para el orden 2

```
> Fi10:=expand(1/3*k110+1/3*k210+1/3*k310);  
  
Fi10 :=  $\frac{2}{3}D_1(f)(x0, y0) + \frac{1}{2}D_2(f)(x0, y0) f(x0, y0)$   
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));  
  
cond2a :=  $\frac{1}{2} = \frac{2}{3}$   
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));  
  
cond2b :=  $\frac{1}{2} = \frac{1}{2}$ 
```

[b) ECUACION AUTONOMA

```
> restart;  
> interface(labeling=false):  
> f:=unapply(g(y),x,y):  
> y1:=f(x,y(x)):  
> y2:=subs(diff(y(x),x)=y1,diff(y1,x)):  
> y3:=subs(diff(y(x),x)=y1,diff(y2,x)):  
> Y1:=subs({y(x)=y0,x=x0},y1):  
> Y2:=subs({y(x)=y0,x=x0},y2):  
> Y3:=subs({y(x)=y0,x=x0},y3):  
> k1_:=f(x0+1/2*h,y0):  
> k2_:=f(x0+1/2*h,y0+1/2*h*k1_):  
> k3_:=f(x0,y0+h*k2_):  
> k1_0:=subs(h=0,k1_):  
> k2_0:=subs(h=0,k2_):  
> k3_0:=subs(h=0,k3_):  
> k11:=diff(k1_,h):  
> k21:=diff(k2_,h):  
> k31:=diff(k3_,h):  
> k110:=subs(h=0,k11):  
> k210:=subs(h=0,k21):  
> k310:=subs(h=0,k31):  
> k12:=diff(k11,h):  
> k22:=diff(k21,h):  
> k32:=diff(k31,h):  
> k120:=subs(h=0,k12):  
> k220:=subs(h=0,k22):  
> k320:=subs(h=0,k32):
```

[condiciones para el orden 1

```
> Fi_0:=expand(1/3*k1_0+1/3*k2_0+1/3*k3_0);  
  
Fi_0 := g(y0)  
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));  
  
cond1a := 1 = 1
```

[condiciones para el orden 2

```
> Fi10:=expand(1/3*k110+1/3*k210+1/3*k310);  
  
Fi10 :=  $\frac{1}{2}D(g)(y0) g(y0)$   
> #cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));  
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = \frac{1}{2}$$

condiciones para el orden 3

```
> Fi20:=expand(1/3*k120+1/3*k220+1/3*k320);
```

$$Fi20 := \frac{5}{12} (D^{(2)})(g)(y0) g(y0)^2 + \frac{1}{3} D(g)(y0)^2 g(y0)$$

```
> #cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

```
> #cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

```
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{5}{24}$$

```
> #cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{6} = \frac{1}{6}$$

b) ECUACION TEST

```
> restart:
```

```
> interface(labeling=false):
```

```
> g:=unapply(lambda*y,y):
```

```
> f:=unapply(g(y),x,y):
```

```
> y1:=f(x,y(x)):
```

```
> y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
```

```
> y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
```

```
> Y1:=subs({y(x)=y0,x=x0},y1):
```

```
> Y2:=subs({y(x)=y0,x=x0},y2):
```

$$Y2 := \lambda^2 y0$$

```
> Y3:=subs({y(x)=y0,x=x0},y3):
```

```
> k1_:=f(x0,y0):
```

```
> k2_:=f(x0+1/2*h,y0+1/2*h*k1_):
```

```
> k3_:=f(x0,y0+h*k2_):
```

```
> k1_0:=subs(h=0,k1_):
```

```
> k2_0:=subs(h=0,k2_):
```

```
> k3_0:=subs(h=0,k3_):
```

```
> k11:=diff(k1_,h):
```

```
> k21:=diff(k2_,h):
```

```
> k31:=diff(k3_,h):
```

```
> k110:=subs(h=0,k11):
```

```
> k210:=subs(h=0,k21):
```

```
> k310:=subs(h=0,k31):
```

```
> k12:=diff(k11,h):
```

```
> k22:=diff(k21,h):
```

```
> k32:=diff(k31,h):
```

```
> k120:=subs(h=0,k12):
```

```
> k220:=subs(h=0,k22):
```

```
> k320:=subs(h=0,k32):
```

condiciones para el orden 1

```
> Fi_0:=expand(1/3*k1_0+1/3*k2_0+1/3*k3_0);
```

$$Fi_0 := \lambda y0$$

```
> cond1a:=coeff(Y1,y0)=coeff(Fi_0,y0);
```

$$cond1a := \lambda = \lambda$$

condiciones para el orden 2

```
> Fi10:=expand(1/3*k110+1/3*k210+1/3*k310);
```

$$Fi10 := \frac{\lambda^2 y0}{2}$$

```
[ > #cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[ > cond2b:=(1/2)*coeff(Y2,y0)=coeff(Fi10,y0);
```

$$\text{cond2b} := \frac{\lambda^2}{2} = \frac{\lambda^2}{2}$$

[condiciones para el orden 3

```
[ > Fi20:=expand(1/3*k120+1/3*k220+1/3*k320);
```

$$\text{Fi20} := \frac{\lambda^3 y_0}{3}$$

```
[ > #cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

```
[ > #cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

```
[ > cond3c:=(1/6)*coeff(Y3,y0)=(1/2)*coeff(Fi20,y0);
```

$$\text{cond3c} := \frac{\lambda^3}{6} = \frac{\lambda^3}{6}$$

```
[ > #cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

```
[ > cond3e:=(1/6)*coeff(Y3,y0)=(1/2)*coeff(Fi20,y0);
```

$$\text{cond3e} := \frac{\lambda^3}{6} = \frac{\lambda^3}{6}$$

[función de estabilidad para este metodo

```
[ > restart;
```

```
[ > interface(labeling=false);
```

```
[ > f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
[ > k1:=f(x0+(1/2)*h,y0,h);
```

$$k1 := \lambda y_0$$

```
[ > k2:=unapply(f(x0+(1/2)*h,y0+(1/2)*h*k1),h);
```

$$k2 := h \rightarrow \lambda \left(y_0 + \frac{1}{2} h \lambda y_0 \right)$$

```
[ > k3:=unapply(f(x0+h,y0+h*k2(h)),h);
```

$$k3 := h \rightarrow \lambda \left(y_0 + h \lambda \left(y_0 + \frac{1}{2} h \lambda y_0 \right) \right)$$

```
[ > y1:=expand(y0+h*((1/3)*k1+(1/3)*k2(h)+(1/3)*k3(h)));
```

$$y1 := y_0 + h \lambda y_0 + \frac{1}{2} h^2 \lambda^2 y_0 + \frac{1}{6} h^3 \lambda^3 y_0$$

```
[ > y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + z + \frac{1}{2} z^2 + \frac{1}{6} z^3 \right) y_0$$

```
[ > r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + z + \frac{1}{2} z^2 + \frac{1}{6} z^3$$

```
[ >
```

```
[ >
```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-16

Para el problema tipo de Cauchy

$$y' = x y + x^2$$

$$y(0) = 1,$$

(de solución exacta $-x + \frac{1}{2} e^{\frac{x^2}{2}} \left(\frac{1}{\pi} \operatorname{erf}\left(\frac{x}{\sqrt{2}}\right) + 1 \right)$, donde erf representa la función

de error $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$) utilícese el método de TAYLOR de orden 3

para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
> restart;  
> interface(labeling=false);  
> Digits:=3;  
La ecuación (aut.) y la verdadera solución  
> f:=(x,y)->x*y+x^2:yv:=x->-x+1/2*exp(1/2*x^2)*Pi^(1/2)*2^(1/2)*erf(1/2*2^(1/2)*x)+1.*exp(1/2*x^2);
```

Los métodos de TAYLOR y el de orden 3

Los métodos de TAYLOR para los primeros órdenes se escriben

$$y_{n+1} = y_n + h * T^{[1]}(x_n, y_n, h)$$

$$y_{n+1} = y_n + h * T^{[2]}(x_n, y_n, h)$$

$$y_{n+1} = y_n + h * T^{[3]}(x_n, y_n, h)$$

donde $T^{[1]}$, $T^{[2]}$ y $T^{[3]}$ representan respectivamente

$$T^{[1]}(x_n, y_n, h) = f^{[0]}(x_n, y_n)$$

$$T^{[2]}(x_n, y_n, h) = f^{[0]}(x_n, y_n) + \frac{h}{2} f^{[1]}(x_n, y_n)$$

$$T^{[3]}(x_n, y_n, h) = f^{[0]}(x_n, y_n) + \frac{h}{2} f^{[1]}(x_n, y_n) + \frac{h^2}{6} f^{[2]}(x_n, y_n)$$

y las $f^{[k]}$ son las pseudoderivadas de $f(x)$ dadas (las primeras) por

$$f^{[0]} = f$$

$$f^{[1]} = f_x + f f_y$$

$$f^{[2]} = f_{xx} + 2f f_{xy} + f_x f_y + f^2 f_{yy} + f f_y^2$$

El último de los métodos de TAYLOR escritos es el de orden 3. $T^{[3]}$ es justamente el desarrollo de $\Delta(x_n, y_n, h)$ hasta dejar el resto en $O(h^3)$

Diferenciales elementales que intervienen

```
> fx:=unapply(diff(f(x,y),x),(x,y));fy:=unapply(diff(f(x,y),y),(x,y));
      fx := (x,y) -> y + 2 x
      fy := (x,y) -> x
> fxx:=unapply(diff(fx(x,y),x),(x,y));fxy:=unapply(diff(fx(x,y),y),(x,y));fyy:=unapply(diff(fy(x,y),y),(x,y));
      fxx := 2
      fxy := 1
      fyy := 0
```

Primer paso

```
> h:=0.1;x0:=0.;y0:=1.;
      x0 := 0.
      y0 := 1.
> printf("      f= %a , fx= %a  fy= %a , fxx= %a
      fxy= %a , fyy= %a
      \n",f(x0,y0),fx(x0,y0),fy(x0,y0),fxx(x0,y0),fxy(x0,y0),fyy(x0,y0));
      f= 0. , fx= 1.  fy= 0. , fxx= 2  fxy= 1 , fyy= 0
> f0:=f(x0,y0);f1:=fx(x0,y0)+f(x0,y0)*fy(x0,y0);f2:=fxx(x0,y0)+2*f(x0,y0)*fxy(x0,y0)+fx(x0,y0)*fy(x0,y0)+f(x0,y0)^2*fyy(x0,y0)+f(x0,y0)*fy(x0,y0)^2;
      f0 := 0.
      f1 := 1.
      f2 := 2.
> T3:=f0+(h/2)*f1+(h^2/6)*f2;
      T3 := 0.0533
> y1:=y0+h*T3;
      y1 := 1.01
```

Segundo paso

```
> x0:=x0+h;y0:=y1;
      x0 := 0.1
      y0 := 1.01
> printf("      f= %a , fx= %a  fy= %a , fxx= %a
      fxy= %a , fyy= %a
      \n",f(x0,y0),fx(x0,y0),fy(x0,y0),fxx(x0,y0),fxy(x0,y0),fyy(x0,y0));
      f= .111 , fx= 1.21  fy= .1 , fxx= 2  fxy= 1 , fyy= 0
> f0:=f(x0,y0);f1:=fx(x0,y0)+f(x0,y0)*fy(x0,y0);f2:=fxx(x0,y0)+2*f(x0,y0)*fxy(x0,y0)+fx(x0,y0)*fy(x0,y0)+f(x0,y0)^2*fyy(x0,y0)+f(x0,y0)*fy(x0,y0)^2;
      f0 := 0.111
      f1 := 1.22
      f2 := 2.34
> T3:=f0+(h/2)*f1+(h^2/6)*f2;
      T3 := 0.176
> y1:=y0+h*T3;
      y1 := 1.03
```

Tercer paso

```
> x0:=x0+h;y0:=y1;
      x0 := 0.2
```

```

[                                     y0 := 1.03
> printf("                             f= %a   , fx= %a   fy= %a   , fxx= %a
  fxy= %a   , fyy= %a
  \n",f(x0,y0),fx(x0,y0),fy(x0,y0),fxx(x0,y0),fxy(x0,y0),fyy(x0,y0));
                                     f= .246   , fx= 1.43   fy= .2   , fxx= 2   fxy= 1   , fyy= 0
> f0:=f(x0,y0);f1:=fx(x0,y0)+f(x0,y0)*fy(x0,y0);f2:=fxx(x0,y0)+2*f(x0,y0)*fxy(x0,y0)+fx(
  x0,y0)*fy(x0,y0)+f(x0,y0)^2*fyy(x0,y0)+f(x0,y0)*fy(x0,y0)^2;
                                     f0 := 0.246
                                     f1 := 1.48
                                     f2 := 2.79
> T3:=f0+(h/2)*f1+(h^2/6)*f2;
                                     T3 := 0.325
> y1:=y0+h*T3;
                                     y1 := 1.06

```

Ejercicios de

(3) Ecuaciones escalares: métodos de RUNGE-KUTTA

Ejercicio 03-17

Para el problema tipo de Cauchy

$$y' = x y + x^2$$

$$y(0) = 1,$$

(de solución exacta $-x + \frac{1 e^{\left(\frac{1x^2}{2}\right) \pi^{\left(\frac{1}{2}\right)} 2^{\left(\frac{1}{2}\right)} \operatorname{erf}\left(\frac{1 2^{\left(\frac{1}{2}\right)} x}{2}\right)}{2} + 1. e^{\left(\frac{1x^2}{2}\right)}$, donde erf representa la función

de error $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$) utilícese el método de Runge-Kutta del tablero

0 |

1/2 | 1/2

1/2 | 0 1/2

1 | 0 0 1

| 1/6 1/3 1/3 1/6

para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=3:
[ La ecuación (aut.) y la verdadera solución
[ > f:=(x,y)->x*y+x^2:yv:=x->-x+1/2*exp(1/2*x^2)*Pi^(1/2)*2^(1/2)*erf(1/2*2^(1/2)*x)+1.*exp(1/2*x^2):
[ RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (e. no aut.)
[ > Rkut4:=proc(f::procedure,ytotalini::list,nit::posint,h::numeric)
[ local x0,y0,k,k1,k2,k3,k4:
[ x0:=ytotalini[1]:y0:=ytotalini[2]:
[ for k from 1 to nit do
[ k1:=f(x0,y0):
[ k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1):
[ k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2):
[ k4:=f(x0+h,y0+h*k3):
[ x0:=x0+h:y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
[ [x0,y0]:
[ od:
[ end:
```

[Primer paso

```
> h:=0.1;xini:=0.;yini:=1.;x0:=xini;y0:=yini;
                                     x0 := 0.
                                     y0 := 1.
> x1:=xini+h;
                                     x1 := 0.1
> k1:=f(x0,y0);
                                     k1 := 0.
> k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1);
                                     k2 := 0.0525
> k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2);
                                     k3 := 0.0525
> k4:=f(x0+h,y0+h*k3);
                                     k4 := 0.111
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
                                     y1 := 1.01
```

[Todos los pasos de $h = 0.1$ desde $x = 0$. hasta $x = 0.3$

```
> h:=0.1;xini:=0.;yini:=1.;ytotalini:=[xini,yini];
                                     h := 0.1
                                     xini := 0.
                                     yini := 1.
                                     ytotalini := [0., 1.]
> for i from 1 to 3 do ytotal1:=Rkut4(f,ytotalini,1,h):print(`[x,y] = `,ytotal1,`
mientras la exacta vale`,evalf(yv(ytotal1[1]))):ytotalini:=ytotal1: od:
[x,y] = , [0.1, 1.01], mientras la exacta vale, 1.01
[x,y] = , [0.2, 1.03], mientras la exacta vale, 1.02
[x,y] = , [0.3, 1.06], mientras la exacta vale, 1.06
```


04. Estimación del error y cambio de paso

- 1 Introducción a la estimación del error
- 2 Pares encajados
- 3 Organización práctica de un par encajado
- 4 Métodos de RUNGE-KUTTA-FEHLBERG
- 5 Estimación basada en la extrapolación de RICHARDSON

Introducción a la estimación del error

Hasta ahora: orden = precisión de los métodos

Mayor orden - mayor número de etapas - mayor coste computacional - Conveniente: no exagerar con el orden

El paso incide decisivamente en el coste computacional

Funcionamiento ideal:

- Paso pequeño cuando la integración resulta difícil
- Paso grande cuando no hay problema

Esencial dotarse de una 'tolerancia' al error.

Dos formas de adoptar una tolerancia:

- 1 Error máximo a soportar en cada paso:

'tolerancia en cada paso (TP)'

- 2 Tolerancia por unidad de intervalo:

'tolerancia unidad (TU)'



Si admitimos como criterio la opción TP (tolerancia en cada paso), en el paso de x_n a $x_n + h = x_{n+1}$ tendremos que comparar el error estimado, pongamos *est*, con el error admitido en este paso (el mismo independientemente del paso de que se trate y de su longitud); o sea, miraremos si

$$est \leq TP$$

o no, y actuaremos según sea el caso.



Si lo que admitimos es el criterio TU (tolerancia unidad), tendremos un valor E para el error total admitido en la integración completa en $[a, b]$ y, entonces, la tolerancia unidad será

$$TU = \frac{E}{b - a}$$

y la tolerancia que corresponde al paso de longitud h entre x_n y $x_n + h = x_{n+1}$ será

$$TU * h;$$

ahora miraremos si

$$est \leq TU * h$$

o no, y actuaremos según sea el caso.

Mecanismo para saber el error que se está cometiendo:

'estimación del error' (estimación=aproximación)

Error que se estima: el error local (LTE)

Este mecanismo devuelve un valor, est , que es una aproximación al error que se está cometiendo.

Dos formas (mecanismos) de estimar el error

- Integración simultánea con dos métodos de distinto orden: **par encajado**
- Empleo de la **extrapolación de RICHARDSON**: integración con un paso h y su mitad $h/2$

Pares encajados

La idea es usar simultáneamente dos métodos para la integración de la ecuación. Utilizar dos métodos parece llevar la dirección contraria de la que pretendemos pero enseguida veremos que no es así. Esta técnica posee dos ventajas:

- La primera (y esperada) es la estimación del error que se está cometiendo en cada paso.
- La segunda (algo más sorprendente) es la capacidad de este sistema de aconsejar, en cada momento, el paso óptimo a emplear; o sea, el más grande posible sin superar el error admitido.

Un **par encajado** de métodos de RUNGE-KUTTA (en la bibliografía en lengua inglesa se cita como 'embedded Runge-Kutta formulae' o algún nombre similar) se compone de dos métodos de RUNGE-KUTTA. El primero, el de orden bajo, tendrá un orden que llamaremos p y un número de etapas que escribiremos como q . El segundo tendrá orden más alto, \hat{p} , y requerirá \hat{q} etapas.

Es esencial que sea $\hat{p} \geq p + 1$ y lo usual es que sea exactamente $\hat{p} = p + 1$. También es esencial que el primero sea de orden exactamente p (y no p o mayor) y que el término principal del error de este método no tenga coeficientes nulos. Veremos enseguida la razón de estos requerimientos.

Veamos lo que hay que hacer cuando ya se ha calculado la aproximación y_n de $y(x_n)$ para la abscisa x_n con paso actual h , y se desea averiguar la idoneidad de la aproximación y_{n+1} para la abscisa $x_{n+1} = x_n + h$.

Mediante el método de orden p calculamos la aproximación, que llamaremos y_{n+1} ; el error local para esta aproximación será

$$T_{n+1}(h) = \mathcal{O}(h^{p+1}).$$

Y mediante el método de orden \hat{p} calculamos la aproximación, que llamaremos \hat{y}_{n+1} ; el error local para esta aproximación será

$$\hat{T}_{n+1}(h) = \mathcal{O}(h^{\hat{p}+1}).$$

Naturalmente, es el error local el que es posible estimar.

Como hemos dicho alguna vez, estos errores locales son los medidos como

$$T_{n+1}(h) = y(x_{n+1}) - y_{n+1} \quad \text{y} \quad \hat{T}_{n+1}(h) = y(x_{n+1}) - \hat{y}_{n+1},$$

para la solución exacta de la ecuación con la condición inicial $y(x_n) = y_n$ en x_n .

Un pequeño artificio nos permite ahora aproximar (lo que llamamos 'estimar') el error local $T_{n+1}(h)$ mediante una simple resta de los valores calculados de y_{n+1} e \hat{y}_{n+1} . Es evidente

$$T_{n+1}(h) = \hat{T}_{n+1} + \hat{y}_{n+1} - y_{n+1},$$

con sólo restar las expresiones anteriores. El orden de $T_{n+1}(h)$ es $\mathcal{O}(h^{p+1})$ y el de $\hat{T}_{n+1}(h)$ es $\mathcal{O}(h^{\hat{p}+1})$, mucho más pequeño, luego una buena aproximación de $T_{n+1}(h)$ es

$$T_{n+1}(h) \sim \hat{y}_{n+1} - y_{n+1},$$

y esta es la estimación del error que nos será útil.

Esta estimación ya nos es de utilidad. Basta ahora hacerse la pregunta de si

$$|\hat{y}_{n+1} - y_{n+1}| \leq TP$$

si nos hemos inclinado por esa modalidad de aceptación del error, o si

$$|\hat{y}_{n+1} - y_{n+1}| \leq TU * h$$

si nos hemos inclinado por la otra.

Pero como hemos anunciado aún hay más, y vamos a ver cómo esta técnica permite aconsejar el tamaño adecuado del paso. Para ello denotaremos por

$$\alpha h$$

cualquier paso posible (mayor o igual que 0, por supuesto). Los valores $\alpha > 1$ darán pasos mayores que h y los valores $\alpha < 1$ darán valores menores que h .

Además, supondremos que el error local se comporta como

$$|T_{n+1}(h)| \sim K h^{p+1},$$

aunque desconoceremos el valor de K que depende de la solución del problema.

Para un paso αh tendremos

$$|T_{n+1}(\alpha h)| \sim K (\alpha h)^{p+1} = \alpha^{p+1} K h^{p+1} = \alpha^{p+1} |T_{n+1}(h)|,$$

lo que significa

$$|T_{n+1}(\alpha h)| \sim \alpha^{p+1} |\hat{y}_{n+1} - y_{n+1}|.$$

El paso adecuado (con el que se comete un error menor que el tolerado pero lo mayor posible) será αh para

$$\alpha^{p+1} |\hat{y}_{n+1} - y_{n+1}| = TP,$$

en la opción 'tolerancia por paso', o

$$\alpha^{p+1} |\hat{y}_{n+1} - y_{n+1}| = TU \alpha h,$$

en la opción de la 'tolerancia unidad'.

Lo que nos lleva en el primer paso al α aconsejado

$$\alpha = \left(\frac{TP}{|\hat{y}_{n+1} - y_{n+1}|} \right)^{1/p+1},$$

y en el segundo, al α aconsejado

$$\alpha = \left(\frac{TU h}{|\hat{y}_{n+1} - y_{n+1}|} \right)^{1/p}.$$

Todo es bastante similar en ambos casos, aunque la fórmula es diferente. La práctica indica que la segunda opción, la de la tolerancia unidad, es más exigente o, si se prefiere, más restrictiva. Particularmente creo que esta segunda toma más en consideración el que el error que es posible cometer en un paso sea algo que dependa del tamaño del paso. O sea, que sea posible hablar de un error tolerado para la integración completa, con independencia del número y tamaño de pasos que tengamos que emplear.

Obsérvese además que el manejo de α permite determinar si se verifica o no la desigualdad

$$|\hat{y}_{n+1} - y_{n+1}| \leq TP$$

o la correspondiente a la otra tolerancia. Es inmediato ver que

$$\alpha \geq 1$$

implica que $|\hat{y}_{n+1} - y_{n+1}| \leq TP$ en el primer caso o $|\hat{y}_{n+1} - y_{n+1}| \leq TU * h$ en el segundo y que el paso h es aceptable (aunque el aconsejado sea αh), y que

$$\alpha < 1$$

indica que el paso debe ser rechazado por producir un error excesivo.

Dado que el objeto de esta técnica es el ahorro de evaluaciones de la función f , la marcha de funcionamiento es la siguiente:

Si $\alpha < 1$, el paso de tamaño h debe ser rechazado, Se toma el paso aconsejado αh y se realiza el nuevo cálculo de y_{n+1} sin ponerlo en duda. Naturalmente que este comportamiento (de no comprobación de la idoneidad de αh) puede llevar a superar en algo el error admitido.

Cuando $\alpha \geq 1$, el paso debería optimizarse al αh , que es más grande. Sin embargo, lo que se hace es aprovechar el valor ya calculado de y_{n+1} para $x_{n+1} = x_n + h$ y, sólo entonces, proceder a cambiar el paso a αh antes de dar el siguiente. Se ahorran así evaluaciones de f aprovechando las ya calculadas, ya que se respeta el error tolerado.

Puede ser útil concluir esta sección con algunas recomendaciones que, sin estar fundamentadas en el caso general, son de uso bastante extendido en la puesta en práctica de un par encajado.

Hemos dicho que, al obtener $\alpha \geq 1$ y aceptar el paso, el valor que se toma de los ya calculados es y_{n+1} , el obtenido con el método de orden más bajo. Es opinión extendida que, en este caso, el valor de \hat{y}_{n+1} producirá aún menos error, al estar obtenido con un método de mayor orden (nótese que el desarrollo teórico indica y_{n+1} como el valor a coger). Cuando se procede de la forma explicada el método se denota por $RK\hat{p}(p)$ y por $RKp(\hat{p})$ en caso contrario; o sea, el orden escrito en primer lugar y fuera del paréntesis es, por convenio, el método con el que 'se avanza'.

La cantidad $\hat{y}_{n+1} - y_{n+1}$ estima el error local, y en la práctica lo hace bastante bien. No obstante es posible que la estimación se haga 'por lo bajo' en algunos problemas.

Siguiendo las indicaciones del especialista en el tema LAWRENCE F. SHAMPINE, hay quien considera prudente substituir el valor de α recomendado por la cantidad $0,9\alpha$.

Otra alternativa muy usada es reemplazar la estimación $\hat{y}_{n+1} - y_{n+1}$ del error por $2(\hat{y}_{n+1} - y_{n+1})$, con idéntico propósito. Pues bien, si en las fórmulas para α hacemos esta substitución y suponemos que p vale por ejemplo 4, que es uno de los valores más usuales, estamos multiplicando a α por 0,87 o 0,84 (según el tipo de tolerancia empleado), lo que es prácticamente lo mismo.

También parece tras alguna práctica que cambios excesivamente bruscos en el h usado pueden llevar a errores. Cuando se desea evitar esto lo que se hace es darse dos valores extremos α_{\min} y α_{\max} que no deben ser superados. Mientras que la no superación de α_{\max} , o sea, tomar

$$\alpha = \min(\alpha, \alpha_{\max})$$

no presenta más que un pequeño problema de ralentización, el análogo

$$\alpha = \max(\alpha, \alpha_{\min})$$

puede llevar a avanzar con errores mayores de los tolerados. Cuando se programa de esta manera, se suelen incluir avisos cuando se llega a usar la limitación del α_{\min} .

Parecidos problemas parece crear el uso de pasos demasiado grandes o pequeños, en particular este último aspecto cuando lleva a la no finalización del proceso de integración con un par encajado. A veces se establecen también dos limitaciones h_{\min} y h_{\max} , proporcionadas al tamaño del intervalo completo de integración, y se toman entonces

$$h = \min(h, h_{\max})$$

o

$$h = \max(h, h_{\min})$$

con la misma problemática cuando es necesario usar la limitación de h_{\min} . No es necesario señalar que es aconsejable también disponer de avisos cuando se llega al uso de h_{\min} .

Naturalmente que estos aspectos que cabría calificar de 'heurísticos' o 'experimentales' dependen enormemente de las dificultades surgidas en el empleo real de estos métodos, por lo que cada cual suele aplicarlos de acuerdo con lo que su experiencia le va indicando

Finalmente, aclaremos la cuestión de que el método de orden bajo no debe anular, si es posible, los coeficientes que presenta en el término principal del error. Si hiciese lo contrario, habría ejemplos en los que las únicas diferenciales elementales no nulas de la ecuación (de entre f , f_x , f_y , f_{xx} , etc.) corresponderían a coeficientes nulos del término de error para el método citado. Para un tal ejemplo, el método de orden bajo se comportaría no como de orden p , sino como de orden $p + 1$. El valor de $\hat{y}_{n+1} - y_{n+1}$ sería entonces prácticamente nulo y no serviría como estimación del error.

La única manera de evitar estos ejemplos 'desafortunados' es conseguir la no nulidad de los coeficientes del término principal del error en el caso del método de orden bajo.

El otro, el de orden alto, por el contrario, conviene que optimice o minimice el error.

Organización práctica de un par encajado

Dada la finalidad del empleo de estos métodos, resulta importante optimizar los elementos que hacen que una integración sea costosa. Por ejemplo, si en un par encajado con órdenes y etapas p , q y \hat{p} , \hat{q} no imponemos limitaciones al carácter de estas etapas, tendremos un número de $q + \hat{q}$ evaluaciones de f en cada paso estudiado (avanzado o no). Gran parte de estas evaluaciones (la mitad, hablando de manera aproximada) se pueden evitar haciendo que ambos métodos 'compartan' etapas. O sea, que exista una sola familia de etapas que sirva para ambos métodos a la vez, con independencia de que algún método pueda prescindir de alguna de las últimas etapas.

No se emplea ningún par encajado que no tenga estas características de compartir etapas. Pero conseguirlo no siempre es posible cuando nos restringimos al mínimo número de etapas para los órdenes p y \hat{p} dados. En la mayor parte de los casos es necesario aumentar algo la cantidad de etapas compartidas para que haya parámetros suficientes que permitan obtener el par que se desea. Empezamos con un estudio de construcción de un par encajado, aunque los órdenes que buscamos son bajos para poder hacer el estudio completo.

Buscamos organizar el par encajado en un tablero de BUTCHER ligeramente reorganizado como

$[c_i]$	$[a_{ij}]$
orden p	$[b_j]$
orden \hat{p}	$[\hat{b}_j]$
estimación	$[d_j]$

en donde los coeficientes de las etapas, $[c_i]$ y $[a_{ij}]$ son compartidos por ambos métodos, que luego tienen cada uno los coeficientes $b_1, b_2 \dots$ el de orden p y $\hat{b}_1, \hat{b}_2 \dots$ el de orden \hat{p} .

Se añade además una fila de coeficientes $d_1, d_2 \dots$ con valores $d_j = \hat{b}_j - b_j$ que sirven para el cálculo efectivo de la estimación del error $\hat{y}_{n+1} - y_{n+1}$ en la forma siguiente. Calculados $k_1, k_2 \dots$ se podrían evaluar

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + \dots) \quad \text{e} \quad \hat{y}_{n+1} = y_n + h(\hat{b}_1 k_1 + \hat{b}_2 k_2 + \dots),$$

calculando luego la diferencia $\hat{y}_{n+1} - y_{n+1}$. Pero de hecho hace menores los errores de redondeo calcular esta diferencia como

$$\hat{y}_{n+1} - y_{n+1} = h(d_1 k_1 + d_2 k_2 + \dots),$$

y así es como realmente se hace. Los coeficientes $d_1, d_2 \dots$ son los que se colocan en la última fila del tablero acompañados del título 'estimación' o 'est'.

Pasemos ya a nuestro ejemplo. Queremos construir un par encajado cuyo método de orden alto \hat{p} sea el conocido como HEUN de tres etapas

0			
$\frac{1}{3}$		$\frac{1}{3}$	
$\frac{2}{3}$		0	$\frac{2}{3}$
		$\frac{1}{4}$	0
		$\frac{3}{4}$	

Para buscar el método acompañante de orden bajo $p = 2$ usaremos las tres etapas del anterior método, acompañadas ahora de coeficientes b_1, b_2, b_3 para $y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3)$. No tiene sentido buscarle de dos etapas, cosa en principio posible, ya que tendremos que hacer necesariamente las tres evaluaciones de f .

Así pues enunciamos el método

$$\begin{aligned}k_1 &= f(x_n, y_n), \\k_2 &= f\left(x_n + \frac{1}{3}h, y_n + \frac{1}{3}h k_1\right) \\k_3 &= f\left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}h k_2\right)\end{aligned}$$

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3),$$

y trataremos de encontrar los valores de los parámetros desconocidos b_1 , b_2 y b_3 para que nos dé un método de orden 2 que además se comporte 'adecuadamente' en cuanto a los coeficientes del término principal del error, en el sentido de su valor no nulo cuya necesidad ya hemos justificado.

Abreviamos, como siempre, poniendo

$$\begin{aligned}(1) &\text{ para } (x_n, y_n), \\(2) &\text{ para } \left(x_n + \frac{1}{3}h, y_n + \frac{1}{3}h k_1\right), \\(3) &\text{ para } \left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}h k_2\right),\end{aligned}$$

y omitiendo normalmente el punto (1) = (x_n, y_n) .



El orden 1 nos lleva como otras veces a

$$\begin{aligned}k_1(h) &= \text{cte} = f \\k_2(h) &= f(2) \\k_3(h) &= f(3); \end{aligned}$$

y a la condición

$$b_1 + b_2 + b_3 = 1.$$

Para el orden 2 se tiene

$$\begin{aligned}k_2'(h) &= \frac{1}{3} f_x(2) + \frac{1}{3} k_1 f_y(2) \\k_3'(h) &= \frac{2}{3} f_x(3) + \left[\frac{2}{3} k_2 + \frac{2}{3} h k_2'\right] f_y(3),\end{aligned}$$

y nos será rentable denotar por $[\]$ la cantidad $[\] = \left[\frac{2}{3} k_2 + \frac{2}{3} h k_2'\right]$ para futuras operaciones.

Lo anterior hace que

$$k'_2(0) = \frac{1}{3} f_x + \frac{1}{3} f f_y$$

y, teniendo en cuenta que

$$[]_{h=0} = \frac{2}{3} f ,$$

a su vez

$$k'_3(0) = \frac{2}{3} f_x + \frac{2}{3} f f_y ;$$

y como

$$\Phi'(0) = b_2 k'_2(0) + b_3 k'_3(0)$$

y

$$\frac{1}{2} f^{(1)} = \frac{1}{2} f_x + \frac{1}{2} f f_y .$$

resulta que para que $\Phi'(0) = \frac{1}{2} f^{(1)}$ necesitamos la condición que aparece por duplicado



$$\frac{1}{3} b_2 + \frac{2}{3} b_3 = \frac{1}{2}$$

junto con la ya citada del orden 1.

Es evidente que podemos encontrar una infinidad de valores de b_1 , b_2 y b_3 que proporcionen métodos de orden 2.

Para el orden 3, que tratamos de evitar, se tiene en primer lugar que

$$\frac{d []}{dh} = \frac{4}{3} k'_2 + \frac{2}{3} h k''_2$$

a lo que sigue

$$k''_2(h) = \frac{1}{9} f_{xx}(2) + \frac{2}{9} k_1 f_{xy}(2) + \frac{1}{9} (k_1)^2 f_{yy}(2)$$

y

$$k''_3(h) = \frac{4}{9} f_{xx}(3) + \frac{4}{3} [] f_{xy}(3) + []^2 f_{yy}(3) + \left(\frac{4}{3} k'_2 + \frac{2}{3} h k''_2 \right) f_y(3) .$$



En $h = 0$ los valores anteriores resultan ser

$$k_2''(0) = \frac{1}{9} f_{xx} + \frac{2}{9} f f_{xy} + \frac{1}{9} (f)^2 f_{yy}$$

y

$$k_3''(h) = \frac{4}{9} f_{xx} + \frac{8}{9} f f_{xy} + \frac{4}{9} (f)^2 f_{yy} + \frac{4}{9} f_x f_y + \frac{4}{9} f (f_y)^2 .$$

Así se obtiene que

$$\begin{aligned} \Phi''(0) &= \left(\frac{1}{9} b_2 + \frac{4}{9} b_3\right) f_{xx} + 2\left(\frac{1}{9} b_2 + \frac{4}{9} b_3\right) f f_{xy} + \left(\frac{1}{9} b_2 + \frac{4}{9} b_3\right) (f)^2 f_{yy} \\ &\quad + \frac{4}{9} b_3 f_x f_y + \frac{4}{9} b_3 f (f_y)^2 , \end{aligned}$$

y como

$$\frac{1}{3} f^{(2)} = \frac{1}{3} f_{xx} + \frac{2}{3} f f_{xy} + \frac{1}{3} (f)^2 f_{yy} + \frac{1}{3} f_x f_y + \frac{1}{3} f (f_y)^2 ,$$

los coeficientes para el término principal del error que aparecen en $\frac{1}{3} f^{(2)} - \Phi''(0)$ son

$$\frac{1}{3} - \frac{1}{9} b_2 - \frac{4}{9} b_3$$

y

$$\frac{1}{3} - \frac{4}{9} b_3$$

repetidos 3 y 2 veces respectivamente.

Para el objetivo buscado de encontrar un método de orden 2 adecuado, podemos elegir, tras algún tanteo, el valor $b_3 = 1/2$, que hace que no se cumpla la segunda de las anteriores condiciones. Las condiciones para el orden 2 nos hacen tomar entonces $b_2 = 1/2$ y $b_1 = 0$. Con estos valores, los 2 coeficientes en que se resumen los 5 del término principal del error valen $1/18$ y $1/9$, dejando en evidencia que hemos encontrado el método adecuado de orden 2 y deja el par encajado como

0			
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	0	$\frac{2}{3}$	
orden 2	0	$\frac{1}{2}$	$\frac{1}{2}$
orden 3	$\frac{1}{4}$	0	$\frac{3}{4}$
est	$\frac{1}{4}$	$-\frac{1}{2}$	$\frac{1}{4}$

Este método está entre los que tienen un cierto uso, y así lo citaremos en la siguiente sección. Los órdenes 2 – 3 son, sin embargo, un poco bajos para los problemas que hacen acudir a los pares encajados. A este respecto, los más usados son los de órdenes 4 – 5 y 7 – 8.

Parece que podríamos imitar este proceso fabricando un $RK3(4)$ con el uso de 4 etapas. No es posible, aunque no probaremos ahora este hecho. Hablaremos sin embargo de otra técnica que permite cierto ahorro suplementario de evaluaciones.

Cuando la última de las etapas de un par encajado posee coeficientes $c_q = 1$ y $a_{q,1}, a_{q,2}, \dots$, que coinciden con los b_1, b_2, \dots , empleados en el cálculo de y_{n+1} si se avanza con el orden p bajo o los $\hat{b}_1, \hat{b}_2, \dots$, empleados en el cálculo de \hat{y}_{n+1} si se avanza con el orden \hat{p} alto, el par encajado se dice que es de tipo **FSAL**. La forma en que se les llama es el acrónimo de la expresión inglesa 'First Same At Last' que describe el funcionamiento del par. Se produce entonces el ahorro de una evaluación por cada uno de los pasos aceptados (salvo en la primera de las ocasiones). Veamos cómo es eso, por ejemplo suponiendo que los $a_{q,1}, a_{q,2}, \dots$, coinciden con los b_1, b_2, \dots , del método de orden bajo p y que avanzamos con ese mismo método.

Supongamos que, después del cálculo de k_1, k_2, \dots, k_q , y de y_{n+1} e \hat{y}_{n+1} , obtenemos $\alpha \geq 1$ y decidimos conservar y_{n+1} como aproximación de $y(x_{n+1}) = y(x_n + h)$, cambiando entonces al nuevo paso $h\alpha$. Para este nuevo paso, es decir, para $x_{n+2} = x_{n+1} + h\alpha$ lo primero que hay que hacer es calcular las correspondientes etapas k_1, k_2, \dots, k_q . La primera de ellas es

$$k_1 = f(x_{n+1}, y_{n+1}).$$

Pero esta cantidad es la que había sido evaluada como

$$k_q = f(x_n + h, y_n + h(a_{q,1} k_1 + a_{q,2} k_2 + \dots + a_{q,q-1} k_{q-1}))$$

(recuérdese que $c_q = 1$ y que $a_{q,1} = b_1, \dots, a_{q,q-1} = b_{q-1}$), por lo que no es necesario realizar esta evaluación de f , sino recuperar el valor de k_q que tendremos almacenado en la oportuna variable.

Hay ejemplos sumamente sencillos de métodos del tipo FSAL. Por ejemplo, el $RK1(2)$ de tablero

0	
1	1
orden 1	1
orden 2	$\frac{1}{2} \quad \frac{1}{2}$
est	$-\frac{1}{2} \quad \frac{1}{2}$

formado por el método de EULER (orden 1) y el método modificado de EULER (orden 2) es FSAL.

Pero estos métodos sólo poseen interés cuando son de mayor orden y el empleo de etapas es más crítico. Daremos algún ejemplo de método FSAL de importancia en la práctica.

Métodos de RUNGE-KUTTA-FEHLBERG

Lo que presentamos aquí son algunos pares encajados que se usan y se han utilizado largamente en la práctica. La mayor parte de ellos se deben a los trabajos de ERWIN FEHLBERG publicados en la década de los 60 y 70 de 1900. Los primeros sirven sobre todo como ejemplos, debido a que su orden no es suficiente para las aplicaciones reales. Sin embargo, los pares con orden más alto son muy eficaces y resultan muy 'económicos' en relación con los métodos de paso fijo, y por lo tanto están entre los más adecuados para la integración de problemas que requieren alta precisión, como los problemas de órbitas perturbadas, por ejemplo.

Además del autor citado, merece la pena citar también a los investigadores JOHN R.DORMAND y P.J. PRINCE por aportar algunos de estos métodos que están entre los más utilizados. Sus principales aportaciones a este tema datan de las década de los 70 y 80 de 1900.



Una lista de los pares encajados, incluso limitada a los más usados, ocuparía ahora demasiado espacio. Nos limitamos a citar algunos de los ejemplos más citados para orden bajo y medio.

Hemos extraído los coeficientes del libro de JOHN R.DORMAND 'Numerical Methods for Differential Equations; A Computational Approach' y también del libro de M. CALVO, J.I. MONTIJANO y L. RÁNDEZ, 'Curso de Análisis Numérico. Métodos de Runge-Kutta para la resolución numérica de ecuaciones diferenciales ordinarias' tras corregir en este algunos de los coeficientes que estaban equivocados.

Como se ve en los ejemplos que siguen, damos los coeficientes de los métodos en su forma racional (a veces larga), que es la única manera de que se empleen en su momento con la precisión requerida. Es por lo que, cuando se necesiten, conviene tomar estos coeficientes de ficheros electrónicos, ya que el menor cambio por error en los mismos desvirtúa completamente el orden que se obtiene.



Empezando con un $RK2(3)$, citemos el par (que hemos construido antes)

0			
$\frac{1}{3}$	$\frac{1}{3}$		
$\frac{2}{3}$	0	$\frac{2}{3}$	
orden 2	0	$\frac{1}{2}$	$\frac{1}{2}$
orden 3	$\frac{1}{4}$	0	$\frac{3}{4}$
est	$\frac{1}{4}$	$-\frac{1}{2}$	$\frac{1}{4}$

par en el que el método de orden 3 es el que minimiza el error, el llamado HEUN de orden 3.

Como hemos comentado, no es posible lograr pares de métodos con etapas comunes del orden $RK3(4)$ cuando nos limitamos a 4 etapas. Sin embargo sí que pueden encontrarse autorizando el empleo de 5 etapas. Un ejemplo es el $RK3(4)$

0					
$\frac{1}{4}$	$\frac{1}{4}$				
$\frac{4}{9}$	$\frac{4}{81}$	$\frac{32}{81}$			
$\frac{6}{7}$	$\frac{57}{98}$	$-\frac{432}{343}$	$\frac{1053}{686}$		
1	$\frac{1}{6}$	0	$\frac{27}{52}$	$\frac{49}{156}$	
orden 3	$\frac{1}{6}$	0	$\frac{27}{52}$	$\frac{49}{156}$	
orden 4	$\frac{43}{288}$	0	$\frac{243}{416}$	$\frac{343}{1872}$	$\frac{1}{12}$
est	$-\frac{5}{288}$	0	$\frac{27}{416}$	$-\frac{245}{1872}$	$\frac{1}{12}$

Se observará que se trata de un par que posee además la propiedad que hemos llamado FSAL.

Un primer par $RK4(5)$

0						
$\frac{2}{9}$	$\frac{2}{9}$					
$\frac{1}{3}$	$\frac{1}{12}$	$\frac{1}{4}$				
$\frac{3}{4}$	$\frac{69}{128}$	$-\frac{243}{128}$	$\frac{135}{64}$			
1	$-\frac{17}{12}$	$\frac{27}{4}$	$-\frac{27}{5}$	$\frac{16}{15}$		
$\frac{5}{6}$	$\frac{65}{432}$	$-\frac{5}{16}$	$\frac{13}{16}$	$\frac{4}{27}$	$\frac{5}{144}$	
orden 4	$\frac{1}{9}$	0	$\frac{9}{20}$	$\frac{16}{45}$	$\frac{1}{12}$	
orden 5	$\frac{47}{450}$	0	$\frac{12}{25}$	$\frac{32}{225}$	$\frac{1}{30}$	$\frac{6}{25}$
est	$-\frac{1}{150}$	0	$\frac{3}{100}$	$-\frac{16}{75}$	$-\frac{1}{20}$	$\frac{6}{25}$

Y otro $RK4(5)$ mas

0						
$\frac{1}{4}$	$\frac{1}{4}$					
$\frac{3}{8}$	$\frac{3}{32}$	$\frac{9}{32}$				
$\frac{12}{13}$	$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$			
1	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		
$\frac{1}{2}$	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	
orden 4	$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	
orden 5	$\frac{16}{135}$	0	$\frac{6656}{12825}$	$\frac{28561}{56430}$	$-\frac{9}{50}$	$\frac{2}{55}$
est	$\frac{1}{360}$	0	$-\frac{128}{4275}$	$-\frac{2197}{75240}$	$\frac{1}{50}$	$\frac{2}{55}$

Para que en estos $RK4(5)$ encontremos la propiedad FSAL hace falta irse hasta las 7 etapas; además, construimos el de orden 5 con las 6 primeras y el de orden 4 con todas ellas;

se obtiene así el par $RK5(4)$ FSAL siguiente

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
orden 4	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$
orden 5	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
est	$\frac{71}{57600}$	0	$-\frac{71}{16695}$	$\frac{71}{1920}$	$-\frac{17253}{339200}$	$\frac{22}{525}$	$-\frac{1}{40}$

No exhibiremos aquí otros métodos de orden más alto, aún cuando algunos se encuentran entre los más usados. Por ejemplo, el método de JOHN R. DORMAND y P.J. PRINCE que se conoce con el nombre de DOPRI8 (par encajado RUNGE-KUTTA con 13 etapas que combina dos métodos de órdenes 7 y 8) y algún otro de parecidas características están, como hemos dicho, entre los que se emplean para problemas que requieren alta precisión en tiempos razonables.

Estimación basada en la extrapolación de RICHARDSON

Cambiando de punto de vista, la extrapolación de RICHARDSON nos permite también estimar el error que se comete durante la integración numérica de un problema. Ahora se emplea sólo un método pero se hace simultáneamente con dos pasos diferentes, usualmente uno mitad del otro. En lo que respecta a sus prestaciones y a su 'gasto computacional' hay que decir dos cosas:

- Sirve para estimar el error que se está cometiendo en cada paso, con un coste del doble aproximado en número de evaluaciones de f . Sin embargo, el empleo de un paso y su mitad, si se programa adecuadamente, reduce básicamente el empleo de evaluaciones a las del propio método exclusivamente.

- También sería posible que este sistema aconsejase, en cada momento, el paso óptimo a emplear. Pero si se hace el cambio de paso a uno nuevo sin relación con el anterior se pierde la ventaja antes citada y que está ligada al paso mitad.

En consecuencia, esta técnica sólo se suele emplear para la estimación del error, sin hacer uso de ninguna recomendación sobre el nuevo tamaño del paso.

Por otra parte su empleo no está ligado al uso de métodos de RUNGE-KUTTA, como ocurría con los pares encajados. En su propia presentación vemos que se puede emplear para métodos casi cualesquiera explícitos de 1 paso, aunque se extiende sin mucho problema a los métodos de paso múltiple.

Denotemos por

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

el método (en principio explícito y de 1 paso, pero no necesariamente de RUNGE-KUTTA) con el que integramos nuestro problema

$$y' = f(x, y), \quad y(a) = \eta,$$

y que supondremos es de orden p . Esto supone que el error local será un $\mathcal{O}(h^{p+1})$.

Este error local es

$$T_{n+1}(h) = y(x_{n+1}) - y_{n+1}$$

para la solución exacta de la ecuación con la condición inicial $y(x_n) = y_n$ en x_n .

Ahora se necesita precisar un poco más el citado infinitésimo $\mathcal{O}(h^{p+1})$. Es, separando el término principal del error,

$$T_{n+1}(h) = y(x_{n+1}) - y_{n+1} = \Psi(x_n, y(x_n)) h^{p+1} + \mathcal{O}(h^{p+2})$$

donde lo importante es la independencia de $\Psi(x_n, y(x_n))$ de h ; sólo depende de x_n directamente y a través de la solución $y(x)$ del problema.

Por ejemplo, para el 'método modificado de EULER', de 2 etapas y orden 2, el error local es $T_{n+1}(h) = \mathcal{O}(h^3)$, pero también

$$T_{n+1}(h) = \frac{1}{2} \left(\frac{1}{3} f^{(2)} - \Phi''(0) \right) h^3 + \mathcal{O}(h^4),$$

que es de la forma indicada para

$$\Psi(x_n, y(x_n)) = \frac{1}{2} \left(-\frac{1}{6} f_{xx} - \frac{1}{3} f f_{xy} - \frac{1}{6} (f)^2 f_{yy} + \frac{1}{3} f_x f_y + \frac{1}{3} f (f_y)^2 \right)$$

(evaluadas todas las diferenciales elementales en $(x_n, y(x_n))$).

Veamos lo que hay que hacer cuando ya se ha calculado la aproximación y_n de $y(x_n)$ para la abscisa x_n con paso actual h , y se desea averiguar la idoneidad de la aproximación y_{n+1} para la abscisa $x_{n+1} = x_n + h$.

Mediante nuestro método y el paso h calculamos y_{n+1} . Se habrá cometido un error local de

$$T_{n+1}(h) = y(x_{n+1}) - y_{n+1} = \Psi(x_n, y(x_n)) h^{p+1} + \mathcal{O}(h^{p+2}).$$

Ahora, con paso $2h$ y desde x_{n-1} calculamos como aproximación en x_{n+1} el valor \hat{y}_{n+1} , con un error local que será

$$\hat{T}_{n+1}(2h) = y(x_{n+1}) - \hat{y}_{n+1} = \Psi(x_{n-1}, y(x_{n-1})) (2h)^{p+1} + \mathcal{O}(h^{p+2}).$$

Ahora, desarrollando $\Psi(x_{n-1}, y(x_{n-1})) = \Psi(x_n - h, y(x_n - h))$ en potencias de h , sólo hasta el segundo término, y alrededor de $h = 0$, se tiene

$$\Psi(x_{n-1}, y(x_{n-1})) = \Psi(x_n, y(x_n)) + \mathcal{O}(h),$$

luego

$$\hat{T}_{n+1}(2h) = y(x_{n+1}) - \hat{y}_{n+1} = \Psi(x_n, y(x_n)) (2h)^{p+1} + \mathcal{O}(h^{p+2})$$

y

$$\hat{T}_{n+1}(2h) = y(x_{n+1}) - \hat{y}_{n+1} = 2^{p+1} \Psi(x_n, y(x_n)) h^{p+1} + \mathcal{O}(h^{p+2}).$$

Basta ahora restar las expresiones de $y(x_{n+1}) - \hat{y}_{n+1}$ y de $y(x_{n+1}) - y_{n+1}$ para obtener

$$y_{n+1} - \hat{y}_{n+1} = (2^{p+1} - 1) \Psi(x_n, y(x_n)) h^{p+1} + \mathcal{O}(h^{p+2}),$$

o sea,

$$y_{n+1} - \hat{y}_{n+1} = (2^{p+1} - 1) T_{n+1}(h) + \mathcal{O}(h^{p+2}).$$

Esto hace que una estimación adecuada venga dada por

$$y_{n+1} - \hat{y}_{n+1} \sim (2^{p+1} - 1) T_{n+1}(h),$$

y, finalmente, que se tenga la estimación

$$T_{n+1}(h) \sim \frac{y_{n+1} - \hat{y}_{n+1}}{2^{p+1} - 1}.$$

Esta estimación es la que hay que comparar con TP para el criterio de 'tolerancia por paso' o con TU^*h para el criterio de la 'tolerancia unidad'.

Si

$$\frac{|y_{n+1} - \hat{y}_{n+1}|}{2^{p+1} - 1} \leq TP \quad \text{o} \quad \leq TU * h,$$

se acepta el paso de longitud h entre x_n y x_{n+1} , se toma y_{n+1} como aproximación a $y(x_{n+1})$ y se dobla el paso a $2h$.

Si la desigualdad es al contrario (o sea, el error estimado parece grande), entonces el paso se divide por 2 hasta $h/2$ y se repite la estimación hasta tener éxito.

Aquí también se emplean en la práctica consideraciones similares a las realizadas en el caso del par encajado al enfrentarse a pasos excesivamente grandes o pequeños.

En el libro de R.L. BURDEN y J.D. FAIRES 'Análisis numérico', en la sección 5.8 'Métodos de extrapolación' de dicho texto, se puede ver un algoritmo que pone en marcha de forma eficaz esta extrapolación, con economía de evaluaciones en la forma que hemos sugerido.

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-01

Se considera el par encajado RK1(2) dado por el método de EULER

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}$$

y por el método

$$\begin{array}{c|cc} 0 & & \\ \frac{2}{3} & \frac{2}{3} & \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

a) Se comprobará que es un par encajado correctamente forjado, diciendo los órdenes que se emplean en efecto y las evaluaciones totales que usa. Se escribirá el conjunto del par o en la forma del cuadro habitual.

b) Seguidamente, se empleará el par para integrar el problema

$$y' = y^2, \quad y(0) = 1,$$

de solución exacta

$$y(x) = \frac{1}{1-x}$$

para una tolerancia de $E = 0.5$ en el intervalo $[0,1]$.

Se darán únicamente los tres primeros pasos.

c) También se empleará el par para integrar el problema

$$y' = -y^2, \quad y(0) = 1,$$

de solución exacta

$$y(x) = \frac{1}{x+1}$$

para una tolerancia de $E = 0.5$ en el intervalo $[0,1]$.

Aquí serán pocos los pasos a dar.

a)

```
[ > restart:
[ > interface(labeling=false):
[ Comprobación del orden del segundo método
```

```
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
```

condiciones para el orden 1

```
[ > Fi_0:=expand((1/4)*k1_0+(3/4)*k2_0+bt[3]*k3_0);
[                                     Fi_0 := f(x0, y0) + bt_3 k3_0
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                     cond1a := 1 = 1
```

condiciones para el orden 2

```
[ > Fi10:=expand((1/4)*k110+(3/4)*k210+bt[3]*k310);
[                                     Fi10 := 1/2 D_1(f)(x0, y0) + 1/2 D_2(f)(x0, y0) f(x0, y0) + bt_3 k310
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                     cond2a := 1/2 = 1/2
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                                     cond2b := 1/2 = 1/2
```

Por lo tanto hemos visto que el segundo método es de orden 2 (de hecho es el método de HEUN de 2 etapas) y hemos obtenido el tablero

0		
2/3	2/3	
ord 1	1	0
ord 2	1/4	3/4
estim	-3/4	3/4

b) el problema $y' = y^2$, $y(0) = 1$

```
> restart;  
> interface(labeling=false):  
> Digits:=3:
```

La ecuación (aut.) y su solución; valores iniciales; la tolerancia

```
> f:=y->y^2;yv:=x->1/(1-x);xini:=0;yini:=1;E:=0.5;
```

```
f:=y -> y^2  
yv:=x -> 1/(1-x)  
xini:=0  
yini:=1  
E:=0.5
```

El par

```
> a[1,2]:=2/3;b[1]:=1;b[2]:=0;bt[1]:=1/4;bt[2]:=3/4;d[1]:=bt[1]-b[1];d[2]:=bt[2]-b[2];
```

```
a1,2:=2/3  
b1:=1  
b2:=0  
bt1:=1/4  
bt2:=3/4  
d1:=-3/4  
d2:=3/4
```

Tolerancia unidad

```
> TU:=E/(1-0);
```

```
TU:=0.5
```

Cálculo de las 2 etapas, el valor de y_1 , y el consejo sobre $\alpha = \left(\frac{TU h}{|y_{t_1} - y_1|} \right)^{1/2} = \frac{TU h}{|est|}$

paso 1 con paso inicial $h = 0.5$

```
> x0:=xini;y0:=yini;h:=solve(h^2*abs(f(y0))=TU*h,h)[2];x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);yt1:=y0+h*(bt[1]*k1+bt[2]*k2);est:=h*(d[1]*k1+d[2]*k2);alpha:=(TU*h)/abs(est);
```

```
x0:=0  
y0:=1  
h:=0.500  
x1:=0.500  
k1:=1  
k2:=1.77  
y1:=1.50  
yt1:=1.79  
est:=0.290  
alpha:=0.862
```

el paso de $h = 0.5$ no se acepta, en su lugar se toma $h = \alpha * h = 0.431$

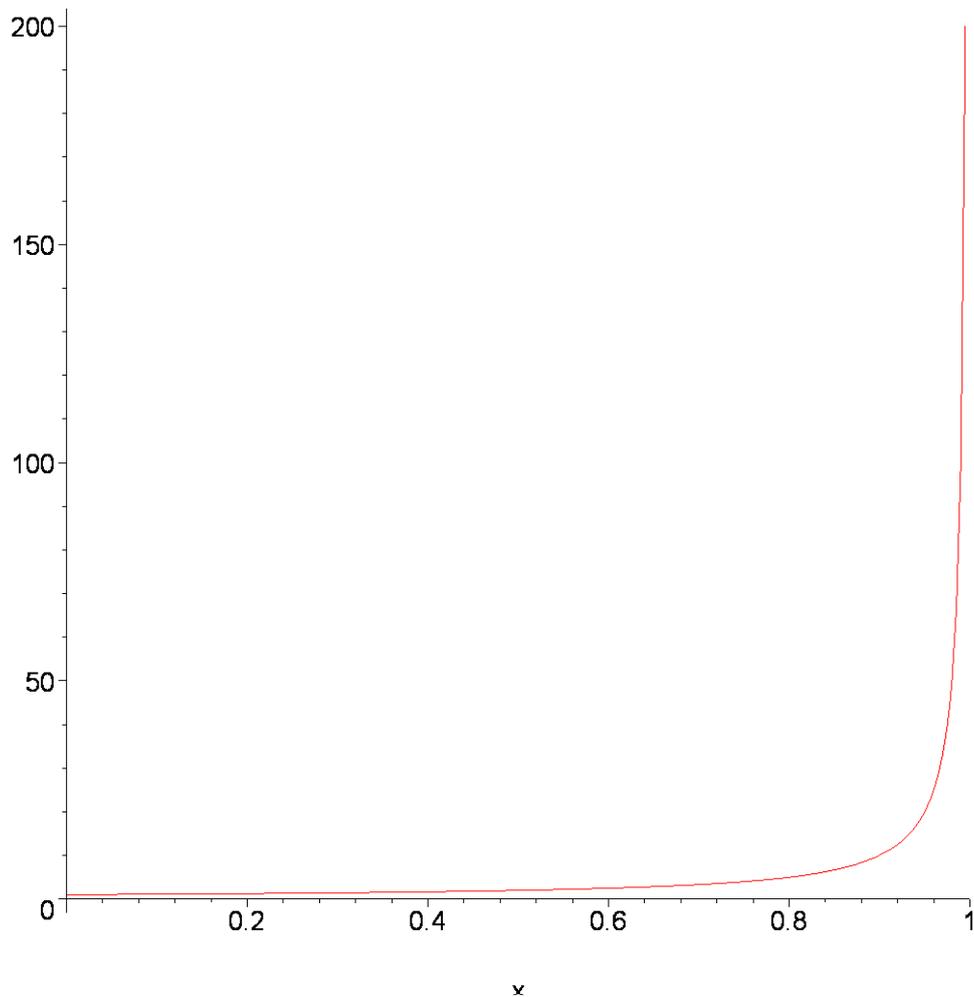
```
> h:=alpha*h;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);
```

```
h:=0.431  
x1:=0.431  
k1:=1  
k2:=1.66
```

```

                                y1 := 1.43
> verdadero_error:=yv(x1)-y1;error_admitido:=TU*(x1-xini);
                                verdadero_error := 0.33
                                error_admitido := 0.216
[ paso 2
> x0:=x0+h;y0:=y1;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);yt
  1:=y0+h*(bt[1]*k1+bt[2]*k2);est:=h*(d[1]*k1+d[2]*k2);alpha:=(TU*h)/abs(est);
                                x0 := 0.431
                                y0 := 1.43
                                x1 := 0.862
                                k1 := 2.04
                                k2 := 4.08
                                y1 := 2.31
                                yt1 := 2.97
                                est := 0.659
                                alpha := 0.328
[ el paso de h = 0.431 no se acepta, en su lugar se toma h = alpha*h = 0.141
> h:=alpha*h;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);
                                h := 0.141
                                x1 := 0.572
                                k1 := 2.04
                                k2 := 2.62
                                y1 := 1.72
> verdadero_error:=yv(x1)-y1;error_admitido:=TU*(x1-xini);
                                verdadero_error := 0.62
                                error_admitido := 0.286
[ paso 3
> x0:=x0+h;y0:=y1;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);yt
  1:=y0+h*(bt[1]*k1+bt[2]*k2);est:=h*(d[1]*k1+d[2]*k2);alpha:=(TU*h)/abs(est);
                                x0 := 0.572
                                y0 := 1.72
                                x1 := 0.713
                                k1 := 2.96
                                k2 := 4.00
                                y1 := 2.14
                                yt1 := 2.25
                                est := 0.110
                                alpha := 0.641
[ el paso de h = 0.141 no se acepta, en su lugar se toma h = alpha*h = 0.0904
> h:=alpha*h;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);
                                h := 0.0904
                                x1 := 0.662
                                k1 := 2.96
                                k2 := 3.61
                                y1 := 1.99
> verdadero_error:=yv(x1)-y1;error_admitido:=TU*(x1-xini);
                                verdadero_error := 0.97
                                error_admitido := 0.331
[ comportamiento en [0,1] de la solución
> plot(yv(x),x=0..1);

```



c) el problema $y' = -y^2$, $y(0) = 1$

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=3:
```

[La ecuación (aut.) y su solución; valores iniciales; la tolerancia

```
[ > f:=y->-y^2;yv:=x->1/(x+1);xini:=0;yini:=1;E:=0.5;
```

$$f := y \rightarrow -y^2$$

$$yv := x \rightarrow \frac{1}{x+1}$$

$$xini := 0$$

$$yini := 1$$

$$E := 0.5$$

[El par

```
[ > a[1,2]:=2/3;b[1]:=1;b[2]:=0;bt[1]:=1/4;bt[2]:=3/4;d[1]:=bt[1]-b[1];d[2]:=bt[2]-b[2];
```

$$a_{1,2} := \frac{2}{3}$$

$$b_1 := 1$$

$$b_2 := 0$$

$$bt_1 := \frac{1}{4}$$

$$bt_2 := \frac{3}{4}$$

$$d_1 := \frac{-3}{4}$$

$$d_2 := \frac{3}{4}$$

Tolerancia unidad

```
> TU:=E/(1-0);
```

TU := 0.5

Cálculo de las 2 etapas, el valor de y_1 , y el consejo sobre $\alpha = \left(\frac{TU h}{|y_{t1} - y_1|} \right)^{\frac{1}{2}} = \frac{TU h}{|est|}$

paso 1 con paso inicial $h = 0.5$

```
> x0:=xini;y0:=yini;h:=solve(h^2*abs(f(y0))=TU*h,h)[2];x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);yt1:=y0+h*(bt[1]*k1+bt[2]*k2);est:=h*(d[1]*k1+d[2]*k2);alpha:=(TU*h)/abs(est);
```

x0 := 0

y0 := 1

h := 0.500

x1 := 0.500

k1 := -1

k2 := -0.445

y1 := 0.500

yt1 := 0.708

est := 0.208

alpha := 1.20

el paso de $h = 0.5$ se acepta provisionalmente, y tomamos y_1 del método de orden 1

```
> verdadero_error:=yv(x1)-y1;error_admitido:=TU*(x1-xini);
```

verdadero_error := 0.167

error_admitido := 0.250

el nuevo paso es $h = \alpha * h = 0.6$, pero como sobrepasa $x_{end} = 1$, $h = 0.5$

```
> h:=alpha*h;h:=1.-x1;
```

h := 0.600

h := 0.500

paso 2

```
> x0:=x0+h;y0:=y1;x1:=x0+h;k1:=f(y0);k2:=f(y0+a[1,2]*h*k1);y1:=y0+h*(b[1]*k1+b[2]*k2);yt1:=y0+h*(bt[1]*k1+bt[2]*k2);est:=h*(d[1]*k1+d[2]*k2);alpha:=(TU*h)/abs(est);
```

x0 := 0.500

y0 := 0.500

x1 := 1.00

k1 := -0.250

k2 := -0.174

y1 := 0.375

yt1 := 0.404

est := 0.0290

alpha := 8.62

el paso de $h = 0.5$ se acepta provisionalmente y hemos terminado la integración; tomamos y_1 del método de orden 1

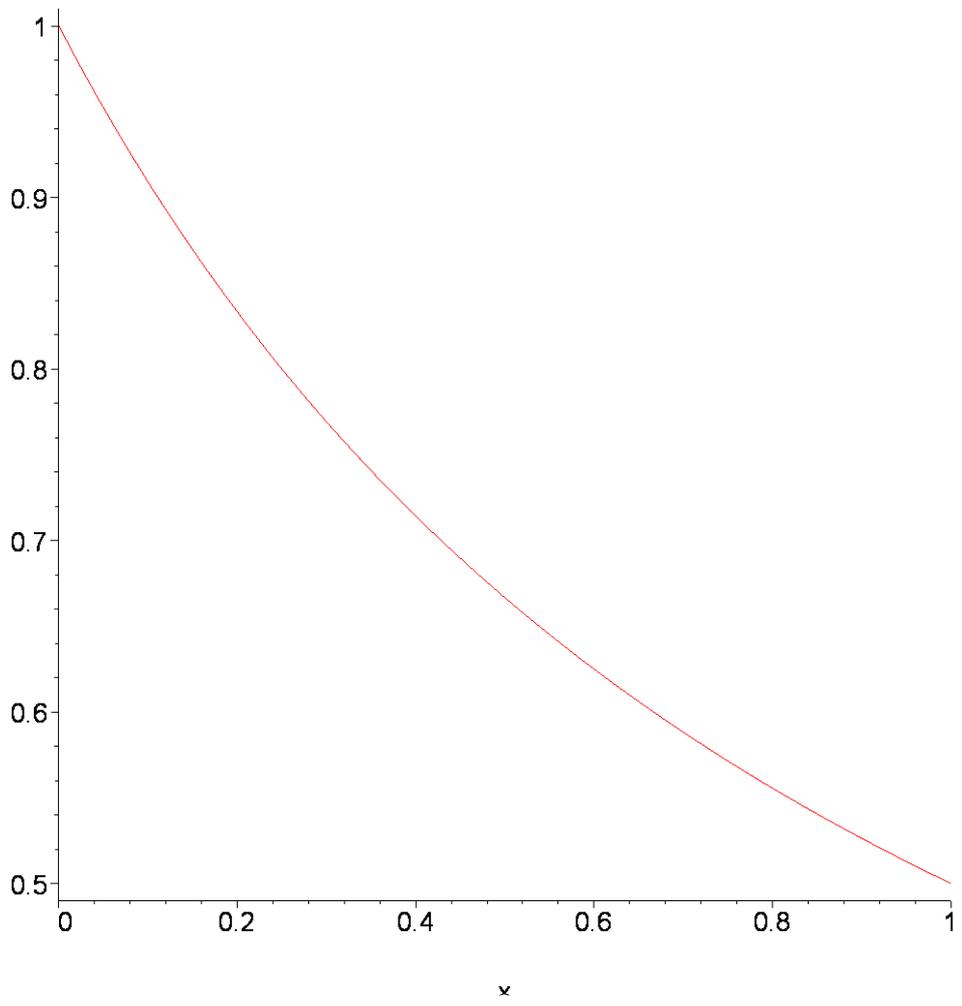
```
> verdadero_error:=yv(x1)-y1;error_admitido:=TU*(x1-xini);
```

verdadero_error := 0.125

error_admitido := 0.500

comportamiento en $[0,1]$ de la solución

```
> plot(yv(x),x=0..1);
```



Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-02

Se desea construir un par encajado RK2(3) . Se quiere que el método de orden 2 que se utiliza sea el 'método del punto medio', dado por el tablero

$$\begin{array}{c|c} 0 & \\ \frac{1}{2} & \frac{1}{2} \\ \hline & 0 \quad 1 \end{array}$$

Búsqese el correspondiente método de orden 3 que complemente el método indicado entre los de tres evaluaciones de la forma

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ c_3 & a_{31} & a_{32} & \\ \hline & bt_1 & bt_2 & bt_3 \end{array}$$

escribiendo el resultado en la forma del cuadro habitual

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ c_3 & a_{31} & a_{32} & \\ \hline \text{ord 2} & 0 & 1 & 0 \\ \hline \text{ord 3} & bt_1 & bt_2 & bt_3 \\ \hline \text{estim} & d_1 & d_2 & d_3 \end{array}$$

Como se podrá ver, una posibilidad simplificadora es hacer $c_3 = a_{31} + a_{32}$

> restart:

```
[ > interface(labeling=false):
```

El 'método del punto medio' y las condiciones para el orden 3

```
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+(1/2)*h,y0+(1/2)*h*k1_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
```

condiciones para el orden 3

```
[ > Fi20:=expand(k220);
```

$$Fi20 := \frac{1}{4} D_{1,1}(f)(x0, y0) + \frac{1}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2$$

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{1}{8}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{1}{4}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{8}$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = 0$$

```
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{6} = 0$$

El método del orden 3 con el que forma un par encajado

```
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+(1/2)*h,y0+(1/2)*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
```

```

[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
[ condiciones para el orden 1
[ > Fi_0:=expand(bt[1]*k1_0+bt[2]*k2_0+bt[3]*k3_0);
      Fi_0 := bt_1 f(x0, y0) + bt_2 f(x0, y0) + bt_3 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
      cond1a := 1 = bt_1 + bt_2 + bt_3
[ condiciones para el orden 2
[ > Fi10:=expand(bt[1]*k110+bt[2]*k210+bt[3]*k310);
      Fi10 := 1/2 bt_2 D_1(f)(x0, y0) + 1/2 bt_2 D_2(f)(x0, y0) f(x0, y0) + bt_3 D_1(f)(x0, y0) c_3 + bt_3 D_2(f)(x0, y0) a_{3,1} f(x0, y0)
      + bt_3 D_2(f)(x0, y0) a_{3,2} f(x0, y0)
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
      cond2a := 1/2 = 1/2 bt_2 + bt_3 c_3
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
      cond2b := 1/2 = 1/2 bt_2 + bt_3 a_{3,1} + bt_3 a_{3,2}
[ condiciones para el orden 3
[ > Fi20:=expand(bt[1]*k120+bt[2]*k220+bt[3]*k320);
      Fi20 := 1/4 bt_2 D_{1,1}(f)(x0, y0) + 1/2 bt_2 D_{1,2}(f)(x0, y0) f(x0, y0) + 1/4 bt_2 D_{2,2}(f)(x0, y0) f(x0, y0)^2 + bt_3 D_{1,1}(f)(x0, y0) c_3^2
      + 2 bt_3 c_3 D_{1,2}(f)(x0, y0) a_{3,1} f(x0, y0) + 2 bt_3 c_3 D_{1,2}(f)(x0, y0) a_{3,2} f(x0, y0) + bt_3 D_{2,2}(f)(x0, y0) a_{3,1}^2 f(x0, y0)^2
      + 2 bt_3 D_{2,2}(f)(x0, y0) a_{3,1} f(x0, y0)^2 a_{3,2} + bt_3 D_{2,2}(f)(x0, y0) a_{3,2}^2 f(x0, y0)^2 + bt_3 D_2(f)(x0, y0) a_{3,2} D_1(f)(x0, y0)
      + bt_3 D_2(f)(x0, y0)^2 a_{3,2} f(x0, y0)
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
      cond3a := 1/6 = 1/8 bt_2 + 1/2 bt_3 c_3^2
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
      cond3b := 1/3 = 1/4 bt_2 + bt_3 c_3 a_{3,1} + bt_3 c_3 a_{3,2}
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
      cond3c := 1/6 = 1/8 bt_2 + 1/2 bt_3 a_{3,1}^2 + bt_3 a_{3,1} a_{3,2} + 1/2 bt_3 a_{3,2}^2
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
      cond3d := 1/6 = 1/2 bt_3 a_{3,2}
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
      cond3e := 1/6 = 1/2 bt_3 a_{3,2}

```

[soluciones para el orden 3

```
> soluciones:=allvalues(solve({c[3]=a[3,1]+a[3,2], cond1a, cond2a, cond2b, cond3a, cond3b, cond3c, cond3d, cond3e}));
```

$$\text{soluciones} := \{bt_1 = \frac{1}{6} \frac{1}{c_3}, bt_2 = \frac{2-2+3c_3}{3-1+2c_3}, bt_3 = \frac{1}{6} \frac{1}{c_3(-1+2c_3)}, a_{3,1} = 3c_3 - 4c_3^2, a_{3,2} = 2c_3(-1+2c_3), c_3 = c_3\}$$

```
> soluciones:=allvalues(solve({bt[3]=1/3, c[3]=a[3,1]+a[3,2], cond1a, cond2a, cond2b, cond3a, cond3b, cond3c, cond3d, cond3e}))[2];
```

$$\text{soluciones} := \{bt_3 = \frac{1}{3}, a_{3,2} = 1, bt_1 = -\frac{1}{6} - \frac{\sqrt{5}}{6}, bt_2 = \frac{5}{6} + \frac{\sqrt{5}}{6}, a_{3,1} = -\frac{3}{4} - \frac{\sqrt{5}}{4}, c_3 = \frac{1}{4} - \frac{\sqrt{5}}{4}\}$$

```
> print(`c[3]=`, subs(soluciones, c[3]));
print(`a[3,1]=`, subs(soluciones, a[3,1]));
print(`a[3,2]=`, subs(soluciones, a[3,2]));
print(`bt[1]=`, subs(soluciones, bt[1]));
print(`bt[2]=`, subs(soluciones, bt[2]));
print(`bt[3]=`, subs(soluciones, bt[3]));
```

$$c[3]=, \frac{1}{4} - \frac{\sqrt{5}}{4}$$

$$a[3,1]=, -\frac{3}{4} - \frac{\sqrt{5}}{4}$$

$$a[3,2]=, 1$$

$$bt[1]=, -\frac{1}{6} - \frac{\sqrt{5}}{6}$$

$$bt[2]=, \frac{5}{6} + \frac{\sqrt{5}}{6}$$

$$bt[3]=, \frac{1}{3}$$

Finalmente hemos obtenido

0				
$\frac{1}{2}$			$\frac{1}{2}$	
$\frac{1+\sqrt{5}}{4}$		$\frac{-3+\sqrt{5}}{4}$		1
ord 2		0	1	0
ord 3		$\frac{-1+\sqrt{5}}{6}$	$\frac{5-\sqrt{5}}{6}$	$\frac{1}{3}$
estim		$\frac{-1+\sqrt{5}}{6}$	$\frac{-1-\sqrt{5}}{6}$	$\frac{1}{3}$

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-03 (del EXAMEN EXTRAORDINARIO de 09JUL08)

Se desea construir un par encajado RK2(3). Se quiere que el método de Runge-Kutta de orden 3 sea el método del tablero

0				
2		2		
2/3		5/9	1/9	

		1/4	0	3/4

Búscuese un método de orden 2, que posea las dos primeras etapas o las tres etapas del método anterior, y que sea adecuado para completar el RK2(3). Compruébese que el método obtenido es efectivamente conveniente para formar un par encajado. Escribábase el resultado en la forma del cuadro habitual para los pares encajados.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1);
[                                     Y1 := f(x0, y0)
[ > Y2:=subs({y(x)=y0,x=x0},y2);
[                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
[ > Y3:=subs({y(x)=y0,x=x0},y3);
[ Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
[       + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+2*h,y0+2*h*k1_):
[ > k3_:=f(x0+(2/3)*h,y0+(5/9)*h*k1_+(1/9)*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[condiciones para el orden 1

```
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0+b[3]*k3_0):  
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));  
[ 
$$cond1a := 1 = b_1 + b_2 + b_3$$

```

[condiciones para el orden 2

```
[ > Fi10:=expand(b[1]*k110+b[2]*k210+b[3]*k310):  
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));  
[ 
$$cond2a := \frac{1}{2} = 2b_2 + \frac{2}{3}b_3$$
  
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = 2b_2 + \frac{2}{3}b_3$$

```
[ > sol2:=solve({cond1a,cond2a,cond2b},{b[1],b[2]});
```

$$sol2 := \{b_1 = \frac{3}{4} - \frac{2}{3}b_3, b_2 = \frac{1}{4} - \frac{1}{3}b_3\}$$

[condiciones para el orden 3

```
[ > Fi20:=expand(b[1]*k120+b[2]*k220+b[3]*k320):  
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = 2b_2 + \frac{2}{9}b_3$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = 4b_2 + \frac{4}{9}b_3$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = 2b_2 + \frac{2}{9}b_3$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = \frac{2}{9}b_3$$

```
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{6} = \frac{2}{9}b_3$$

```
[ > sol3:=solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
```

$$sol3 := \{b_1 = \frac{1}{4}, b_2 = 0, b_3 = \frac{3}{4}\}$$

[prueba de soluciones (solo 2 etapas)

```
[ > b[3]:=0;  
[ 
$$b_3 := 0$$

```

```
[ > b[1]:=subs(sol2,b[1]);b[2]:=subs(sol2,b[2]);
```

$$b_1 := \frac{3}{4}$$

$$b_2 := \frac{1}{4}$$

```
[ > op(1,cond3a)-op(2,cond3a);  
[ op(1,cond3b)-op(2,cond3b);  
[ op(1,cond3c)-op(2,cond3c);  
[ op(1,cond3d)-op(2,cond3d);  
[ op(1,cond3e)-op(2,cond3e);
```

$$\frac{-1}{3}$$

$$\frac{-2}{3}$$
$$\frac{-1}{3}$$
$$\frac{1}{6}$$
$$\frac{1}{6}$$

[prueba de soluciones (ahora un caso de 3 etapas)

> **b[3]:=1/2;**

$$b_3 := \frac{1}{2}$$

> **b[1]:=subs(sol2,b[1]);b[2]:=subs(sol2,b[2]);**

$$b_1 := \frac{5}{12}$$

$$b_2 := \frac{1}{12}$$

> **op(1,cond3a)-op(2,cond3a);**
op(1,cond3b)-op(2,cond3b);
op(1,cond3c)-op(2,cond3c);
op(1,cond3d)-op(2,cond3d);
op(1,cond3e)-op(2,cond3e);

$$\frac{-1}{9}$$
$$\frac{-2}{9}$$
$$\frac{-1}{9}$$
$$\frac{1}{18}$$
$$\frac{1}{18}$$

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-04 (del EXAMEN EXTRAORDINARIO DE 15JUL09)

Se desea construir un par encajado RK2(3) . Se quiere que el método de orden 3 que se utiliza sea el dado por el tablero

$$\begin{array}{ccc|c} 0 & & & \\ \frac{1}{2} & & & \frac{1}{2} \\ \frac{1+\sqrt{5}}{4} & | & \frac{-3+\sqrt{5}}{4} & 1 \\ \hline & | & \frac{-1+\sqrt{5}}{6} & \frac{5-\sqrt{5}}{6} & \frac{1}{3} \end{array}$$

Búsqese el correspondiente método de orden 2 que complemente el método indicado entre los de dos o tres evaluaciones de la forma

$$\begin{array}{ccc|c} 0 & & & \\ \frac{1}{2} & & & \frac{1}{2} \\ \frac{1+\sqrt{5}}{4} & | & \frac{-3+\sqrt{5}}{4} & 1 \\ \hline \text{ord 2} & | & b_1 & b_2 & b_3 \end{array}$$

escribiendo el resultado en la forma del cuadro habitual

$$\begin{array}{ccc|c} 0 & & & \\ \frac{1}{2} & & & \frac{1}{2} \\ \frac{1+\sqrt{5}}{4} & | & \frac{-3+\sqrt{5}}{4} & 1 \\ \hline \end{array}$$

ord 2		b_1	b_2	b_3

ord 3		$\frac{-1+\sqrt{5}}{6}$	$\frac{5-\sqrt{5}}{6}$	$\frac{1}{3}$

estim		d_1	d_2	d_3

y comprobando que el método de orden 2 obtenido es adecuado para el par encajado.

```
> restart;
> interface(labeling=false):
```

Empezamos probando con sólo 2 evaluaciones, ya que hay genéricamente pares encajados de esas características. Como por obligación hay que formar un tablero de la forma

0		
$\frac{1}{2}$		$\frac{1}{2}$

ord 2		b_1 b_2

y el método buscado es de orden 2, o sea, consistente, necesariamente

$$b_1 + b_2 = 1$$

lo que será, naturalmente, la primera condición. Busquemos la necesidad de otras.

Condiciones para que el método sea de orden 2 y no sea de orden 3

```
> y1:=f(x,y(x));
                                     y1 := f(x, y(x))
> y2:=subs(diff(y(x),x)=y1,diff(y1,x));
                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
> y3:=subs(diff(y(x),x)=y1,diff(y2,x));
y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
      + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
> Y1:=subs({y(x)=y0,x=x0},y1);
                                     Y1 := f(x0, y0)
> Y2:=subs({y(x)=y0,x=x0},y2);
                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
> Y3:=subs({y(x)=y0,x=x0},y3);
Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
      + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
> k1_:=f(x0,y0);
                                     k1_ := f(x0, y0)
> k2_:=f(x0+(1/2)*h,y0+(1/2)*h*k1_);
                                     k2_ := f(x0 + 1/2 h, y0 + 1/2 h f(x0, y0))
> k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_);
                                     k3_ := f(x0 + c3 h, y0 + a3,1 h f(x0, y0) + a3,2 h f(x0 + 1/2 h, y0 + 1/2 h f(x0, y0)))
> k1_0:=subs(h=0,k1_);
                                     k1_0 := f(x0, y0)
> k2_0:=subs(h=0,k2_);
```

```

[
> k3_0:=subs(h=0,k3_);                                k2_0 := f(x0, y0)
[
> k11:=diff(k1_,h);                                    k3_0 := f(x0, y0)
[
> k21:=diff(k2_,h);                                    k11 := 0
[
> k31:=diff(k3_,h);
k21 :=  $\frac{1}{2} D_1(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_2(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0)$ 
[
> k31:=diff(k3_,h);
k31 :=  $D_1(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) c_3 +$ 
 $D_2(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) \left( a_{3,1} f(x0, y0) \right.$ 
 $+ a_{3,2} f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right)$ 
 $+ a_{3,2} h \left( \frac{1}{2} D_1(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_2(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0) \right)$ 
[
> k110:=subs(h=0,k11);                                k110 := 0
[
> k210:=subs(h=0,k21);
k210 :=  $\frac{1}{2} D_1(f)(x0, y0) + \frac{1}{2} D_2(f)(x0, y0) f(x0, y0)$ 
[
> k310:=subs(h=0,k31);
k310 :=  $D_1(f)(x0, y0) c_3 + D_2(f)(x0, y0) (a_{3,1} f(x0, y0) + a_{3,2} f(x0, y0))$ 
[
> k12:=diff(k11,h);                                    k12 := 0
[
> k22:=diff(k21,h);
k22 :=  $\frac{1}{4} D_{1,1}(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{4} D_{1,2}(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0)$ 
 $+ \frac{1}{2} \left( \frac{1}{2} D_{1,2}(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_{2,2}(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0) \right) f(x0, y0)$ 
[
> k32:=diff(k31,h);
k32 :=  $\left( D_{1,1}(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) c_3 +$ 
 $D_{1,2}(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) \left( a_{3,1} f(x0, y0) \right.$ 
 $+ a_{3,2} f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right)$ 
 $+ a_{3,2} h \left( \frac{1}{2} D_1(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_2(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0) \right)$ 
 $\left. \right) c_3 + \left($ 
 $D_{1,2}(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) c_3 +$ 
 $D_{2,2}(f) \left( x0 + c_3 h, y0 + a_{3,1} h f(x0, y0) + a_{3,2} h f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) \right) \left( a_{3,1} f(x0, y0) \right.$ 
 $+ a_{3,2} f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right)$ 
 $+ a_{3,2} h \left( \frac{1}{2} D_1(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_2(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0) \right)$ 
 $\left. \right) \left( a_{3,1} f(x0, y0) \right.$ 
 $+ a_{3,2} f \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right)$ 
 $+ a_{3,2} h \left( \frac{1}{2} D_1(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) + \frac{1}{2} D_2(f) \left( x0 + \frac{h}{2}, y0 + \frac{1}{2} h f(x0, y0) \right) f(x0, y0) \right)$ 
 $\left. \right) +$ 

```

$$D_2(f)\left(x_0 + c_3 h, y_0 + a_{3,1} h f(x_0, y_0) + a_{3,2} h f\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right)\right) \left(2 a_{3,2} \left(\frac{1}{2} D_{1,1}(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) + \frac{1}{2} D_2(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) f(x_0, y_0) \right) + a_{3,2} h \left(\frac{1}{4} D_{1,1}(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) + \frac{1}{4} D_{1,2}(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) f(x_0, y_0) + \frac{1}{2} \left(\frac{1}{2} D_{1,2}(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) + \frac{1}{2} D_{2,2}(f)\left(x_0 + \frac{h}{2}, y_0 + \frac{1}{2} h f(x_0, y_0)\right) f(x_0, y_0) \right) f(x_0, y_0) \right) \right)$$

> **k120:=subs(h=0,k12);**

$$k120 := 0$$

> **k220:=subs(h=0,k22);**

$$k220 := \frac{1}{4} D_{1,1}(f)(x_0, y_0) + \frac{1}{4} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{2} \left(\frac{1}{2} D_{1,2}(f)(x_0, y_0) + \frac{1}{2} D_{2,2}(f)(x_0, y_0) f(x_0, y_0) \right) f(x_0, y_0)$$

> **k320:=subs(h=0,k32);**

$$k320 := (D_{1,1}(f)(x_0, y_0) c_3 + D_{1,2}(f)(x_0, y_0) (a_{3,1} f(x_0, y_0) + a_{3,2} f(x_0, y_0))) c_3 + (D_{1,2}(f)(x_0, y_0) c_3 + D_{2,2}(f)(x_0, y_0) (a_{3,1} f(x_0, y_0) + a_{3,2} f(x_0, y_0))) (a_{3,1} f(x_0, y_0) + a_{3,2} f(x_0, y_0)) + 2 D_2(f)(x_0, y_0) a_{3,2} \left(\frac{1}{2} D_1(f)(x_0, y_0) + \frac{1}{2} D_2(f)(x_0, y_0) f(x_0, y_0) \right)$$

condiciones para el orden 1

> **Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);**

$$Fi_0 := b_1 f(x_0, y_0) + b_2 f(x_0, y_0)$$

> **cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));**

$$cond1a := 1 = b_1 + b_2$$

condiciones para el orden 2

> **Fi10:=expand(b[1]*k110+b[2]*k210);**

$$Fi10 := \frac{1}{2} b_2 D_1(f)(x_0, y_0) + \frac{1}{2} b_2 D_2(f)(x_0, y_0) f(x_0, y_0)$$

> **cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));**

$$cond2a := \frac{1}{2} = \frac{1}{2} b_2$$

> **cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));**

$$cond2b := \frac{1}{2} = \frac{1}{2} b_2$$

> **soluciones:=solve({cond1a,cond2a,cond2b});**

$$soluciones := \{ b_1 = 0, b_2 = 1 \}$$

o sea, esto nos obliga a que el método de orden 2, de entre los de 2 evaluaciones buscados sea el método 'del punto medio'

0 |

$\frac{1}{2}$ | $\frac{1}{2}$

| 0 1

comprobaremos que este método es uno de los adecuados para el par encajado

> **b[1]:=0;b[2]:=1;**

$$b_1 := 0$$

$$b_2 := 1$$

condiciones para el orden 3 y coeficientes del término principal del error

> **Fi20:=expand(b[1]*k120+b[2]*k220);**

$$Fi20 := \frac{1}{4} D_{1,1}(f)(x0, y0) + \frac{1}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2$$

```
> cond3a := (1/6)*coeff(Y3,D[1,1](f)(x0,y0)) = (1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
coef3a := (1/6)*coeff(Y3,D[1,1](f)(x0,y0)) - (1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{1}{8}$$

$$coef3a := \frac{1}{24}$$

```
> cond3b := (1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)), f(x0,y0)) = (1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)), f(x0,y0));
coef3b := (1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)), f(x0,y0)) - (1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)), f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{1}{4}$$

$$coef3b := \frac{1}{12}$$

```
> cond3c := (1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)), f(x0,y0)^2) = (1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)), f(x0,y0)^2);
coef3c := (1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)), f(x0,y0)^2) - (1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)), f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{8}$$

$$coef3c := \frac{1}{24}$$

```
> cond3d := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)), D[1](f)(x0,y0)) = (1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)), D[1](f)(x0,y0));
coef3d := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)), D[1](f)(x0,y0)) - (1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)), D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = 0$$

$$coef3d := \frac{1}{6}$$

```
> cond3e := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2), f(x0,y0)) = (1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2), f(x0,y0));
coef3e := (1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2), f(x0,y0)) - (1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2), f(x0,y0));
```

$$cond3e := \frac{1}{6} = 0$$

$$coef3e := \frac{1}{6}$$

```
>
```

Lo que es razonable, por lo que se obtiene el par encajado

0			
$\frac{1}{2}$		$\frac{1}{2}$	
$\frac{1+\sqrt{5}}{4}$		$\frac{-3+\sqrt{5}}{4}$	1
ord 2		0	1
		0	

ord 3		$\frac{-1+\sqrt{5}}{6}$	$\frac{5-\sqrt{5}}{6}$	$\frac{1}{3}$
--------------	--	-------------------------	------------------------	---------------

estim		$\frac{-1+\sqrt{5}}{6}$	$\frac{-1-\sqrt{5}}{6}$	$\frac{1}{3}$
--------------	--	-------------------------	-------------------------	---------------

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-05 (del EXAMEN ORDINARIO DE 09FEB10)

La idea es construir un par encajado RK2(3) empleando un máximo de 3 etapas.

Se quiere que el método de orden 3 sea el dado por el tablero

0			
$\frac{2}{3}$		$\frac{2}{3}$	
0		-1	1

		0	$\frac{3}{4}$ $\frac{1}{4}$

Compruébese que, limitándonos a las 3 etapas del método anterior, no es posible conseguir completar el par encajado que buscamos, o sea, que no es posible encontrar un método de orden 2 que sea adecuado.

```
> restart;
> interface(labeling=false);
El método de el orden 3 del enunciado
> y1:=f(x,y(x));
                                y1 := f(x, y(x))
> y2:=subs(diff(y(x),x)=y1,diff(y1,x));
                                y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
> y3:=subs(diff(y(x),x)=y1,diff(y2,x));
y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
      + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
> Y1:=subs({y(x)=y0,x=x0},y1);
                                Y1 := f(x0, y0)
> Y2:=subs({y(x)=y0,x=x0},y2);
                                Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
> Y3:=subs({y(x)=y0,x=x0},y3);
Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
      + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
> k1_:=f(x0,y0);
                                k1_ := f(x0, y0)
> k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_);
                                k2_ := f(x0 + 2/3 h, y0 + 2/3 h f(x0, y0))
> k3_:=f(x0,y0-h*k1_+h*k2_);
                                k3_ := f(x0, y0 - h f(x0, y0) + h f(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)))
> k1_0:=subs(h=0,k1_);
                                k1_0 := f(x0, y0)
```

```

> k2_0:=subs(h=0,k2_);
k2_0 := f(x0, y0)
> k3_0:=subs(h=0,k3_);
k3_0 := f(x0, y0)
> k11:=diff(k1_,h);
k11 := 0
> k21:=diff(k2_,h);
k21 :=  $\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)$ 
> k31:=diff(k3_,h);
k31 :=  $D_2(f)\left(x0, y0 - h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)\left(-f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
+  $h\left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right)$ 
> k110:=subs(h=0,k11);
k110 := 0
> k210:=subs(h=0,k21);
k210 :=  $\frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$ 
> k310:=subs(h=0,k31);
k310 := 0
> k12:=diff(k11,h);
k12 := 0
> k22:=diff(k21,h);
k22 :=  $\frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{4}{9} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)$ 
+  $\frac{2}{3}\left(\frac{2}{3} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_{2,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) f(x0, y0)$ 
> k32:=diff(k31,h);
k32 :=  $D_{2,2}(f)\left(x0, y0 - h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)\left(-f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
+  $h\left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right)^2$ 
+  $D_2(f)\left(x0, y0 - h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)\left(\frac{4}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
+  $\frac{4}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0) + h\left(\frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
+  $\frac{4}{9} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)$ 
+  $\frac{2}{3}\left(\frac{2}{3} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_{2,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) f(x0, y0)$ 
> k120:=subs(h=0,k12);
k120 := 0
> k220:=subs(h=0,k22);
k220 :=  $\frac{4}{9} D_{1,1}(f)(x0, y0) + \frac{4}{9} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{2}{3}\left(\frac{2}{3} D_{1,2}(f)(x0, y0) + \frac{2}{3} D_{2,2}(f)(x0, y0) f(x0, y0)\right) f(x0, y0)$ 
> k320:=subs(h=0,k32);
k320 :=  $D_2(f)(x0, y0)\left(\frac{4}{3} D_1(f)(x0, y0) + \frac{4}{3} D_2(f)(x0, y0) f(x0, y0)\right)$ 
condiciones para el orden 1
> Fi_0:=expand((3/4)*k2_0+(1/4)*k3_0);
Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
cond1a := 1 = 1

```

condiciones para el orden 2

> **Fi10:=expand((3/4)*k210+(1/4)*k310);**

$$Fi10 := \frac{1}{2} D_1(f)(x0, y0) + \frac{1}{2} D_2(f)(x0, y0) f(x0, y0)$$

> **cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));**

$$cond2a := \frac{1}{2} = \frac{1}{2}$$

> **cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));**

$$cond2b := \frac{1}{2} = \frac{1}{2}$$

condiciones para el orden 3

> **Fi20:=expand((3/4)*k220+(1/4)*k320);**

$$Fi20 := \frac{1}{3} D_{1,1}(f)(x0, y0) + \frac{2}{3} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{3} D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{1}{3} D_2(f)(x0, y0) D_1(f)(x0, y0) + \frac{1}{3} D_2(f)(x0, y0)^2 f(x0, y0)$$

> **cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));**

$$cond3a := \frac{1}{6} = \frac{1}{6}$$

> **cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));**

$$cond3b := \frac{1}{3} = \frac{1}{3}$$

> **cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);**

$$cond3c := \frac{1}{6} = \frac{1}{6}$$

> **cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));**

$$cond3d := \frac{1}{6} = \frac{1}{6}$$

> **cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));**

$$cond3e := \frac{1}{6} = \frac{1}{6}$$

Método de orden 2 con 2 etapas

> **y1:=f(x,y(x));**

$$y1 := f(x, y(x))$$

> **y2:=subs(diff(y(x),x)=y1,diff(y1,x));**

$$y2 := D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x))$$

> **y3:=subs(diff(y(x),x)=y1,diff(y2,x));**

$$y3 := D_{1,1}(f)(x, y(x)) + D_{1,2}(f)(x, y(x)) f(x, y(x)) + (D_{1,2}(f)(x, y(x)) + D_{2,2}(f)(x, y(x)) f(x, y(x))) f(x, y(x)) + D_2(f)(x, y(x)) (D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x)))$$

> **Y1:=subs({y(x)=y0,x=x0},y1);**

$$Y1 := f(x0, y0)$$

> **Y2:=subs({y(x)=y0,x=x0},y2);**

$$Y2 := D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0)$$

> **Y3:=subs({y(x)=y0,x=x0},y3);**

$$Y3 := D_{1,1}(f)(x0, y0) + D_{1,2}(f)(x0, y0) f(x0, y0) + (D_{1,2}(f)(x0, y0) + D_{2,2}(f)(x0, y0) f(x0, y0)) f(x0, y0) + D_2(f)(x0, y0) (D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0))$$

> **k1_:=f(x0,y0);**

$$k1_ := f(x0, y0)$$

```

> k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_);
                                k2_ := f(x0 + 2/3 h, y0 + 2/3 h f(x0, y0))
> k1_0:=subs(h=0,k1_);
                                k1_0 := f(x0, y0)
> k2_0:=subs(h=0,k2_);
                                k2_0 := f(x0, y0)
> k11:=diff(k1_,h);
                                k11 := 0
> k21:=diff(k2_,h);
                                k21 := 2/3 D1(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) + 2/3 D2(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) f(x0, y0)
> k110:=subs(h=0,k11);
                                k110 := 0
> k210:=subs(h=0,k21);
                                k210 := 2/3 D1(f)(x0, y0) + 2/3 D2(f)(x0, y0) f(x0, y0)
> k12:=diff(k11,h);
                                k12 := 0
> k22:=diff(k21,h);
                                k22 := 4/9 D1,1(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) + 4/9 D1,2(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) f(x0, y0)
                                + 2/3 (2/3 D1,2(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) + 2/3 D2,2(f)(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)) f(x0, y0)) f(x0, y0)
> k120:=subs(h=0,k12);
                                k120 := 0
> k220:=subs(h=0,k22);
                                k220 := 4/9 D1,1(f)(x0, y0) + 4/9 D1,2(f)(x0, y0) f(x0, y0) + 2/3 (2/3 D1,2(f)(x0, y0) + 2/3 D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
condiciones para el orden 1
> Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
                                Fi_0 := b1 f(x0, y0) + b2 f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                cond1a := 1 = b1 + b2
condiciones para el orden 2
> Fi10:=expand(b[1]*k110+b[2]*k210);
                                Fi10 := 2/3 b2 D1(f)(x0, y0) + 2/3 b2 D2(f)(x0, y0) f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                cond2a := 1/2 = 2/3 b2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                cond2b := 1/2 = 2/3 b2
soluciones para el orden 2
> soluciones2:=allvalues(solve({cond1a,cond2a,cond2b}));
                                soluciones2 := { b2 = 3/4, b1 = 1/4 }
como quedan las condiciones para el orden 3 con ese método
> Fi20:=expand(b[1]*k120+b[2]*k220);
                                Fi20 := 4/9 b2 D1,1(f)(x0, y0) + 8/9 b2 D1,2(f)(x0, y0) f(x0, y0) + 4/9 b2 D2,2(f)(x0, y0) f(x0, y0)^2
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));subs(soluc

```

iones2,cond3a);

$$cond3a := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));subs(soluciones2,cond3b);

$$cond3b := \frac{1}{3} = \frac{4}{9} b_2$$

$$\frac{1}{3} = \frac{1}{3}$$

> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);subs(soluciones2,cond3c);

$$cond3c := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));subs(soluciones2,cond3d);

$$cond3d := \frac{1}{6} = 0$$

$$\frac{1}{6} = 0$$

> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));subs(soluciones2,cond3e);

$$cond3e := \frac{1}{6} = 0$$

$$\frac{1}{6} = 0$$

evidentemente el método de orden 2 que hemos buscado

no es adecuado para formar un par encajado (anula 3 de

los 5 coeficientes del término principal del error).

Buscamos un método más adecuado entre los de 3 etapas

Método de el orden 2 con las tres etapas del otro método

> y1:=f(x,y(x));

$$y1 := f(x, y(x))$$

> y2:=subs(diff(y(x),x)=y1,diff(y1,x));

$$y2 := D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x))$$

> y3:=subs(diff(y(x),x)=y1,diff(y2,x));

$$y3 := D_{1,1}(f)(x, y(x)) + D_{1,2}(f)(x, y(x)) f(x, y(x)) + (D_{1,2}(f)(x, y(x)) + D_{2,2}(f)(x, y(x)) f(x, y(x))) f(x, y(x)) + D_2(f)(x, y(x)) (D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x)))$$

> Y1:=subs({y(x)=y0,x=x0},y1);

$$Y1 := f(x0, y0)$$

> Y2:=subs({y(x)=y0,x=x0},y2);

$$Y2 := D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0)$$

> Y3:=subs({y(x)=y0,x=x0},y3);

$$Y3 := D_{1,1}(f)(x0, y0) + D_{1,2}(f)(x0, y0) f(x0, y0) + (D_{1,2}(f)(x0, y0) + D_{2,2}(f)(x0, y0) f(x0, y0)) f(x0, y0) + D_2(f)(x0, y0) (D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0))$$

> k1_:=f(x0,y0);

$$k1_ := f(x0, y0)$$

> k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_);

```

[
    k2_ := f(x0 + 2h/3, y0 + 2h/3 f(x0, y0))
]
> k3_ := f(x0, y0 - h*k1_ + h*k2_);
    k3_ := f(x0, y0 - h f(x0, y0) + h f(x0 + 2h/3, y0 + 2h/3 f(x0, y0)))
]
> k1_0 := subs(h=0, k1_);
    k1_0 := f(x0, y0)
]
> k2_0 := subs(h=0, k2_);
    k2_0 := f(x0, y0)
]
> k3_0 := subs(h=0, k3_);
    k3_0 := f(x0, y0)
]
> k11 := diff(k1_, h);
    k11 := 0
]
> k21 := diff(k2_, h);
    k21 := 2/3 D1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 2/3 D2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0)
]
> k31 := diff(k3_, h);
    k31 := D2(f)(x0, y0 - h f(x0, y0) + h f(x0 + 2h/3, y0 + 2h/3 f(x0, y0))) (-f(x0, y0) + f(x0 + 2h/3, y0 + 2h/3 f(x0, y0)))
    + h (2/3 D1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 2/3 D2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0))
]
> k110 := subs(h=0, k11);
    k110 := 0
]
> k210 := subs(h=0, k21);
    k210 := 2/3 D1(f)(x0, y0) + 2/3 D2(f)(x0, y0) f(x0, y0)
]
> k310 := subs(h=0, k31);
    k310 := 0
]
> k12 := diff(k11, h);
    k12 := 0
]
> k22 := diff(k21, h);
    k22 := 4/9 D1,1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 4/9 D1,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0)
    + 2/3 (2/3 D1,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 2/3 D2,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0)) f(x0, y0)
]
> k32 := diff(k31, h);
    k32 := D2,2(f)(x0, y0 - h f(x0, y0) + h f(x0 + 2h/3, y0 + 2h/3 f(x0, y0))) (-f(x0, y0) + f(x0 + 2h/3, y0 + 2h/3 f(x0, y0)))
    + h (2/3 D1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 2/3 D2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0))^2 +
    D2(f)(x0, y0 - h f(x0, y0) + h f(x0 + 2h/3, y0 + 2h/3 f(x0, y0))) (4/3 D1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0))
    + 4/3 D2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0) + h (4/9 D1,1(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0))
    + 4/9 D1,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0))
    + 2/3 (2/3 D1,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) + 2/3 D2,2(f)(x0 + 2h/3, y0 + 2h/3 f(x0, y0)) f(x0, y0)) f(x0, y0))
]
> k120 := subs(h=0, k12);
    k120 := 0
]
> k220 := subs(h=0, k22);
    k220 := 4/9 D1,1(f)(x0, y0) + 4/9 D1,2(f)(x0, y0) f(x0, y0) + 2/3 (2/3 D1,2(f)(x0, y0) + 2/3 D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
]
> k320 := subs(h=0, k32);

```

$$k320 := D_2(f)(x0, y0) \left(\frac{4}{3} D_1(f)(x0, y0) + \frac{4}{3} D_2(f)(x0, y0) f(x0, y0) \right)$$

condiciones para el orden 1

> `Fi_0:=expand(b[1]*k1_0+b[2]*k2_0+b[3]*k3_0);`

$$Fi_0 := b_1 f(x0, y0) + b_2 f(x0, y0) + b_3 f(x0, y0)$$

> `cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));`

$$cond1a := 1 = b_1 + b_2 + b_3$$

condiciones para el orden 2

> `Fi10:=expand(b[1]*k110+b[2]*k210+b[3]*k310);`

$$Fi10 := \frac{2}{3} b_2 D_1(f)(x0, y0) + \frac{2}{3} b_2 D_2(f)(x0, y0) f(x0, y0)$$

> `cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));`

$$cond2a := \frac{1}{2} = \frac{2}{3} b_2$$

> `cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));`

$$cond2b := \frac{1}{2} = \frac{2}{3} b_2$$

soluciones para el orden 2

> `soluciones2:=allvalues(solve({cond1a,cond2a,cond2b}));`

$$soluciones2 := \{ b_2 = \frac{3}{4}, b_3 = \frac{1}{4} - b_1, b_1 = b_1 \}$$

como quedan las condiciones para el orden 3 con ese método

> `Fi20:=expand(b[1]*k120+b[2]*k220+b[3]*k320);`

$$Fi20 := \frac{4}{9} b_2 D_{1,1}(f)(x0, y0) + \frac{8}{9} b_2 D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{4}{9} b_2 D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{4}{3} b_3 D_2(f)(x0, y0) D_1(f)(x0, y0) + \frac{4}{3} b_3 D_2(f)(x0, y0)^2 f(x0, y0)$$

> `cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));subs(soluciones2,cond3a);`

$$cond3a := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));subs(soluciones2,cond3b);`

$$cond3b := \frac{1}{3} = \frac{4}{9} b_2$$

$$\frac{1}{3} = \frac{1}{3}$$

> `cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);subs(soluciones2,cond3c);`

$$cond3c := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));subs(soluciones2,cond3d);`

$$cond3d := \frac{1}{6} = \frac{2}{3} b_3$$

$$\frac{1}{6} = \frac{1}{6} - \frac{2}{3} b_1$$

> `cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));subs(soluciones2,cond3e);`

$$cond3e := \frac{1}{6} = \frac{2}{3} b_3$$

$$\frac{1}{6} = \frac{1}{6} - \frac{2}{3} b_1$$

Ocurre lo mismo que antes: no es posible encontrar ningún método de orden 2 que no anule al menos 3 de los coeficientes del término principal del error. Por lo tanto, limitándose a las 3 etapas del método del enunciado, no es posible completar el par con ningún método de orden 2

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-06 (del EXAMEN ORDINARIO DE 19ENE12)

La idea es construir un par encajado RK2(3) empleando un máximo de 3 etapas.

Se quiere que el método de orden 3 sea el dado por el tablero

0			
$\frac{2}{3}$		$\frac{2}{3}$	
$\frac{2}{3}$		$-\frac{1}{3}$	1

		$\frac{1}{4}$	$\frac{1}{2}$
		$\frac{1}{4}$	$\frac{1}{4}$

Compruébese que, limitándonos a las 3 etapas del método anterior, no es posible conseguir completar el par encajado que buscamos, o sea, que no es posible encontrar un método de orden 2 que sea adecuado.

```
> restart;
> interface(labeling=false):
El método de el orden 3 del enunciado
> y1:=f(x,y(x));
                                     y1 := f(x, y(x))
> y2:=subs(diff(y(x),x)=y1,diff(y1,x));
                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
> y3:=subs(diff(y(x),x)=y1,diff(y2,x));
y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
      + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
> Y1:=subs({y(x)=y0,x=x0},y1);
                                     Y1 := f(x0, y0)
> Y2:=subs({y(x)=y0,x=x0},y2);
                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
> Y3:=subs({y(x)=y0,x=x0},y3);
Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
      + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
> k1_:=f(x0,y0);
                                     k1_ := f(x0, y0)
> k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_);
                                     k2_ := f(x0 + 2/3 h, y0 + 2/3 h f(x0, y0))
> k3_:=f(x0+(2/3)*h,y0+(-1/3)*h*k1_+h*k2_);
                                     k3_ := f(x0 + 2/3 h, y0 - 1/3 h f(x0, y0) + h f(x0 + 2/3 h, y0 + 2/3 h f(x0, y0)))
> k1_0:=subs(h=0,k1_);
```

```

[
> k2_0:=subs(h=0,k2_);                                k1_0 := f(x0, y0)
[
> k3_0:=subs(h=0,k3_);                                k2_0 := f(x0, y0)
[
> k11:=diff(k1_,h);                                   k3_0 := f(x0, y0)
[
> k21:=diff(k2_,h);                                   k11 := 0
[
k21 :=  $\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)$ 
[
> k31:=diff(k3_,h);
k31 :=  $\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) +$ 
 $D_2(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) \left(-\frac{1}{3} f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
 $+ h \left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right)$ 
[
> k110:=subs(h=0,k11);                                k110 := 0
[
> k210:=subs(h=0,k21);
k210 :=  $\frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$ 
[
> k310:=subs(h=0,k31);
k310 :=  $\frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$ 
[
> k12:=diff(k11,h);                                   k12 := 0
[
> k22:=diff(k21,h);
k22 :=  $\frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{4}{9} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)$ 
 $+ \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_{2,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) f(x0, y0)$ 
[
> k32:=diff(k31,h);
k32 :=  $\frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) + \frac{2}{3}$ 
 $D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) \left(-\frac{1}{3} f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
 $+ h \left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) + \left(\frac{2}{3} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) +$ 
 $D_{2,2}(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) \left(-\frac{1}{3} f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
 $+ h \left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) \left(-\frac{1}{3} f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
 $+ f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)$ 
 $+ h \left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0)\right) +$ 
 $D_2(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right) \left(\frac{4}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 
 $+ \frac{4}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right) f(x0, y0) + h \left(\frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0)\right)\right)$ 

```

$$+ \frac{4}{9} D_{1,2}(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f(x_0, y_0) \right) f(x_0, y_0) \\ + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f(x_0, y_0) \right) + \frac{2}{3} D_{2,2}(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f(x_0, y_0) \right) f(x_0, y_0) \right) f(x_0, y_0) \Bigg)$$

> **k120:=subs(h=0,k12);**

$$k120 := 0$$

> **k220:=subs(h=0,k22);**

$$k220 := \frac{4}{9} D_{1,1}(f)(x_0, y_0) + \frac{4}{9} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x_0, y_0) + \frac{2}{3} D_{2,2}(f)(x_0, y_0) f(x_0, y_0) \right) f(x_0, y_0)$$

> **k320:=subs(h=0,k32);**

$$k320 := \frac{4}{9} D_{1,1}(f)(x_0, y_0) + \frac{4}{9} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x_0, y_0) + \frac{2}{3} D_{2,2}(f)(x_0, y_0) f(x_0, y_0) \right) f(x_0, y_0) \\ + D_2(f)(x_0, y_0) \left(\frac{4}{3} D_1(f)(x_0, y_0) + \frac{4}{3} D_2(f)(x_0, y_0) f(x_0, y_0) \right)$$

condiciones para el orden 1

> **Fi_0:=expand((1/4)*k1_0+(1/2)*k2_0+(1/4)*k3_0);**

$$Fi_0 := f(x_0, y_0)$$

> **cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));**

$$cond1a := 1 = 1$$

condiciones para el orden 2

> **Fi10:=expand((1/4)*k110+(1/2)*k210+(1/4)*k310);**

$$Fi10 := \frac{1}{2} D_1(f)(x_0, y_0) + \frac{1}{2} D_2(f)(x_0, y_0) f(x_0, y_0)$$

> **cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));**

$$cond2a := \frac{1}{2} = \frac{1}{2}$$

> **cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));**

$$cond2b := \frac{1}{2} = \frac{1}{2}$$

condiciones para el orden 3

> **Fi20:=expand((1/4)*k120+(1/2)*k220+(1/4)*k320);**

$$Fi20 := \frac{1}{3} D_{1,1}(f)(x_0, y_0) + \frac{2}{3} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{3} D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2 + \frac{1}{3} D_2(f)(x_0, y_0) D_1(f)(x_0, y_0) \\ + \frac{1}{3} D_2(f)(x_0, y_0)^2 f(x_0, y_0)$$

> **cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));**

$$cond3a := \frac{1}{6} = \frac{1}{6}$$

> **cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));**

$$cond3b := \frac{1}{3} = \frac{1}{3}$$

> **cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);**

$$cond3c := \frac{1}{6} = \frac{1}{6}$$

> **cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));**

$$cond3d := \frac{1}{6} = \frac{1}{6}$$

> **cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));**

$$cond3e := \frac{1}{6} = \frac{1}{6}$$

Método de orden 2 con 2 etapas

> **y1:=f(x,y(x));**

$$y1 := f(x, y(x))$$

> **y2:=subs(diff(y(x),x)=y1,diff(y1,x));**

$$y2 := D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x))$$

> **y3:=subs(diff(y(x),x)=y1,diff(y2,x));**

$$y3 := D_{1,1}(f)(x, y(x)) + D_{1,2}(f)(x, y(x)) f(x, y(x)) + (D_{1,2}(f)(x, y(x)) + D_{2,2}(f)(x, y(x)) f(x, y(x))) f(x, y(x)) + D_2(f)(x, y(x)) (D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x)))$$

> **Y1:=subs({y(x)=y0,x=x0},y1);**

$$Y1 := f(x0, y0)$$

> **Y2:=subs({y(x)=y0,x=x0},y2);**

$$Y2 := D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0)$$

> **Y3:=subs({y(x)=y0,x=x0},y3);**

$$Y3 := D_{1,1}(f)(x0, y0) + D_{1,2}(f)(x0, y0) f(x0, y0) + (D_{1,2}(f)(x0, y0) + D_{2,2}(f)(x0, y0) f(x0, y0)) f(x0, y0) + D_2(f)(x0, y0) (D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0))$$

> **k1_:=f(x0,y0);**

$$k1_ := f(x0, y0)$$

> **k2_:=f(x0+(2/3)*h,y0+(2/3)*h*k1_);**

$$k2_ := f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)$$

> **k1_0:=subs(h=0,k1_);**

$$k1_0 := f(x0, y0)$$

> **k2_0:=subs(h=0,k2_);**

$$k2_0 := f(x0, y0)$$

> **k11:=diff(k1_,h);**

$$k11 := 0$$

> **k21:=diff(k2_,h);**

$$k21 := \frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) f(x0, y0)$$

> **k110:=subs(h=0,k11);**

$$k110 := 0$$

> **k210:=subs(h=0,k21);**

$$k210 := \frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$$

> **k12:=diff(k11,h);**

$$k12 := 0$$

> **k22:=diff(k21,h);**

$$k22 := \frac{4}{9} D_{1,1}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) + \frac{4}{9} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) f(x0, y0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) + \frac{2}{3} D_{2,2}(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) f(x0, y0) \right) f(x0, y0)$$

> **k120:=subs(h=0,k12);**

$$k120 := 0$$

> **k220:=subs(h=0,k22);**

$$k220 := \frac{4}{9} D_{1,1}(f)(x0, y0) + \frac{4}{9} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x0, y0) + \frac{2}{3} D_{2,2}(f)(x0, y0) f(x0, y0) \right) f(x0, y0)$$

condiciones para el orden 1

> **Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);**

$$Fi_0 := b_1 f(x0, y0) + b_2 f(x0, y0)$$

> **cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));**

$$cond1a := 1 = b_1 + b_2$$

[condiciones para el orden 2

> `Fi10:=expand(b[1]*k110+b[2]*k210);`

$$Fi10 := \frac{2}{3} b_2 D_1(f)(x_0, y_0) + \frac{2}{3} b_2 D_2(f)(x_0, y_0) f(x_0, y_0)$$

> `cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));`

$$cond2a := \frac{1}{2} = \frac{2}{3} b_2$$

> `cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));`

$$cond2b := \frac{1}{2} = \frac{2}{3} b_2$$

[soluciones para el orden 2

> `soluciones2:=allvalues(solve({cond1a,cond2a,cond2b}));`

$$soluciones2 := \{b_1 = \frac{1}{4}, b_2 = \frac{3}{4}\}$$

[como quedan las condiciones para el orden 3 con ese método

> `Fi20:=expand(b[1]*k120+b[2]*k220);`

$$Fi20 := \frac{4}{9} b_2 D_{1,1}(f)(x_0, y_0) + \frac{8}{9} b_2 D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{4}{9} b_2 D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2$$

> `cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));subs(soluciones2,cond3a);`

$$cond3a := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));subs(soluciones2,cond3b);`

$$cond3b := \frac{1}{3} = \frac{4}{9} b_2$$

$$\frac{1}{3} = \frac{1}{3}$$

> `cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);subs(soluciones2,cond3c);`

$$cond3c := \frac{1}{6} = \frac{2}{9} b_2$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));subs(soluciones2,cond3d);`

$$cond3d := \frac{1}{6} = 0$$

$$\frac{1}{6} = 0$$

> `cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));subs(soluciones2,cond3e);`

$$cond3e := \frac{1}{6} = 0$$

$$\frac{1}{6} = 0$$

evidentemente el método de orden 2 que hemos buscado

no es adecuado para formar un par encajado (anula 3 de

los 5 coeficientes del término principal del error).

Buscamos un método más adecuado entre los de 3 etapas

Método de el orden 2 con las tres etapas del otro método

> $y1 := f(x, y(x));$

$$y1 := f(x, y(x))$$

> $y2 := \text{subs}(\text{diff}(y(x), x) = y1, \text{diff}(y1, x));$

$$y2 := D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x))$$

> $y3 := \text{subs}(\text{diff}(y(x), x) = y1, \text{diff}(y2, x));$

$$y3 := D_{1,1}(f)(x, y(x)) + D_{1,2}(f)(x, y(x)) f(x, y(x)) + (D_{1,2}(f)(x, y(x)) + D_{2,2}(f)(x, y(x)) f(x, y(x))) f(x, y(x)) \\ + D_2(f)(x, y(x)) (D_1(f)(x, y(x)) + D_2(f)(x, y(x)) f(x, y(x)))$$

> $Y1 := \text{subs}(\{y(x) = y0, x = x0\}, y1);$

$$Y1 := f(x0, y0)$$

> $Y2 := \text{subs}(\{y(x) = y0, x = x0\}, y2);$

$$Y2 := D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0)$$

> $Y3 := \text{subs}(\{y(x) = y0, x = x0\}, y3);$

$$Y3 := D_{1,1}(f)(x0, y0) + D_{1,2}(f)(x0, y0) f(x0, y0) + (D_{1,2}(f)(x0, y0) + D_{2,2}(f)(x0, y0) f(x0, y0)) f(x0, y0) \\ + D_2(f)(x0, y0) (D_1(f)(x0, y0) + D_2(f)(x0, y0) f(x0, y0))$$

> $k1_ := f(x0, y0);$

$$k1_ := f(x0, y0)$$

> $k2_ := f(x0 + (2/3)*h, y0 + (2/3)*h*k1_);$

$$k2_ := f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)$$

> $k3_ := f(x0 + (2/3)*h, y0 - (1/3)*h*k1_ + h*k2_);$

$$k3_ := f\left(x0 + \frac{2h}{3}, y0 - \frac{1h}{3} f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)\right)$$

> $k1_0 := \text{subs}(h=0, k1_);$

$$k1_0 := f(x0, y0)$$

> $k2_0 := \text{subs}(h=0, k2_);$

$$k2_0 := f(x0, y0)$$

> $k3_0 := \text{subs}(h=0, k3_);$

$$k3_0 := f(x0, y0)$$

> $k11 := \text{diff}(k1_ , h);$

$$k11 := 0$$

> $k21 := \text{diff}(k2_ , h);$

$$k21 := \frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) f(x0, y0)$$

> $k31 := \text{diff}(k3_ , h);$

$$k31 := \frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1h}{3} f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)\right) + \\ D_2(f)\left(x0 + \frac{2h}{3}, y0 - \frac{1h}{3} f(x0, y0) + h f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)\right) \left(-\frac{1}{3} f(x0, y0) + f\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right)\right) \\ + h \left(\frac{2}{3} D_1(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) + \frac{2}{3} D_2(f)\left(x0 + \frac{2h}{3}, y0 + \frac{2h}{3} f(x0, y0)\right) f(x0, y0)\right)$$

> $k110 := \text{subs}(h=0, k11);$

$$k110 := 0$$

> $k210 := \text{subs}(h=0, k21);$

$$k210 := \frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$$

> $k310 := \text{subs}(h=0, k31);$

$$k310 := \frac{2}{3} D_1(f)(x0, y0) + \frac{2}{3} D_2(f)(x0, y0) f(x0, y0)$$

> $k12 := \text{diff}(k11, h);$

$$k12 := 0$$

> $k22 := \text{diff}(k21, h);$

$$k22 := \frac{4}{9} D_{1,1}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{4}{9} D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \\ + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{2}{3} D_{2,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \right) f(x0, y0)$$

> **k32:=diff(k31,h);**

$$k32 := \frac{4}{9} D_{1,1}(f) \left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) + \frac{2}{3} \\ D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \left(-\frac{1}{3} f(x0, y0) + f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \\ + h \left(\frac{2}{3} D_1(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{2}{3} D_2(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \right) + \left(\right. \\ \left. \frac{2}{3} D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) + \right. \\ \left. D_{2,2}(f) \left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \left(-\frac{1}{3} f(x0, y0) + f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \right) \\ + h \left(\frac{2}{3} D_1(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{2}{3} D_2(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \right) \left(-\frac{1}{3} f(x0, y0) \right) \\ + f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \\ + h \left(\frac{2}{3} D_1(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{2}{3} D_2(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \right) + \\ D_2(f) \left(x0 + \frac{2h}{3}, y0 - \frac{1}{3} h f(x0, y0) + h f \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \left(\frac{4}{3} D_1(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \\ + \frac{4}{3} D_2(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) + h \left(\frac{4}{9} D_{1,1}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) \right) \\ + \frac{4}{9} D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \\ + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) + \frac{2}{3} D_{2,2}(f) \left(x0 + \frac{2h}{3}, y0 + \frac{2}{3} h f(x0, y0) \right) f(x0, y0) \right) f(x0, y0) \left. \right)$$

> **k120:=subs(h=0,k12);**

$$k120 := 0$$

> **k220:=subs(h=0,k22);**

$$k220 := \frac{4}{9} D_{1,1}(f)(x0, y0) + \frac{4}{9} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x0, y0) + \frac{2}{3} D_{2,2}(f)(x0, y0) f(x0, y0) \right) f(x0, y0)$$

> **k320:=subs(h=0,k32);**

$$k320 := \frac{4}{9} D_{1,1}(f)(x0, y0) + \frac{4}{9} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x0, y0) + \frac{2}{3} D_{2,2}(f)(x0, y0) f(x0, y0) \right) f(x0, y0) \\ + D_2(f)(x0, y0) \left(\frac{4}{3} D_1(f)(x0, y0) + \frac{4}{3} D_2(f)(x0, y0) f(x0, y0) \right)$$

condiciones para el orden 1

> **Fi_0:=expand(b[1]*k1_0+b[2]*k2_0+b[3]*k3_0);**

$$Fi_0 := b_1 f(x0, y0) + b_2 f(x0, y0) + b_3 f(x0, y0)$$

> **cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));**

$$cond1a := 1 = b_1 + b_2 + b_3$$

condiciones para el orden 2

> **Fi10:=expand(b[1]*k110+b[2]*k210+b[3]*k310);**

$$Fi10 := \frac{2}{3} b_2 D_1(f)(x0, y0) + \frac{2}{3} b_2 D_2(f)(x0, y0) f(x0, y0) + \frac{2}{3} b_3 D_1(f)(x0, y0) + \frac{2}{3} b_3 D_2(f)(x0, y0) f(x0, y0)$$

> **cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));**

$$cond2a := \frac{1}{2} = \frac{2}{3} b_2 + \frac{2}{3} b_3$$

> **cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));**

$$\text{cond2b} := \frac{1}{2} = \frac{2}{3}b_2 + \frac{2}{3}b_3$$

[soluciones para el orden 2

> `soluciones2:=allvalues(solve({cond1a,cond2a,cond2b}));`

$$\text{soluciones2} := \{b_1 = \frac{1}{4}, b_2 = \frac{3}{4} - b_3, b_3 = b_3\}$$

[como quedan las condiciones para el orden 3 con ese método

> `Fi20:=expand(b[1]*k120+b[2]*k220+b[3]*k320);`

$$\begin{aligned} \text{Fi20} := & \frac{4}{9}b_2 D_{1,1}(f)(x0, y0) + \frac{8}{9}b_2 D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{4}{9}b_2 D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{4}{9}b_3 D_{1,1}(f)(x0, y0) \\ & + \frac{8}{9}b_3 D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{4}{9}b_3 D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{4}{3}b_3 D_2(f)(x0, y0) D_1(f)(x0, y0) \\ & + \frac{4}{3}b_3 D_2(f)(x0, y0)^2 f(x0, y0) \end{aligned}$$

> `cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));subs(soluciones2,cond3a);`

$$\text{cond3a} := \frac{1}{6} = \frac{2}{9}b_2 + \frac{2}{9}b_3$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));subs(soluciones2,cond3b);`

$$\text{cond3b} := \frac{1}{3} = \frac{4}{9}b_2 + \frac{4}{9}b_3$$

$$\frac{1}{3} = \frac{1}{3}$$

> `cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);subs(soluciones2,cond3c);`

$$\text{cond3c} := \frac{1}{6} = \frac{2}{9}b_2 + \frac{2}{9}b_3$$

$$\frac{1}{6} = \frac{1}{6}$$

> `cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));subs(soluciones2,cond3d);`

$$\text{cond3d} := \frac{1}{6} = \frac{2}{3}b_3$$

$$\frac{1}{6} = \frac{2}{3}b_3$$

> `cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));subs(soluciones2,cond3e);`

$$\text{cond3e} := \frac{1}{6} = \frac{2}{3}b_3$$

$$\frac{1}{6} = \frac{2}{3}b_3$$

[>
[>
[>
[>
[>
[>
[>
[>
[>
[>
[>

[Ocurre lo mismo que antes: no es posible encontrar ningún método

de orden 2 que no anule al menos 3 de los coeficientes del término principal del error. Por lo tanto, limitándose a las 3 etapas del método del enunciado, no es posible completar el par con ningún método de orden 2

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-07 (del EXAMEN ORDINARIO de 01FEB13)

La idea es construir un par encajado RK2(3) empleando las 3 etapas del conocido método de orden 3 dado por el tablero

0				
$\frac{1}{3}$		$\frac{1}{3}$		
$\frac{2}{3}$		0	$\frac{2}{3}$	

ord 2		b_1	b_2	b_3

ord 3		$\frac{1}{4}$	0	$\frac{3}{4}$

estim		d_1	d_2	d_3

Compruébese que, con esas 3 etapas, se pueden encontrar infinitos métodos de orden 2 para completar el par, que se describirán por el sistema de ecuaciones que deban cumplir.

Es sabido, sin embargo, que no todos son válidos para completar un buen par encajado. ¿Qué condiciones se deben comprobar para que el método de orden 2 esté entre los que resultan adecuados? Busca un ejemplo concreto de método de orden 2, con las etapas dadas, y que permita formar un buen par encajado con el método de orden 3 del enunciado.

```
[ > restart;
[ > interface(labeling=false);
[ Condiciones para que el método sea de orden 2 y no sea de orden 3
[ > y1:=f(x,y(x));
[                                     y1 := f(x, y(x))
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x));
[                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x));
[ y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
```

```

+ D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
> Y1:=subs({y(x)=y0,x=x0},Y1);
Y1 := f(x0, y0)
> Y2:=subs({y(x)=y0,x=x0},Y2);
Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
> Y3:=subs({y(x)=y0,x=x0},Y3);
Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
+ D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
> k1_:=f(x0,y0);
k1_ := f(x0, y0)
> k2_:=f(x0+(1/3)*h,y0+(1/3)*h*k1_);
k2_ := f(x0 + h/3, y0 + 1/3 h f(x0, y0))
> k3_:=f(x0+(2/3)*h,y0+(2/3)*h*k2_);
k3_ := f(x0 + 2h/3, y0 + 2/3 h f(x0 + h/3, y0 + 1/3 h f(x0, y0)))
> k1_0:=subs(h=0,k1_);
k1_0 := f(x0, y0)
> k2_0:=subs(h=0,k2_);
k2_0 := f(x0, y0)
> k3_0:=subs(h=0,k3_);
k3_0 := f(x0, y0)
> k11:=diff(k1_,h);
k11 := 0
> k21:=diff(k2_,h);
k21 := 1/3 D1(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 1/3 D2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) f(x0, y0)
> k31:=diff(k3_,h);
k31 := 2/3 D1(f)(x0 + 2h/3, y0 + 2/3 h f(x0 + h/3, y0 + 1/3 h f(x0, y0))) + D2(f)(x0 + 2h/3, y0 + 2/3 h f(x0 + h/3, y0 + 1/3 h f(x0, y0)))
( 2/3 f(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 2/3 h (1/3 D1(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 1/3 D2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) f(x0, y0)) )
> k110:=subs(h=0,k11);
k110 := 0
> k210:=subs(h=0,k21);
k210 := 1/3 D1(f)(x0, y0) + 1/3 D2(f)(x0, y0) f(x0, y0)
> k310:=subs(h=0,k31);
k310 := 2/3 D1(f)(x0, y0) + 2/3 D2(f)(x0, y0) f(x0, y0)
> k12:=diff(k11,h);
k12 := 0
> k22:=diff(k21,h);
k22 := 1/9 D1,1(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 1/9 D1,2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) f(x0, y0)
+ 1/3 (1/3 D1,2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 1/3 D2,2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) f(x0, y0)) f(x0, y0)
> k32:=diff(k31,h);
k32 := 4/9 D1,1(f)(x0 + 2h/3, y0 + 2/3 h f(x0 + h/3, y0 + 1/3 h f(x0, y0))) + 2/3
D1,2(f)(x0 + 2h/3, y0 + 2/3 h f(x0 + h/3, y0 + 1/3 h f(x0, y0)))
( 2/3 f(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 2/3 h (1/3 D1(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) + 1/3 D2(f)(x0 + h/3, y0 + 1/3 h f(x0, y0)) f(x0, y0)) )

```

$$\begin{aligned}
& + \left(\frac{2}{3} D_{1,2}(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right) + D_{2,2}(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right) \right) \left(\frac{2}{3} f \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) + \frac{2}{3} h \left(\frac{1}{3} D_1(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) + \frac{1}{3} D_2(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) f(x_0, y_0) \right) \right) \\
& \left(\frac{2}{3} f \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) + \frac{2}{3} h \left(\frac{1}{3} D_1(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) + \frac{1}{3} D_2(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) f(x_0, y_0) \right) \right) \\
& + D_2(f) \left(x_0 + \frac{2h}{3}, y_0 + \frac{2}{3} h f \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right) \left(\frac{4}{9} D_1(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right. \\
& \left. + \frac{4}{9} D_2(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) f(x_0, y_0) + \frac{2}{3} h \left(\frac{1}{9} D_{1,1}(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right. \right. \\
& \left. \left. + \frac{1}{9} D_{1,2}(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) \right) f(x_0, y_0) \right) \\
& \left. + \frac{1}{3} \left(\frac{1}{3} D_{1,2}(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) + \frac{1}{3} D_{2,2}(f) \left(x_0 + \frac{h}{3}, y_0 + \frac{1}{3} h f(x_0, y_0) \right) f(x_0, y_0) \right) f(x_0, y_0) \right)
\end{aligned}$$

> **k120:=subs(h=0,k12);**

$$k120 := 0$$

> **k220:=subs(h=0,k22);**

$$k220 := \frac{1}{9} D_{1,1}(f)(x_0, y_0) + \frac{1}{9} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{3} \left(\frac{1}{3} D_{1,2}(f)(x_0, y_0) + \frac{1}{3} D_{2,2}(f)(x_0, y_0) f(x_0, y_0) \right) f(x_0, y_0)$$

> **k320:=subs(h=0,k32);**

$$\begin{aligned}
k320 := & \frac{4}{9} D_{1,1}(f)(x_0, y_0) + \frac{4}{9} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{2}{3} \left(\frac{2}{3} D_{1,2}(f)(x_0, y_0) + \frac{2}{3} D_{2,2}(f)(x_0, y_0) f(x_0, y_0) \right) f(x_0, y_0) \\
& + D_2(f)(x_0, y_0) \left(\frac{4}{9} D_1(f)(x_0, y_0) + \frac{4}{9} D_2(f)(x_0, y_0) f(x_0, y_0) \right)
\end{aligned}$$

condiciones para el orden 1

> **Fi_0:=expand(b[1]*k1_0+b[2]*k2_0+b[3]*k3_0);**

$$Fi_0 := b_1 f(x_0, y_0) + b_2 f(x_0, y_0) + b_3 f(x_0, y_0)$$

> **cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));**

$$cond1a := 1 = b_1 + b_2 + b_3$$

condiciones para el orden 2

> **Fi10:=expand(b[1]*k110+b[2]*k210+b[3]*k310);**

$$Fi10 := \frac{1}{3} b_2 D_1(f)(x_0, y_0) + \frac{1}{3} b_2 D_2(f)(x_0, y_0) f(x_0, y_0) + \frac{2}{3} b_3 D_1(f)(x_0, y_0) + \frac{2}{3} b_3 D_2(f)(x_0, y_0) f(x_0, y_0)$$

> **cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));**

$$cond2a := \frac{1}{2} = \frac{1}{3} b_2 + \frac{2}{3} b_3$$

> **cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));**

$$cond2b := \frac{1}{2} = \frac{1}{3} b_2 + \frac{2}{3} b_3$$

> **orden2:=solve({cond1a,cond2a,cond2b});**

$$orden2 := \{ b_2 = \frac{1}{2} - 2 b_1, b_3 = \frac{1}{2} + b_1, b_1 = b_1 \}$$

condiciones para el orden 3 y coeficientes del término principal del error

> **Fi20:=expand(b[1]*k120+b[2]*k220+b[3]*k320);**

$$\begin{aligned}
Fi20 := & \frac{1}{9} b_2 D_{1,1}(f)(x_0, y_0) + \frac{2}{9} b_2 D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{9} b_2 D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2 + \frac{4}{9} b_3 D_{1,1}(f)(x_0, y_0) \\
& + \frac{8}{9} b_3 D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{4}{9} b_3 D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2 + \frac{4}{9} b_3 D_2(f)(x_0, y_0) D_1(f)(x_0, y_0) \\
& + \frac{4}{9} b_3 D_2(f)(x_0, y_0)^2 f(x_0, y_0)
\end{aligned}$$

```
> cond3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))=coeff(Fi20,D[1,1](f)(x0,y0));
coef3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))-coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{3} = \frac{1}{9}b_2 + \frac{4}{9}b_3$$

$$coef3a := \frac{1}{3} - \frac{1}{9}b_2 - \frac{4}{9}b_3$$

```
> cond3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
coef3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))-coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{2}{3} = \frac{2}{9}b_2 + \frac{8}{9}b_3$$

$$coef3b := \frac{2}{3} - \frac{2}{9}b_2 - \frac{8}{9}b_3$$

```
> cond3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
coef3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)-coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{3} = \frac{1}{9}b_2 + \frac{4}{9}b_3$$

$$coef3c := \frac{1}{3} - \frac{1}{9}b_2 - \frac{4}{9}b_3$$

```
> cond3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
coef3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{3} = \frac{4}{9}b_3$$

$$coef3d := \frac{1}{3} - \frac{4}{9}b_3$$

```
> cond3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
coef3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{3} = \frac{4}{9}b_3$$

$$coef3e := \frac{1}{3} - \frac{4}{9}b_3$$

```
> b[1]:=0;b[2]:=1/2;b[3]:=1/2;
```

$$b_1 := 0$$

$$b_2 := \frac{1}{2}$$

$$b_3 := \frac{1}{2}$$

```
> coef3a;coef3b;coef3c;coef3d;coef3e;
```

$$\frac{1}{18}$$

$$\frac{1}{9}$$

$$\frac{1}{18}$$

$$\frac{1}{9}$$

$$\frac{1}{9}$$

Lo que es razonable, por lo que se obtiene el par encajado

0			
$\frac{1}{3}$		$\frac{1}{3}$	
$\frac{2}{3}$		0	$\frac{2}{3}$

ord 2		0	$\frac{1}{2}$	$\frac{1}{2}$
--------------	--	----------	---------------	---------------

ord 3		$\frac{1}{4}$	0	$\frac{3}{4}$
--------------	--	---------------	----------	---------------

estim		d_1	d_2	d_3
--------------	--	-------	-------	-------

[>
[>

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-08

Se considera el par encajado RK4(5) de FEHLBERG dado por

0						
$\frac{1}{4}$		$\frac{1}{4}$				
$\frac{3}{8}$		$\frac{3}{32}$	$\frac{9}{32}$			
$\frac{12}{13}$		$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$		
1		$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$	
$\frac{1}{2}$		$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$

orden 4		$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	0
---------	--	------------------	---	---------------------	---------------------	----------------	---

orden 5		$\frac{16}{135}$	0	$\frac{6656}{12825}$	$\frac{28561}{56430}$	$-\frac{9}{50}$	$\frac{2}{55}$
---------	--	------------------	---	----------------------	-----------------------	-----------------	----------------

est		$\frac{1}{360}$	0	$-\frac{128}{4275}$	$-\frac{2197}{75240}$	$\frac{1}{50}$	$\frac{2}{55}$
-----	--	-----------------	---	---------------------	-----------------------	----------------	----------------

a) Se comprobará mediante las gráficas de eficiencia correspondientes que se trata efectivamente de métodos de orden 4 y 5. Para ello se utilizará en $[0,0.8]$ el problema

$$y' = y^2, \quad y(0) = 1,$$

de solución exacta

$$y(x) = \frac{1}{1-x}$$

b) También se empleará el par para integrar el mismo problema con una tolerancia de $E = 0.0001$ en el intervalo $[0,0.8]$. Se calcularán las evaluaciones realizadas y (ya que se conoce la solución exacta) se comparará el error E admitido con el verdadero error cometido. Se estimará además el orden efectivo que muestra el par encajado.

a)

```
> restart;
> Digits:=30;
> interface(labeling=false);
RK4(5) -> RUNGE-KUTTA-FEHLBERG órdenes 4 y 5 clásico (e. aut.)
> a[2,1]:=1/4:a[3,1]:=3/32:a[3,2]:=9/32:a[4,1]:=1932/2197:a[4,2]:=-7200/2197:a[4,3]:=729
6/2197:a[5,1]:=439/216:a[5,2]:=-8:a[5,3]:=3680/513:a[5,4]:=-845/4104:a[6,1]:=-8/27:a[6
,2]:=2:a[6,3]:=-3544/2565:a[6,4]:=1859/4104:a[6,5]:=-11/40:
> b[1]:=25/216:b[3]:=1408/2565:b[4]:=2197/4104:b[5]:=-1/5:
> bt[1]:=16/135:bt[3]:=6656/12825:bt[4]:=28561/56430:bt[5]:=-9/50:bt[6]:=2/55:
> d[1]:=1/360:d[3]:=-128/4275:d[4]:=-2197/75240:d[5]:=1/50:d[6]:=2/55:
> a[2,1]:=evalf(a[2,1]):a[3,1]:=evalf(a[3,1]):a[3,2]:=evalf(a[3,2]):a[4,1]:=evalf(a[4,1]
):a[4,2]:=evalf(a[4,2]):a[4,3]:=evalf(a[4,3]):a[5,1]:=evalf(a[5,1]):a[5,2]:=evalf(a[5,
2]):a[5,3]:=evalf(a[5,3]):a[5,4]:=evalf(a[5,4]):a[6,1]:=evalf(a[6,1]):a[6,2]:=evalf(a[
6,2]):a[6,3]:=evalf(a[6,3]):a[6,4]:=evalf(a[6,4]):a[6,5]:=evalf(a[6,5]):
> b[1]:=evalf(b[1]):b[3]:=evalf(b[3]):b[4]:=evalf(b[4]):b[5]:=evalf(b[5]):
> bt[1]:=evalf(bt[1]):bt[3]:=evalf(bt[3]):bt[4]:=evalf(bt[4]):bt[5]:=evalf(bt[5]):bt[6]:
=evalf(bt[6]):
> d[1]:=evalf(d[1]):d[3]:=evalf(d[3]):d[4]:=evalf(d[4]):d[5]:=evalf(d[5]):d[6]:=evalf(d[
6]):
> RKF45etapas:=proc(f::procedure,y0::numeric,h::numeric)
global a,k1,k2,k3,k4,k5,k6:
k1:=f(y0):
k2:=f(y0+h*a[2,1]*k1):
k3:=f(y0+h*(a[3,1]*k1+a[3,2]*k2)):
k4:=f(y0+h*(a[4,1]*k1+a[4,2]*k2+a[4,3]*k3)):
k5:=f(y0+h*(a[5,1]*k1+a[5,2]*k2+a[5,3]*k3+a[5,4]*k4)):
k6:=f(y0+h*(a[6,1]*k1+a[6,2]*k2+a[6,3]*k3+a[6,4]*k4+a[6,5]*k5)):
end:
```

La ecuación (e. aut.) y la verdadera solución

```
> f:=y->y^2;yv:=x->1/(1-x);xini:=0;xend:=0.8;yini:=1;
```

$$f := y \rightarrow y^2$$

$$yv := x \rightarrow \frac{1}{1-x}$$

$$xini := 0$$

$$xend := 0.8$$

$$yini := 1$$

Las gráficas de eficiencia, para el orden 4

```
> listanumpasos:=[1024,2048]:listapaso:=[(xend-xini)/evalf(listanumpasos[1]),0.8/evalf(1
istanumpasos[2])]:listalogpaso:=[evalf(log[10](listapaso[1])),evalf(log[10](listapaso[
2]))]:listalogerror:=[0,0]:
> x0:=xini:y0:=yini:h:=listapaso[1]:for k from 1 to listanumpasos[1] do
RKF45etapas(f,y0,h);y1:=y0+h*(b[1]*k1+b[3]*k3+b[4]*k4+b[5]*k5);x0:=x0+h;y0:=y1;
od:listalogerror[1]:=log[10](abs(yv(x0)-y0)):
> x0:=xini:y0:=yini:h:=listapaso[2]:for k from 1 to listanumpasos[2] do
RKF45etapas(f,y0,h);y1:=y0+h*(b[1]*k1+b[3]*k3+b[4]*k4+b[5]*k5);x0:=x0+h;y0:=y1;
od:listalogerror[2]:=log[10](abs(yv(x0)-y0)):
> (listalogerror[2]-listalogerror[1])/(listalogpaso[2]-listalogpaso[1]);
```



```

; x0:=x0+h; y0:=y1; h:=min(h, xend-x0); fi;
od:
listalogerror[2]:=log[10](abs(yv(xend)-y0)); listalogeval[2]:=log[10](evalf(cont)):
> print(`el numero de evaluaciones para E=`, E, ` es `, cont); print(`y ademas
hay`, conr, `evaluaciones fallidas`); print(`el error verdadero cometido para E=`, E, ` es
`, abs(yv(xend)-y0));
                                     el numero de evaluaciones para E=, 0.00001, es , 84
                                     y ademas hay, 72, evaluaciones fallidas
                                     el error verdadero cometido para E=, 0.00001, es , 0.00006463593781018747442532480
> (listalogerror[2]-listalogerror[1])/(listalogeval[2]-listalogeval[1]);
                                     -3.87517738583822147471419789138

```

y el par se muestra con un orden efectivo cercano a 4 ; no hemos empleado factores de corrección del error estimado, y tampoco hemos puesto en discusión el paso recomendado (que son cosas que en la práctica se hacen)

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-09 (del EXAMEN EXTRAORDINARIO de 14JUL04)

Para un problema escalar se diseñan tres métodos numéricos de integración de ecuaciones diferenciales que designaremos con los nombres de M1 , M2 y M3 .

Para determinar cuál es el de orden más alto, se realiza la integración de un problema test entre 0 y 1 . Los dos primeros métodos son de paso fijo. Para ellos se emplean los pasos

$$\frac{1}{2^4} = 0.0625 , \frac{1}{2^5} = 0.0313 , \frac{1}{2^6} = 0.0156 , \frac{1}{2^7} = 0.0078 , \frac{1}{2^8} = 0.0039$$

Los números de pasos son, en todos los casos,

16 , 32 , 64 , 128 , 256

Los números de evaluaciones son para el M1

32 , 64 , 128 , 256 , 512

y para el M2

48 , 96 , 192 , 384 , 768

El método M3 es un par encajado. Se prueba con tolerancias

0.001 , 0.0005 , 0.0001 , 0.00005

y para ellas se han necesitado en el cálculo los siguientes números de evaluaciones

128 , 220 , 336 , 448

Para cada método se calcula el \log_{10} del error cometido en la aproximación de la solución en $x=1$, error medido en módulo. Los logaritmos de los errores que se obtienen son, para el M1

- 3.849 , - 4.701 , - 5.381 , - 6.014 , - 6.630

para el M2

- 3.278 , - 4.861 , - 6.814 , - 7.743 , - 8.659

y para el M3

- 7.561 , - 8.014 , - 8.643 , - 8.959

Con estos datos, se debe construir la 'gráfica de eficiencia' que mezcla los resultados de M1 , M2 y M3 exhibiendo 'número de evaluaciones versus \log_{10} del error'. Además de dirá qué método es más interesante en cuanto al orden efectivo que se obtiene.

[> restart:

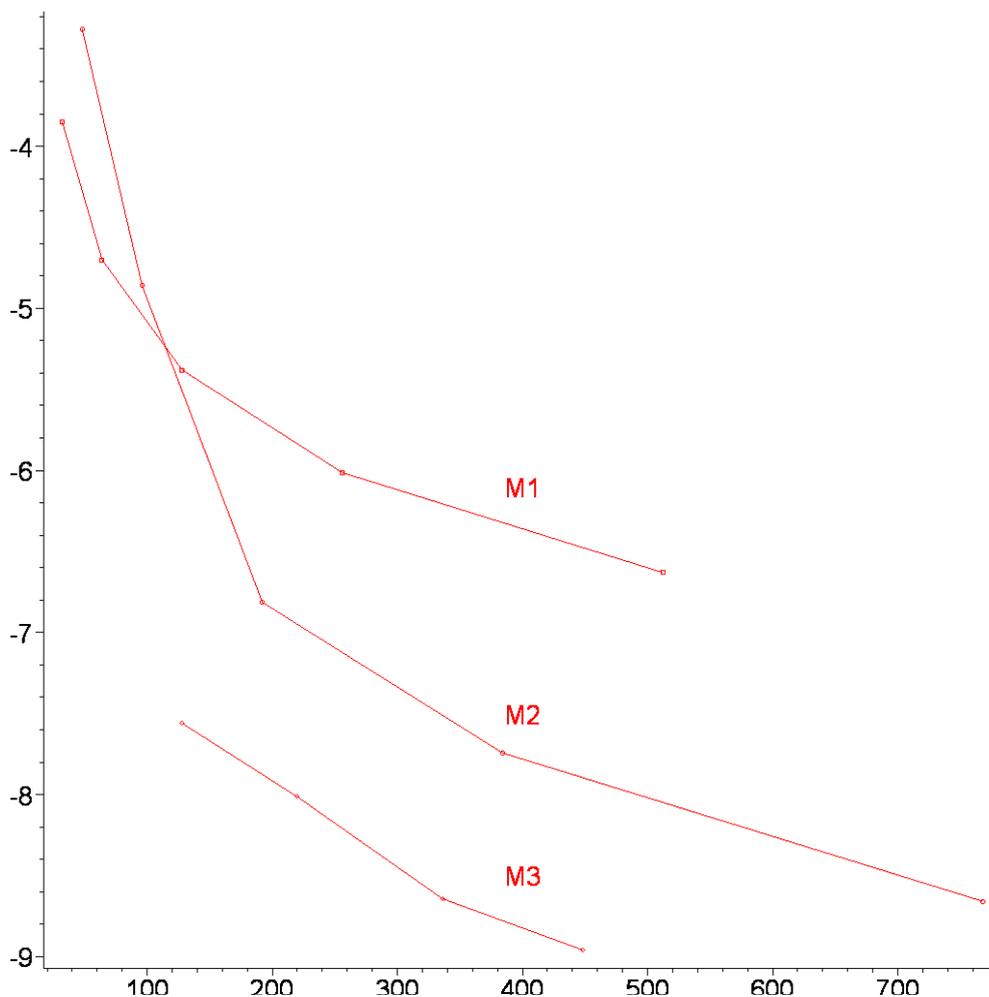
```
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listaeval:=[32 , 64 , 128 , 256 , 512]:
[ > listalog1eval:=[0,0,0,0,0]:
[ > for i from 1 to 5 do listalogeval[i]:=log[10](listaeval[i]): od:
[ > listalogerro:=[ -3.849, -4.701, -5.381, -6.014, -6.630]:
[ > lista2eval:=[48 , 96 , 192 , 384 , 768]:
[ > listalog2eval:=[0,0,0,0,0]:
[ > for i from 1 to 5 do lista2logeval[i]:=log[10](lista2eval[i]): od:
[ > lista2logerro:=[-3.278, -4.861, -6.814, -7.743, -8.659]:
[ > lista3eval:=[128 , 220 , 336 , 448]:
[ > listalog3eval:=[0,0,0,0]:
[ > for i from 1 to 4 do lista3logeval[i]:=log[10](lista3eval[i]): od:
[ > lista3logerro:=[-7.561, -8.014, -8.643, -8.959]:
```

[Gráficas comparadas 'número de evaluaciones versus log[10] del error'

```
[ > listagra1:=[seq([listaeval[i],listalogerro[i]],i=1..5)]:
[ > gra11:=plot(listagra1,style=LINE):
[ > gra12:=plot(listagra1,style=POINT,symbol=BOX):
[ > gra13:=textplot([400., -6.1,`M1`]):
[ > listagra2:=[seq([lista2eval[i],lista2logerro[i]],i=1..5)]:
[ > gra21:=plot(listagra2,style=LINE):
[ > gra22:=plot(listagra2,style=POINT,symbol=CIRCLE):
[ > gra23:=textplot([400., -7.5,`M2`]):
[ > listagra3:=[seq([lista3eval[i],lista3logerro[i]],i=1..4)]:
[ > gra31:=plot(listagra3,style=LINE):
[ > gra32:=plot(listagra3,style=POINT,symbol=DIAMOND):
[ > gra33:=textplot([400, -8.5,`M3`]):
[ > display(gra11,gra12,gra13,gra21,gra22,gra23,gra31,gra32,gra33);
```



Comparación 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya sabemos de otros problemas:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los mas representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del numero de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima

a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

$$\begin{aligned}\log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2)\end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

Orden efectivo de los métodos, empleando la pendiente del último segmento.

```
> pendm1:=evalf((lista1logerro[5]-lista1logerro[4])/(lista1logeval[5]-lista1logeval[4]))
;
pendm1 := -2.0463077064506152057
> pendm2:=evalf((lista2logerro[5]-lista2logerro[4])/(lista2logeval[5]-lista2logeval[4]))
;
pendm2 := -3.0428861349168239108
> pendm3:=evalf((lista3logerro[4]-lista3logerro[3])/(lista3logeval[4]-lista3logeval[3]))
;
pendm3 := -2.5292395983697455602
```

Método más interesante en cuanto al orden efectivo

El método M2 presenta el mejor orden efectivo de los tres métodos.

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-10 (del EXAMEN EXTRAORDINARIO de 09JUL03)

Para comparar dos métodos de tipo RUNGE-KUTTA, que denotaremos por m_1 y m_2 , el primero de paso fijo y el segundo de paso variable, se integra un problema test y se comparan los resultados numéricos obtenidos con los dos métodos.

Para el primero, m_1 , se emplean los pasos

$$\frac{1}{2} = 0.5, \frac{1}{2^2} = 0.25, \frac{1}{2^3} = 0.125, \frac{1}{2^4} = 0.0625, \frac{1}{2^5} = 0.0313, \frac{1}{2^6} = 0.0156$$

lo que significa que los números de evaluaciones de función son

390, 790, 1580, 3160, 6330, 12670

y se obtienen los errores

$0.107100612 \cdot 10^{-1}$, $0.324754737 \cdot 10^{-2}$, $0.173265513 \cdot 10^{-3}$, $0.921034842 \cdot 10^{-4}$,
 $0.405803845 \cdot 10^{-5}$, $0.164174164 \cdot 10^{-6}$

Para m_2 se emplean las tolerancias

10^{-3} , 10^{-4} , 10^{-5} , 10^{-6} , 10^{-7} , 10^{-8} , 10^{-9} , 10^{-10} , 10^{-11} ;

para ellas los números de evaluaciones de función son

180, 308, 524, 816, 1332, 2264, 3896, 6824, 11932

y se obtienen los errores

$0.120075671 \cdot 10^{-1}$, $0.224069086 \cdot 10^{-2}$, $0.293453270 \cdot 10^{-3}$, $0.325178066 \cdot 10^{-4}$,
 $0.397930221 \cdot 10^{-5}$, $0.436613469 \cdot 10^{-6}$, $0.445225052 \cdot 10^{-7}$, $0.455563685 \cdot 10^{-8}$,
 $0.473519591 \cdot 10^{-9}$

Se puede deducir de los datos anteriores con qué orden están funcionando los métodos ?

Cuál es el que tiene mejor comportamiento ?

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > lista1eval:=[0,0,0,0,0,0]:
[ > lista1erro:=[0,0,0,0,0,0]:
[ > lista1logeval:=[0,0,0,0,0,0]:
[ > lista1logerro:=[0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0,0]:
```

```
[ > lista2erro:=[0,0,0,0,0,0,0,0,0]:
[ > lista2logeval:=[0,0,0,0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0,0,0,0]:
[ Datos de m1
[ > listaleval:=[390 , 790 , 1580 , 3160 , 6330 , 12670]:
[ > listalerror:=[ 0.107100612 * 10^(-1) , 0.324754737 * 10^(-2) , 0.173265513 * 10^(-3) ,
[ 0.921034842 * 10^(-4) , 0.405803845 * 10^(-5) , 0.164174164 * 10^(-6)]:
[ > for i from 1 to 6 do listallogeval[i]:=log[10](listaleval[i]): od:
[ > for i from 1 to 6 do listallogerro[i]:=log[10](listalerror[i]): od:
[ > evalf(listallogeval,4);evalf(listallogerro,4);
[
[ 2.591, 2.897, 3.198, 3.499, 3.801, 4.102]
[
[ -1.970, -2.488, -3.761, -4.036, -5.392, -6.785]
```

```
[ Datos de m2
[ > lista2eval:=[180 , 308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:
[ > lista2erro:=[0.120075671 * 10^(-1) , 0.224069086 * 10^(-2) , 0.293453270 * 10^(-3) ,
[ 0.325178066 * 10^(-4) , 0.397930221 * 10^(-5) , 0.436613469 * 10^(-6) , 0.445225052 *
[ 10^(-7) , 0.455563685 * 10^(-8) , 0.473519591 * 10^(-9)]:
[ > for i from 1 to 9 do lista2logeval[i]:=log[10](lista2eval[i]): od:
[ > for i from 1 to 9 do lista2logerro[i]:=log[10](lista2erro[i]): od:
[ > evalf(lista2logeval,4);evalf(lista2logerro,4);
[
[ 2.255, 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]
[
[ -1.921, -2.650, -3.532, -4.488, -5.400, -6.360, -7.351, -8.341, -9.325]
```

[Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya sabemos de otros problemas:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los más representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del número de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

$$\log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) = \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) =$$

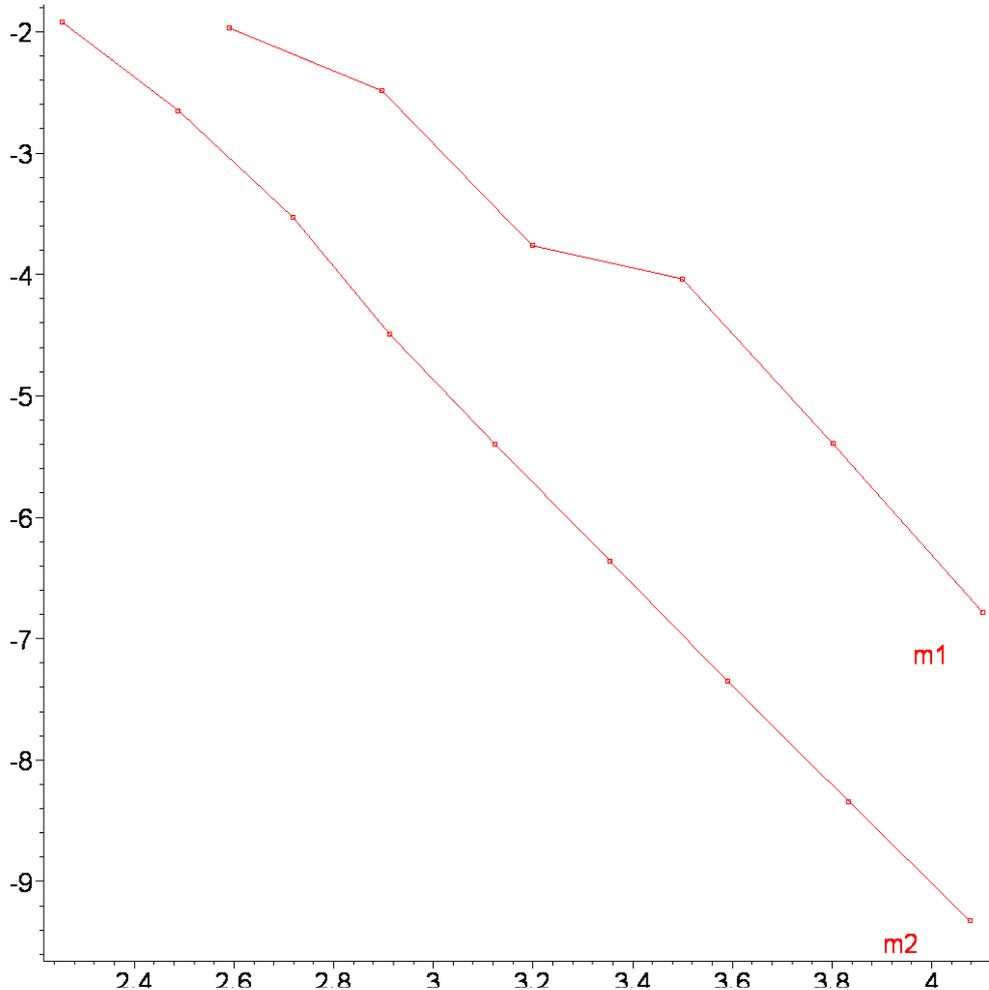
$$= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2)$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

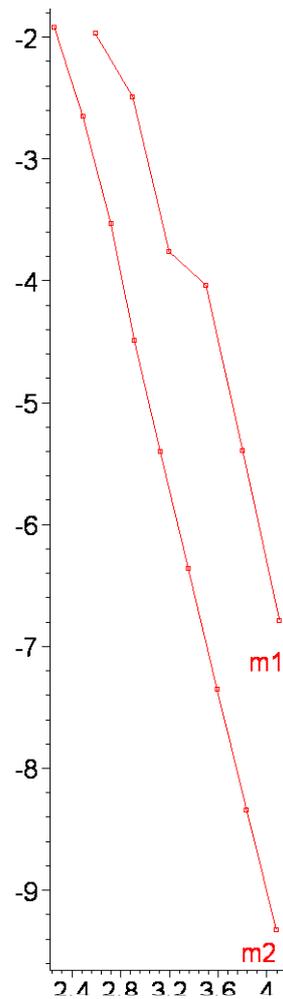
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
> lista1grafb:=[seq([lista1logeval[i],lista1logerro[i]],i=1..6)]:
> gra1b1:=plot(lista1grafb,style=LINE):
> gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):
> #gra1b3:=textplot([3.999, -11.78, `m1`]):
> gra1b3:=textplot([3.999, -7.12, `m1`]):
> lista2grafb:=[seq([lista2logeval[i],lista2logerro[i]],i=1..9)]:
> gra2b1:=plot(lista2grafb,style=LINE):
> gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):
> gra2b3:=textplot([3.938, -9.505, `m2`]):
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3);
```



```
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,scaling='CONSTRAINED');
```



a) Orden efectivo de los métodos

```
> pendm1:=evalf((lista1logerro[6]-lista1logerro[5])/(lista1logeval[6]-lista1logeval[5]))
;
pendm1 := -4.6222182608419113758
> pendm2:=evalf((lista2logerro[9]-lista2logerro[8])/(lista2logeval[9]-lista2logeval[8]))
;
pendm2 := -4.0515680608587727185
```

El orden efectivo de m1 es de 4.6 , el de m2 es de 4 .

b) Método con mejor comportamiento

El método m1 parece tener algo mejor comportamiento de cara al error cuando el paso se hace adecuadamente pequeño, dentro de un orden parecido para ambos métodos.

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-11

Para comprobar la eficiencia de un método de tipo RUNGE-KUTTA, que denotaremos por m_1 , se integra el problema test conocido como problema de BESSEL

$$y'' + 100y = -\frac{y}{4t^2}, \quad t \in [0.1, 10]$$

y se comparan los resultados numéricos con otros dos métodos, uno de ellos de paso variable, que denotaremos por m_2 , y el otro de paso fijo, que denotaremos por m_3 . Para m_1 se emplearon los pasos

$$\frac{1}{2} = 0.5, \quad \frac{1}{2^2} = 0.25, \quad \frac{1}{2^3} = 0.125, \quad \frac{1}{2^4} = 0.0625, \quad \frac{1}{2^5} = 0.0313, \quad \frac{1}{2^6} = 0.0156$$

lo que significa que los números de evaluaciones de función fueron

390, 790, 1580, 3160, 6330, 12670

y se obtuvieron los errores

$0.107100612 \cdot 10^{-1}$, $0.324754737 \cdot 10^{-4}$, $0.173265513 \cdot 10^{-6}$, $0.921034842 \cdot 10^{-9}$,
 $0.405803845 \cdot 10^{-11}$, $0.164174164 \cdot 10^{-13}$

Para m_2 se emplearon las tolerancias

$$10^{(-3)}, \quad 10^{(-4)}, \quad 10^{(-5)}, \quad 10^{(-6)}, \quad 10^{(-7)}, \quad 10^{(-8)}, \quad 10^{(-9)}, \quad 10^{(-10)}, \quad 10^{(-11)};$$

para ellas los números de evaluaciones de función fueron

180, 308, 524, 816, 1332, 2264, 3896, 6824, 11932

y se obtuvieron los errores

$0.120075671 \cdot 10^{-1}$, $0.224069086 \cdot 10^{-2}$, $0.293453270 \cdot 10^{-3}$, $0.325178066 \cdot 10^{-4}$,
 $0.397930221 \cdot 10^{-5}$, $0.436613469 \cdot 10^{-6}$, $0.445225052 \cdot 10^{-7}$, $0.455563685 \cdot 10^{-8}$,
 $0.473519591 \cdot 10^{-9}$

Finalmente, para m_3 se emplearon los pasos

$$\frac{1}{2} = 0.5, \quad \frac{1}{2^2} = 0.25, \quad \frac{1}{2^3} = 0.125, \quad \frac{1}{2^4} = 0.0625, \quad \frac{1}{2^5} = 0.0313, \quad \frac{1}{2^6} = 0.0156, \quad \frac{1}{2^7} = 0.0078,$$

con números de evaluaciones

156, 316, 632, 1264, 2532, 5068, 10136

y se obtuvieron los errores

0.188422582 10⁽⁻¹⁾ , 0.501136903 10⁽⁻²⁾ , 0.430771249 10⁽⁻³⁾ , 0.299114127 10⁽⁻⁴⁾ ,
0.185532367 10⁽⁻⁵⁾ , 0.114097541 10⁽⁻⁶⁾ , 0.703332178 10⁽⁻⁸⁾

Se puede deducir de los datos anteriores con qué orden están funcionando los métodos ?

Cuál es el que tiene mejor comportamiento ?

```
[ > restart:  
[ > interface(labeling=false):  
[ > Digits:=20:  
[ > with(linalg):with(plots):  
Warning, the protected names norm and trace have been redefined and unprotected  
  
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listaleval:=[0,0,0,0,0,0]:  
[ > listalerro:=[0,0,0,0,0,0]:  
[ > lista1logeval:=[0,0,0,0,0,0]:  
[ > lista1logerro:=[0,0,0,0,0,0]:  
[ > lista2eval:=[0,0,0,0,0,0,0,0]:  
[ > lista2erro:=[0,0,0,0,0,0,0,0]:  
[ > lista2logeval:=[0,0,0,0,0,0,0,0]:  
[ > lista2logerro:=[0,0,0,0,0,0,0,0]:  
[ > lista3eval:=[0,0,0,0,0,0,0]:  
[ > lista3erro:=[0,0,0,0,0,0,0]:  
[ > lista3logeval:=[0,0,0,0,0,0,0]:  
[ > lista3logerro:=[0,0,0,0,0,0,0]:
```

[Datos de m1

```
[ > listaleval:=[390 , 790 , 1580 , 3160 , 6330 , 12670]:  
[ > listalerro:=[ 0.107100612 * 10(-1) , 0.324754737 * 10(-4) , 0.173265513 * 10(-6) ,  
[ 0.921034842 * 10(-9) , 0.405803845 * 10(-11) , 0.164174164 * 10(-13)]:  
[ > for i from 1 to 6 do lista1logeval[i]:=log[10](listaleval[i]): od:  
[ > for i from 1 to 6 do lista1logerro[i]:=log[10](listalerro[i]): od:  
[ > evalf(lista1logeval,4);evalf(lista1logerro,4);  
[ 2.591, 2.897, 3.198, 3.499, 3.801, 4.102]  
[ -1.970, -4.488, -6.761, -9.036, -11.39, -13.78]
```

[Datos de m2

```
[ > lista2eval:=[180 , 308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:  
[ > lista2erro:=[0.120075671 * 10(-1) , 0.224069086 * 10(-2) , 0.293453270 * 10(-3) ,  
[ 0.325178066 * 10(-4) , 0.397930221 * 10(-5) , 0.436613469 * 10(-6) , 0.445225052 *  
[ 10(-7) , 0.455563685 * 10(-8) , 0.473519591 * 10(-9)]:  
[ > for i from 1 to 9 do lista2logeval[i]:=log[10](lista2eval[i]): od:  
[ > for i from 1 to 9 do lista2logerro[i]:=log[10](lista2erro[i]): od:  
[ > evalf(lista2logeval,4);evalf(lista2logerro,4);  
[ 2.255, 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]  
[ -1.921, -2.650, -3.532, -4.488, -5.400, -6.360, -7.351, -8.341, -9.325]
```

[Datos de m3

```
[ > lista3eval:=[156 , 316 , 632 , 1264 , 2532 , 5068 , 10136]:  
[ > lista3erro:=[0.188422582 * 10(-1) , 0.501136903 * 10(-2) , 0.430771249 * 10(-3) ,  
[ 0.299114127 * 10(-4) , 0.185532367 * 10(-5) , 0.114097541 * 10(-6) , 0.703332178  
[ * 10(-8)]:  
[ > for i from 1 to 7 do lista3logeval[i]:=log[10](lista3eval[i]): od:  
[ > for i from 1 to 7 do lista3logerro[i]:=log[10](lista3erro[i]): od:  
[ > evalf(lista3logeval,4);evalf(lista3logerro,4);  
[ 2.193, 2.499, 2.800, 3.101, 3.403, 3.704, 4.005]  
[ -1.725, -2.300, -3.366, -4.524, -5.732, -6.943, -8.153]
```

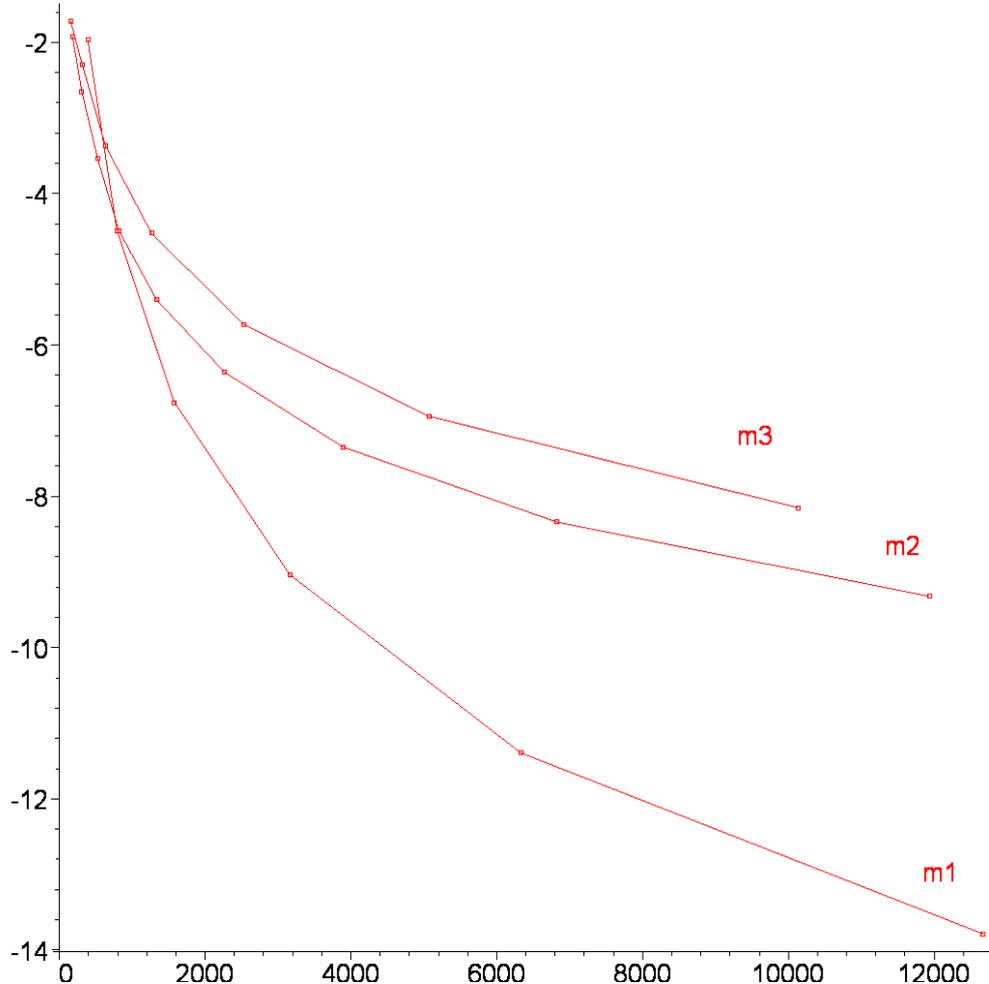
[Gráficas comparadas 'evaluaciones versus log[10] del error'

```
[ > lista1grafa:=seq([listaleval[i],lista1logerro[i]],i=1..6):  
[ > gral1:=plot(lista1grafa,style=LINE):
```

```

[ > gra1a2:=plot(lista1grafa,style=POINT,symbol=BOX):
[ > gra1a3:=textplot([1.209e+004, -12.95,`m1`]):
[ > lista2grafa:=[seq([lista2eval[i],lista2logerro[i]],i=1..9)]:
[ > gra2a1:=plot(lista2grafa,style=LINE):
[ > gra2a2:=plot(lista2grafa,style=POINT,symbol=BOX):
[ > gra2a3:=textplot([1.158e+004, -8.642,`m2`]):
[ > lista3grafa:=[seq([lista3eval[i],lista3logerro[i]],i=1..7)]:
[ > gra3a1:=plot(lista3grafa,style=LINE):
[ > gra3a2:=plot(lista3grafa,style=POINT,symbol=BOX):
[ > gra3a3:=textplot([9552, -7.183,`m3`]):
[ > display(gra1a1,gra1a2,gra1a3,gra2a1,gra2a2,gra2a3,gra3a1,gra3a2,gra3a3);

```



[Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya dijimos en el problema precedente:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los más representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del número de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

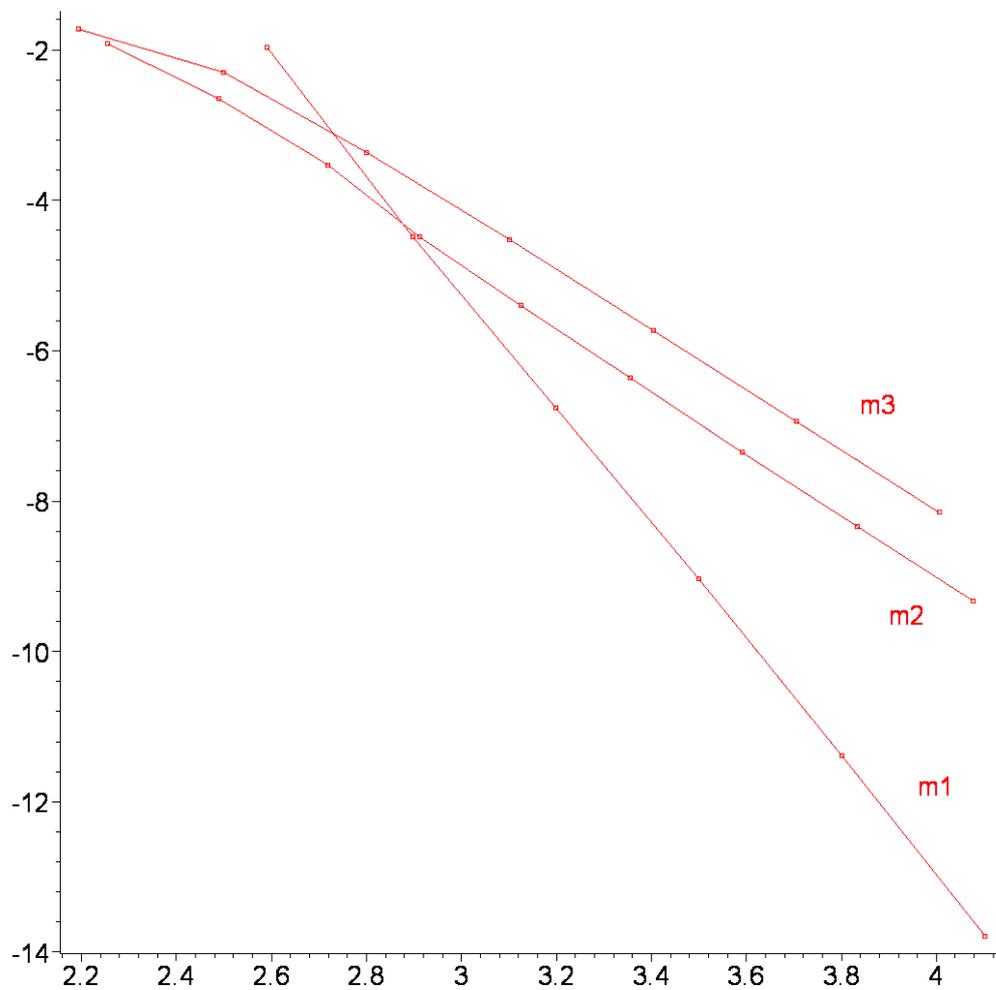
$$\begin{aligned} \log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2) \end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

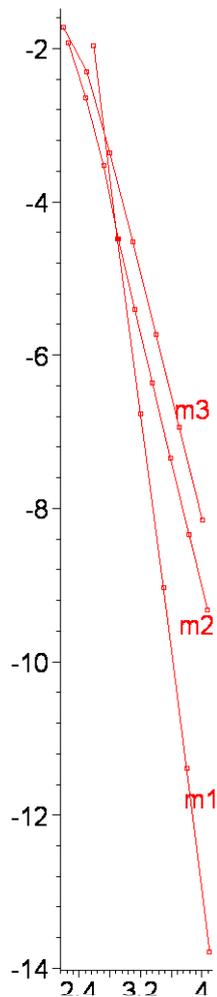
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
[ > lista1grafb:=[seq([lista1logeval[i],lista1logerro[i]],i=1..6)]:
[ > gra1b1:=plot(lista1grafb,style=LINE):
[ > gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):
[ > gra1b3:=textplot([3.999, -11.78, `m1`]):
[ > lista2grafb:=[seq([lista2logeval[i],lista2logerro[i]],i=1..9)]:
[ > gra2b1:=plot(lista2grafb,style=LINE):
[ > gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):
[ > gra2b3:=textplot([3.938, -9.505, `m2`]):
[ > lista3grafb:=[seq([lista3logeval[i],lista3logerro[i]],i=1..7)]:
[ > gra3b1:=plot(lista3grafb,style=LINE):
[ > gra3b2:=plot(lista3grafb,style=POINT,symbol=BOX):
[ > gra3b3:=textplot([3.877, -6.701, `m3`]):
[ > display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,gra3b1,gra3b2,gra3b3);
```



```
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,gra3b1,gra3b2,gra3b3,scaling='CONSTRAINED');
```



a) Orden efectivo de los métodos

```

> pendm1:=evalf((lista1logerro[6]-lista1logerro[5])/(lista1logeval[6]-lista1logeval[5]))
;
pendm1 := -7.9403665875614143211
> pendm2:=evalf((lista2logerro[9]-lista2logerro[8])/(lista2logeval[9]-lista2logeval[8]))
;
pendm2 := -4.0515680608587727185
> pendm3:=evalf((lista3logerro[7]-lista3logerro[6])/(lista3logeval[7]-lista3logeval[6]))
;
pendm3 := -4.0199176659068501718

```

El orden efectivo de m1 es de 8. Los de m2 y m3 son de 4.

b) Método con mejor comportamiento

El método m1 parece tener mejor comportamiento de cara al error cuando el paso se hace adecuadamente pequeño. Además está lo que hemos dicho del orden de los métodos.

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-12 (del EXAMEN EXTRAORDINARIO de 01JUL05)

Para comparar dos métodos de paso variable, que denotaremos por m_1 y m_2 , se integra un problema test y se comparan los resultados numéricos obtenidos con los dos métodos.

Para el primero, m_1 , se emplean las tolerancias

$$10^{(-3)}, 10^{(-6)}, 10^{(-9)}, 10^{(-12)} ;$$

para ellas los números de evaluaciones de función son

192, 882, 3808, 12426

y se obtienen los errores

$$0.102646532 \cdot 10^{(-1)}, 0.028463542 \cdot 10^{(-5)}, 0.185225052 \cdot 10^{(-8)}, 0.253251780 \cdot 10^{(-9)},$$

Para m_2 se emplean las tolerancias

$$10^{(-3)}, 10^{(-4)}, 10^{(-5)}, 10^{(-6)}, 10^{(-7)}, 10^{(-8)}, 10^{(-9)}, 10^{(-10)}, 10^{(-11)} ;$$

para ellas los números de evaluaciones de función son

180, 308, 524, 816, 1332, 2264, 3896, 6824, 11932

y se obtienen los errores

$$0.120075671 \cdot 10^{(-1)}, 0.224069086 \cdot 10^{(-2)}, 0.293453270 \cdot 10^{(-3)}, 0.325178066 \cdot 10^{(-4)},$$

$$0.397930221 \cdot 10^{(-5)}, 0.436613469 \cdot 10^{(-6)}, 0.445225052 \cdot 10^{(-7)}, 0.455563685 \cdot 10^{(-8)},$$

$$0.473519591 \cdot 10^{(-9)}$$

Se puede deducir de los datos anteriores con qué orden están funcionando los métodos ?

Cuál es el que tiene mejor comportamiento ?

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listaeval:=[0,0,0,0]:
[ > listaerro:=[0,0,0,0]:
[ > lista1logeval:=[0,0,0,0]:
[ > lista1logerro:=[0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0,0]:
[ > lista2erro:=[0,0,0,0,0,0,0,0,0]:
[ > lista2logeval:=[0,0,0,0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0,0,0,0]:
```

[Datos de m1

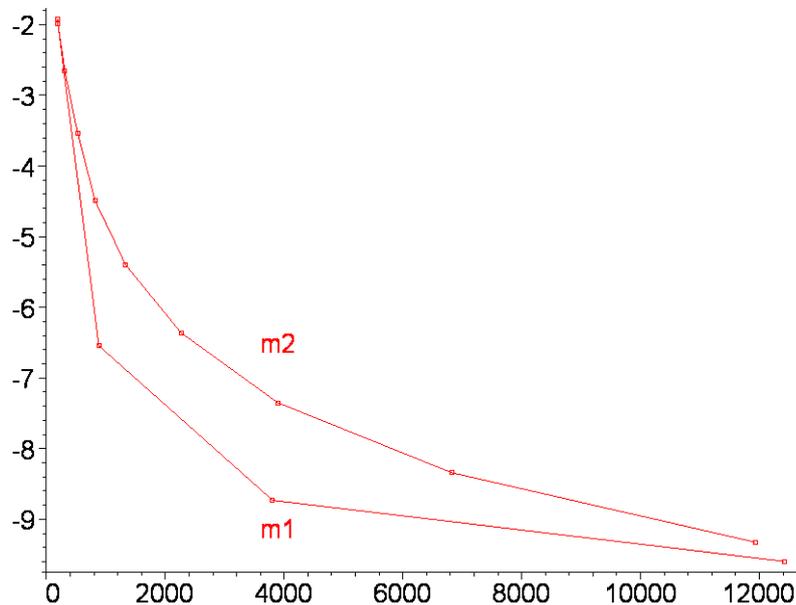
```
[ > listaeval:=[192 , 882 , 3808 , 12426]:  
[ > listaerro:=[ 0.102646532 * 10^(-1) , 0.028463542 * 10^(-5) , 0.185225052 * 10^(-8) ,  
[ 0.253251780 * 10^(-9)]:  
[ > for i from 1 to 4 do listaallogeval[i]:=log[10](listaeval[i]): od:  
[ > for i from 1 to 4 do listaallogerro[i]:=log[10](listaerro[i]): od:  
[ > evalf(listaallogeval,4);evalf(listaallogerro,4);  
[ 2.283, 2.945, 3.580, 4.094]  
[ -1.989, -6.546, -8.732, -9.596]
```

[Datos de m2

```
[ > lista2eval:=[180 , 308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:  
[ > lista2erro:=[0.120075671 * 10^(-1) , 0.224069086 * 10^(-2) , 0.293453270 * 10^(-3) ,  
[ 0.325178066 * 10^(-4) , 0.397930221 * 10^(-5) , 0.436613469 * 10^(-6) , 0.445225052 *  
[ 10^(-7) , 0.455563685 * 10^(-8) , 0.473519591 * 10^(-9)]:  
[ > for i from 1 to 9 do lista2logeval[i]:=log[10](lista2eval[i]): od:  
[ > for i from 1 to 9 do lista2logerro[i]:=log[10](lista2erro[i]): od:  
[ > evalf(lista2logeval,4);evalf(lista2logerro,4);  
[ 2.255, 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]  
[ -1.921, -2.650, -3.532, -4.488, -5.400, -6.360, -7.351, -8.341, -9.325]
```

[Gráficas comparadas 'número de evaluaciones versus log[10] del error'

```
[ > lista1grafb:=[seq([listaeval[i],listaallogerro[i]],i=1..4]):  
[ > gra1b1:=plot(lista1grafb,style=LINE):  
[ > gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):  
[ > gra1b3:=textplot([3908, -9.12, `m1`]):  
[ > lista2grafb:=[seq([lista2eval[i],lista2logerro[i]],i=1..9]):  
[ > gra2b1:=plot(lista2grafb,style=LINE):  
[ > gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):  
[ > gra2b3:=textplot([3908, -6.50, `m2`]):  
[ > display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3);
```



```
[ >  
[ >
```

[Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya sabemos de otros problemas:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los mas representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el $\log[10]$ del numero de evaluaciones y en el de ordenadas el $\log[10]$ del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

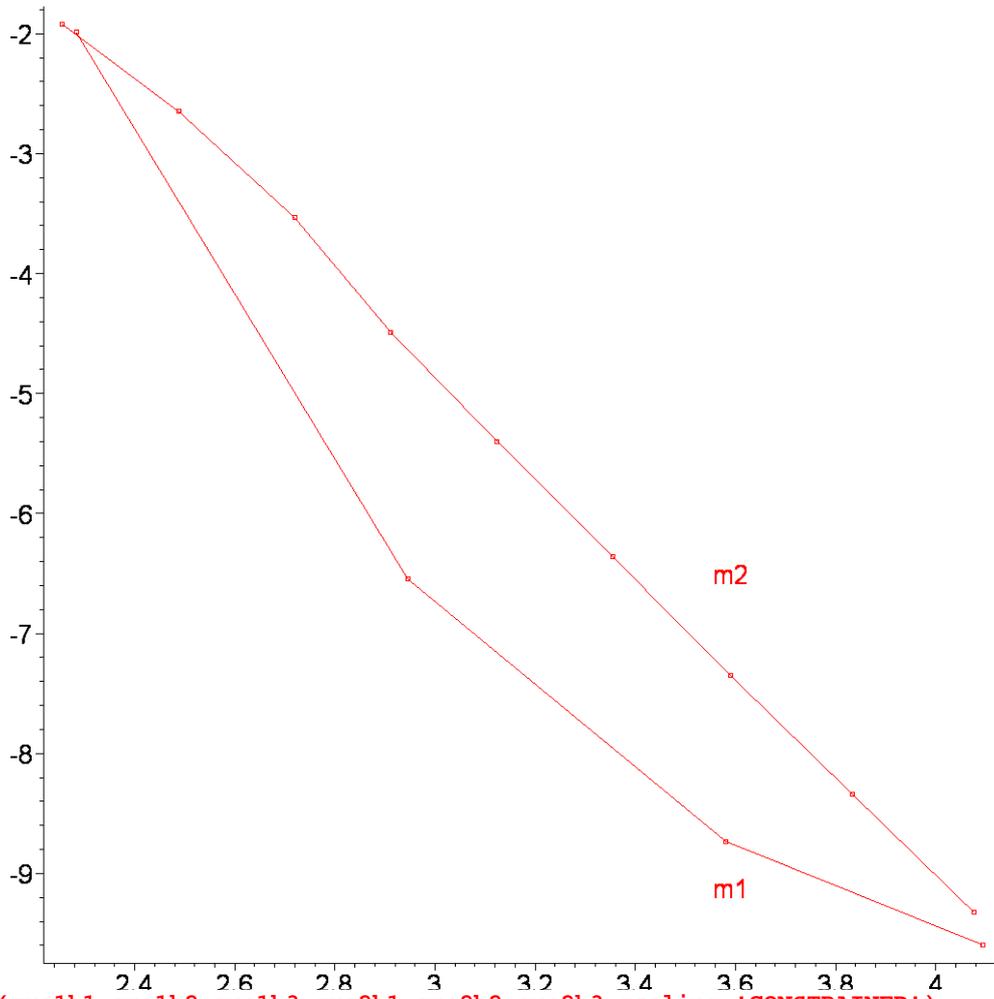
$$\begin{aligned} \log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2) \end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

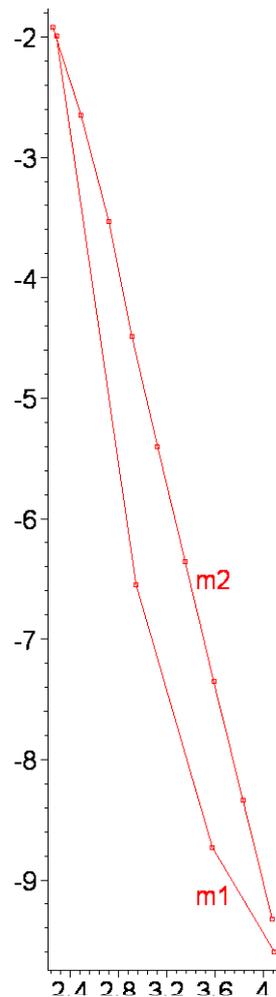
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas ' $\log[10]$ del paso versus $\log[10]$ del error', que carecen ahora de sentido, sino las gráficas ' $\log[10]$ del número de evaluaciones versus $\log[10]$ del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
> lista1grafb:=seq([lista1logeval[i],lista1logerro[i]],i=1..4):
[ > gra1b1:=plot(lista1grafb,style=LINE):
[ > gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):
[ > gra1b3:=textplot([log[10](3908), -9.12,`m1`]):
[ > lista2grafb:=seq([lista2logeval[i],lista2logerro[i]],i=1..9):
[ > gra2b1:=plot(lista2grafb,style=LINE):
[ > gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):
[ > gra2b3:=textplot([log[10](3908), -6.50,`m2`]):
[ > display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3);
```



```
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,scaling='CONSTRAINED');
```



a) Orden efectivo de los métodos

```
> pendm1:=evalf((lista1logerro[4]-lista1logerro[3])/(lista1logeval[4]-lista1logeval[3]))
;
```

pendm1 := -1.6824169151647295462

```
> pendm2:=evalf((lista2logerro[9]-lista2logerro[8])/(lista2logeval[9]-lista2logeval[8]))
;
```

pendm2 := -4.0515680608587727185

El orden efectivo de m1 es de 1.7 , el de m2 es de 4.1

b) Método con mejor comportamiento

El método m1 parece tener algo mejor comportamiento al principio, pero, cuando se mira el orden (o sea, lo que sucede cuando el paso se convierte en pequeño) resulta claramente mejor el método m2

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-13 (del EXAMEN EXTRAORDINARIO de 04JUL06)

Para comprobar la eficiencia de un método de tipo RUNGE-KUTTA y de paso variable, que denotaremos por $m1$, se integra un problema test con este método y con otro de paso fijo, que denotaremos por $m2$.

Para $m1$, el de paso variable, se emplearon las tolerancias

$$10^{(-3)}, 10^{(-4)}, 10^{(-5)}, 10^{(-6)}, 10^{(-7)}, 10^{(-8)}, 10^{(-9)}, 10^{(-10)}, 10^{(-11)} ;$$

para ellas los números de evaluaciones de función fueron

$$180, 308, 524, 816, 1332, 2264, 3896, 6824, 11932$$

y se obtuvieron los errores

$$0.120075671 \cdot 10^{(-1)}, 0.224069086 \cdot 10^{(-2)}, 0.293453270 \cdot 10^{(-3)}, 0.325178066 \cdot 10^{(-4)}, \\ 0.397930221 \cdot 10^{(-5)}, 0.436613469 \cdot 10^{(-6)}, 0.445225052 \cdot 10^{(-7)}, 0.455563685 \cdot 10^{(-8)}, \\ 0.473519591 \cdot 10^{(-9)}$$

Para $m2$, el de paso fijo, se emplearon los pasos

$$\frac{1}{2} = 0.5, \frac{1}{2^2} = 0.25, \frac{1}{2^3} = 0.125, \frac{1}{2^4} = 0.0625, \frac{1}{2^5} = 0.0313, \frac{1}{2^6} = 0.0156$$

lo que significa que los números de evaluaciones de función fueron

$$390, 790, 1580, 3160, 6330, 12670$$

y se obtuvieron los errores

$$0.107100612 \cdot 10^{(-1)}, 0.324754737 \cdot 10^{(-4)}, 0.173265513 \cdot 10^{(-6)}, 0.921034842 \cdot 10^{(-9)}, \\ 0.405803845 \cdot 10^{(-11)}, 0.164174164 \cdot 10^{(-13)}$$

Se puede deducir de los datos anteriores con qué orden están funcionando los métodos ?

Cuál es el que tiene mejor comportamiento ?

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

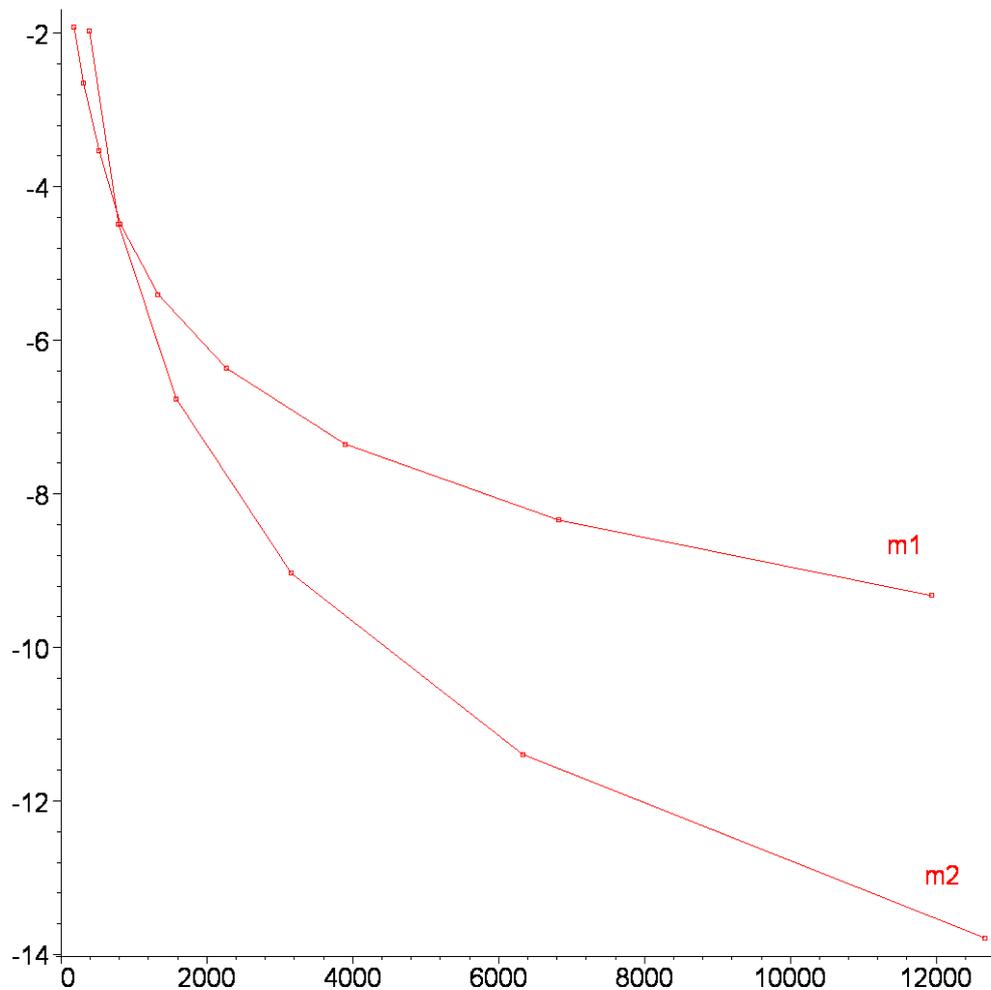
[Variables lista

```
[ > lista1eval:=[0,0,0,0,0,0,0,0,0,0]:
[ > lista1erro:=[0,0,0,0,0,0,0,0,0,0]:
[ > lista1logeval:=[0,0,0,0,0,0,0,0,0,0]:
[ > lista1logerro:=[0,0,0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0,0,0]:
```

```

[ > lista2erro:=[0,0,0,0,0,0]:
[ > lista2logeval:=[0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0]:
[ Datos de m1
[ > listaleval:=[180 , 308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:
[ > lista1erro:=[0.120075671 * 10^(-1) , 0.224069086 * 10^(-2) , 0.293453270 * 10^(-3) ,
0.325178066 * 10^(-4) , 0.397930221 * 10^(-5) , 0.436613469 * 10^(-6) , 0.445225052 *
10^(-7) , 0.455563685 * 10^(-8) , 0.473519591 * 10^(-9)]:
[ > for i from 1 to 9 do lista1logeval[i]:=log[10](listaleval[i]): od:
[ > for i from 1 to 9 do lista1logerro[i]:=log[10](lista1erro[i]): od:
[ > evalf(lista1logeval,4);evalf(lista1logerro,4);
[
[2.255, 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]
[-1.921, -2.650, -3.532, -4.488, -5.400, -6.360, -7.351, -8.341, -9.325]
[ Datos de m2
[ > lista2eval:=[390 , 790 , 1580 , 3160 , 6330 , 12670]:
[ > lista2erro:=[ 0.107100612 * 10^(-1) , 0.324754737 * 10^(-4) , 0.173265513 * 10^(-6) ,
0.921034842 * 10^(-9) , 0.405803845 * 10^(-11) , 0.164174164 * 10^(-13)]:
[ > for i from 1 to 6 do lista2logeval[i]:=log[10](lista2eval[i]): od:
[ > for i from 1 to 6 do lista2logerro[i]:=log[10](lista2erro[i]): od:
[ > evalf(lista2logeval,4);evalf(lista2logerro,4);
[
[2.591, 2.897, 3.198, 3.499, 3.801, 4.102]
[-1.970, -4.488, -6.761, -9.036, -11.39, -13.78]
[ Gráficas comparadas 'evaluaciones versus log[10] del error'
[ > lista1grafa:=[seq([listaleval[i],lista1logerro[i]],i=1..9)]:
[ > grala1:=plot(lista1grafa,style=LINE):
[ > grala2:=plot(lista1grafa,style=POINT,symbol=BOX):
[ > grala3:=textplot([1.158e+004, -8.642, `m1`]):
[ > lista2grafa:=[seq([lista2eval[i],lista2logerro[i]],i=1..6)]:
[ > gra2a1:=plot(lista2grafa,style=LINE):
[ > gra2a2:=plot(lista2grafa,style=POINT,symbol=BOX):
[ > gra2a3:=textplot([1.209e+004, -12.95, `m2`]):
[ > display(grala1,grala2,grala3,gra2a1,gra2a2,gra2a3);

```



Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya dijimos en el problema precedente:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los mas representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del numero de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima

a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

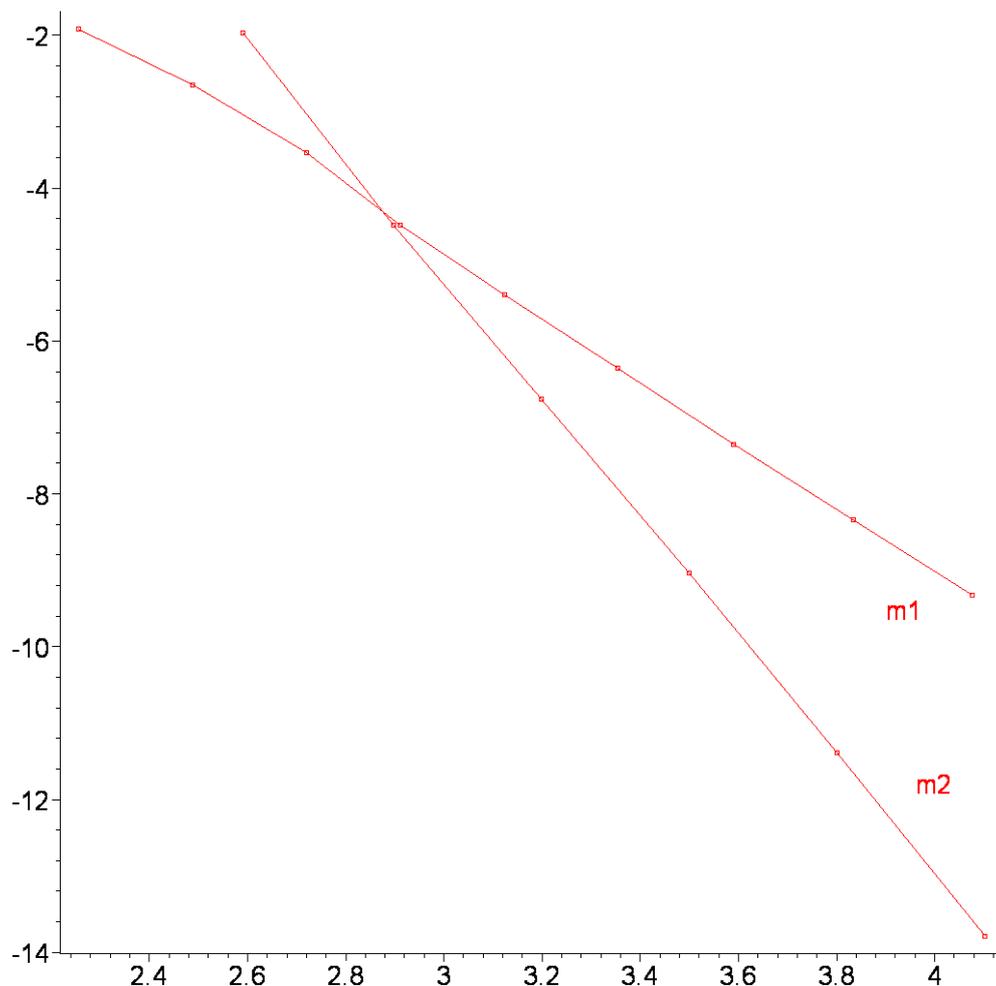
$$\begin{aligned}\log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2)\end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

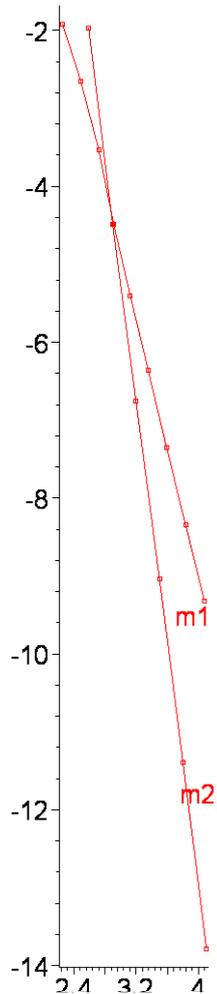
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
[ > lista1grafb:=seq([lista1logeval[i],lista1logerro[i]],i=1..9]):  
[ > gra1b1:=plot(lista1grafb,style=LINE):  
[ > gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):  
[ > gra1b3:=textplot([3.938, -9.505,`m1`]):  
[ > lista2grafb:=seq([lista2logeval[i],lista2logerro[i]],i=1..6]):  
[ > gra2b1:=plot(lista2grafb,style=LINE):  
[ > gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):  
[ > gra2b3:=textplot([3.999, -11.78,`m2`]):  
[ > display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3);
```



```
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,scaling='CONSTRAINED');
```



Orden efectivo de los métodos

```
> pendm1:=evalf((lista1logerro[9]-lista1logerro[8])/(lista1logeval[9]-lista1logeval[8]))
;
```

pendm1 := -4.0515680608587727185

```
> pendm2:=evalf((lista2logerro[6]-lista2logerro[5])/(lista2logeval[6]-lista2logeval[5]))
;
```

pendm2 := -7.9403665875614143211

El orden efectivo de m1 es de 4 . El de m2 es de 8 .

Método con mejor comportamiento

El método m2, el de paso fijo, tiene mejor comportamiento tanto en los errores efectivamente calculados como en lo que toca al orden de los métodos.

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-14 (del EXAMEN EXTRAORDINARIO de 03JUL10)

Para un problema escalar se diseñan dos métodos numéricos de integración de ecuaciones diferenciales que designaremos con los nombres de M1 y M2 .

Para determinar cuál es el más apto, se realiza la integración de un problema test entre 0 y 1 . El primer método es de paso fijo. Se ensaya con los pasos

$$\frac{1}{2^4} = 0.0625, \frac{1}{2^5} = 0.0313, \frac{1}{2^6} = 0.0156, \frac{1}{2^7} = 0.0078, \frac{1}{2^8} = 0.0039$$

siendo el número de pasos para el M1

16 , 32 , 64 , 128 , 256

y el número de evaluaciones

32 , 64 , 128 , 256 , 512

en cada caso. El método M2 es un par encajado. Se prueba con tolerancias

0.001 , 0.0005 , 0.0001 , 0.00005 , 0.00001 ,

y para ellas se han necesitado en el cálculo los siguientes números de evaluaciones

48 , 96 , 192 , 384 , 768

Para cada método se calcula el \log_{10} del error cometido en la aproximación de la solución en $x=1$. Los logaritmos de los errores que se obtienen son, para el M1

- 3.929 , - 4.640 , - 5.271 , - 6.014 , - 6.642

y para el M2

- 3.168 , - 4.722 , - 6.862 , - 7.801 , - 8.639

Con estos datos, se debe construir la 'gráfica de eficiencia' que mezcla los

resultados de M1 y M2 exhibiendo 'número de evaluaciones versus

\log_{10} del error'. Además de dirá qué método es más interesante en la zona de integración empleada.

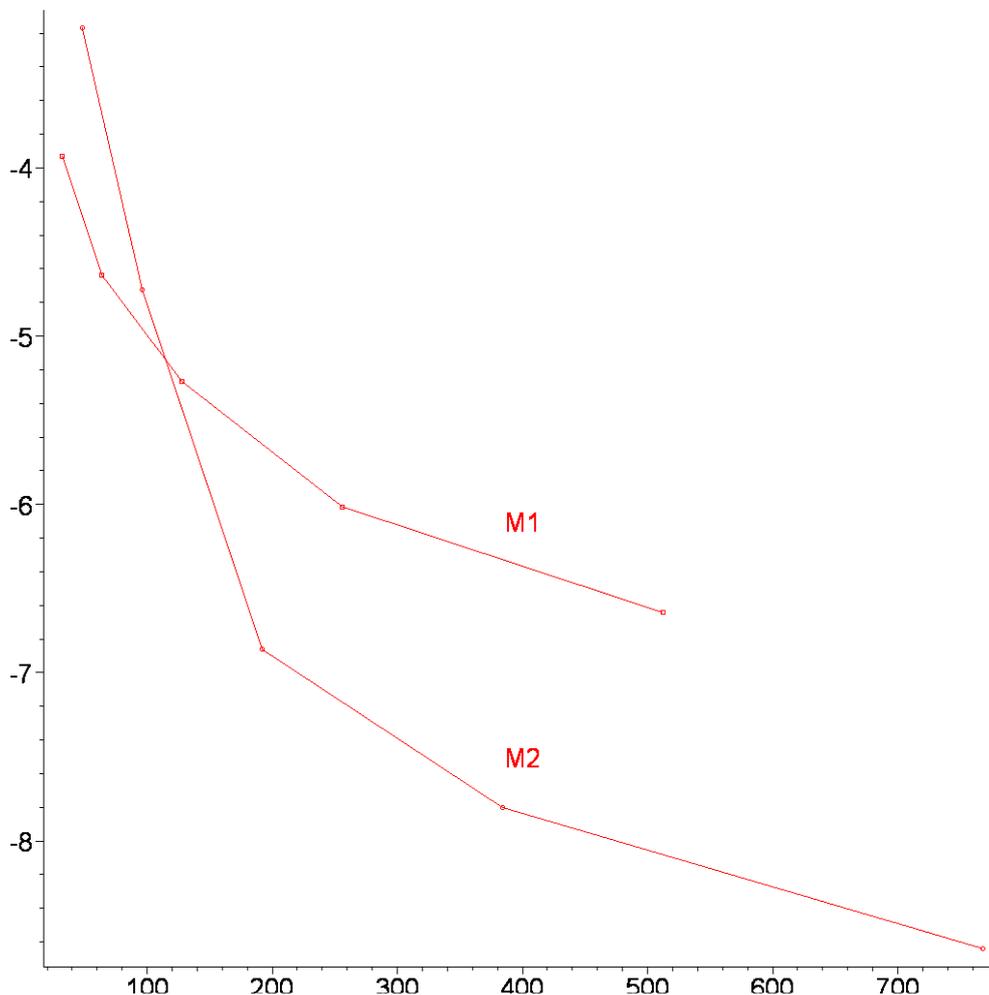
¿Se puede deducir de los datos anteriores con qué orden están funcionando los métodos?; de ser así, ¿cuál es el que tiene mejor comportamiento?

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
[ Variables lista
[ > listaeval:=[32 , 64 , 128 , 256 , 512]:
```

```

[ > listallogeval:= [0,0,0,0,0]:
[ > for i from 1 to 5 do listallogeval[i]:=log[10](listaleval[i]): od:
[ > lista1logerro:= [ - 3.929 , - 4.640 , - 5.271 , - 6.014 , - 6.642]:
[ > lista2eval:= [48 , 96 , 192 , 384 , 768]:
[ > listalog2eval:= [0,0,0,0,0]:
[ > for i from 1 to 5 do lista2logeval[i]:=log[10](lista2eval[i]): od:
[ > lista2logerro:= [- 3.168 , - 4.722 , - 6.862 , - 7.801 , - 8.639]:
[ Gráficas comparadas 'número de evaluaciones versus log[10] del error'
[ > listagra1:= [seq([lista1eval[i],lista1logerro[i]],i=1..5)]:
[ > gra11:=plot(listagra1,style=LINE):
[ > gra12:=plot(listagra1,style=POINT,symbol=BOX):
[ > gra13:=textplot([400., -6.1, `M1`]):
[ > listagra2:= [seq([lista2eval[i],lista2logerro[i]],i=1..5)]:
[ > gra21:=plot(listagra2,style=LINE):
[ > gra22:=plot(listagra2,style=POINT,symbol=CIRCLE):
[ > gra23:=textplot([400., -7.5, `M2`]):
[ > display(gra11,gra12,gra13,gra21,gra22,gra23);

```



Método más interesante en la zona de integración empleada

El método M2 está más abajo en la zona derecha de la gráfica y se comporta mejor en la zona de integración empleada.

Comparación 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya sabemos de otros problemas:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar

en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los más representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del número de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

$$\begin{aligned} \log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2) \end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

Orden efectivo de los métodos, empleando la pendiente del último segmento.

```
> pendm1:=evalf((lista1logerro[5]-lista1logerro[4])/(lista1logeval[5]-lista1logeval[4]))
;
pendm1 := -2.0861708435892635545
> pendm2:=evalf((lista2logerro[5]-lista2logerro[4])/(lista2logeval[5]-lista2logeval[4]))
;
pendm2 := -2.7837757435156096476
```

Método más interesante en cuanto al orden efectivo

El método M2 presenta el mejor orden efectivo de los dos métodos; respecto al orden, es el de mejor comportamiento

>

Ejercicios de

(4) Estimación del error y cambio de paso

Ejercicio 04-15 (del EXAMEN ORDINARIO DE 09FEB09)

Para comparar la eficiencia de tres métodos de tipo RUNGE-KUTTA-FEHLBERG, que denotaremos por m_1 , m_2 y m_3 , se integra un problema test del que se conoce la solución exacta. Para ello se realiza la integración del problema en el intervalo $[0,1]$ para distintos valores de la tolerancia, anotando para cada uno de los métodos los errores obtenidos

$0.42007 \cdot 10^{-2}$, $0.22406 \cdot 10^{-2}$, $0.29345 \cdot 10^{-3}$, $0.32517 \cdot 10^{-4}$,
 $0.39793 \cdot 10^{-5}$, $0.43661 \cdot 10^{-6}$, $0.64522 \cdot 10^{-7}$, $0.85556 \cdot 10^{-8}$

para m_1 ,

$0.10973 \cdot 10^{-1}$, $0.99718 \cdot 10^{-2}$, $0.82375 \cdot 10^{-3}$, $0.31979 \cdot 10^{-4}$,
 $0.43661 \cdot 10^{-5}$, $0.14522 \cdot 10^{-6}$, $0.85556 \cdot 10^{-8}$, $0.97351 \cdot 10^{-10}$

para m_2 , y finalmente

$0.38188 \cdot 10^{-3}$, $0.18672 \cdot 10^{-3}$, $0.57947 \cdot 10^{-4}$, $0.89295 \cdot 10^{-5}$,
 $0.26507 \cdot 10^{-5}$, $0.39692 \cdot 10^{-6}$, $0.72924 \cdot 10^{-7}$, $0.13920 \cdot 10^{-8}$

para m_3 . Asimismo, el número de evaluaciones empleado en cada uno de los tres casos viene dado por

390, 790, 1580, 3160, 6330, 12670, 19000, 25340

para m_1 y

308, 524, 816, 1332, 2264, 3896, 6824, 11932

para m_2 y m_3 . Se pide:

- Construir una gráfica de eficiencia que recoja simultáneamente los resultados obtenidos para m_1 , m_2 y m_3 (en doble escala logarítmica).
- Deducir el orden efectivo de cada uno de los tres métodos para este problema test. ¿Se puede deducir de lo anterior el orden de los métodos (razónese la respuesta)?
- ¿Se puede estimar de algún modo la proporción que guardan entre sí las partes principales del error de los métodos m_1 y m_3 para este problema? De ser posible, hágase esta estimación.
- ¿Se comporta siempre mejor (para todos los valores de la tolerancia) el método de orden más alto? Razónese a qué se debe lo que se observa en la gráfica.

```
[ > restart:  
[ > interface(labeling=false):  
[ > Digits:=20:  
[
```

```
[ > with(linalg):with(plots):
```

[Variables lista

```
[ > lista1eval:=[0,0,0,0,0,0,0,0]:listalog1eval:=[0,0,0,0,0,0,0,0]:
[ > lista1erro:=[0,0,0,0,0,0,0,0]:listalog1erro:=[0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0]:listalog2eval:=[0,0,0,0,0,0,0,0]:
[ > lista2erro:=[0,0,0,0,0,0,0,0]:listalog2erro:=[0,0,0,0,0,0,0,0]:
[ > lista3eval:=[0,0,0,0,0,0,0,0]:listalog3eval:=[0,0,0,0,0,0,0,0]:
[ > lista3erro:=[0,0,0,0,0,0,0,0]:listalog3erro:=[0,0,0,0,0,0,0,0]:
```

[Datos de m1

```
[ > lista1eval:=[390 , 790 , 1580 , 3160 , 6330 , 12670 , 19000 , 25340]:
[ > lista1erro:=[0.42007*10^(-2) , 0.22406*10^(-2) , 0.29345*10^(-3) , 0.32517*10^(-4) ,
[ 0.39793*10^(-5) , 0.43661*10^(-6) , 0.64522*10^(-7) , 0.85556*10^(-8)]:
[ > for i from 1 to 8 do listalog1eval[i]:=log[10](lista1eval[i]): od:
[ > for i from 1 to 8 do listalog1erro[i]:=log[10](lista1erro[i]): od:
[ > evalf(listalog1eval,4);evalf(listalog1erro,4);
[
[ 2.591, 2.897, 3.198, 3.499, 3.801, 4.102, 4.278, 4.403]
[ -2.377, -2.650, -3.532, -4.488, -5.400, -6.360, -7.190, -8.068]
```

[Datos de m2

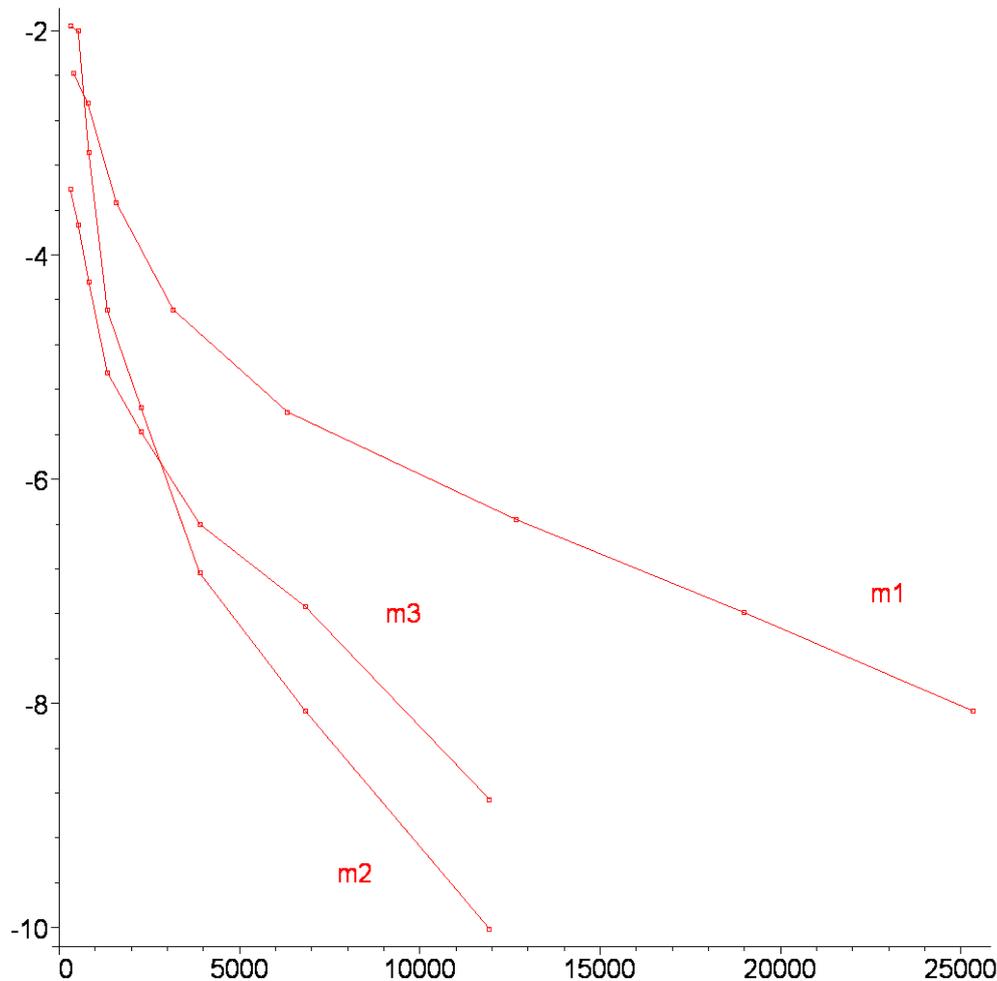
```
[ > lista2eval:=[308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:
[ > lista2erro:=[0.10973*10^(-1) , 0.99718*10^(-2) , 0.82375*10^(-3) , 0.31979*10^(-4) ,
[ 0.43661*10^(-5) , 0.14522*10^(-6) , 0.85556*10^(-8) , 0.97351*10^(-10)]:
[ > for i from 1 to 8 do listalog2eval[i]:=log[10](lista2eval[i]): od:
[ > for i from 1 to 8 do listalog2erro[i]:=log[10](lista2erro[i]): od:
[ > evalf(listalog2eval,4);evalf(listalog2erro,4);
[
[ 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]
[ -1.960, -2.001, -3.084, -4.495, -5.360, -6.838, -8.068, -10.01]
```

[Datos de m3

```
[ > lista3eval:=[308 , 524 , 816 , 1332 , 2264 , 3896 , 6824 , 11932]:
[ > lista3erro:=[0.38188*10^(-3) , 0.18672*10^(-3) , 0.57947*10^(-4) , 0.89295*10^(-5) ,
[ 0.26507*10^(-5) , 0.39692*10^(-6) , 0.72924*10^(-7) , 0.13920*10^(-8)]:
[ > for i from 1 to 8 do listalog3eval[i]:=log[10](lista3eval[i]): od:
[ > for i from 1 to 8 do listalog3erro[i]:=log[10](lista3erro[i]): od:
[ > evalf(listalog3eval,4);evalf(listalog3erro,4);
[
[ 2.488, 2.719, 2.911, 3.124, 3.354, 3.590, 3.833, 4.076]
[ -3.418, -3.729, -4.237, -5.049, -5.577, -6.401, -7.137, -8.856]
```

[Gráficas comparadas 'evaluaciones versus log[10] del error'

```
[ > lista1grafa:=[seq([lista1eval[i],listalog1erro[i]],i=1..8)]:
[ > gral1:=plot(lista1grafa,style=LINE):
[ > gral2:=plot(lista1grafa,style=POINT,symbol=BOX):
[ > gral3:=textplot([2.3e+004, -7.0, `m1`]):
[ > lista2grafa:=[seq([lista2eval[i],listalog2erro[i]],i=1..8)]:
[ > gra2a1:=plot(lista2grafa,style=LINE):
[ > gra2a2:=plot(lista2grafa,style=POINT,symbol=BOX):
[ > gra2a3:=textplot([8.2e+003, -9.5, `m2`]):
[ > lista3grafa:=[seq([lista3eval[i],listalog3erro[i]],i=1..8)]:
[ > gra3a1:=plot(lista3grafa,style=LINE):
[ > gra3a2:=plot(lista3grafa,style=POINT,symbol=BOX):
[ > gra3a3:=textplot([9552, -7.183, `m3`]):
[ > display(gra1a1,gral2,gral3,gra2a1,gra2a2,gra2a3,gra3a1,gra3a2,gra3a3);
```



Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

Recordemos lo que ya dijimos en el problema precedente:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los mas representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el log[10] del numero de evaluaciones y en el de ordenadas el log[10] del error. Entonces también el aspecto de la gráfica se aproxima

a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

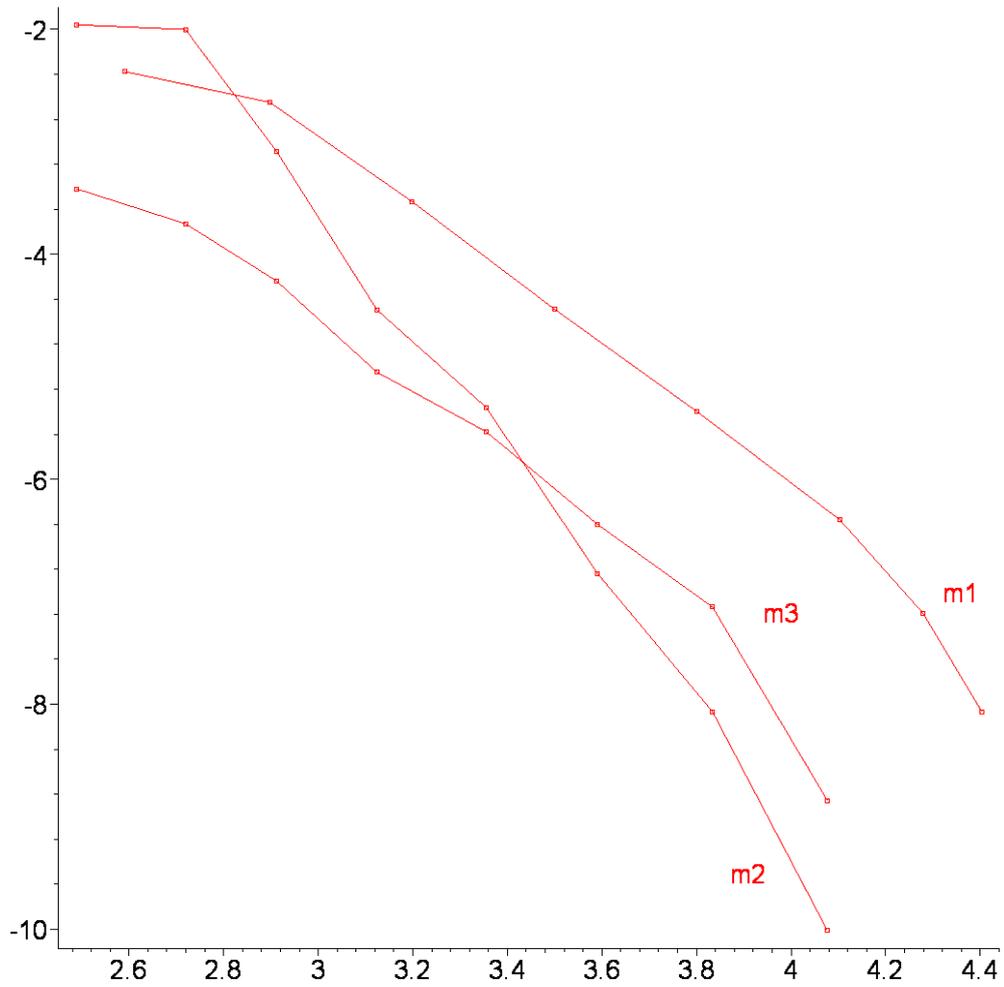
$$\begin{aligned} \log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2) \end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

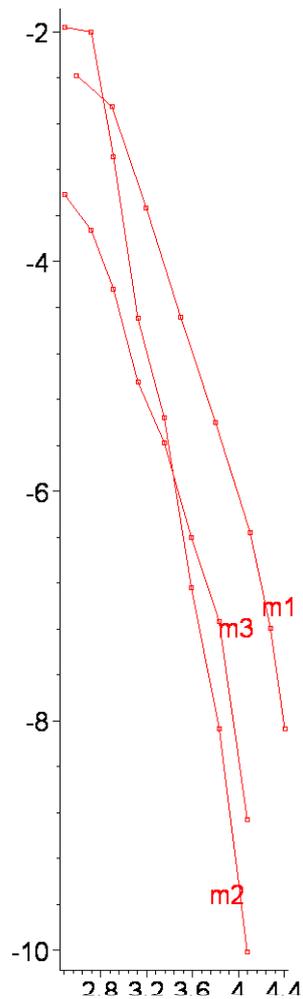
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas 'log[10] del paso versus log[10] del error', que carecen ahora de sentido, sino las gráficas 'log[10] del número de evaluaciones versus log[10] del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
[ > lista1grafb:=seq([listalog1eval[i],listalog1erro[i]],i=1..8):
[ > gra1b1:=plot(lista1grafb,style=LINE):
[ > gra1b2:=plot(lista1grafb,style=POINT,symbol=BOX):
[ > gra1b3:=textplot([4.36, -7.,`m1`]):
[ > lista2grafb:=seq([listalog2eval[i],listalog2erro[i]],i=1..8):
[ > gra2b1:=plot(lista2grafb,style=LINE):
[ > gra2b2:=plot(lista2grafb,style=POINT,symbol=BOX):
[ > gra2b3:=textplot([3.91, -9.5,`m2`]):
[ > lista3grafb:=seq([listalog3eval[i],listalog3erro[i]],i=1..8):
[ > gra3b1:=plot(lista3grafb,style=LINE):
[ > gra3b2:=plot(lista3grafb,style=POINT,symbol=BOX):
[ > gra3b3:=textplot([3.98, -7.183,`m3`]):
[ > display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,gra3b1,gra3b2,gra3b3);
```



```
> display(gra1b1,gra1b2,gra1b3,gra2b1,gra2b2,gra2b3,gra3b1,gra3b2,gra3b3,scaling='CONSTRAINED');
```



a) Orden efectivo de los métodos

```

> pendm1:=evalf((listalog1erro[8]-listalog1erro[7])/(listalog1eval[8]-listalog1eval[7]))
;
pendm1 := -7.0166831799694800746
> pendm2:=evalf((listalog2erro[8]-listalog2erro[7])/(listalog2eval[8]-listalog2eval[7]))
;
pendm2 := -8.0103688084028781068
> pendm3:=evalf((listalog3erro[8]-listalog3erro[7])/(listalog3eval[8]-listalog3eval[7]))
;
pendm3 := -7.0845234440415768611

```

El orden efectivo de m2 es de 8. Los de m1 y m3 son de 7.

- [>
- [>
- [>

05. Sistemas autónomos: métodos de RUNGE-KUTTA

- 1 Presentación de los sistemas
- 2 Notaciones y organización
- 3 Aumento del número de condiciones

Presentación de los sistemas

Ya dijimos que , cuando el problema planteado es un sistema

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

los vectores \mathbf{y} o $\mathbf{y}(x)$ e \mathbf{y}' o $\mathbf{y}'(x)$ como función incógnita y su derivada son de tamaño m (hablaremos de sistemas sólo en el caso de sistemas de dimensión $m \geq 2$, reservando la palabra 'ecuación' o el término 'ecuación escalar' para el caso $m = 1$) y la función

$$\mathbf{f} : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$$

devuelve valores vectoriales también de m componentes. Los valores de la variable independiente x son siempre números reales con valor en algún intervalo $[a, b]$ de \mathbb{R} .

También en el primer capítulo explicamos que íbamos a usar la notación ${}^i y(x)$ para representar la componente número i de $\mathbf{y}(x)$ y lo mismo ${}^i y_n$ para representar la componente número i de la aproximación \mathbf{y}_n a $\mathbf{y}(x_n)$. Por su parte, ${}^i f$ representaba la componente número i de la función vectorial \mathbf{f} .

Otro aspecto sobre el que hablamos fue que sólo íbamos a razonar con sistemas autónomos, lo que simplificaba notablemente la programación, a pesar de que, si un sistema de tamaño m no es autónomo, su reconvertido a autónomo es de tamaño $m + 1$, aunque esto no representa ninguna penalización importante en el caso $m \geq 2$ (si que lo es cuando el no autónomo original es de tamaño $m = 1$, o sea una ecuación escalar, pero este aspecto ha sido tratado extensamente antes).

Consideremos lo que hemos llamado un problema tipo autónomo, consistente ahora en un sistema autónomo de primer orden en la forma normal, y una condición inicial

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

donde la función \mathbf{f} es

$$\mathbf{f} : \mathbb{R}^m \rightarrow \mathbb{R}^m$$

y en su dominio que ahora es todo \mathbb{R}^m es Lipschitziana (respecto de la variable \mathbf{y}), con constante de Lipschitz L .

Ya dijimos que, en estas condiciones, en un intervalo $[a, b]$ existe una única solución vectorial $\mathbf{y}(x)$ del problema, que es la solución que intentamos aproximar.

En lo que respecta a la búsqueda de aproximación numérica para la solución $\mathbf{y}(x)$ de nuestro problema, suponemos que nos movemos en la red usual de nodos

$$x_n = a + n h, \quad n = 0, \dots, N,$$

que dividen el intervalo $[a, b]$ en N subintervalos iguales y de longitud h . Representamos ahora por \mathbf{y}_n la aproximación

$$\mathbf{y}_n \sim \mathbf{y}(x_n)$$

al valor $\mathbf{y}(x_n)$ de la solución en x_n , o sea, componente a componente,

$$^i y_n \sim ^i y(x_n)$$

Un aspecto práctico más. En los desarrollos teóricos y prácticos que hagamos vamos a hacerlo con el valor concreto $m = 2$ del tamaño. Generalmente, esto no va a restar generalidad a los resultados que exponamos (si algo deja de ser general, presentaremos el resultado para m general, sin esta limitación).

A cambio, la presentación y la interpretación de los resultados será bastante más clara. Luego, en la programación de un método cualquiera, bastará con substituir 2 por m en las sentencias de programación.

Así, con esta simplificación,

$$\mathbf{y} = \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

será la función incógnita,

$$\mathbf{y}' = \begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix}$$

su derivada, y

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} {}^1f({}^1y, {}^2y) \\ {}^2f({}^1y, {}^2y) \end{pmatrix}, \quad \begin{bmatrix} {}^1y(a) \\ {}^2y(a) \end{bmatrix} = \begin{bmatrix} {}^1\eta \\ {}^2\eta \end{bmatrix}$$

un problema dado por un sistema autónomo y su condición inicial, sistema que se escribe suprimiendo toda aparición de vectores como

$$\begin{cases} {}^1y' = {}^1f({}^1y, {}^2y), & {}^1y(a) = {}^1\eta \\ {}^2y' = {}^2f({}^1y, {}^2y), & {}^2y(a) = {}^2\eta, \end{cases}$$

Consideremos un método

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \Phi(\mathbf{y}_n, h)$$

(lo estamos escribiendo, claro, en el caso autónomo). El error local sigue siendo

$$\mathbf{T}_{n+1}(h) = \mathbf{y}(x_{n+1}) - \mathbf{y}(x_n) - h \Phi(\mathbf{y}(x_n), h),$$

o mejor,

$$\mathbf{T}(h) = \mathbf{y}(x+h) - \mathbf{y}(x) - h \Phi(\mathbf{y}(x), h),$$

y también seguimos poniendo

$$\frac{\mathbf{T}(h)}{h} = \Delta(h) - \Phi(h),$$

con

$$\Delta(h) = \frac{\mathbf{y}(x+h) - \mathbf{y}(x)}{h}.$$



También como antes

$$\Delta(h) = \mathbf{y}'(x) + \frac{h}{2} \mathbf{y}''(x) + \frac{h^2}{6} \mathbf{y}'''(x) + \frac{h^3}{24} \mathbf{y}^{(4)}(x) + \dots,$$

y

$$\Phi(h) = \Phi(0) + h \Phi'(0) + \frac{h^2}{2} \Phi''(0) + \frac{h^3}{6} \Phi'''(0) + \dots,$$

y el objetivo de los métodos es cancelar en lo posible las potencias más bajas de h en la diferencia $\Delta(h) - \Phi(h)$ obteniendo un método de orden p cuando

$$\frac{\mathbf{T}(h)}{h} = \Delta(h) - \Phi(h) = \mathcal{O}(h^p),$$

o, lo que es lo mismo,

$$\mathbf{T}(h) = \mathcal{O}(h^{p+1}).$$



La notación $\mathbf{f}^{(k)}(\mathbf{y})$ para $\mathbf{y}^{(k+1)}(x)$

Con el cambio que representa el que la ecuación $\mathbf{y}' = \mathbf{f}(\mathbf{y})$ no dependa explícitamente de x , volvemos a tener la necesidad de expresar las derivadas $\mathbf{y}^{(k)}(x)$ en términos de la función \mathbf{f} .

También es posible hacerlo y de una manera semejante al caso escalar. Pero ahora debemos definir por recurrencia

$$\begin{aligned}\mathbf{f}^{(0)}(\mathbf{y}) &= \mathbf{f}(\mathbf{y}), \\ \mathbf{f}^{(k+1)}(\mathbf{y}) &= \mathbf{f}_{\mathbf{y}}^{(k)}(\mathbf{y}) \mathbf{f}(\mathbf{y}),\end{aligned}\tag{0}$$

haciendo notar que la expresión $\mathbf{f}_{\mathbf{y}}^{(k)}(\mathbf{y}) \mathbf{f}(\mathbf{y})$ es el vector (la función vectorial) producto de una matriz y un vector: la matriz es la matriz jacobiana $\mathbf{f}_{\mathbf{y}}^{(k)}(\mathbf{y})$ de la función vectorial $\mathbf{f}^{(k)}$ respecto de su variable, también vectorial, \mathbf{y} , y el vector es la función vectorial \mathbf{f} . Y también debemos notar que, ahora, no es posible intercambiar el orden de $\mathbf{f}_{\mathbf{y}}^{(k)}(\mathbf{y})$ y de \mathbf{f} , que no se pueden multiplicar en otro orden.



La comprobación de que $\mathbf{y}'(x) = \mathbf{f}^{(0)}(\mathbf{y}(x))$ consiste en observar el sistema; no es difícil ver que $\mathbf{y}''(x) = \mathbf{f}^{(1)}(\mathbf{y}(x))$, y se puede comprobar por recurrencia que

$$\mathbf{y}^{(k+1)}(x) = \mathbf{f}^{(k)}(\mathbf{y}(x)), \quad k = 0, 1, 2, \dots,$$

que es la descripción de $\mathbf{y}^{(k+1)}(x)$ que buscábamos.

Ahora volvemos a tener que

$$\Delta(h) = \mathbf{f}^{(0)} + \frac{h}{2} \mathbf{f}^{(1)} + \frac{h^2}{6} \mathbf{f}^{(2)} + \frac{h^3}{24} \mathbf{f}^{(3)} + \dots,$$

para comparar con

$$\Phi(h) = \Phi(0) + h \Phi'(0) + \frac{h^2}{2} \Phi''(0) + \frac{h^3}{6} \Phi'''(0) + \dots,$$

y se llega finalmente a las mismas ecuaciones

$$\Phi^{(p-1)}(0) = \frac{1}{p} \mathbf{f}^{(p-1)},$$

ahora vectoriales, eso sí, que nos van garantizando los sucesivos órdenes de los métodos.



Veamos cuáles son algunos de los primeros $\mathbf{f}^{(k)}$ que nos permitan, al menos, hacer alguna prueba. El primero es

$$\mathbf{f}^{(0)}(\mathbf{y}) = \mathbf{f}(\mathbf{y})$$

o mejor, sin incluir el valor de la variable,

$$\mathbf{f}^{(0)} = \mathbf{f} = \begin{pmatrix} {}^1f \\ {}^2f \end{pmatrix}$$

El segundo es

$$\mathbf{f}^{(1)} = \mathbf{f}_y \mathbf{f} = \begin{pmatrix} {}^1f_1 & {}^1f_2 \\ {}^2f_1 & {}^2f_2 \end{pmatrix} \begin{pmatrix} {}^1f \\ {}^2f \end{pmatrix} = \begin{pmatrix} {}^1f_1 {}^1f + {}^1f_2 {}^2f \\ {}^2f_1 {}^1f + {}^2f_2 {}^2f \end{pmatrix}$$

Una notación, llamada generalmente 'tensorial' permite reducir un poco esta fórmula y, sobre todo, las que siguen.

La repetición en

$${}^i f_j {}^j f$$

del índice j en posiciones opuestas (arriba y abajo) hace que debamos interpretar esta expresión como la suma de términos

$${}^i f_j {}^j f = \sum_{j=1}^m {}^i f_j {}^j f = {}^i f_1 {}^1 f + {}^i f_2 {}^2 f$$

(lo último en 'nuestro caso' de $m = 2$). Así, tenemos abreviadamente

$$\mathbf{f}^{(1)} = \begin{pmatrix} {}^1 f_j {}^j f \\ {}^2 f_j {}^j f \end{pmatrix}$$

El siguiente es $\mathbf{f}^{(2)} = \mathbf{f}_y^{(1)} \mathbf{f}$, o sea

$$\begin{aligned} \mathbf{f}^{(2)} &= \begin{pmatrix} {}^1 f_j j f \\ {}^2 f_j j f \end{pmatrix}_y \begin{pmatrix} {}^1 f \\ {}^2 f \end{pmatrix} = \begin{pmatrix} \left({}^1 f_j j f \right)_1 & \left({}^1 f_j j f \right)_2 \\ \left({}^2 f_j j f \right)_1 & \left({}^2 f_j j f \right)_2 \end{pmatrix} \begin{pmatrix} {}^1 f \\ {}^2 f \end{pmatrix} \\ &= \begin{pmatrix} {}^1 f_{j_1} j f + {}^1 f_j j f_1 & {}^1 f_{j_2} j f + {}^1 f_j j f_2 \\ {}^2 f_{j_1} j f + {}^2 f_j j f_1 & {}^2 f_{j_2} j f + {}^2 f_j j f_2 \end{pmatrix} \begin{pmatrix} {}^1 f \\ {}^2 f \end{pmatrix} \\ &= \begin{pmatrix} \left({}^1 f_{j_1} j f + {}^1 f_j j f_1 \right) {}^1 f + \left({}^1 f_{j_2} j f + {}^1 f_j j f_2 \right) {}^2 f \\ \left({}^2 f_{j_1} j f + {}^2 f_j j f_1 \right) {}^1 f + \left({}^2 f_{j_2} j f + {}^2 f_j j f_2 \right) {}^2 f \end{pmatrix} \\ &= \begin{pmatrix} {}^1 f_{j_k} j f k f + {}^1 f_j j f_k k f \\ {}^2 f_{j_k} j f k f + {}^2 f_j j f_k k f \end{pmatrix} \end{aligned}$$

y, en definitiva,

$$i f^{(2)} = i f_{j_k} j f k f + i f_j j f_k k f,$$

todo ello usando la notación tensorial, lo que significa que

$$i f^{(2)} = \sum_{j=1}^m \sum_{k=1}^m \left(i f_{j_k} j f k f + i f_j j f_k k f \right),$$

o sea, la suma de $2m^2$ términos (8 términos cuando $m = 2$, que es la tónica de nuestra presentación).

Aunque las expresiones de los $f^{(k)}$ iban siendo cada vez mucho más complejas, ahora, en el caso de $\mathbf{f}^{(k)}$, todo se vuelve más difícil por la irrupción de los índices de coordenada. Solo la que hemos llamado 'notación tensorial' consigue dejar las fórmulas en un tamaño que se pueda considerar moderado.

Por el momento, al menos, no seguiremos más adelante en esta vía.

Métodos de RUNGE-KUTTA para sistemas

Para un problema tipo con un sistema autónomo

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

y un método de RUNGE-KUTTA

$$y_{n+1} = y_n + h \Phi(y_n, h)$$

de los que hemos estudiado, la trasposición al caso de un sistema autónomo es muy sencilla. Basta escribir el método en la forma vectorial

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \Phi(\mathbf{y}_n, h)$$

y llevarlo a cabo, lo que, para escalares, consiste en organizar los cálculos como

$$\begin{cases} {}^1y_{n+1} = {}^1y_n + h {}^1\Phi({}^1y_n, {}^2y_n, h), \\ {}^2y_{n+1} = {}^2y_n + h {}^2\Phi({}^1y_n, {}^2y_n, h), \end{cases}$$



Por ejemplo, para el 'método modificado de EULER'

$$\begin{aligned} k_1 &= f(y_n), \\ k_2 &= f(y_n + h k_1) \end{aligned}$$

$$y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2),$$

se escribe como

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n), \\ \mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + h \mathbf{k}_1) \end{aligned}$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2),$$



o sea, se hace

$${}^1k_1 = {}^1f({}^1y_n, {}^2y_n),$$

$${}^2k_1 = {}^2f({}^1y_n, {}^2y_n),$$

$${}^1k_2 = {}^1f({}^1y_n + h {}^1k_1, {}^2y_n + h {}^2k_1),$$

$${}^2k_2 = {}^2f({}^1y_n + h {}^1k_1, {}^2y_n + h {}^2k_1),$$

$${}^1y_{n+1} = {}^1y_n + \frac{h}{2} ({}^1k_1 + {}^1k_2),$$

$${}^2y_{n+1} = {}^2y_n + \frac{h}{2} ({}^2k_1 + {}^2k_2),$$

sencillo, aunque lioso desde el punto de vista de los índices de las coordenadas (y eso que sólo lo escribimos con 2 coordenadas).

Análogamente, el método de RUNGE-KUTTA 'clásico'

$$k_1 = f(y_n),$$

$$k_2 = f(y_n + \frac{1}{2} h k_1)$$

$$k_3 = f(y_n + \frac{1}{2} h k_2)$$

$$k_4 = f(y_n + h k_3)$$

$$y_{n+1} = y_n + \frac{1}{6} h (k_1 + 2 k_2 + 2 k_3 + k_4).$$

es ahora

$$\mathbf{k}_1 = f(\mathbf{y}_n),$$

$$\mathbf{k}_2 = f(\mathbf{y}_n + \frac{1}{2} h \mathbf{k}_1)$$

$$\mathbf{k}_3 = f(\mathbf{y}_n + \frac{1}{2} h \mathbf{k}_2)$$

$$\mathbf{k}_4 = f(\mathbf{y}_n + h \mathbf{k}_3)$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{1}{6} h (\mathbf{k}_1 + 2 \mathbf{k}_2 + 2 \mathbf{k}_3 + \mathbf{k}_4).$$

y eso significa

$$\begin{aligned}
 {}^1k_1 &= {}^1f({}^1y_n, {}^2y_n), \\
 {}^2k_1 &= {}^2f({}^1y_n, {}^2y_n), \\
 {}^1k_2 &= {}^1f\left({}^1y_n + \frac{1}{2}h {}^1k_1, {}^2y_n + \frac{1}{2}h {}^2k_1\right) \\
 {}^2k_2 &= {}^2f\left({}^1y_n + \frac{1}{2}h {}^1k_1, {}^2y_n + \frac{1}{2}h {}^2k_1\right) \\
 {}^1k_3 &= {}^1f\left({}^1y_n + \frac{1}{2}h {}^1k_2, {}^2y_n + \frac{1}{2}h {}^2k_2\right) \\
 {}^2k_3 &= {}^2f\left({}^1y_n + \frac{1}{2}h {}^1k_2, {}^2y_n + \frac{1}{2}h {}^2k_2\right) \\
 {}^1k_4 &= {}^1f\left({}^1y_n + h {}^1k_3, {}^2y_n + h {}^2k_3\right) \\
 {}^2k_4 &= {}^2f\left({}^1y_n + h {}^1k_3, {}^2y_n + h {}^2k_3\right) \\
 {}^1y_{n+1} &= {}^1y_n + \frac{1}{6}h ({}^1k_1 + 2 {}^1k_2 + 2 {}^1k_3 + {}^1k_4) \\
 {}^2y_{n+1} &= {}^2y_n + \frac{1}{6}h ({}^2k_1 + 2 {}^2k_2 + 2 {}^2k_3 + {}^2k_4).
 \end{aligned}$$

Pero si escribir un método en el caso de un sistema autónomo no parece difícil, lo que no está claro es el orden que ahora puede obtenerse con cada número de etapas, ya que la multiplicidad de coordenadas hace que se multipliquen las diferenciales que intervienen y, consecuentemente, las condiciones a imponer para un orden dado.

Vamos a ver que, para los órdenes bajos, no ocurre nada de eso, y que sí ocurre en cuanto llegamos a orden 4.

Veamos, por ejemplo, que para orden 2 las cosas son de lo más tranquilas. Para un método explícito de RUNGE-KUTTA de dos etapas, tendremos ahora

$$\begin{aligned}
 \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n), \\
 \mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + a_{2,1}h \mathbf{k}_1) \\
 \mathbf{y}_{n+1} &= \mathbf{y}_n + h(b_1 \mathbf{k}_1 + b_2 \mathbf{k}_2),
 \end{aligned}$$

lo que significa

$$\begin{aligned} {}^1k_1 &= {}^1f({}^1y_n, {}^2y_n), \\ {}^2k_1 &= {}^2f({}^1y_n, {}^2y_n), \\ {}^1k_2 &= {}^1f({}^1y_n + a_{2,1}h {}^1k_1, {}^2y_n + a_{2,1}h {}^2k_1), \\ {}^2k_2 &= {}^2f({}^1y_n + a_{2,1}h {}^1k_1, {}^2y_n + a_{2,1}h {}^2k_1), \\ {}^1y_{n+1} &= {}^1y_n + h(b_1 {}^1k_1 + b_2 {}^1k_2), \\ {}^2y_{n+1} &= {}^2y_n + h(b_1 {}^2k_1 + b_2 {}^2k_2), \end{aligned}$$

siendo entonces la función (vectorial) $\Phi(h) = b_1 \mathbf{k}_1 + b_2 \mathbf{k}_2(h)$, o sea,

$$\begin{aligned} {}^1\Phi(h) &= b_1 {}^1k_1 + b_2 {}^1k_2(h), \\ {}^2\Phi(h) &= b_1 {}^2k_1 + b_2 {}^2k_2(h). \end{aligned}$$

Ponemos

$$\begin{aligned} (1) & \text{ para } ({}^1y_n, {}^2y_n), \\ (2) & \text{ para } ({}^1y_n + a_{2,1}h {}^1k_1, {}^2y_n + a_{2,1}h {}^2k_1). \end{aligned}$$

omitiendo el punto (1) cuando se presente.

Como

$$\begin{aligned} {}^1k_1 &= {}^1f \\ {}^2k_1 &= {}^2f \\ {}^1k_2(h) &= {}^1f(2) \\ {}^2k_2(h) &= {}^2f(2), \end{aligned}$$

resulta

$$\begin{aligned} {}^1k_1(0) &= {}^1f \\ {}^2k_1(0) &= {}^2f \\ {}^1k_2(0) &= {}^1f \\ {}^2k_2(0) &= {}^2f, \end{aligned}$$

luego

$$\begin{aligned} {}^1\Phi(0) &= (b_1 + b_2) {}^1f \\ {}^2\Phi(0) &= (b_1 + b_2) {}^2f. \end{aligned}$$

mientras que

$$\mathbf{f}^{(0)} = \mathbf{f} = \begin{pmatrix} {}^1f \\ {}^2f \end{pmatrix}.$$

La condición para el orden 1 nos lleva a

$$(1a) \quad b_1 + b_2 = 1$$

repetida 2 (= m) veces.

Recordemos para lo que sigue que ${}^i f_j$ representa la derivada respecto de ${}^j y$ de la componente número i de \mathbf{f} o, lo que es lo mismo, la componente número i de la derivada respecto de ${}^j y$ de la función vectorial \mathbf{f} .

$$\begin{aligned} {}^1k'_2(h) &= {}^1f_1(2)a_{2,1} {}^1k_1 + {}^1f_2(2)a_{2,1} {}^2k_1 \\ {}^2k'_2(h) &= {}^2f_1(2)a_{2,1} {}^1k_1 + {}^2f_2(2)a_{2,1} {}^2k_1, \end{aligned}$$

y, para $h = 0$,

$$\begin{aligned} {}^1k'_2(0) &= a_{2,1} {}^1f_1 {}^1f + a_{2,1} {}^1f_2 {}^2f \\ {}^2k'_2(0) &= a_{2,1} {}^2f_1 {}^1f + a_{2,1} {}^2f_2 {}^2f, \end{aligned}$$

luego, como $\Phi'(h) = b_2 \mathbf{k}'_2(h)$,

$$\begin{aligned} {}^1\Phi'(0) &= b_2 a_{2,1} {}^1f_1 {}^1f + b_2 a_{2,1} {}^1f_2 {}^2f \\ {}^2\Phi'(0) &= b_2 a_{2,1} {}^2f_1 {}^1f + b_2 a_{2,1} {}^2f_2 {}^2f, \end{aligned}$$

mientras que

$$\frac{1}{2} \mathbf{f}^{(1)} = \begin{pmatrix} \frac{1}{2} {}^1f_1 {}^1f + \frac{1}{2} {}^1f_2 {}^2f \\ \frac{1}{2} {}^2f_1 {}^1f + \frac{1}{2} {}^2f_2 {}^2f \end{pmatrix}.$$

La condición para el orden 2

$$\Phi'(0) = \frac{1}{2} \mathbf{f}^{(1)}$$

nos lleva a

$$(2b) \quad b_2 a_{2,1} = 1/2$$

repetida 4 ($= m^2$) veces.

De manera que la diferencial elemental f aparece 'repetida' 2 ($= m$) veces como 1f y 2f , y la diferencial elemental f_y aparece 'repetida' 4 ($= m^2$) veces como 1f_1 1f , 1f_2 2f , 2f_1 1f y 2f_2 2f . Pero no hemos modificado de hecho las condiciones para el orden 2 que aparecían en el caso escalar autónomo.

Aunque es más pesado de escribir, lo mismo sucede también ahora con el caso de 3 etapas y orden 3. Se llega a las mismas condiciones todavía, o sea **(1a)**, ahora repetida 2 ($= m$) veces, y **(2b)**, ahora repetida 4 ($= m^2$) veces para el orden 2 y

$$(3c) \quad b_2 (a_{2,1})^2 + b_3 (a_{3,1} + a_{3,2})^2 = 1/3$$

$$(3e) \quad b_3 a_{2,1} a_{3,2} = 1/6.$$

ahora repetidas 8 ($= m^3$) veces para el orden 3, también máximo.

Un ejemplo

Es de nuevo al estudiar los métodos de orden 4 cuando nos encontramos que existen diferencias entre el caso de sistemas (autónomos) y el caso escalar autónomo que ya estudiamos. Consideremos los métodos de 4 etapas que, para el caso autónomo (escalar o vectorial) tienen el tablero

	a_{21}			
	a_{31}	a_{32}		
	a_{41}	a_{42}	a_{43}	
	b_1	b_2	b_3	b_4

y la organización

$$\begin{aligned}
 \mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n), \\
 \mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + a_{2,1} h \mathbf{k}_1) \\
 \mathbf{k}_3 &= \mathbf{f}(\mathbf{y}_n + a_{3,1} h \mathbf{k}_1 + a_{3,2} h \mathbf{k}_2) \\
 \mathbf{k}_4 &= \mathbf{f}(\mathbf{y}_n + a_{4,1} h \mathbf{k}_1 + a_{4,2} h \mathbf{k}_2 + a_{4,3} h \mathbf{k}_3) \\
 \mathbf{y}_{n+1} &= \mathbf{y}_n + h (b_1 \mathbf{k}_1 + b_2 \mathbf{k}_2 + b_3 \mathbf{k}_3 + b_4 \mathbf{k}_4).
 \end{aligned}$$

De nuevo nos ayudamos de los cálculos que 'Maple' hace por nosotros en el programa **Ruk44.mws** que se encarga de analizar los métodos de 4 etapas para ecuaciones autónomas y no autónomas y para sistemas.

También nos desentendemos de las condiciones para los órdenes 1 a 3, que sabemos que se repiten en el caso de sistemas autónomos respecto de las del caso escalar autónomo. Sin embargo, si nos fijamos en las condiciones (4f), (4h) y (4k) que ya vimos para el caso escalar autónomo, y que correspondían a las diferenciales

$$(f)^3 f_{yyy}, \quad (f)^2 f_y f_{yy} \quad \text{y} \quad f (f_y)^3$$

respectivamente, ahora, a causa de las componentes 1f , 2f y sus derivadas sucesivas se bifurcan en una variedad de condiciones que resultan más difíciles de cumplir.

El mismo ejemplo que entonces citamos viene ahora en nuestro auxilio. En efecto, es fácil comprobar que el método

	1/2			
	1/2	-3/2		
	0	4/3	-1/3	
	1/6	2/3	0	1/6

propuesto por J.D. LAMBERT y que ya citamos, no es de orden 4 más que para el caso escalar autónomo, siendo sólo de orden 3 para un verdadero sistema autónomo, como podemos ver en el programa 'Maple' llamado **Rkut44_contraejemplo.mws** que se encarga de comprobar el hecho citado.

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-01

Para el problema de segundo orden

$$y'' + y = 0$$

$$y(0) = 0, \quad y'(0) = 1$$

y convertido en el problema tipo autónomo de Cauchy

$$y'_1 = y_2$$

$$y'_2 = -y_1$$

$$y_1(0) = 0, \quad y_2(0) = 1$$

(de solución exacta $y_1(x) = \sin(x)$, $y_2(x) = \cos(x)$, o sea $y(x) = \sin(x)$) utilícese el método de

TAYLOR de orden 3 para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:  
[ > interface(labeling=false):  
[ > Digits:=3:
```

El sistema (aut.) y la verdadera solución

```
[ > f[1]:=unapply(y2,y1,y2):  
[   f[2]:=unapply(-y1,y1,y2):  
[ > yv[1]:=unapply(sin(x),x):  
[   yv[2]:=unapply(cos(x),x):
```

Los métodos de TAYLOR y el de orden 3

Los métodos de TAYLOR para sistemas autónomos y en los primeros

órdenes se escriben vectorialmente

$$y_{n+1} = y_n + h * T^{[1]}(y_n, h)$$

$$y_{n+1} = y_n + h * T^{[2]}(y_n, h)$$

$$y_{n+1} = y_n + h * T^{[3]}(y_n, h)$$

donde $T^{[1]}$, $T^{[2]}$ y $T^{[3]}$ representan respectivamente

$$T^{[1]}(y_n, h) = f^{[0]}(y_n)$$

$$T^{[2]}(y_n, h) = f^{[0]}(y_n) + \frac{h}{2} f^{[1]}(y_n)$$

$$T^{[3]}(y_n, h) = f^{[0]}(y_n) + \frac{h}{2} f^{[1]}(y_n) + \frac{h^2}{6} f^{[2]}(y_n)$$

y las $f^{[k]}$ son las derivadas de $f(y(x))$ substituyendo $y'(x)$ por f . Vienen dadas (las primeras)

para cada componente j por

$$f^{[0](j)} = f^j$$

$$f^{[1](j)} = f_k^j f^k \quad (= \sum_{k=1}^m f_k^j f^k)$$

$$f^{[2](j)} = f_{kl}^j f^k f^l + f_k^j f_l^k f^l$$

El último de los métodos de TAYLOR escritos es el de orden 3. $T^{[3]}$ es justamente el desarrollo de $\Delta(x_n, y_n, h)$ hasta dejar el resto en $O(h^3)$

Diferenciales elementales que intervienen

Para $f^{[0]}$ ya están $f[1]$ y $f[2]$

```
> F0[1]:=unapply(f[1](y1,y2),y1,y2):
  F0[2]:=unapply(f[2](y1,y2),y1,y2):
```

Para $f^{[1]}$

```
> f1[1]:=unapply(diff(f[1](y1,y2),y1),y1,y2):
  f1[2]:=unapply(diff(f[2](y1,y2),y1),y1,y2):
  f2[1]:=unapply(diff(f[1](y1,y2),y2),y1,y2):
  f2[2]:=unapply(diff(f[2](y1,y2),y2),y1,y2):
  F1[1]:=unapply(f1[1](y1,y2)*f[1](y1,y2)+f2[1](y1,y2)*f[2](y1,y2),y1,y2):
  F1[2]:=unapply(f1[2](y1,y2)*f[1](y1,y2)+f2[2](y1,y2)*f[2](y1,y2),y1,y2):
```

Para $f^{[2]}$

```
> f11[1]:=unapply(diff(f1[1](y1,y2),y1),y1,y2):
  f11[2]:=unapply(diff(f1[2](y1,y2),y1),y1,y2):
  f12[1]:=unapply(diff(f1[1](y1,y2),y2),y1,y2):
  f12[2]:=unapply(diff(f1[2](y1,y2),y2),y1,y2):
  f22[1]:=unapply(diff(f2[1](y1,y2),y2),y1,y2):
  f22[2]:=unapply(diff(f2[2](y1,y2),y2),y1,y2):
  F2[1]:=unapply(f11[1](y1,y2)*f[1](y1,y2)*f[1](y1,y2)+2*f12[1](y1,y2)*f[1](y1,y2)*f[2](y1,y2)+f22[1](y1,y2)*f[2](y1,y2)*f[2](y1,y2)+f1[1](y1,y2)*f1[1](y1,y2)*f[1](y1,y2)+f2[1](y1,y2)*f1[2](y1,y2)*f[1](y1,y2)+f1[1](y1,y2)*f2[1](y1,y2)*f[2](y1,y2)+f2[1](y1,y2)*f2[2](y1,y2)*f[2](y1,y2),y1,y2):
  F2[2]:=unapply(f11[2](y1,y2)*f[1](y1,y2)*f[1](y1,y2)+2*f12[2](y1,y2)*f[1](y1,y2)*f[2](y1,y2)+f22[2](y1,y2)*f[2](y1,y2)*f[2](y1,y2)+f1[2](y1,y2)*f1[1](y1,y2)*f[1](y1,y2)+f2[2](y1,y2)*f1[2](y1,y2)*f[1](y1,y2)+f1[2](y1,y2)*f2[1](y1,y2)*f[2](y1,y2)+f2[2](y1,y2)*f2[2](y1,y2)*f[2](y1,y2),y1,y2):
```

ahora $T^{[3]}(y_n, h)$ es

```
> T3[1]:=unapply(F0[1](y1,y2)+(h/2)*F1[1](y1,y2)+(h^2/6)*F2[1](y1,y2),y1,y2):
  T3[2]:=unapply(F0[2](y1,y2)+(h/2)*F1[2](y1,y2)+(h^2/6)*F2[2](y1,y2),y1,y2):
```

Funciones que intervienen

```
> print(`1f=` ,f[1](y[1],y[2])):print(`2f=` ,f[2](y[1],y[2])):
                                     1f=, y2
                                     2f=, -y1
> print(`1f_1=` ,f1[1](y[1],y[2])):print(`2f_1=` ,f1[2](y[1],y[2])):print(`1f_2=` ,f2[1](y[1],y[2])):print(`1f_1=` ,f2[2](y[1],y[2])):
                                     1f_1=, 0
                                     2f_1=, -1
                                     1f_2=, 1
                                     1f_1=, 0
```

```

> print(`1f_11=`,f11[1](y[1],y[2])):print(`2f_11=`,f11[2](y[1],y[2])):print(`1f_12=`,f12
[1](y[1],y[2])):print(`2f_12=`,f12[2](y[1],y[2])):print(`1f_22=`,f22[1](y[1],y[2])):pr
int(`2f_22=`,f22[2](y[1],y[2])):

1f_11=, 0
2f_11=, 0
1f_12=, 0
2f_12=, 0
1f_22=, 0
2f_22=, 0

```

Estamos en condiciones de comenzar con el método

```

> x0:=0;y0:=[0.,1.];h:=0.1;

x0 := 0
y0 := [0., 1.]
h := 0.1

```

Primer paso

```

> x1:=x0+h;

x1 := 0.1
> print(`1f=`,f[1](y0[1],y0[2])):print(`2f=`,f[2](y0[1],y0[2])):

1f=, 1.
2f=, -0.
> print(`1f_1=`,f1[1](y0[1],y0[2])):print(`2f_1=`,f1[2](y0[1],y0[2])):print(`1f_2=`,f2[1
](y0[1],y0[2])):print(`1f_1=`,f2[2](y0[1],y0[2])):

1f_1=, 0
2f_1=, -1
1f_2=, 1
1f_1=, 0
> print(`1f_11=`,f11[1](y0[1],y0[2])):print(`2f_11=`,f11[2](y0[1],y0[2])):print(`1f_12=`,
f12[1](y0[1],y0[2])):print(`2f_12=`,f12[2](y0[1],y0[2])):print(`1f_22=`,f22[1](y0[1],
y0[2])):print(`2f_22=`,f22[2](y0[1],y0[2])):

1f_11=, 0
2f_11=, 0
1f_12=, 0
2f_12=, 0
1f_22=, 0
2f_22=, 0
> print(`1f[0]=`,F0[1](y0[1],y0[2])):print(`2f[0]=`,F0[2](y0[1],y0[2])):

1f[0]=, 1.
2f[0]=, -0.
> print(`1f[1]=`,F1[1](y0[1],y0[2])):print(`2f[1]=`,F1[2](y0[1],y0[2])):

1f[1]=, -0.
2f[1]=, -1.
> print(`1f[2]=`,F2[1](y0[1],y0[2])):print(`2f[2]=`,F2[2](y0[1],y0[2])):

1f[2]=, -1.
2f[2]=, 0.
> T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);

0.998
-0.0500
> y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);

y1_1 := 0.0998
y1_2 := 0.995

```

Segundo paso

```

> y0:=y1;x1:=x1+h;

y0 := y1

```

```

[                                     x1 := 0.2
> T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);
                                     0.988
                                     -0.150
> y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);
                                     yI1 := 0.199
                                     yI2 := 0.970
Tercer paso
> y0:=y1;x1:=x1+h;
                                     y0 := y1
                                     x1 := 0.3
> T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);
                                     0.958
                                     -0.248
> y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);
                                     yI1 := 0.295
                                     yI2 := 0.936
> yv[1](x1)-y1[1];yv[2](x1)-y1[2];
                                     0.001
                                     0.019

```

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-02

Para el problema tipo autónomo de Cauchy

$$y'_1 = \frac{-y_1^2 + y_2^4}{y_2^2}$$

$$y'_2 = -\frac{y_1}{y_2}$$

$$y_1(0) = 0, y_2(0) = 1$$

(de solución exacta $y_1(x) = \sin(x) \cos(x)$, $y_2(x) = \cos(x)$) utilícese el método de

TAYLOR de orden 3 para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:  
[ > interface(labeling=false):  
[ > Digits:=3:
```

El sistema (aut.) y la verdadera solución

```
[ > f[1]:=unapply((-y1^2+y2^4)/y2^2,y1,y2):  
[ f[2]:=unapply(-y1/y2,y1,y2):  
[ > yv[1]:=unapply(sin(x)*cos(x),x):  
[ yv[2]:=unapply(cos(x),x):
```

Los métodos de TAYLOR y el de orden 3

Los métodos de TAYLOR para sistemas autónomos y en los primeros

órdenes se escriben vectorialmente

$$y_{n+1} = y_n + h * T^{[1]}(y_n, h)$$

$$y_{n+1} = y_n + h * T^{[2]}(y_n, h)$$

$$y_{n+1} = y_n + h * T^{[3]}(y_n, h)$$

donde $T^{[1]}$, $T^{[2]}$ y $T^{[3]}$ representan respectivamente

$$T^{[1]}(y_n, h) = f^{[0]}(y_n)$$

$$T^{[2]}(y_n, h) = f^{[0]}(y_n) + \frac{h}{2} f^{[1]}(y_n)$$

$$T^{[3]}(y_n, h) = f^{[0]}(y_n) + \frac{h}{2} f^{[1]}(y_n) + \frac{h^2}{6} f^{[2]}(y_n)$$

y las $f^{[k]}$ son las derivadas de $f(y(x))$ substituyendo $y'(x)$ por f . Vienen dadas (las primeras)

para cada componente j por

$$f^{[0](j)} = f^j$$

$$f^{[1](j)} = f_k^j f^k \quad (= \sum_{k=1}^m f_k^j f^k)$$

$$f^{[2](j)} = f_{kl}^j f^k f^l + f_k^j f_l^k f^l$$

El último de los métodos de TAYLOR escritos es el de orden 3. $T^{[3]}$ es justamente el desarrollo de $\Delta(x_n, y_n, h)$ hasta dejar el resto en $O(h^3)$

Diferenciales elementales que intervienen

Para $f^{[0]}$ ya están f[1] y f[2]

```
> F0[1]:=unapply(f[1](y1,y2),y1,y2):
F0[2]:=unapply(f[2](y1,y2),y1,y2):
```

Para $f^{[1]}$

```
> f1[1]:=unapply(diff(f[1](y1,y2),y1),y1,y2):
f1[2]:=unapply(diff(f[2](y1,y2),y1),y1,y2):
f2[1]:=unapply(diff(f[1](y1,y2),y2),y1,y2):
f2[2]:=unapply(diff(f[2](y1,y2),y2),y1,y2):
F1[1]:=unapply(f1[1](y1,y2)*f[1](y1,y2)+f2[1](y1,y2)*f[2](y1,y2),y1,y2):
F1[2]:=unapply(f1[2](y1,y2)*f[1](y1,y2)+f2[2](y1,y2)*f[2](y1,y2),y1,y2):
```

Para $f^{[2]}$

```
> f11[1]:=unapply(diff(f1[1](y1,y2),y1),y1,y2):
f11[2]:=unapply(diff(f1[2](y1,y2),y1),y1,y2):
f12[1]:=unapply(diff(f1[1](y1,y2),y2),y1,y2):
f12[2]:=unapply(diff(f1[2](y1,y2),y2),y1,y2):
f22[1]:=unapply(diff(f2[1](y1,y2),y2),y1,y2):
f22[2]:=unapply(diff(f2[2](y1,y2),y2),y1,y2):
F2[1]:=unapply(f11[1](y1,y2)*f[1](y1,y2)*f[1](y1,y2)+2*f12[1](y1,y2)*f[1](y1,y2)*f[2](y1,y2)+f22[1](y1,y2)*f[2](y1,y2)*f[2](y1,y2)+f1[1](y1,y2)*f1[1](y1,y2)*f[1](y1,y2)+f2[1](y1,y2)*f1[2](y1,y2)*f[1](y1,y2)+f1[1](y1,y2)*f2[1](y1,y2)*f[2](y1,y2)+f2[1](y1,y2)*f2[2](y1,y2)*f[2](y1,y2),y1,y2):
F2[2]:=unapply(f11[2](y1,y2)*f[1](y1,y2)*f[1](y1,y2)+2*f12[2](y1,y2)*f[1](y1,y2)*f[2](y1,y2)+f22[2](y1,y2)*f[2](y1,y2)*f[2](y1,y2)+f1[2](y1,y2)*f1[1](y1,y2)*f[1](y1,y2)+f2[2](y1,y2)*f1[2](y1,y2)*f[1](y1,y2)+f1[2](y1,y2)*f2[1](y1,y2)*f[2](y1,y2)+f2[2](y1,y2)*f2[2](y1,y2)*f[2](y1,y2),y1,y2):
```

ahora $T^{[3]}(y_n, h)$ es

```
> T3[1]:=unapply(F0[1](y1,y2)+(h/2)*F1[1](y1,y2)+(h^2/6)*F2[1](y1,y2),y1,y2):
T3[2]:=unapply(F0[2](y1,y2)+(h/2)*F1[2](y1,y2)+(h^2/6)*F2[2](y1,y2),y1,y2):
```

Funciones que intervienen

```
> print(`1f=` ,f[1](y[1],y[2])):print(`2f=` ,f[2](y[1],y[2])):
```

$$1f =, \frac{-y_1^2 + y_2^4}{y_2}$$

$$2f =, -\frac{y_1}{y_2}$$

```
> print(`1f_1=` ,f1[1](y[1],y[2])):print(`2f_1=` ,f1[2](y[1],y[2])):print(`1f_2=` ,f2[1](y[1],y[2])):print(`1f_1=` ,f2[2](y[1],y[2])):
```

$$1f_1 =, -2 \frac{y_1}{y_2}$$

$$2f_1 =, -\frac{1}{y_2}$$

$$lf_{-2} = 4 y_2 - \frac{2(-y_1^2 + y_2^4)}{y_2^3}$$

$$lf_{-1} = \frac{y_1}{y_2}$$

```
> print(`1f_11=` , f11[1](y[1],y[2])):print(`2f_11=` , f11[2](y[1],y[2])):print(`1f_12=` , f12
[1](y[1],y[2])):print(`2f_12=` , f12[2](y[1],y[2])):print(`1f_22=` , f22[1](y[1],y[2])):pr
int(`2f_22=` , f22[2](y[1],y[2])):
```

$$lf_{11} = -2 \frac{1}{y_2}$$

$$2f_{11} = 0$$

$$lf_{12} = 4 \frac{y_1}{y_2^3}$$

$$2f_{12} = \frac{1}{y_2}$$

$$lf_{22} = -4 + \frac{6(-y_1^2 + y_2^4)}{y_2^4}$$

$$2f_{22} = -2 \frac{y_1}{y_2^3}$$

Estamos en condiciones de comenzar con el método

```
> x0:=0;y0:=[0.,1.];h:=0.1;
```

```

x0 := 0
y0 := [0., 1.]
h := 0.1
```

Primer paso

```
> x1:=x0+h;
```

```
x1 := 0.1
```

```
> print(`1f=` , f[1](y0[1],y0[2])):print(`2f=` , f[2](y0[1],y0[2])):
```

```
1f =, 1.00
```

```
2f =, -0.
```

```
> print(`1f_1=` , f1[1](y0[1],y0[2])):print(`2f_1=` , f1[2](y0[1],y0[2])):print(`1f_2=` , f2[1]
(y0[1],y0[2])):print(`1f_1=` , f2[2](y0[1],y0[2])):
```

```
1f_1 =, -0.
```

```
2f_1 =, -1.
```

```
1f_2 =, 2.00
```

```
1f_1 =, 0.
```

```
> print(`1f_11=` , f11[1](y0[1],y0[2])):print(`2f_11=` , f11[2](y0[1],y0[2])):print(`1f_12=`
 , f12[1](y0[1],y0[2])):print(`2f_12=` , f12[2](y0[1],y0[2])):print(`1f_22=` , f22[1](y0[1],
y0[2])):print(`2f_22=` , f22[2](y0[1],y0[2])):
```

```
1f_11 =, -2.
```

```
2f_11 =, 0
```

```
1f_12 =, 0.
```

```
2f_12 =, 1.
```

```
1f_22 =, 2.00
```

```
2f_22 =, -0.
```

```
> print(`1f[0]=` , F0[1](y0[1],y0[2])):print(`2f[0]=` , F0[2](y0[1],y0[2])):
```

```
1f[0] =, 1.00
```

```
2f[0] =, -0.
```

```
> print(`1f[1]=` , F1[1](y0[1],y0[2])):print(`2f[1]=` , F1[2](y0[1],y0[2])):
```

```
1f[1] =, -0.
```

```

[
  > print(`1f[2]=`,F2[1](y0[1],y0[2]));print(`2f[2]=`,F2[2](y0[1],y0[2])):
  2f[1]=, -1.00
  1f[2]=, -4.00
  2f[2]=, -0.
[
  > T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);
  0.993
  -0.0500
[
  > y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);
  yI1 := 0.0993
  yI2 := 0.995
Segundo paso
[
  > y0:=y1;x1:=x1+h;
  y0 := yI
  x1 := 0.2
[
  > T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);
  0.953
  -0.150
[
  > y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);
  yI1 := 0.195
  yI2 := 0.970
Tercer paso
[
  > y0:=y1;x1:=x1+h;
  y0 := yI
  x1 := 0.3
[
  > T3[1](y0[1],y0[2]);T3[2](y0[1],y0[2]);
  0.855
  -0.250
[
  > y1[1]:=y0[1]+h*T3[1](y0[1],y0[2]);y1[2]:=y0[2]+h*T3[2](y0[1],y0[2]);
  yI1 := 0.280
  yI2 := 0.936
[
  > yv[1](x1)-y1[1];yv[2](x1)-y1[2];
  0.003
  0.019

```

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-03

Para el problema de segundo orden

$$y'' + y = 0$$

$$y(0) = 0, y'(0) = 1$$

y convertido en el problema tipo autónomo de Cauchy

$$y'_1 = y_2$$

$$y'_2 = -y_1$$

$$y_1(0) = 0, y_2(0) = 1$$

(de solución exacta $y_1(x) = \sin(x)$, $y_2(x) = \cos(x)$, o sea $y(x) = \sin(x)$) utilícese el método de

RUNGE-KUTTA del tablero

0					
1/2		1/2			
1/2		0	1/2		
1		0	0	1	

		1/6	1/3	1/3	1/6

para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=3:
[ El sistema (aut.) y la verdadera solución
[ > f:=unapply([y[2],-y[1]],y):
[ > yv[1]:=unapply(sin(x),x):
[   yv[2]:=unapply(cos(x),x):
[ RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. aut.)
[ > Rkut4:=proc(f::procedure,yini::list,nit::posint,h::numeric)
[   local k,k1,k2,k3,k4:
[   global y0:
[   y0:=yini:
[   for k from 1 to nit do
[   k1:=f(y0):
[   k2:=f(y0+(1./2.)*h*k1):
[   k3:=f(y0+(1./2.)*h*k2):
[   k4:=f(y0+h*k3):
[   y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
[   od:
```

```
end:
```

```
Estamos en condiciones de comenzar con el método
```

```
> x0:=0;y0:=[0.,1.];h:=0.1;
```

```
      x0 := 0
      y0 := [0., 1.]
      h := 0.1
```

```
Primer paso
```

```
> x1:=x0+h;
```

```
      x1 := 0.1
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 1.
      2k1=, -0.
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 1.
      2k2=, -0.0500
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.998
      2k3=, -0.0500
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
      1k4=, 0.995
      2k4=, -0.0998
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
      y1 := [0.0998, 0.995]
```

```
Segundo paso
```

```
> y0:=y1;x1:=x1+h;
```

```
      y0 := [0.0998, 0.995]
      x1 := 0.2
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.995
      2k1=, -0.0998
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.990
      2k2=, -0.150
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.988
      2k3=, -0.149
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
      1k4=, 0.980
      2k4=, -0.199
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
      y1 := [0.199, 0.980]
```

```
Tercer paso
```

```
> y0:=y1;x1:=x1+h;
```

```
      y0 := [0.199, 0.980]
      x1 := 0.3
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.980
      2k1=, -0.199
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.970
      2k2=, -0.248
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```

|                                     1k3=, 0.968
|                                     2k3=, -0.248
| > k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
|                                     1k4=, 0.955
|                                     2k4=, -0.296
| > y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
|                                     y1 := [0.296, 0.955]
| > yv[1](x1)-y1[1];yv[2](x1)-y1[2];
|                                     0.
|                                     0.

```

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-04

Para el problema tipo autónomo de Cauchy

$$y'_1 = \frac{-y_1^2 + y_2^4}{y_2^2}$$

$$y'_2 = -\frac{y_1}{y_2}$$

$$y_1(0) = 0, y_2(0) = 1$$

(de solución exacta $y_1(x) = \sin(x) \cos(x)$, $y_2(x) = \cos(x)$) utilícese el método de

RUNGE-KUTTA del tablero

0					
1/2		1/2			
1/2		0	1/2		
1		0	0	1	

		1/6	1/3	1/3	1/6

para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=3:
[ El sistema (aut.) y la verdadera solución
[ > f:=unapply([(-y[1]^2+y[2]^4)/y[2]^2,-y[1]/y[2]],y):
[ > yv[1]:=unapply(sin(x)*cos(x),x):
[ > yv[2]:=unapply(cos(x),x):
[ RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. aut.)
[ > Rkut4:=proc(f::procedure,yini::list,nit::posint,h::numeric)
[ local k,k1,k2,k3,k4:
[ global y0:
[ y0:=yini:
[ for k from 1 to nit do
[ k1:=f(y0):
[ k2:=f(y0+(1./2.)*h*k1):
[ k3:=f(y0+(1./2.)*h*k2):
[ k4:=f(y0+h*k3):
[ y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
[ od:
[ end:
```

Estamos en condiciones de comenzar con el método

```
> x0:=0;y0:=[0.,1.];h:=0.1;
```

```
      x0 := 0
      y0 := [0., 1.]
      h := 0.1
```

Primer paso

```
> x1:=x0+h;
```

```
      x1 := 0.1
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 1.00
      2k1=, -0.
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.998
      2k2=, -0.0500
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.994
      2k3=, -0.0500
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
      1k4=, 0.980
      2k4=, -0.0999
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
      y1 := [0.0994, 0.995]
```

Segundo paso

```
> y0:=y1;x1:=x1+h;
```

```
      y0 := [0.0994, 0.995]
```

```
      x1 := 0.2
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.980
      2k1=, -0.0999
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.958
      2k2=, -0.149
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.954
      2k3=, -0.149
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
      1k4=, 0.921
      2k4=, -0.199
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
      y1 := [0.195, 0.980]
```

Tercer paso

```
> y0:=y1;x1:=x1+h;
```

```
      y0 := [0.195, 0.980]
```

```
      x1 := 0.3
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.921
      2k1=, -0.199
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.879
      2k2=, -0.248
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.876
      2k3=, -0.247
```

```
[ > k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):  
                                1k4=, 0.825  
                                2k4=, -0.296  
[ > y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);  
                                y1 := [0.283, 0.955]  
[ > yv[1](x1)-y1[1];yv[2](x1)-y1[2];  
                                0.  
                                0.
```

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-05

Para el problema tipo no autónomo

$$y'_1 = y_1 + 2y_2 + x$$

$$y'_2 = y_1 - x$$

$$y_1(0) = 0, y_2(0) = 0$$

(de solución exacta $y_1(x) = x - 1 + e^{(-x)}$, $y_2(x) = -x + 1 - e^{(-x)}$) utilícese el método de

RUNGE-KUTTA del tablero

0					
1/2		1/2			
1/2		0	1/2		
1		0	0	1	

		1/6	1/3	1/3	1/6

para aproximar la solución en $x = 0.3$ tomando amplitud de paso $h = 0.1$

Luego se convertirá el sistema en autónomo, pero de tamaño 3, y se repetirán los cálculos, comparando los datos obtenidos.

Realícense las operaciones con 3 cifras significativas y redondeo

```
[ > restart:
[ > simplify(dsolve({diff(y1(x),x)=y1(x)+2*y2(x)+x,diff(y2(x),x)=y1(x)-x,y1(0)=0,y2(0)=0},
[ {y1(x),y2(x)})));
[ {y2(x) = 1 - e(-x) - x, y1(x) = -1 + e(-x) + x}
[ > simplify(dsolve({diff(y1(x),x)=y1(x)+2*y2(x)+y3(x),diff(y2(x),x)=y1(x)-y3(x),diff(y3(x)
[ ),x)=1,y1(0)=0,y2(0)=0,y3(0)=0},{y1(x),y2(x),y3(x)})));
[ {y2(x) = 1 - e(-x) - x, y1(x) = -1 + e(-x) + x, y3(x) = x}
```

Caso no autónomo

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=3:
[ El sistema (no aut.) y la verdadera solución
[ > f:=unapply([y[1]+2*y[2]+x,y[1]-x],x,y):
[ > yv[1]:=unapply(x-1+exp(-x),x):
[ yv[2]:=unapply(-x+1-exp(-x),x):
```

RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. no aut.)

```
[ > Rkut4:=proc(f::procedure,xini::numeric,yini::list,nit::posint,h::numeric)
```

```

local k,k1,k2,k3,k4;
global x0,y0;
x0:=xini;
y0:=yini;
for k from 1 to nit do
k1:=f(x0,y0);
k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1);
k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2);
k4:=f(x0+h,y0+h*k3);
x0:=x0+h;
y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
[x0,y0];
od:
end:

```

```
> xini:=0.;yini:=[0.,0.];solu:=Rkut4(f,xini,yini,10,0.03);
```

```
      xini := 0.
```

```
      yini := [0., 0.]
```

```
      solu := [0.30, [0.0409, -0.0409]]
```

```
> [solu[2,1],solu[2,2]];[yv[1](solu[1]),yv[2](solu[1])];[solu[2,1]-yv[1](solu[1]),solu[2,2]-yv[2](solu[1])];
```

```
      [0.0409, -0.0409]
```

```
      [0.041, -0.041]
```

```
      [-0.0001, 0.0001]
```

Estamos en condiciones de comenzar con el método

```
> x0:=0;y0:=[0.,0.];h:=0.1;
```

```
      x0 := 0
```

```
      y0 := [0., 0.]
```

```
      h := 0.1
```

Primer paso

```
> x1:=x0+h;
```

```
      x1 := 0.1
```

```
> k1:=f(x0,y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.
```

```
      2k1=, 0.
```

```
> k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.0500
```

```
      2k2=, -0.0500
```

```
> k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
      1k3=, 0.0475
```

```
      2k3=, -0.0475
```

```
> k4:=f(x0+h,y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
      1k4=, 0.0952
```

```
      2k4=, -0.0952
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
      y1 := [0.00483, -0.00483]
```

Segundo paso

```
> x0:=x1;y0:=y1;x1:=x1+h;
```

```
      y0 := [0.00483, -0.00483]
```

```
      x1 := 0.2
```

```
> k1:=f(x0,y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
      1k1=, 0.0952
```

```
      2k1=, -0.0952
```

```
> k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
      1k2=, 0.140
```

```
      2k2=, -0.140
```

```

> k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
      Ik3=, 0.138
      2k3=, -0.138
> k4:=f(x0+h,y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
      Ik4=, 0.181
      2k4=, -0.181
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
      y1 := [0.0186, -0.0186]

```

Tercer paso

```

> x0:=x1:y0:=y1;x1:=x1+h;
      y0 := [0.0186, -0.0186]
      x1 := 0.3
> k1:=f(x0,y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
      Ik1=, 0.181
      2k1=, -0.181
> k2:=f(x0+(1./2.)*h,y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
      Ik2=, 0.222
      2k2=, -0.222
> k3:=f(x0+(1./2.)*h,y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
      Ik3=, 0.220
      2k3=, -0.220
> k4:=f(x0+h,y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
      Ik4=, 0.259
      2k4=, -0.259
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
      y1 := [0.0406, -0.0406]

```

Al cabo de los tres pasos

```

> [y1[1],y1[2]];[yv[1](x1),yv[2](x1)];[y1[1]-yv[1](x1),y1[2]-yv[2](x1)];
      [0.0406, -0.0406]
      [0.041, -0.041]
      [-0.0004, 0.0004]

```

Caso autónomo

```

> restart:
> interface(labeling=false):
> Digits:=3:

```

El sistema (aut.) y la verdadera solución

```

> f:=unapply([y[1]+2*y[2]+y[3],y[1]-y[3],1],y):
> yv[1]:=unapply(x-1+exp(-x),x):
> yv[2]:=unapply(-x+1-exp(-x),x):
> yv[3]:=unapply(x,x):

```

RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. aut.)

```

> Rkut4:=proc(f::procedure,yini::list,nit::posint,h::numeric)
  local k,k1,k2,k3,k4:
  global y0:
  y0:=yini:
  for k from 1 to nit do
    k1:=f(y0):
    k2:=f(y0+(1./2.)*h*k1):
    k3:=f(y0+(1./2.)*h*k2):
    k4:=f(y0+h*k3):
    y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
  od:
end:
> yini:=[0.,0.,0.];solu:=Rkut4(f,yini,10,0.03);

```

```

                                yini := [0., 0., 0.]
                                solu := [0.0409, -0.0409, 0.300]
> [solu[1],solu[2],solu[3]];[yv[1](solu[3]),yv[2](solu[3]),yv[3](solu[3])];[solu[1]-yv[1]
  ](solu[3]),solu[2]-yv[2](solu[3]),solu[3]-yv[3](solu[3])];
                                [0.0409, -0.0409, 0.300]
                                [0.041, -0.041, 0.300]
                                [-0.0001, 0.0001, 0.]

```

Estamos en condiciones de comenzar con el método

```
> x0:=0;y0:=[0.,0.,0.];h:=0.1;
```

```

                                x0 := 0
                                y0 := [0., 0., 0.]
                                h := 0.1

```

Primer paso

```
> x1:=x0+h;
```

```
                                x1 := 0.1
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
                                1k1=, 0.
```

```
                                2k1=, 0.
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
                                1k2=, 0.0500
```

```
                                2k2=, -0.0500
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
                                1k3=, 0.0475
```

```
                                2k3=, -0.0475
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
                                1k4=, 0.0952
```

```
                                2k4=, -0.0952
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
                                y1 := [0.00483, -0.00483, 0.100]
```

Segundo paso

```
> y0:=y1;x1:=x1+h;
```

```
                                y0 := [0.00483, -0.00483, 0.100]
```

```
                                x1 := 0.2
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```

```
                                1k1=, 0.0952
```

```
                                2k1=, -0.0952
```

```
> k2:=f(y0+(1./2.)*h*k1):print(`1k2=`,k2[1]):print(`2k2=`,k2[2]):
```

```
                                1k2=, 0.140
```

```
                                2k2=, -0.140
```

```
> k3:=f(y0+(1./2.)*h*k2):print(`1k3=`,k3[1]):print(`2k3=`,k3[2]):
```

```
                                1k3=, 0.138
```

```
                                2k3=, -0.138
```

```
> k4:=f(y0+h*k3):print(`1k4=`,k4[1]):print(`2k4=`,k4[2]):
```

```
                                1k4=, 0.181
```

```
                                2k4=, -0.181
```

```
> y1:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4);
```

```
                                y1 := [0.0186, -0.0186, 0.200]
```

Tercer paso

```
> y0:=y1;x1:=x1+h;
```

```
                                y0 := [0.0186, -0.0186, 0.200]
```

```
                                x1 := 0.3
```

```
> k1:=f(y0):print(`1k1=`,k1[1]):print(`2k1=`,k1[2]):
```


Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-06

Para un problema (en este caso de Inteligencia Artificial) se diseñan dos métodos numéricos de integración de ecuaciones diferenciales que designaremos con los nombres de M1 y M2 .

Para determinar cual es el más adecuado, se realiza la integración de un problema test entre 0 y 1 y se emplean los pasos

$$1, \frac{1}{2} = 0.5, \frac{1}{2^2} = 0.25, \frac{1}{2^3} = 0.125, \frac{1}{2^4} = 0.0625, \frac{1}{2^5} = 0.0313, \frac{1}{2^6} = 0.0156, \frac{1}{2^7} = 0.0078, \frac{1}{2^8} = 0.0039$$

Los números de pasos son, en todos los casos,

1, 2, 4, 8, 16, 32, 64, 128, 256

Los números de evaluaciones son para el M1

2, 4, 8, 16, 32, 64, 128, 256, 512

y para el M2

3, 6, 12, 24, 48, 96, 192, 384, 768

Para cada método se calcula el \log_{10} del error cometido en la aproximación de la solución en $x=1$, error medido en norma2. Los errores que se obtienen son, para el M1

1.716, 1.912, .8447, -2.149, -3.849, -4.701, -5.381, -6.014, -6.630

y para el M2

2.180, 2.202, .04186, -4.047, -4.878, -5.861, -6.814, -7.743, -8.659

Con estos datos, se deben construir las siguientes 'gráficas de eficiencia' que mezclen los resultados de M1 y M2 :

- La gráfica 'paso versus \log_{10} del error'.
- La gráfica 'número de evaluaciones versus \log_{10} del error'.
- Finalmente, se dirá qué método parece más adecuado.

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listanumpasos:=[0,0,0,0,0,0,0,0,0,0]:
[
```

```

[ > listapaso:=[0,0,0,0,0,0,0,0,0]:
[ > listaeval:=[0,0,0,0,0,0,0,0,0]:
[ > lista1loggerro:=[0,0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0,0]:
[ > lista2loggerro:=[0,0,0,0,0,0,0,0,0]:
[ El sistema (aut.)
[ > listanumpasos:=[1 , 2 , 4 , 8 , 16 , 32 , 64 , 128 , 256]:
[ > for i from 1 to 9 do listapaso[i]=1./listanumpasos[i]; od:
[ > f:=[x->-10*x[1]+x[2],x->-10*x[2]]:yini:=[1.,1.]:
[ > u:=x->[exp(-10*x)+x*exp(-10*x),exp(-10*x)]:
[ M1 -> Runge-Kutta 2 evaluaciones y orden 2 'Heun' (s. aut.)
[ > for i from 1 to 9 do listaeval[i]:=2*listanumpasos[i]; od:
[ > Heun2:=proc(f::list,yini::list,nit::posint,h::numeric)
  local k,k1,k2:
  global y0:
  y0:=yini:
  for k from 1 to nit do
  k1:=f(y0):
  k2:=f(y0+(2./3.)*h*k1):
  y0:=y0+h*((1./4.)*k1+(3./4.)*k2):
  od:
  end:
[ > for i from 1 to 9 do y1:=Heun2(f,yini,listanumpasos[i],evalf(listapaso[i])):
  lista1loggerro[i]:=log[10](linalg[norm](y1-u(1.),2)): od:
[ > evalf(lista1loggerro,4);
[
[1.716, 1.912, 0.8447, -2.149, -3.849, -4.701, -5.381, -6.014, -6.630]

```

```

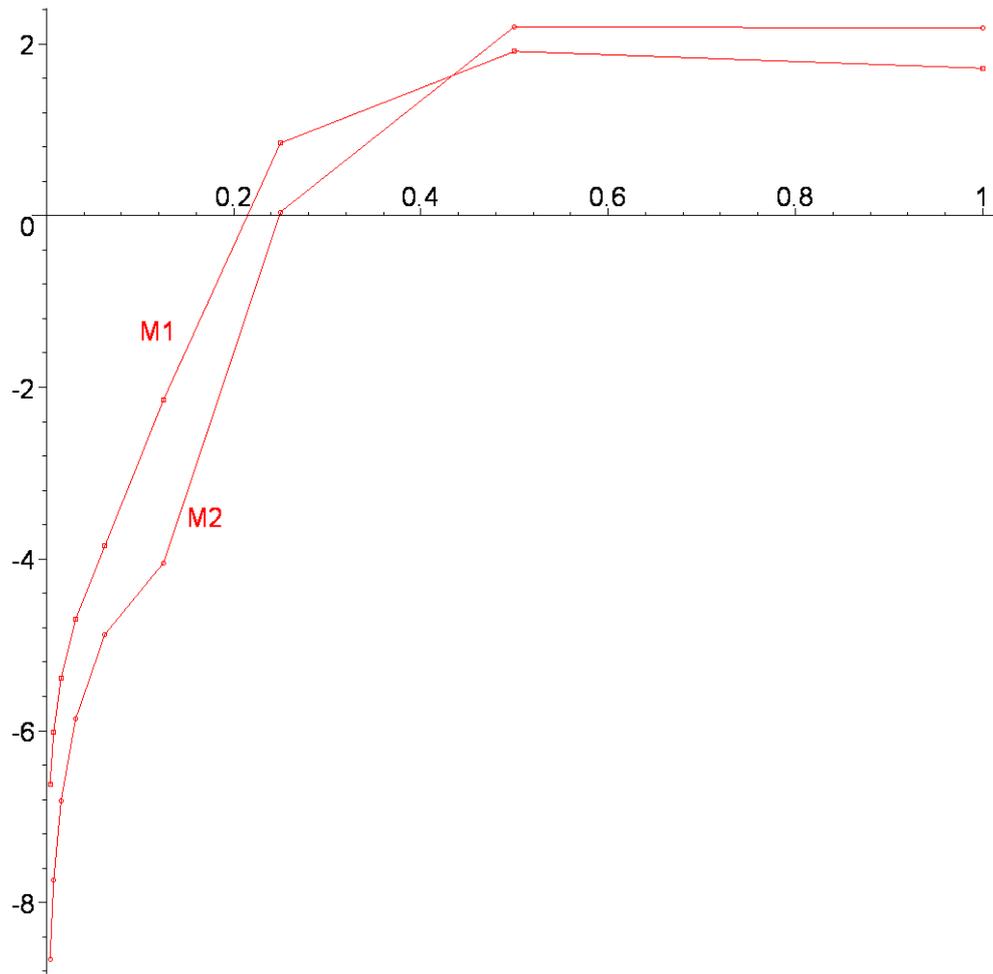
[ M2 -> Runge-Kutta 3 evaluaciones y orden 3 'Heun' (s. aut.)
[ > for i from 1 to 9 do lista2eval[i]:=3*listanumpasos[i]; od:
[ > Heun3:=proc(f::list,yini::list,nit::posint,h::numeric)
  local k,k1,k2,k3:
  global y0:
  y0:=yini:
  for k from 1 to nit do
  k1:=f(y0):
  k2:=f(y0+(1./3.)*h*k1):
  k3:=f(y0+(2./3.)*h*k2):
  y0:=y0+h*((1./4.)*k1+(3./4.)*k3):
  od:
  end:
[ > for i from 1 to 9 do y1:=Heun3(f,yini,listanumpasos[i],evalf(listapaso[i])):
  lista2loggerro[i]:=log[10](linalg[norm](y1-u(1.),2)): od:
[ > evalf(lista2loggerro,4);
[
[2.180, 2.202, 0.04186, -4.047, -4.878, -5.861, -6.814, -7.743, -8.659]

```

```

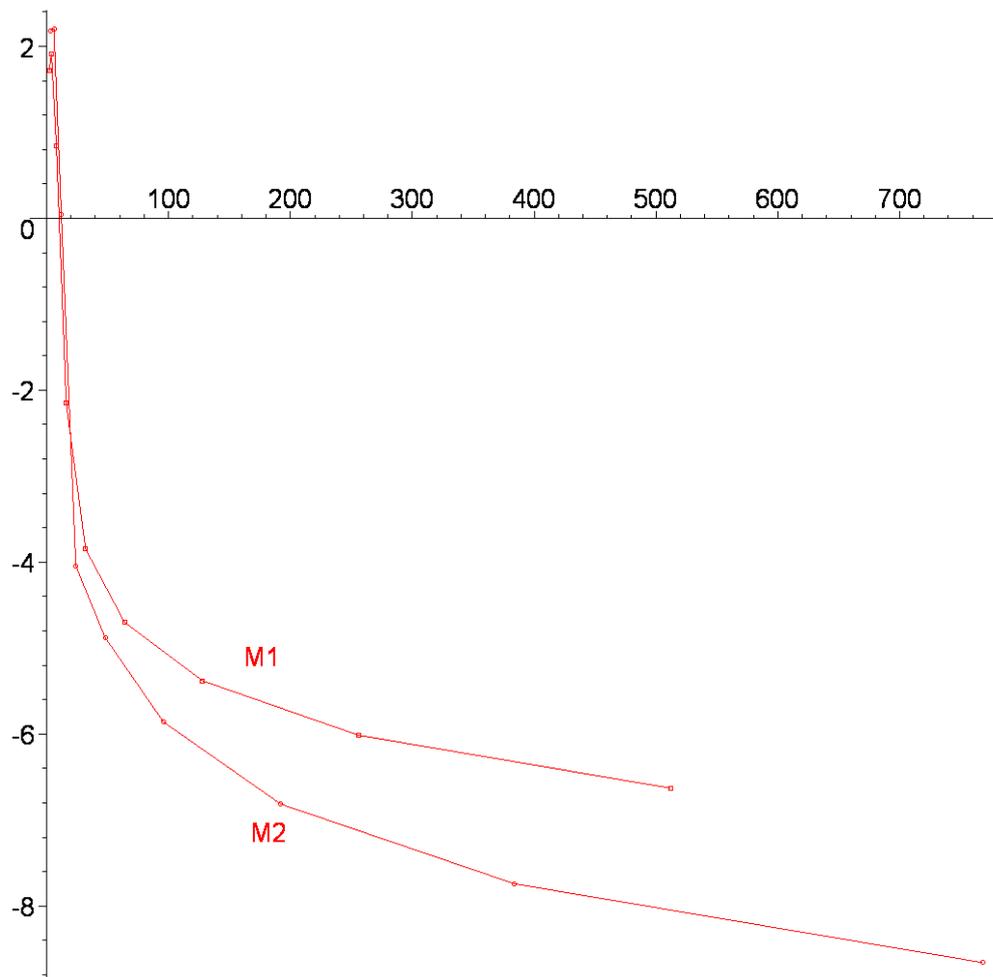
[ a) Gráficas comparadas 'paso versus log[10] del error'
[ > listagraa1:=[seq([listapaso[i],lista1loggerro[i]],i=1..9)]:
[ > graa11:=plot(listagraa1,style=LINE):
[ > graa12:=plot(listagraa1,style=POINT,symbol=BOX):
[ > graa13:=textplot([0.1186, -1.335, `M1`]):
[ > listagraa2:=[seq([listapaso[i],lista2loggerro[i]],i=1..9)]:
[ > graa21:=plot(listagraa2,style=LINE):
[ > graa22:=plot(listagraa2,style=POINT,symbol=CIRCLE):
[ > graa23:=textplot([0.1693, -3.503, `M2`]):
[ > display(graa11,graa12,graa13,graa21,graa22,graa23);

```



b) Gráficas comparadas 'número de evaluaciones versus \log_{10} del error'

```
[ > listagrab1:=seq([lista1eval[i],lista1logerro[i]],i=1..9]):
[ > grab11:=plot(listagrab1,style=LINE):
[ > grab12:=plot(listagrab1,style=POINT,symbol=BOX):
[ > grab13:=textplot([176.5, -5.081,`M1`]):
[ > listagrab2:=seq([lista2eval[i],lista2logerro[i]],i=1..9]):
[ > grab21:=plot(listagrab2,style=LINE):
[ > grab22:=plot(listagrab2,style=POINT,symbol=CIRCLE):
[ > grab23:=textplot([182, -7.132,`M2`]):
[ > display(grab11,grab12,grab13,grab21,grab22,grab23);
```



c) Método más adecuado

El método M2 funciona mejor en cuanto el paso se hace adecuadamente pequeño, lo que corresponde al mayor número de evaluaciones de la función.

Ejercicios de

(5) Sistemas autónomos: métodos de RUNGE-KUTTA

Ejercicio 05-07

El problema de dos cuerpos está finalmente gobernado por el siguiente sistema de primer orden

$$y'_1 = y_3$$

$$y'_2 = y_4$$

$$y'_3 = -\frac{y_1}{\sqrt{y_1^2 + y_2^2}^3}$$

$$y'_4 = -\frac{y_2}{\sqrt{y_1^2 + y_2^2}^3}$$

donde, como es habitual, (y_1, y_2) representa la posición de la partícula y (y_3, y_4) representa su velocidad. Para las condiciones iniciales

$$y_1(0) = 1, y_2(0) = 0, y_3(0) = 0, y_4(0) = 1$$

la solución exacta, que es la que se busca numéricamente, es una órbita circular de período 2π ,

exactamente la órbita $(\cos x, \sin x, -\sin x, \cos x)$. Se integra numéricamente en el

intervalo $[0, 4\pi]$ (dos vueltas) con los métodos RK2, RK3 y RK4, que son los siguientes métodos de Runge-Kutta de 2, 3 y 4 evaluaciones, y órdenes 2, 3 y 4:

- el método de Heun de dos evaluaciones, con tablero

$$\begin{array}{c|c} 0 & \\ \hline 2/3 & 2/3 \\ \hline & \\ \hline & | 1/4 \quad 3/4 \end{array}$$

- el método de Heun de tres evaluaciones, con tablero

$$\begin{array}{c|c} 0 & \\ \hline 1/3 & 1/3 \\ \hline 2/3 & 0 \quad 2/3 \\ \hline & \\ \hline & | 1/4 \quad 0 \quad 3/4 \end{array}$$

- el método clásico de Runge-Kutta de cuatro evaluaciones, con tablero

$$\begin{array}{c|c} 0 & \\ \hline 1/2 & 1/2 \end{array}$$

$$1/2 \mid 0 \ 1/2$$

$$1 \mid 0 \ 0 \ 1$$

$$\mid 1/6 \ 1/3 \ 1/3 \ 1/6$$

Las listas de 8 datos que se consideran corresponden a los ocho pasos

$$\frac{\pi}{2^2} = 0.7854, \frac{\pi}{2^3} = 0.3827, \frac{\pi}{2^4} = 0.1963, \frac{\pi}{2^5} = 0.0982, \frac{\pi}{2^6} = 0.0491, \frac{\pi}{2^7} = 0.0245, \frac{\pi}{2^8} = 0.0123, \frac{\pi}{2^9} = 0.0061$$

Los números de pasos son, en todos los casos,

16, 32, 64, 128, 256, 512, 1024, 2048

Los números de evaluaciones son para el RK2

32, 64, 128, 256, 512, 1024, 2048, 4096

para el RK3

48, 96, 192, 384, 768, 1536, 3072, 6144

y para el RK4

64, 128, 256, 512, 1024, 2048, 4096, 8192

Para cada método se calcula el \log_{10} del error cometido en la aproximación de

$$(y_1(4\pi), y_2(4\pi), y_3(4\pi), y_4(4\pi))$$

error medido en norma2. Como podemos ver, para nuestros métodos, vamos

obteniendo los siguientes logaritmos de los errores; para el RK2

0.4343, 0.3456, -0.1695, -0.7862, -1.412, -2.030, -2.641, -3.248

para el RK3

-0.008691, -1.459, -3.037, -4.912, -6.062, -6.770, -6.782, -7.7859

y para el RK4

0.3978, -1.267, -2.707, -4.096, -5.445, -6.973, -7.007, -8.084

Con estos datos, se deben construir las siguientes 'gráficas de eficiencia' que mezcles los resultados de RK2, RK3 y RK4 :

- La gráfica 'paso versus \log_{10} del error'.
- La gráfica 'número de evaluaciones versus \log_{10} del error'.
- La gráfica ' \log_{10} del paso versus \log_{10} del error'. En ésta se puede comprobar la tendencia a dar rectas con pendiente igual en módulo al orden, usando bien la recta de regresión de los datos, o mejor la pendiente de alguno de los últimos segmentos que componen la recta.
- La gráfica ' \log_{10} del número de evaluaciones versus \log_{10} del error'. Convendrá establecer las mismas comprobaciones que en el apartado precedente.

```
[ > restart:
[ > interface(labeling=false):
[ > Digits:=20:
[ > with(stats):with(linalg):with(plots):
Warning, the protected names norm and trace have been redefined and unprotected

Warning, the name changecoords has been redefined
```

[Variables lista

```
[ > listanumpasos:=[0,0,0,0,0,0,0,0]:
[ > listpaso:=[0,0,0,0,0,0,0,0]:
[ > listalogpaso:=[0,0,0,0,0,0,0,0]:
[ > lista2eval:=[0,0,0,0,0,0,0,0]:
[ > lista2logeval:=[0,0,0,0,0,0,0,0]:
[ > lista2logerro:=[0,0,0,0,0,0,0,0]:
[ > lista3eval:=[0,0,0,0,0,0,0,0]:
[ > lista3logeval:=[0,0,0,0,0,0,0,0]:
[ > lista3logerro:=[0,0,0,0,0,0,0,0]:
[ > lista4eval:=[0,0,0,0,0,0,0,0]:
[ > lista4logeval:=[0,0,0,0,0,0,0,0]:
[ > lista4logerro:=[0,0,0,0,0,0,0,0]:
```

[El sistema (aut.)

```
[ > listanumpasos:=[16,32,64,128,256,512,1024,2048]:
[ > for i from 1 to 8 do listapaso[i]:=4*Pi/listanumpasos[i]; od:
[ > for i from 1 to 8 do listalogpaso[i]:=evalf(log[10](listapaso[i])); od:
[ > f:=[x->x[3],x->x[4],x->-(x[1]/sqrt(x[1]^2+x[2]^2)^3),x->-(x[2]/sqrt(x[1]^2+x[2]^2)^3)]
: yini:=[1.,0.,0.,1.]:
[ > u:=x->[cos(x),sin(x),-sin(x),cos(x)]:
```

[RK2 -> Runge-Kutta 2 evaluaciones y orden 2 'Heun' (s. aut.)

```
[ > for i from 1 to 8 do lista2eval[i]:=2*listanumpasos[i]; od:
[ > for i from 1 to 8 do lista2logeval[i]:=evalf(log[10](lista2eval[i])); od:
[ > Heun2:=proc(f::list,yini::list,nit::posint,h::numeric)
local k,k1,k2:
global y0:
y0:=yini:
for k from 1 to nit do
k1:=f(y0):
k2:=f(y0+(2./3.)*h*k1):
y0:=y0+h*((1./4.)*k1+(3./4.)*k2):
od:
end:
[ > for i from 1 to 8 do y1:=Heun2(f,yini,listanumpasos[i],evalf(listapaso[i])):
lista2logerro[i]:=log[10](linalg[norm](y1-u(4*Pi),2)): od:
```

[RK3 -> Runge-Kutta 3 evaluaciones y orden 3 'Heun' (s. aut.)

```
[ > for i from 1 to 8 do lista3eval[i]:=3*listanumpasos[i]; od:
[ > for i from 1 to 8 do lista3logeval[i]:=evalf(log[10](lista3eval[i])); od:
[ > Heun3:=proc(f::list,yini::list,nit::posint,h::numeric)
local k,k1,k2,k3:
global y0:
y0:=yini:
for k from 1 to nit do
k1:=f(y0):
k2:=f(y0+(1./3.)*h*k1):
k3:=f(y0+(2./3.)*h*k2):
y0:=y0+h*((1./4.)*k1+(3./4.)*k3):
od:
end:
[ > for i from 1 to 8 do y1:=Heun3(f,yini,listanumpasos[i],evalf(listapaso[i])):
lista3logerro[i]:=log[10](linalg[norm](y1-u(4*Pi),2)): od:
```

[RK4 -> Runge-Kutta 4 evaluaciones y orden 4 clásico (s. aut.)

```
[ > for i from 1 to 8 do lista4eval[i]:=4*listanumpasos[i]; od:
[
```

```

[ > for i from 1 to 8 do lista4logeval[i]:=evalf(log[10](lista4eval[i])); od:
[ > Rkut4:=proc(f::list,yini::list,nit::posint,h::numeric)
  local k,k1,k2,k3,k4:
  global y0:
  y0:=yini:
  for k from 1 to nit do
  k1:=f(y0):
  k2:=f(y0+(1./2.)*h*k1):
  k3:=f(y0+(1./2.)*h*k2):
  k4:=f(y0+h*k3):
  y0:=y0+h*((1./6.)*k1+(1./3.)*k2+(1./3.)*k3+(1./6.)*k4):
  od:
  end:
[ > for i from 1 to 8 do y1:=Rkut4(f,yini,listanumpasos[i],evalf(listapaso[i])):
  lista4logerro[i]:=log[10](linalg[norm](y1-u(4*Pi),2)): od:

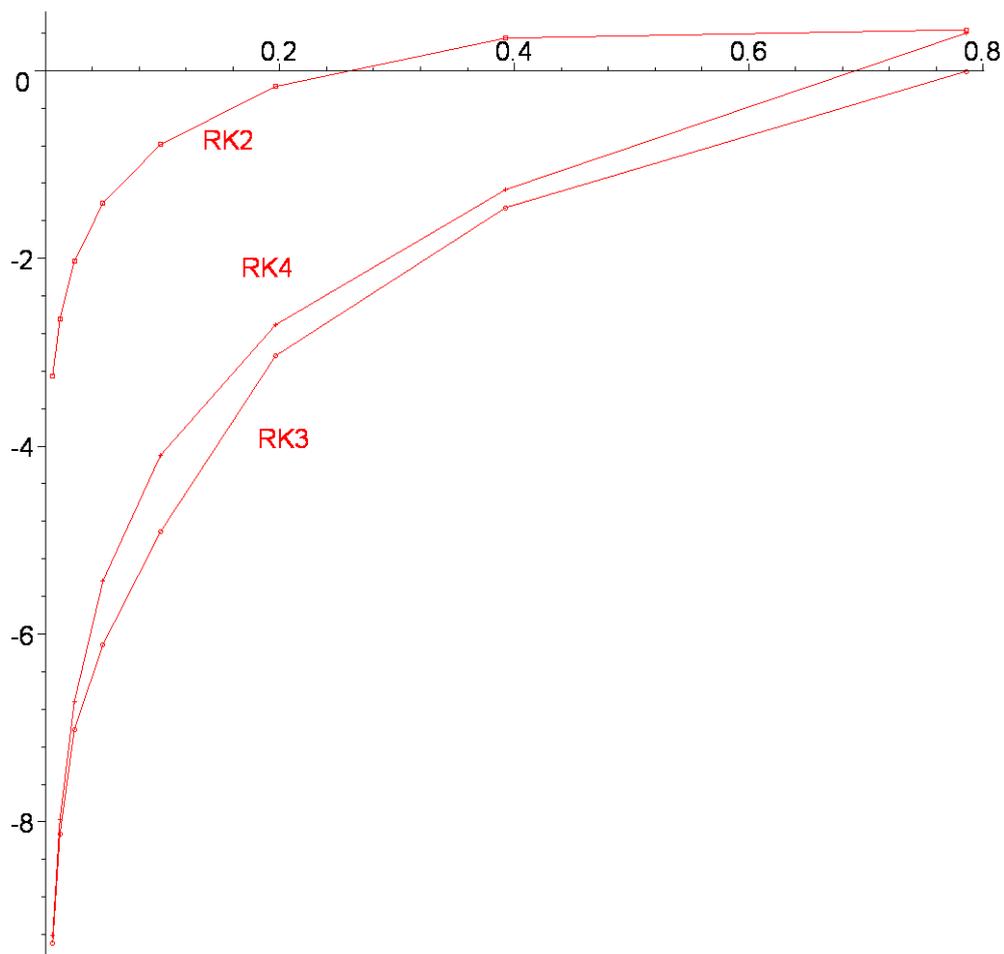
```

a) Gráficas comparadas 'paso versus log[10] del error'

```

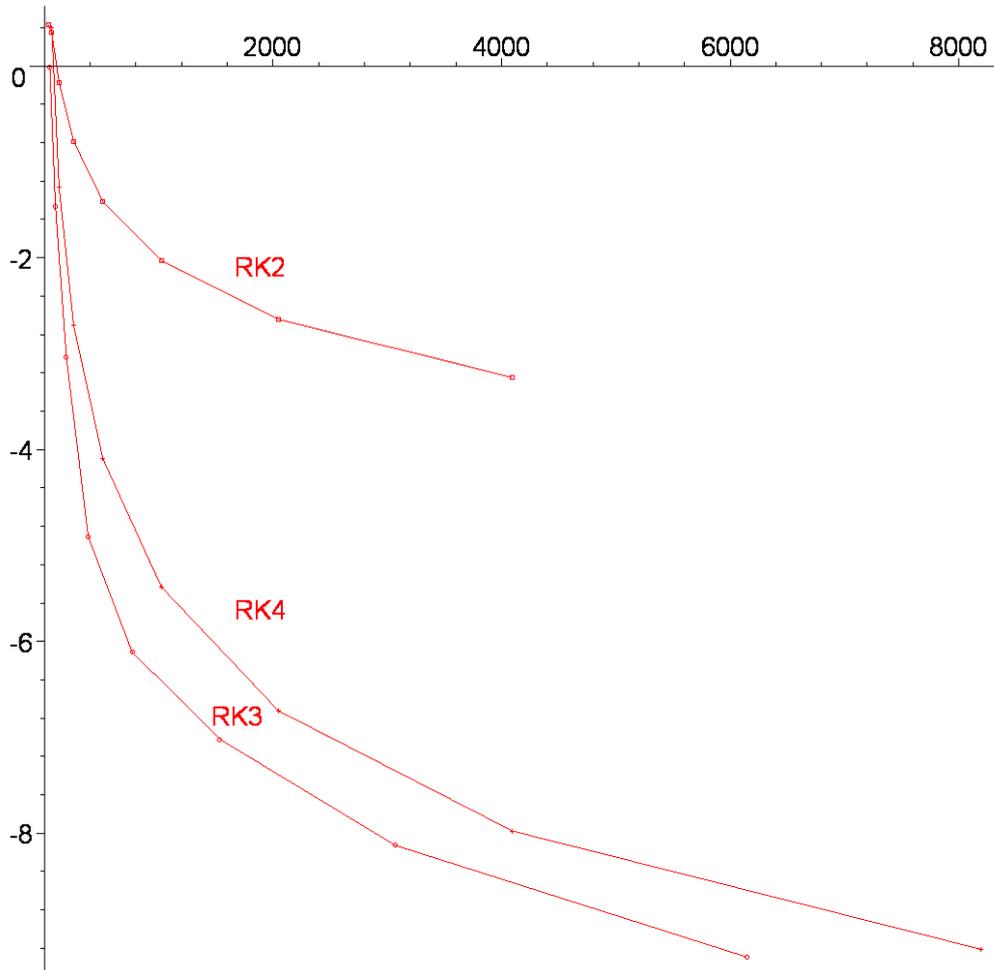
[ > listagraa2:=seq([listapaso[i],lista2logerro[i]],i=1..8):
[ > graa21:=plot(listagraa2,style=LINE):
[ > graa22:=plot(listagraa2,style=POINT,symbol=BOX):
[ > graa23:=textplot([0.1567, -0.7317,`RK2`]):
[ > listagraa3:=seq([listapaso[i],lista3logerro[i]],i=1..8):
[ > graa31:=plot(listagraa3,style=LINE):
[ > graa32:=plot(listagraa3,style=POINT,symbol=CIRCLE):
[ > graa33:=textplot([0.2036, -3.904,`RK3`]):
[ > listagraa4:=seq([listapaso[i],lista4logerro[i]],i=1..8):
[ > graa41:=plot(listagraa4,style=LINE):
[ > graa42:=plot(listagraa4,style=POINT,symbol=CROSS):
[ > graa43:=textplot([0.1892, -2.086,`RK4`]):
[ > display(graa21,graa22,graa23,graa31,graa32,graa33,graa41,graa42,graa43);

```



[b) Gráficas comparadas 'número de evaluaciones versus log[10] del error'

```
[ > listagrab2:=[seq([lista2eval[i],lista2logerro[i]],i=1..8)]:  
[ > grab21:=plot(listagrab2,style=LINE):  
[ > grab22:=plot(listagrab2,style=POINT,symbol=BOX):  
[ > grab23:=textplot([1892, -2.086,`RK2`]):  
[ > listagrab3:=[seq([lista3eval[i],lista3logerro[i]],i=1..8)]:  
[ > grab31:=plot(listagrab3,style=LINE):  
[ > grab32:=plot(listagrab3,style=POINT,symbol=CIRCLE):  
[ > grab33:=textplot([1687, -6.77,`RK3`]):  
[ > listagrab4:=[seq([lista4eval[i],lista4logerro[i]],i=1..8)]:  
[ > grab41:=plot(listagrab4,style=LINE):  
[ > grab42:=plot(listagrab4,style=POINT,symbol=CROSS):  
[ > grab43:=textplot([1890, -5.66,`RK4`]):  
[ > display(grab21,grab22,grab23,grab31,grab32,grab33,grab41,grab42,grab43);
```



[c) Gráficas comparadas 'log[10] del paso versus log[10] del error'

Recordemos lo que ya sabemos de otros problemas:

Hay un tipo de gráficas que sirve para comprobar el orden 'efectivo' de los métodos, o sea, el orden con que, en la práctica, parecen comportarse. En estas gráficas se debe presentar en el eje de abscisas el log[10] del paso empleado y en el de ordenadas el log[10] del error.

Entonces el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). La pendiente de esa recta es el orden con que se comporta el método.

En efecto, si el método es de orden p , el error será $O(h^p)$. Considerando que, aproximadamente

se tiene $E(h) = K h^p$, resulta que el cociente incremental (la pendiente aproximada de la recta) entre dos puntos correspondientes a pasos $h_1 < h_2$ es

$$\frac{\log_{10}(E(h_2)) - \log_{10}(E(h_1))}{\log_{10}(h_2) - \log_{10}(h_1)} = \frac{\log_{10}(K) + p \log_{10}(h_2) - \log_{10}(K) - p \log_{10}(h_1)}{\log_{10}(h_2) - \log_{10}(h_1)} = p$$

Esta pendiente se puede estudiar cuando se utilizan los segmentos correspondientes a los menores valores de h , los mas representativos del efecto del orden.

Algo semejante ocurre cuando se presenta en el eje de abscisas el $\log[10]$ del numero de evaluaciones y en el de ordenadas el $\log[10]$ del error. Entonces también el aspecto de la gráfica se aproxima a una recta (o se puede aproximar utilizando la regresión lineal). Pero la pendiente de esa recta es ahora igual a menos el orden con que se comporta el método, debido a que en el cociente antes expuesto el denominador es ahora

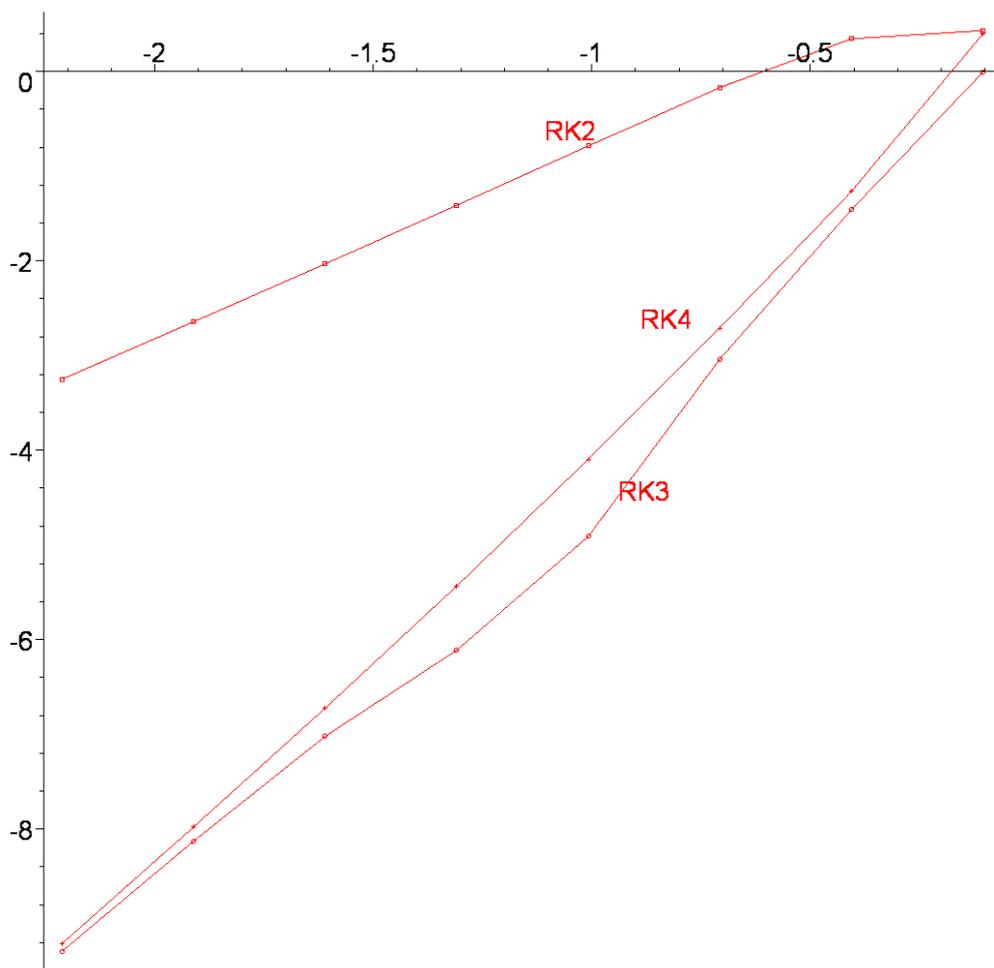
$$\begin{aligned} \log_{10}(\text{eval}(h_2)) - \log_{10}(\text{eval}(h_1)) &= \log_{10}\left(\frac{m}{h_2}\right) - \log_{10}\left(\frac{m}{h_1}\right) = \\ &= \log_{10}(m) - \log_{10}(h_2) - \log_{10}(m) + \log_{10}(h_1) = \log_{10}(h_1) - \log_{10}(h_2) \end{aligned}$$

en el supuesto de que m es el número de evaluaciones por paso del método en cuestión.

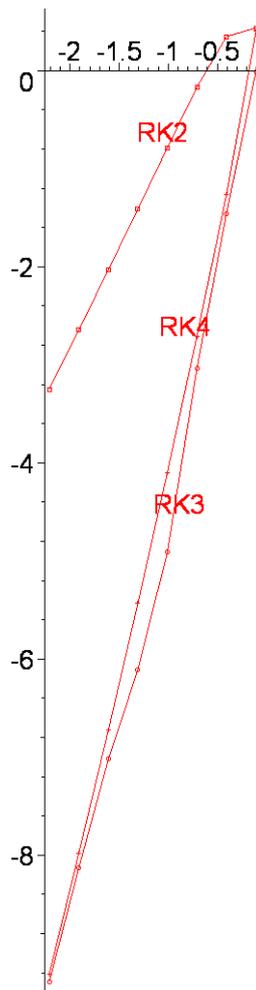
Esta idea se emplea especialmente cuando el método es de paso variable y se desea averiguar su orden efectivo, ya que, entonces, no tiene sentido hablar del paso del método, pero sí que lo tiene hablar de las evaluaciones realizadas por el mismo.

Y esto es justamente lo que sucede en este caso en el que hay que comparar con métodos de paso variable; por ello no emplearemos las gráficas ' $\log[10]$ del paso versus $\log[10]$ del error', que carecen ahora de sentido, sino las gráficas ' $\log[10]$ del número de evaluaciones versus $\log[10]$ del error', que son las que ahora pueden servir para detectar el 'orden efectivo'.

```
[ > listagrac2:=[seq([listalogpaso[i],lista2logerro[i]],i=1..8)]:
[ > grac21:=plot(listagrac2,style=LINE):
[ > grac22:=plot(listagrac2,style=POINT,symbol=BOX):
[ > grac23:=textplot([-1.048, -0.6154,`RK2`]):
[ > listagrac3:=[seq([listalogpaso[i],lista3logerro[i]],i=1..8)]:
[ > grac31:=plot(listagrac3,style=LINE):
[ > grac32:=plot(listagrac3,style=POINT,symbol=CIRCLE):
[ > grac33:=textplot([-0.8809, -4.416,`RK3`]):
[ > listagrac4:=[seq([listalogpaso[i],lista4logerro[i]],i=1..8)]:
[ > grac41:=plot(listagrac4,style=LINE):
[ > grac42:=plot(listagrac4,style=POINT,symbol=CROSS):
[ > grac43:=textplot([-0.8283, -2.6,`RK4`]):
[ > display(grac21,grac22,grac23,grac31,grac32,grac33,grac41,grac42,grac43);
```



```
> display(grac21,grac22,grac23,grac31,grac32,grac33,grac41,grac42,grac43,scaling='CONSTRAINED');
```



Rectas de regresión

```
> fit[leastsquare[[x,y]]]([listalogpaso,lista2logerro]);
      y = 0.96142505469558678610 + 1.8559225880462144870 x
> fit[leastsquare[[x,y]]]([listalogpaso,lista3logerro]);
      y = 0.11105446214318435735 + 4.4079924671362720487 x
> fit[leastsquare[[x,y]]]([listalogpaso,lista4logerro]);
      y = 0.60442944554465036897 + 4.5156389522284374802 x
```

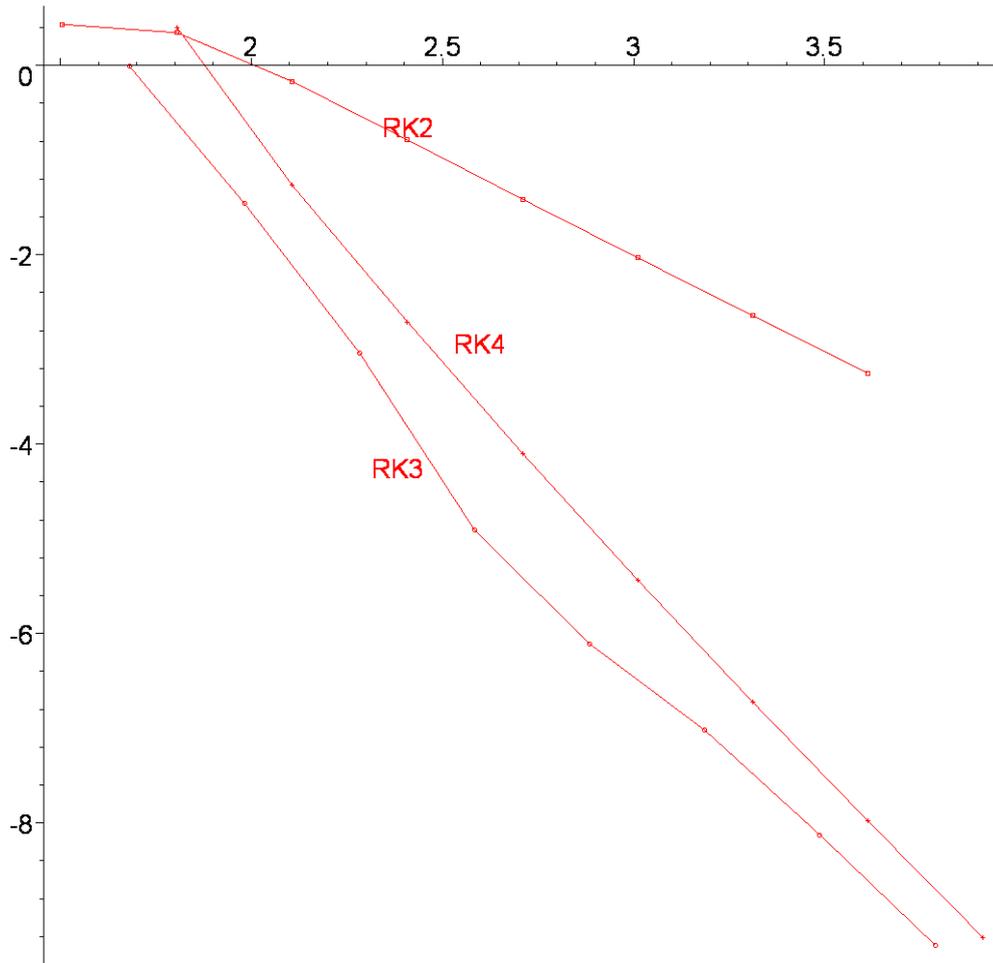
Pendiente del último segmento

```
> (lista2logerro[8]-lista2logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      2.0159076898864243822
> (lista3logerro[8]-lista3logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      3.8614045452902369117
> (lista4logerro[8]-lista4logerro[7])/(listalogpaso[8]-listalogpaso[7]);
      4.0914880055539869109
```

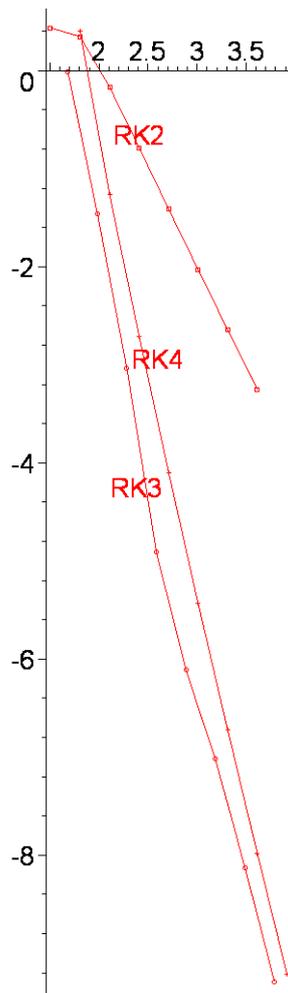
d) Gráficas comparadas 'log[10] del número de evaluaciones versus log[10] del error'

```
> listagrad2:= [seq([lista2logeval[i],lista2logerro[i]],i=1..8)]:
> grad21:=plot(listagrad2,style=LINE):
> grad22:=plot(listagrad2,style=POINT,symbol=BOX):
> grad23:=textplot([2.412, -0.649, `RK2`]):
> listagrad3:= [seq([lista3logeval[i],lista3logerro[i]],i=1..8)]:
> grad31:=plot(listagrad3,style=LINE):
> grad32:=plot(listagrad3,style=POINT,symbol=CIRCLE):
> grad33:=textplot([2.384, -4.248, `RK3`]):
> listagrad4:= [seq([lista4logeval[i],lista4logerro[i]],i=1..8)]:
> grad41:=plot(listagrad4,style=LINE):
> grad42:=plot(listagrad4,style=POINT,symbol=CROSS):
> grad43:=textplot([2.597, -2.936, `RK4`]):
```

```
> display(grad21,grad22,grad23,grad31,grad32,grad33,grad41,grad42,grad43);
```



```
> display(grad21,grad22,grad23,grad31,grad32,grad33,grad41,grad42,grad43,scaling='CONSTRAINED');
```



[Rectas de regresión

```
[ > fit[leastsquare[[x,y]]]([lista2logeval,lista2logerro]);
      y = 3.5601618389696398627 - 1.8559225880462144869 x
[ > fit[leastsquare[[x,y]]]([lista3logeval,lista3logerro]);
      y = 7.0595101592693490126 - 4.4079924671362720469 x
[ > fit[leastsquare[[x,y]]]([lista4logeval,lista4logerro]);
      y = 8.2867498726154145443 - 4.5156389522284374829 x
```

[Pendiente del último segmento

```
[ > (lista2logerro[8]-lista2logerro[7])/(lista2logeval[8]-lista2logeval[7]);
      -2.0159076898864243822
[ > (lista3logerro[8]-lista3logerro[7])/(lista3logeval[8]-lista3logeval[7]);
      -3.8614045452902369117
[ > (lista4logerro[8]-lista4logerro[7])/(lista4logeval[8]-lista4logeval[7]);
      -4.0914880055539869109
```


06. Ideas sobre convergencia, consistencia y 0-estabilidad

- 1 ¿Convergencia o consistencia?
- 2 Convergencia igual a consistencia

Convergencia; consistencia

Comentaremos algún resultado para los problemas como

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(a) = \boldsymbol{\eta},$$

aunque, para ser más intuitivos, razonaremos en el caso escalar

$$y' = f(x, y), \quad y(a) = \eta,$$

supuestas esas condiciones que nos han llevado a calificar un problema de valores iniciales como 'problema tipo', lo que nos garantizaba la existencia de solución única en un intervalo $[a, b]$. Nuestro estudio se ha encaminado hasta ahora a comprobar la posible consistencia de los métodos, o sea, el que con nuestras notaciones

$$\lim_{h \rightarrow 0} \frac{T(h)}{h} = 0,$$

donde $T(h)$ era el llamado 'error local' del método numérico

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h),$$

y donde y_n se refiere a la aproximación $y_n \sim y(x_n)$ para la red de nodos $x_0, x_1, x_2, \dots, x_N$ repartidos en el intervalo $[a, b]$ y separados por el paso $h = (b - a)/N$

Pero, seguramente, nuestra preferencia 'a priori' hubiera sido sin duda la de que

$$\lim_{h \rightarrow 0 \text{ o } N \rightarrow \infty} \max_{n=1 \dots N} |y(x_n) - y_n| = 0.$$

Esto último es lo que usualmente se llama **convergencia** del método numérico.

Pues bien, vamos a explicar que consistencia equivale a convergencia cuando se trabaja con un método que tenga la propiedad de ser 'estable'.

La estabilidad es una propiedad que en matemáticas significa que métodos cercanos o parecidos llevan a datos también parecidos. En el caso de uno de nuestros métodos

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h),$$

diremos que es estable o mejor **0-estable** cuando la diferencia de las sucesiones resultantes y_n y z_n de este método y del 'cercano'

$$z_{n+1} = z_n + h (\Phi(x_n, y_n, h) + \varepsilon_n(x_n, y_n, h)),$$

con $z_0 = y_0$, están acotadas en la forma

$$\max_{n=1 \dots N} |y_n - z_n| \leq M \max_{n=0 \dots N-1} |\varepsilon_n|,$$

para alguna constante positiva M .

Empleamos el término '0-estabilidad' para distinguir esta estabilidad de otras que trataremos más adelante, como la 'estabilidad lineal' o la 'estabilidad para métodos de E.D.P.' Lo importante por ahora es convencernos de que, de manera general, estamos actuando con métodos que presentan esta característica de la 0-estabilidad. No es fácil comprobar este aspecto empleando la definición directamente, pero es más sencillo si usamos el resultado que citamos a continuación:

El método

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

es 0-estable si la función $\Phi(x, y, h)$ es lipschitziana (con constante de LIPSCHITZ digamos Λ) respecto de la variable y (para $x \in [a, b]$ y $h \leq h_0$), o sea

$$|\Phi(x, y, h) - \Phi(x, y^*, h)| \leq \Lambda |y - y^*|.$$

Aceptaremos este resultado (aunque no lo probaremos) y procederemos a convencernos de que nuestra trayectoria de métodos se viene refiriendo a métodos de este tipo, y que por lo tanto son 0-estables. Esta afirmación se basa en el resultado

Todos los métodos explícitos de RUNGE-KUTTA cumplen la anterior condición de LIPSCHITZ y son , por lo tanto, 0-estables.

En la línea de lo que venimos haciendo no probaremos este hecho y nos limitaremos a ver en un ejemplo que así sucede, aunque este ejemplo nos dará una idea de la forma de razonar el resultado si lo deseamos.

El método de EULER $y_{n+1} = y_n + h f(x_n, y_n)$ es de este tipo, ya que $\Phi(x, y, h) = f(x, y)$, por lo que la propiedad de LIPSCHITZ es cierta para $\Lambda = L$.

Verlo en el caso que sigue es algo más costoso, pero más ilustrativo. El 'método del punto medio', que ya hemos visto, tiene

$$\Phi(x, y, h) = f\left(x + \frac{1}{2}h, y + \frac{1}{2}hf(x, y)\right).$$

En consecuencia,

$$\begin{aligned} & |\Phi(x, y, h) - \Phi(x, y^*, h)| \\ &= \left| f\left(x + \frac{1}{2}h, y + \frac{1}{2}hf(x, y)\right) - f\left(x + \frac{1}{2}h, y^* + \frac{1}{2}hf(x, y^*)\right) \right| \\ &\leq L \left| y + \frac{1}{2}hf(x, y) - y^* - \frac{1}{2}hf(x, y^*) \right| \\ &\leq L \left(|y - y^*| + \frac{1}{2}h |f(x, y) - f(x, y^*)| \right) \\ &\leq L \left(|y - y^*| + \frac{1}{2}hL |y - y^*| \right) \\ &= \left(L + \frac{1}{2}hL^2 \right) |y - y^*|, \end{aligned}$$

lo que sirve para probar que $\Phi(x, y, h)$ cumple la propiedad de LIPSCHITZ para $\Lambda = L(1 + \frac{1}{2}hL)$, donde L es la constante de LIPSCHITZ para f .

Terminamos comprobando lo antes dicho: para un método 0-estable, la consistencia garantiza la convergencia. En efecto, si nuestro método

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

es 0-estable y consistente, la consistencia significa que

$$\lim_{h \rightarrow 0} \frac{T(h)}{h} = 0,$$

o sea, cuando $h \rightarrow 0$ o $N \rightarrow \infty$

$$\Delta(h) - \Phi(h) = \frac{y(x_{n+1}) - y(x_n)}{h} - \Phi(x_n, y(x_n), h) \rightarrow 0.$$

Pongamos

$$\varepsilon_n(x_n, y(x_n), h) = \frac{y(x_{n+1}) - y(x_n)}{h} - \Phi(x_n, y(x_n), h).$$

Operando, se encuentra enseguida que

$$y(x_{n+1}) = y(x_n) + h (\Phi(x_n, y(x_n), h) + \varepsilon_n(x_n, y(x_n), h)).$$



La 0-estabilidad del método hace entonces que

$$\max_{n=1 \dots N} |y_n - y(x_n)| \leq M \max_{n=0 \dots N-1} |\varepsilon_n|,$$

y ya que cuando $h \rightarrow 0$ o $N \rightarrow \infty$

$$\varepsilon_n(x_n, y(x_n), h) \rightarrow 0,$$

se tiene también que

$$\max_{n=1 \dots N} |y_n - y(x_n)| \rightarrow 0,$$

que es justamente la convergencia del método.



Hemos comprobado así la implicación que más nos interesa de un resultado importante y clásico que suele expresarse como

Un método

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h)$$

es convergente para el que hemos llamado 'problema tipo' si y sólo si es a la vez consistente y 0-estable (también para el 'problema tipo')

Terminamos aquí estas observaciones que tratan de hacer ver que nuestra limitación al estudio de la consistencia y el orden de consistencia no lo es tanto. Hemos visto que este tipo de estudio nos permite realizar un análisis bastante completo y que, de hecho, por este sistema estudiamos al mismo tiempo la convergencia de los métodos, que es, en cierta manera, el objetivo final.

Ejercicios de

(6) Ideas sobre convergencia, consistencia y estabilidad

07. Métodos de RUNGE-KUTTA implícitos

- 1 Los métodos implícitos
- 2 Métodos implícitos y métodos de GAUSS
- 3 Ejemplos

Los métodos implícitos

En lugar del método de RUNGE-KUTTA explícito de 3 etapas

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1) \\ k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2) \end{aligned}$$

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3),$$

cuyo tablero de BUTCHER era

0				
c_2		a_{21}		
c_3		a_{31}	a_{32}	
		b_1	b_2	b_3

vamos a considerar ahora uno de igual número de etapas pero de la forma



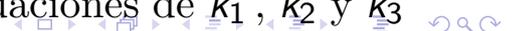
$$\begin{aligned} k_1 &= f(x_n + c_1 h, y_n + a_{1,1} h k_1 + a_{1,2} h k_2 + a_{1,3} h k_3) \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1 + a_{2,2} h k_2 + a_{2,3} h k_3) \\ k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2 + a_{3,3} h k_3) \end{aligned}$$

$$y_{n+1} = y_n + h(b_1 k_1 + b_2 k_2 + b_3 k_3),$$

cuyo tablero de BUTCHER es ahora

c_1		a_{11}	a_{12}	a_{13}
c_2		a_{21}	a_{22}	a_{23}
c_3		a_{31}	a_{32}	a_{33}
		b_1	b_2	b_3

Es obvio que la programación de tal método no se puede hacer mediante asignaciones directas, como en los métodos explícitos, sino resolviendo 'de alguna manera' las ecuaciones de k_1 , k_2 y k_3



en las que las etapas k_i aparecen no sólo a la izquierda del signo $=$, sino también a su derecha e incluidos en las variables de una función f que en nuestro estudio no es ni fácil ni complicada sino sólo una 'simple letra'.

En otras palabras; los valores de las etapas $k_1, k_2 \dots$ que antes se obtenían de forma (matemáticamente) exacta mediante asignaciones, ahora sólo se pueden obtener en principio de manera aproximada y eso a base de muchas evaluaciones de f .

Lo más usual es proceder con la llamada 'iteración de punto fijo' para el cálculo de aproximaciones de las etapas $k_1, k_2 \dots$. Consiste en darse valores 'arbitrarios' de k_1, k_2 y k_3 (en el caso de tres etapas), valores que llamaremos $k_{1,0}, k_{2,0}$ y $k_{3,0}$, y proceder a refinar sucesivamente estos valores mediante la repetida iteración de

$$\begin{aligned} k_{1,i+1} &= f(x_n + c_1 h, y_n + a_{1,1} h k_{1,i} + a_{1,2} h k_{2,i} + a_{1,3} h k_{3,i}) \\ k_{2,i+1} &= f(x_n + c_2 h, y_n + a_{2,1} h k_{1,i} + a_{2,2} h k_{2,i} + a_{2,3} h k_{3,i}) \\ k_{3,i+1} &= f(x_n + c_3 h, y_n + a_{3,1} h k_{1,i} + a_{3,2} h k_{2,i} + a_{3,3} h k_{3,i}) \end{aligned}$$

Naturalmente, elegir estos valores 'arbitrarios' cercanos a los verdaderos, aunque desconocidos, k_1, k_2 y k_3 , hace que obtengamos buenas aproximaciones con muy pocas iteraciones. Se procura generalmente que la cercanía valores iniciales - valores verdaderos sea grande para que una sola iteración sea suficiente (otra cosa aumentaría de manera desproporcionada el número de evaluaciones de f).

Cada uno de los métodos que veremos posee una o varias formas de encontrar valores iniciales convenientes de las etapas, lo que permite entonces que se conviertan en métodos realmente utilizables. Algunos ejemplos nos llevarán enseguida a tener práctica en esta organización.

Pero, ¿por qué acudir a estos métodos de puesta en práctica tan compleja cuando disponemos de métodos sencillos de implementar (explícitos) de orden tan alto como queramos? Algo más adelante veremos que hay un cierto tipo de ecuaciones, que designaremos con el calificativo de 'stiff' o de rígidas, que son bastante usuales y que se integran muy mal con cualquier método explícito de RUNGE-KUTTA. Lo que no ocurre con los métodos implícitos que, pese a su mayor inconveniente de programación, tienen un comportamiento idóneo con este tipo de ecuaciones. El tema de la 'estabilidad lineal' nos aclarará este comportamiento, justificando la molestia que ahora nos tomamos.

Comenzamos por definir de forma más general estos métodos implícitos, que hasta ahora hemos expuesto sólo para 3 etapas.

Pero de manera más general, un **Un método de RUNGE-KUTTA implícito o no necesariamente explícito de q etapas** tiene etapas dadas por

$$\begin{aligned} k_1 &= f(x_n + c_1 h, y_n + a_{1,1} h k_1 + a_{1,2} h k_2 + \cdots + a_{1,q-1} h k_{q-1} + a_{1,q} h k_q) \\ k_2 &= f(x_n + c_2 h, y_n + a_{2,1} h k_1 + a_{2,2} h k_2 + \cdots + a_{2,q-1} h k_{q-1} + a_{2,q} h k_q) \\ k_3 &= f(x_n + c_3 h, y_n + a_{3,1} h k_1 + a_{3,2} h k_2 + \cdots + a_{3,q-1} h k_{q-1} + a_{3,q} h k_q) \\ &\vdots \\ k_q &= f(x_n + c_q h, y_n + a_{q,1} h k_1 + a_{q,2} h k_2 + \cdots + a_{q,q-1} h k_{q-1} + a_{q,q} h k_q) \end{aligned}$$

con los k_1, k_2, \dots dados por ecuaciones implícitas, y luego obtiene el valor de y_{n+1} como

$$y_{n+1} = y_n + h (b_1 k_1 + b_2 k_2 + \cdots + b_q k_q).$$

Por lo tanto se resume mediante el Tablero de BUTCHER

c_1	a_{11}	a_{12}	a_{13}	\cdots	$a_{1\ q-1}$	$a_{1\ q}$
c_2	a_{21}	a_{22}	a_{23}	\cdots	$a_{2\ q-1}$	$a_{2\ q}$
c_3	a_{31}	a_{32}	a_{33}	\cdots	$a_{3\ q-1}$	$a_{3\ q}$
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots
c_{q-1}	$a_{q-1\ 1}$	$a_{q-1\ 2}$	$a_{q-1\ 3}$	\cdots	$a_{q-1\ q-1}$	$a_{q-1\ q}$
c_q	$a_{q\ 1}$	$a_{q\ 2}$	$a_{q\ 3}$	\cdots	$a_{q\ q-1}$	$a_{q\ q}$
	b_1	b_2	b_3	\cdots	b_{q-1}	b_q

La regularidad de la organización de estos métodos hace que sea mucho más sencilla la caracterización del orden que se puede alcanzar con un número dado de etapas. Así el resultado que tenemos es ahora:

Los métodos de RUNGE-KUTTA implícitos de q etapas alcanzan, de una sola manera, el orden máximo $2q$. El único método de q etapas que alcanza este orden de $2q$ recibe el nombre de **método de GAUSS** de q etapas.

El nombre se debe a que las abscisas c_1, c_2, \dots que emplean estos métodos son las mismas que se usan en la integración (óptima para el orden de exactitud) de GAUSS.

En la búsqueda del orden de los métodos implícitos que, de momento, estudiamos en el caso escalar, sirven las técnicas que ya hemos usado con los explícitos. Ahora sigue siendo igualmente válido poner uno de estos métodos como

$$y_{n+1} = y_n + h \Phi(x_n, y_n, h),$$

y describir $\Phi(h)$ como

$$\Phi(h) = b_1 k_1(h) + b_2 k_2(h) + \cdots + b_q k_q(h),$$

haciendo notar que ahora $k_1(h)$ depende efectivamente de h . Las condiciones de orden se obtienen también sencillamente de las igualdades

$$\Phi^{(p-1)}(0) = \frac{1}{p} f^{(p-1)}$$

donde $\Phi^{(p)}(0)$ se prepara mediante el cálculo de los valores $k_1^{(p)}(0)$, $k_2^{(p)}(0)$...



Es fácil observar que, en el cálculo de $k_i^{(p)}(h)$ se encuentra el propio valor de $k_i^{(p)}(h)$ en alguna parte a la derecha de $k_i^{(p)}(h) =$. No obstante, se encuentra multiplicado siempre por h , por lo que al calcular $k_i^{(p)}(0)$, que es lo que finalmente interesa, dicho término desaparece y el valor de $k_i^{(p)}(0)$ se asigna directamente sin resolver ninguna ecuación.

Otra advertencia especial es que ahora no es ninguno de los puntos

- (1) $(x_n + c_1 h, y_n + a_{1,1} h k_1 + a_{1,2} h k_2 + \cdots + a_{1,q} h k_q)$,
- (2) $(x_n + c_2 h, y_n + a_{2,1} h k_1 + a_{2,2} h k_2 + \cdots + a_{2,q} h k_q)$,
- (3)

el más repetido. Ahora lo es el punto (x_n, y_n) , que denotaremos por

$$(0) \quad (x_n, y_n)$$

y que omitiremos en general en las expresiones para mayor brevedad.



Métodos implícitos de 1 etapa

Ahora los métodos (implícitos) de una etapa tienen tablero

$$\begin{array}{c|c} c_1 & a_{11} \\ \hline & b_1 \end{array}$$

y tres parámetros a determinar. Se escriben

$$k_1 = f(x_n + c_1 h, y_n + a_{1,1} h k_1)$$

$$y_{n+1} = y_n + h b_1 k_1,$$

y se tiene

$$\Phi(h) = b_1 k_1(h).$$

Usaremos las formas de acortar para los puntos

- (0) para (x_n, y_n) , y
- (1) para $(x_n + c_1 h, y_n + a_{1,1} h k_1)$.



$$k_1(h) = f(1),$$

y (1) se transforma en (0), que no escribimos, cuando $h = 0$, luego $k_1(0) = f$ y $\Phi(0) = b_1 f$. Como $f^{(0)} = f$, la única posibilidad (para f genérica) de que $\Phi(0) = f^{(0)}$ es que

$$b_1 = 1$$

aunque ahora esto deja dos parámetros libres c_1 y $a_{1,1}$, lo que permite ir a buscar mayor orden.

$$k_1'(h) = c_1 f_x(1) + f_y(1)[a_{1,1} k_1 + a_{1,1} h k_1'];$$

denotaremos por $[\]$ la cantidad $[\] = [a_{1,1} k_1 + a_{1,1} h k_1']$ para futuras operaciones. Teniendo en cuenta que

$$[\]_{h=0} = a_{1,1} f,$$

resultan

$$k_1'(0) = c_1 f_x + a_{1,1} f f_y \quad \text{y} \quad \Phi'(0) = b_1 c_1 f_x + b_1 a_{1,1} f f_y.$$



Recordando que

$$\frac{1}{2} f^{(1)} = \frac{1}{2} f_x + \frac{1}{2} f f_y,$$

el orden 2 necesita la ecuación ya establecida y, además,

$$\begin{aligned} b_1 c_1 &= 1/2 \\ b_1 a_{1,1} &= 1/2. \end{aligned}$$

Esto deja

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

como único método de 1 etapa y orden 2: el método de GAUSS de 1 etapa. Recordando la afirmación que hicimos sobre el orden alcanzable, no es necesario buscar orden más alto para este método.

La puesta en práctica de este método

$$k_1 = f(x_n + \frac{1}{2} h, y_n + \frac{1}{2} h k_1)$$

$$y_{n+1} = y_n + h k_1,$$

se apoya en las etapas explícitas del 'método del punto medio'

$$\begin{aligned} \hat{k}_1 &= f(x_n, y_n), \\ \hat{k}_2 &= f(x_n + \frac{1}{2} h, y_n + \frac{1}{2} h \hat{k}_1), \end{aligned}$$

método de orden 2, porque es natural tomar \hat{k}_2 como buena aproximación $k_{1,0}$ de k_1 (evaluación de f para la misma primera variable y una segunda de estructura análoga). Cuando así se hace, se efectúa una sola iteración de punto fijo para mejorar $k_{1,0}$. O sea, la secuencia de operaciones queda como

$$\begin{aligned}\hat{k}_1 &= f(x_n, y_n), \\ k_{1,0} &= f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}h\hat{k}_1\right), \\ k_1 &= f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_{1,0}\right), \\ y_{n+1} &= y_n + hk_1,\end{aligned}$$

realizando un conjunto de 3 evaluaciones de f por paso para el cálculo efectivo (y sólo aproximado) de la única etapa del método.

Que la solución de este método (y de hecho de todos los implícitos) sea sólo aproximada puede tener consecuencias sobre el orden alcanzado y sobre el comportamiento cara a las ecuaciones 'stiff' de las que hablaremos más adelante. Si por el momento nos referimos al orden, y teniendo en cuenta que el procedimiento de arranque (aproximación de alguna de las k_i) se hace con un método explícito, es importante que el orden del método empleado no sea inferior al del método implícito.

No obstante, no es el orden la única cualidad que buscamos, ya que nos interesamos también por el comportamiento de cara a las ecuaciones rígidas. Se emplea mucho para 1 sola etapa el llamado 'método de EULER implícito', con tablero

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

y que sólo posee orden 1. Como es

$$k_1 = f(x_n + h, y_n + hk_1)$$

$$y_{n+1} = y_n + hk_1,$$

a veces se le escribe en la forma más simple y equivalente

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}),$$

donde ahora el carácter implícito se traslada al cálculo de y_{n+1} .

De manera análoga a lo hecho antes, la puesta en práctica de este método se apoya en las etapas explícitas del 'método modificado de EULER'

$$\begin{aligned} \hat{k}_1 &= f(x_n, y_n), \\ \hat{k}_2 &= f(x_n + h, y_n + h \hat{k}_1), \end{aligned}$$

tomando \hat{k}_2 como buena aproximación $k_{1,0}$ de k_1 , y dejando el método en forma práctica como

$$\begin{aligned} \hat{k}_1 &= f(x_n, y_n), \\ k_{1,0} &= f(x_n + h, y_n + h \hat{k}_1), \\ k_1 &= f(x_n + h, y_n + h k_{1,0}), \end{aligned}$$

$$y_{n+1} = y_n + h k_1,$$

con 3 evaluaciones de f por paso para el cálculo efectivo de la única etapa del método implícito de EULER.

Métodos implícitos de 2 etapas

En primer lugar presentamos el método de GAUSS de 2 etapas, con orden 4, dejando su deducción (algo pesada) para algún ejercicio. Sus coeficientes son menos sencillos e irracionales. Tiene por tablero

$\frac{3 - \sqrt{3}}{6}$	$\frac{1}{4}$	$\frac{3 - 2\sqrt{3}}{12}$
$\frac{3 + \sqrt{3}}{6}$	$\frac{3 + 2\sqrt{3}}{12}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

y es el único de 2 etapas con ese orden. $c_1 \sim 0,21$ y $c_2 \sim 0,79$ son las abscisas para la integración de GAUSS en $[0, 1]$ con 2 abscisas.

Pero, para 2 etapas, uno de los más usados es el llamado 'método trapezoidal', que sólo alcanza orden 2. Se le denomina también 'LOBATTO IIIa', como miembro de una familia de métodos emparentados con los de integración numérica de LOBATTO. Sólo es implícito en la segunda etapa k_2 . Su tablero es

0	0	0
1	1/2	1/2
	1/2	1/2

y se escribe

$$\begin{aligned}
 k_1 &= f(x_n, y_n) \\
 k_2 &= f(x_n + h, y_n + \frac{1}{2} h k_1 + \frac{1}{2} h k_2) \\
 y_{n+1} &= y_n + h \left(\frac{1}{2} k_1 + \frac{1}{2} k_2 \right).
 \end{aligned}$$

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↻ 🔍 ↺

De nuevo el 'método modificado de EULER'

$$\begin{aligned}
 \hat{k}_1 &= f(x_n, y_n), \\
 \hat{k}_2 &= f(x_n + h, y_n + h \hat{k}_1), \\
 \hat{y}_{n+1} &= y_n + h \left(\frac{1}{2} \hat{k}_1 + \frac{1}{2} \hat{k}_2 \right).
 \end{aligned}$$

tiene etapas que nos permiten lanzar el 'método trapezoidal'. Es posible tomar $k_{2,0} = f(x_{n+1}, \hat{y}_{n+1})$ como buena aproximación de k_2 , realizando entonces una sola mejora de k_2 y dejando el método en forma práctica como

$$\begin{aligned}
 k_1 = \hat{k}_1 &= f(x_n, y_n), \\
 \hat{k}_2 &= f(x_n + h, y_n + h \hat{k}_1), \\
 \hat{y}_{n+1} &= y_n + h \left(\frac{1}{2} \hat{k}_1 + \frac{1}{2} \hat{k}_2 \right), \\
 k_{2,0} &= f(x_{n+1}, \hat{y}_{n+1}), \\
 k_2 &= f(x_n + h, y_n + \frac{1}{2} h k_1 + \frac{1}{2} h k_{2,0}), \\
 y_{n+1} &= y_n + h \left(\frac{1}{2} k_1 + \frac{1}{2} k_2 \right).
 \end{aligned}$$

con 4 evaluaciones de f por paso para el cálculo efectivo de las dos etapas implícitas.

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ↻ 🔍 ↺

Estos procedimientos de arranque para los métodos implícitos empleando métodos explícitos marchan bien cuando se integra un problema 'normal' (no especialmente rígido). Sin embargo, su uso con problemas cuyo carácter 'stiff' sea muy fuerte lleva a comportamientos similares a los de los métodos explícitos, que es justamente lo que se pretendía evitar con los implícitos.

En estos casos, la única solución consiste en iterar realmente (y no una sola vez) con algún método de convergencia rápida, ya que el número de evaluaciones es una dificultad importante. Así se usa, por ejemplo el 'método de NEWTON' o el 'método modificado de NEWTON' y, en ocasiones, aún estos métodos tardan en llegar a una convergencia adecuada.

Y es que, finalmente, los problemas de integración difícil no admiten recetas generales y requieren correcciones heurísticas hasta dar con un tratamiento y un método adecuados.

El capítulo siguiente comienza a ocuparse de estos problemas cuyo tratamiento es más difícil.

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-01

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

1		1	
c_2		0	a_{22}

		b_1	b_2

b) Pruébese que, entre los métodos obtenidos, se encuentra el método de GAUSS de 1 sola etapa

(desde aquí es Ejercicio del capítulo 8)

c) Para los métodos obtenidos, calcúlese la función de estabilidad

d) Calcúlese la función de estabilidad para el citado método de GAUSS, y compruébese que es A-estable

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+h,y0+h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+c[2]*h,y0+a[2,2]*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
```

```

[
    Fi_0 := b1 f(x0, y0) + b2 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
    cond1a := 1 = b1 + b2
[ > solve(cond1a);
    { b2 = 1 - b1, b1 = b1 }
[ condiciones para el orden 2
[ > Fi10:=expand(b[1]*k110+b[2]*k210);
    Fi10 := b1 D1(f)(x0, y0) + b1 D2(f)(x0, y0) f(x0, y0) + b2 D1(f)(x0, y0) c2 + b2 D2(f)(x0, y0) a2,2 f(x0, y0)
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
    cond2a := 1/2 = b1 + b2 c2
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
    cond2b := 1/2 = b1 + b2 a2,2
[ > solucion:=solve({cond1a,cond2a,cond2b},{b[1],c[2],a[2,2]});
    solucion := { c2 = 1/2 * (-1 + 2 b2) / b2, b1 = 1 - b2, a2,2 = 1/2 * (-1 + 2 b2) / b2 }
]

```

```

[ condiciones para el orden 3
[ > Fi20:=expand(b[1]*k120+b[2]*k220);
    Fi20 := b1 D1,1(f)(x0, y0) + 2 b1 D1,2(f)(x0, y0) f(x0, y0) + b1 D2,2(f)(x0, y0) f(x0, y0)^2 + 2 b1 D2(f)(x0, y0) D1(f)(x0, y0)
    + 2 b1 D2(f)(x0, y0)^2 f(x0, y0) + b2 D1,1(f)(x0, y0) c2^2 + 2 b2 c2 D1,2(f)(x0, y0) a2,2 f(x0, y0)
    + b2 a2,2^2 f(x0, y0)^2 D2,2(f)(x0, y0) + 2 b2 D2(f)(x0, y0) a2,2 D1(f)(x0, y0) c2 + 2 b2 D2(f)(x0, y0)^2 a2,2^2 f(x0, y0)
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
    cond3a := 1/6 = 1/2 b1 + 1/2 b2 c2^2
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
    cond3b := 1/3 = b1 + b2 c2 a2,2
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
    cond3c := 1/6 = 1/2 b1 + 1/2 b2 a2,2^2
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
    cond3d := 1/6 = b1 + b2 c2 a2,2
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
    cond3e := 1/6 = b1 + b2 a2,2^2
[ > solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
]

```

Ni siquiera es posible igualar simultáneamente cond3a y cond3d, por ejemplo, si se acepta que $c_2 = a_{2,2}$, como se desprende de cond2a y cond2b

```

[ > solve({c[2]=a[2,2],cond3a,cond3d});
[ b)
[ > solucion1:=subs(b[2]=1,solucion) union {b[2]=1};
    solucion1 := { b1 = 0, b2 = 1, c2 = 1/2, a2,2 = 1/2 }
]

```

La solución obtenida corresponde al tablero

$$\frac{1}{2} \mid 0 \quad \frac{1}{2}$$

$$\mid 0 \quad 1$$

que no hace intervenir la primera etapa y que coincide, por lo tanto con

$$\frac{1}{2} \mid \frac{1}{2}$$

$$\mid 1$$

que no es sino el método de Gauss de 1 etapa

c)

función de estabilidad

> `f := (x, y) -> lambda * y;`

$$f := (x, y) \rightarrow \lambda y$$

> `ka1 := unapply(f(x0+h, y0+h*kp1(h)), h);`

$$ka1 := h \rightarrow \lambda (y0 + h \text{ kp1}(h))$$

> `ka2 := unapply(f(x0+c[2]*h, y0+a[2,2]*h*kp2(h)), h);`

$$ka2 := h \rightarrow \lambda (y0 + a_{2,2} h \text{ kp2}(h))$$

> `test := solve({k1=subs(kp1(h)=k1, ka1(h)), k2=subs(kp2(h)=k2, ka2(h))}, {k1, k2});`

$$\text{test} := \left\{ k2 = -\frac{\lambda y0}{-1 + \lambda a_{2,2} h}, k1 = -\frac{\lambda y0}{-1 + \lambda h} \right\}$$

> `k1 := subs(test, k1); k2 := subs(test, k2);`

$$k1 := -\frac{\lambda y0}{-1 + \lambda h}$$

$$k2 := -\frac{\lambda y0}{-1 + \lambda a_{2,2} h}$$

> `y1 := expand(y0+h*(b[1]*k1+b[2]*k2));`

$$y1 := y0 - \frac{h b_1 \lambda y0}{-1 + \lambda h} - \frac{h b_2 \lambda y0}{-1 + \lambda a_{2,2} h}$$

> `y1 := collect(subs(lambda=z/h, y1), y0);`

$$y1 := \left(1 - \frac{b_1 z}{-1 + z} - \frac{b_2 z}{-1 + z a_{2,2}} \right) y0$$

> `r := unapply(collect(simplify(y1/y0), z), z);`

$$r := z \rightarrow -\frac{-1 + (-a_{2,2} + b_1 a_{2,2} + b_2) z^2 + (a_{2,2} + 1 - b_1 - b_2) z}{(-1 + z)(-1 + z a_{2,2})}$$

que es la función de estabilidad de la familia general propuesta

d)

Para el método de GAUSS de 1 etapa, obtenido en el conjunto-solucion solucion1

tenemos como función de estabilidad

> `r1 := unapply(simplify(subs(solucion1, r(z))), z);`

$$r1 := z \rightarrow -\frac{z+2}{-2+z}$$

El dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r1(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2. Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-02

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

0		0	
c_2		a_{21}	a_{22}

		b_1	b_2

b) Compruébese qué se obtiene cuando se fuerza el que $c_2 = 1$, o sea, que las abscisas implicadas sean 0 y 1

(desde aquí es Ejercicio del capítulo 8)

c) Para los métodos de orden máximo obtenidos, calcúlese la función de estabilidad

d) Calcúlese la función de estabilidad para los métodos del apartado b)

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0,y0),h):
[ > k2_:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1_(h)+a[2,2]*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
[                               Fi_0 := b1 f(x0, y0) + b2 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                               cond1a := 1 = b1 + b2
```

```

> solve(cond1a);
                                     { b2 = 1 - b1, b1 = b1 }
condiciones para el orden 2
> Fi10:=expand(b[1]*k110+b[2]*k210);
      Fi10 := b2 D1(f)(x0, y0) c2 + b2 D2(f)(x0, y0) a2,1 f(x0, y0) + b2 D2(f)(x0, y0) a2,2 f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a := 1/2 = b2 c2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b := 1/2 = b2 a2,1 + b2 a2,2
> solve({cond1a,cond2a,cond2b},{b[1],c[2],a[2,1]});
                                     { c2 = 1/2 * 1/b2, b1 = 1 - b2, a2,1 = -1/2 * (-1 + 2*b2*a2,2)/b2 }
condiciones para el orden 3
> Fi20:=expand(b[1]*k120+b[2]*k220);
      Fi20 := b2 D1,1(f)(x0, y0) c2^2 + 2 b2 c2 D1,2(f)(x0, y0) a2,1 f(x0, y0) + 2 b2 c2 D1,2(f)(x0, y0) a2,2 f(x0, y0)
      + b2 D2,2(f)(x0, y0) a2,1^2 f(x0, y0)^2 + 2 b2 D2,2(f)(x0, y0) a2,1 f(x0, y0)^2 a2,2 + b2 D2,2(f)(x0, y0) a2,2^2 f(x0, y0)^2
      + 2 b2 D2(f)(x0, y0) a2,2 D1(f)(x0, y0) c2 + 2 b2 D2(f)(x0, y0)^2 a2,2 a2,1 f(x0, y0) + 2 b2 D2(f)(x0, y0)^2 a2,2^2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
                                     cond3a := 1/6 = 1/2 * b2 c2^2
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
                                     cond3b := 1/3 = b2 c2 a2,1 + b2 c2 a2,2
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
                                     cond3c := 1/6 = 1/2 * b2 a2,1^2 + b2 a2,1 a2,2 + 1/2 * b2 a2,2^2
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
                                     cond3d := 1/6 = b2 c2 a2,2
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
                                     cond3e := 1/6 = b2 a2,1 a2,2 + b2 a2,2^2
> solucion:=solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
                                     solucion := { c2 = 2/3, b1 = 1/4, b2 = 3/4, a2,1 = 1/3, a2,2 = 1/3 }

```

Es decir, existe una única solución de orden 3. Se trata del método dado por el tablero

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

[b)

[forzando a que $c_2 = 1$

```
> solucion1:=solve({cond1a,cond2a,cond2b,c[2]=1},{c[2],a[2,1],b[1],b[2]});  
  
solucion1 := {c2 = 1, b1 = 1/2, b2 = 1/2, a2,1 = 1 - a2,2}
```

Se obtiene la familia de orden 2 dada por el tablero

$$\begin{array}{c|cc} 0 & & 0 \\ 1 & 1 - a_{22} & a_{22} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

uno de cuyos métodos es

```
> solucion2:=subs(a[2,2]=1/2,solucion1) union {a[2,2]=1/2};  
  
solucion2 := {c2 = 1, b1 = 1/2, b2 = 1/2, a2,2 = 1/2, a2,1 = 1/2}
```

el conocido como LOBATTO IIIA de 2 etapas

$$\begin{array}{c|cc} 0 & & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

[c)

[función de estabilidad

```
> f:=(x,y)->lambda*y;  
  
f := (x, y) -> lambda y  
  
> ka1:=unapply(f(x0,y0),h);  
  
ka1 := h -> lambda y0  
  
> ka2:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)),h);  
  
ka2 := h -> lambda (y0 + a2,1 h kp1(h) + a2,2 h kp2(h))  
  
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))}),{k1,k2});  
  
test := {k1 = lambda y0, k2 = -frac(lambda y0 (1 + lambda a2,1 h), -1 + lambda a2,2 h)}  
  
> k1:=subs(test,k1);k2:=subs(test,k2);  
  
k1 := lambda y0  
k2 := -frac(lambda y0 (1 + lambda a2,1 h), -1 + lambda a2,2 h)  
  
> y1:=expand(y0+h*(b[1]*k1+b[2]*k2));  
  
y1 := y0 + h b1 lambda y0 -frac(h b2 lambda y0, -1 + lambda a2,2 h) -frac(b2 lambda^2 y0 a2,1 h^2, -1 + lambda a2,2 h)  
  
> y1:=collect(subs(lambda=z/h,y1),y0);  
  
y1 := (1 + b1 z -frac(b2 z, -1 + z a2,2) -frac(b2 z^2 a2,1, -1 + z a2,2)) y0  
  
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow -\frac{1 + (-b_1 a_{2,2} + b_2 a_{2,1}) z^2 + (-a_{2,2} + b_1 + b_2) z}{-1 + z a_{2,2}}$$

que es la función de estabilidad de la familia general propuesta, mientras que

> `r1:=unapply(simplify(subs(solucion,r(z))),z);`

$$r1 := z \rightarrow -\frac{1}{2} \frac{6 + z^2 + 4z}{-3 + z}$$

lo es de la solución de orden 3 obtenida. Ahora, el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r1(z)$ sea inferior a 1 no será todo el semiplano de parte real negativa, ya que la función de estabilidad tiende a infinito cuando z tiende a infinito en cualquier dirección. O sea, el método no es A-estable

d)

función de estabilidad para métodos de b)

> `r2:=unapply(simplify(subs(solucion1,r(z))),z);`

$$r2 := z \rightarrow \frac{1}{2} \frac{-2 + 2z^2 a_{2,2} - z^2 - 2z + 2z a_{2,2}}{-1 + z a_{2,2}}$$

y, a priori, nada podemos decir de la A-estabilidad. Sin embargo, para el método LOBATTO IIIA de 2 etapas

> `r3:=unapply(simplify(subs(solucion2,r(z))),z);`

$$r3 := z \rightarrow -\frac{2+z}{-2+z}$$

el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r3(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2. Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-03

Búsqese el método de GAUSS de 2 etapas (es decir, el método implícito de RUNGE-KUTTA con dos etapas y orden 4)

$$c_1 \mid a_{11} \quad a_{12}$$

$$c_2 \mid a_{21} \quad a_{22}$$

$$\mid b_1 \quad b_2$$

(desde aquí es Ejercicio del capítulo 8)

Calcúlese la función de estabilidad del método y compruébese que es A-estable

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp10(h)+a[1,2]*h*kp20(h)),h):
[ > k20:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp10(h)+a[2,2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand(b[1]*k100+b[2]*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand(b[1]*k110+b[2]*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
x
pand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
x
pand(k220):
[ > F12:=expand(b[1]*k120+b[2]*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[
```

```

> k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
> Fi3:=expand(b[1]*k130+b[2]*k230):
# consistencia y orden 1
> cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
cond1a := b1 + b2 = 1
# condiciones para el orden 2
> cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
cond2a := b1 c1 + b2 c2 = 1/2
> cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0),
f(x0,y0)));
cond2b := b1 a1,1 + b1 a1,2 + b2 a2,1 + b2 a2,2 = 1/2
# condiciones para el orden 3
> cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
cond3a := b1 c1^2 + b2 c2^2 = 1/3
> cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
y0)),f(x0,y0));
cond3b := 2 b1 c1 a1,1 + 2 b1 c1 a1,2 + 2 b2 c2 a2,1 + 2 b2 c2 a2,2 = 2/3
> cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
0,y0)),f(x0,y0)^2);
cond3c := b1 a1,1^2 + 2 b1 a1,1 a1,2 + b1 a1,2^2 + b2 a2,1^2 + 2 b2 a2,1 a2,2 + b2 a2,2^2 = 1/3
> cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
0,y0)),D[1](f)(x0,y0));
cond3d := 2 b1 c1 a1,1 + 2 b1 a1,2 c2 + 2 b2 a2,1 c1 + 2 b2 c2 a2,2 = 1/3
> cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
^2),f(x0,y0));
cond3e :=
2 b1 a1,1^2 + 2 b1 a1,1 a1,2 + 2 b1 a1,2 a2,1 + 2 b1 a1,2 a2,2 + 2 b2 a2,1 a1,1 + 2 b2 a1,2 a2,1 + 2 b2 a2,1 a2,2 + 2 b2 a2,2^2 = 1/3
# condiciones para el orden 4
> cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
cond4a := b2 c2^3 + b1 c1^3 = 1/4
> cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
(x0,y0)),f(x0,y0));
cond4b := 3 b1 c1^2 a1,1 + 3 b1 c1^2 a1,2 + 3 b2 c2^2 a2,1 + 3 b2 c2^2 a2,2 = 3/4
> cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](
f)(x0,y0)),f(x0,y0)^2);
cond4c := 3 b1 c1 a1,1^2 + 3 b1 c1 a1,2^2 + 6 b1 c1 a1,1 a1,2 + 3 b2 c2 a2,1^2 + 3 b2 c2 a2,2^2 + 6 b2 c2 a2,1 a2,2 = 3/4
> cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](
f)(x0,y0)),D[1](f)(x0,y0));
cond4d := 6 b2 c2^2 a2,2 + 6 b2 c2 a2,1 c1 + 6 b1 c1 a1,2 c2 + 6 b1 c1^2 a1,1 = 3/4
> cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(
coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));
cond4e := 6 b2 a2,1 c1 a1,1 + 6 b2 c1 a1,2 a2,1 + 6 b2 a1,2 c2 a2,1 + 6 b2 c2 a2,1 a1,1 + 12 b2 c2 a2,1 a2,2 + 12 b1 c1 a1,1^2
+ 6 b1 c1 a1,2 a2,1 + 12 b1 c1 a1,1 a1,2 + 6 b1 c1 a1,2 a2,2 + 6 b1 a1,2 c2 a2,1 + 6 b1 a1,2 c2 a2,2 + 12 b2 c2 a2,2^2 = 5/4

```

```
> cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);
```

$$\text{cond4f} := 3 b_1 a_{1,1}^2 a_{1,2} + 3 b_1 a_{1,1} a_{1,2}^2 + b_1 a_{1,1}^3 + b_1 a_{1,2}^3 + b_2 a_{2,1}^3 + b_2 a_{2,2}^3 + 3 b_2 a_{2,1}^2 a_{2,2} + 3 b_2 a_{2,1} a_{2,2}^2 = \frac{1}{4}$$

```
> cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));
```

```
cond4g :=
```

$$6 b_2 c_2 a_{2,1} a_{2,2} + 6 b_2 c_2 a_{2,2}^2 + 6 b_1 a_{1,2}^2 c_2 + 6 b_1 c_1 a_{1,1}^2 + 6 b_1 c_1 a_{1,1} a_{1,2} + 6 b_1 a_{1,2} c_2 a_{1,1} + 6 b_2 a_{2,1} c_1 a_{2,2} + 6 b_2 a_{2,1}^2 c_1 = \frac{3}{4}$$

```
> cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);
```

$$\begin{aligned} \text{cond4h} := & 9 b_1 a_{1,1}^3 + 9 b_2 a_{2,2}^3 + 6 b_2 a_{2,1} a_{1,1} a_{2,2} + 18 b_2 a_{2,1} a_{2,2}^2 + 3 b_2 a_{2,1} a_{1,1}^2 + 6 b_2 a_{1,2} a_{2,1}^2 + 18 b_1 a_{1,1}^2 a_{1,2} \\ & + 6 b_1 a_{1,2}^2 a_{2,1} + 9 b_1 a_{1,1} a_{1,2}^2 + 6 b_1 a_{1,2}^2 a_{2,2} + 6 b_1 a_{1,2} a_{2,1} a_{1,1} + 6 b_1 a_{1,2} a_{2,2} a_{1,1} + 9 b_2 a_{2,1}^2 a_{2,2} + 3 b_2 a_{1,2}^2 a_{2,1} \\ & + 3 b_1 a_{1,2} a_{2,1}^2 + 6 b_1 a_{1,2} a_{2,1} a_{2,2} + 3 b_1 a_{1,2} a_{2,2}^2 + 6 b_2 a_{2,1}^2 a_{1,1} + 6 b_2 a_{1,2} a_{2,1} a_{1,1} + 6 b_2 a_{1,2} a_{2,1} a_{2,2} = 1 \end{aligned}$$

```
> cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));
```

$$\text{cond4i} := 3 b_1 c_1^2 a_{1,1} + 3 b_1 a_{1,2} c_2^2 + 3 b_2 a_{2,1} c_1^2 + 3 b_2 c_2^2 a_{2,2} = \frac{1}{4}$$

```
> cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);
```

$$\begin{aligned} \text{cond4j} := & 6 b_1 c_1 a_{1,1}^2 + 6 b_1 a_{1,2} c_2 a_{1,1} + 6 b_1 c_1 a_{1,2} a_{2,1} + 6 b_1 a_{1,2} c_2 a_{2,2} + 6 b_2 c_2 a_{2,2}^2 + 6 b_2 a_{2,1} c_1 a_{1,1} + 6 b_2 a_{1,2} c_2 a_{2,1} \\ & + 6 b_2 a_{2,1} c_1 a_{2,2} = \frac{1}{4} \end{aligned}$$

```
> cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));
```

$$\begin{aligned} \text{cond4k} := & 6 b_1 a_{1,1}^3 + 6 b_2 a_{2,2}^3 + 6 b_1 a_{1,1}^2 a_{1,2} + 6 b_1 a_{1,2}^2 a_{2,1} + 6 b_1 a_{1,2} a_{2,2}^2 + 12 b_1 a_{1,2} a_{2,1} a_{1,1} + 6 b_1 a_{1,2} a_{2,2} a_{1,1} \\ & + 6 b_1 a_{1,2} a_{2,1} a_{2,2} + 6 b_2 a_{1,2} a_{2,1} a_{1,1} + 12 b_2 a_{1,2} a_{2,1} a_{2,2} + 6 b_2 a_{2,1} a_{1,1}^2 + 6 b_2 a_{1,2} a_{2,1}^2 + 6 b_2 a_{2,1} a_{2,2}^2 \\ & + 6 b_2 a_{2,1} a_{1,1} a_{2,2} = \frac{1}{4} \end{aligned}$$

```
# soluciones para el método
```

```
> metodo:=allvalues(solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e,cond4a,cond4b,cond4c,cond4d,cond4f,c[1]=a[1,1]+a[1,2],c[2]=a[2,1]+a[2,2]}))[2];
```

$$\text{metodo} := \left\{ c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}, c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}, a_{1,2} = \frac{1}{4} - \frac{\sqrt{3}}{6}, a_{2,1} = \frac{1}{4} + \frac{\sqrt{3}}{6}, b_1 = \frac{1}{2}, b_2 = \frac{1}{2}, a_{1,1} = \frac{1}{4}, a_{2,2} = \frac{1}{4} \right\}$$

```
> c[1]:=subs(metodo,c[1]);c[2]:=subs(metodo,c[2]);a[1,1]:=subs(metodo,a[1,1]);a[1,2]:=subs(metodo,a[1,2]);a[2,1]:=subs(metodo,a[2,1]);a[2,2]:=subs(metodo,a[2,2]);b[1]:=subs(metodo,b[1]);b[2]:=subs(metodo,b[2]);
```

$$c_1 := \frac{1}{2} - \frac{\sqrt{3}}{6}$$

$$c_2 := \frac{1}{2} + \frac{\sqrt{3}}{6}$$

$$a_{1,1} := \frac{1}{4}$$

$$a_{1,2} := \frac{1}{4} - \frac{\sqrt{3}}{6}$$

$$a_{2,1} := \frac{1}{4} + \frac{\sqrt{3}}{6}$$

$$a_{2,2} := \frac{1}{4}$$

$$b_1 := \frac{1}{2}$$

$$b_2 := \frac{1}{2}$$

Significa esto que el método de GAUSS de 2 etapas es

$$\begin{array}{c|cc} \frac{3-\sqrt{3}}{6} & \frac{1}{4} & \frac{3-2\sqrt{3}}{12} \\ \frac{3+\sqrt{3}}{6} & \frac{3+2\sqrt{3}}{12} & \frac{1}{4} \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{2} & \frac{1}{2} \end{array}$$

funcion de estabilidad

> f := (x, y) -> lambda * y;

$$f := (x, y) \rightarrow \lambda y$$

> ka1 := unapply(f(x0+c[1]*h, y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)), h);

$$ka1 := h \rightarrow \lambda \left(y0 + \frac{1}{4} h kp1(h) + \left(\frac{1}{4} - \frac{\sqrt{3}}{6} \right) h kp2(h) \right)$$

> ka2 := unapply(f(x0+c[2]*h, y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)), h);

$$ka2 := h \rightarrow \lambda \left(y0 + \left(\frac{1}{4} + \frac{\sqrt{3}}{6} \right) h kp1(h) + \frac{1}{4} h kp2(h) \right)$$

> test := solve({k1=subs({kp1(h)=k1, kp2(h)=k2}, ka1(h)), k2=subs({kp1(h)=k1, kp2(h)=k2}, ka2(h))}, {k1, k2});

$$test := \left\{ k2 = \frac{2\sqrt{3}\lambda y0(\lambda h + 2\sqrt{3})}{12 - 6\lambda h + \lambda^2 h^2}, k1 = -\frac{2\sqrt{3}\lambda y0(-2\sqrt{3} + \lambda h)}{12 - 6\lambda h + \lambda^2 h^2} \right\}$$

> k1 := subs(test, k1); k2 := subs(test, k2);

$$k1 := -\frac{2\sqrt{3}\lambda y0(-2\sqrt{3} + \lambda h)}{12 - 6\lambda h + \lambda^2 h^2}$$

$$k2 := \frac{2\sqrt{3}\lambda y0(\lambda h + 2\sqrt{3})}{12 - 6\lambda h + \lambda^2 h^2}$$

> y1 := expand(y0+h*(b[1]*k1+b[2]*k2));

$$y1 := y0 + \frac{12 h \lambda y0}{12 - 6\lambda h + \lambda^2 h^2}$$

> y1 := collect(subs(lambda=z/h, y1), y0);

$$y1 := \left(1 + \frac{12 z}{12 - 6 z + z^2} \right) y0$$

> r := unapply(collect(simplify(y1/y0), z), z);

$$r := z \rightarrow \frac{12 + 6 z + z^2}{12 - 6 z + z^2}$$

que es la función de estabilidad del método de GAUSS de 2 etapas.

Como las raíces del numerador de r son $-3 + i\sqrt{3}$ y las del

denominador son $3 + i\sqrt{3}$, tenemos que

$$\frac{(z + 3 - i\sqrt{3})(z + 3 + i\sqrt{3})}{(z - 3 - i\sqrt{3})(z - 3 + i\sqrt{3})} = \frac{(z + 3)^2 + 3}{(z - 3)^2 + 3}$$

y es fácil ver que los complejos con $\text{Re}(z) < 0$ hacen menor el módulo del numerador que el del denominador, por lo que el método de GAUSS es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-04 (del EXAMEN ORDINARIO de 09FEB04)

Para los métodos de RUNGE-KUTTA implícitos del siguiente tablero

$$\begin{array}{c|cc} \frac{1}{3} & & \frac{1}{3} \\ \hline 1 & \alpha & 0 \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

- dígase qué valores de α proporcionan métodos de orden 2 ;
- dígase si existen valores de α que proporcionen métodos de orden 3 ;
- dígase si existen valores de α que proporcionen métodos de orden 4 .

(desde aquí es Ejercicio del capítulo 8)

- Calcúlese la función de estabilidad para los métodos de la familia.
- Discútase sobre la A-estabilidad de los métodos de la familia.
- Calcúlese la función de estabilidad para los métodos obtenidos en b) y c).

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+(1/3)*h,y0+(1/3)*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+h,y0+alpha*h*kp1_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

[

```

> Fi_0:=expand((3/4)*k1_0+(1/4)*k2_0);
                                     Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                     cond1a := 1 = 1

```

El orden es siempre de 1 al menos

condiciones para el orden 2

```

> Fi10:=expand((3/4)*k110+(1/4)*k210);
                                     Fi10 := 1/2 D1(f)(x0, y0) + 1/4 D2(f)(x0, y0) f(x0, y0) + 1/4 D2(f)(x0, y0) alpha f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a := 1/2 = 1/2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b := 1/2 = 1/4 + alpha/4
> solucion:=solve({cond2a,cond2b});
                                     solucion := {alpha = 1}

```

El valor $\alpha = 1$ es el único que proporciona orden 2

b) condiciones para el orden 3

```

> Fi20:=expand((3/4)*k120+(1/4)*k220);
                                     Fi20 := 1/3 D1,1(f)(x0, y0) + 1/6 D1,2(f)(x0, y0) f(x0, y0) + 1/12 D2,2(f)(x0, y0) f(x0, y0)^2 + 1/6 D2(f)(x0, y0) D1(f)(x0, y0)
                                     + 1/6 D2(f)(x0, y0)^2 f(x0, y0) + 1/2 D1,2(f)(x0, y0) alpha f(x0, y0) + 1/4 alpha^2 f(x0, y0)^2 D2,2(f)(x0, y0)
                                     + 1/6 D2(f)(x0, y0) alpha D1(f)(x0, y0) + 1/6 D2(f)(x0, y0)^2 alpha f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0)):cond3a:=subs(solucion,cond3a);
                                     cond3a := 1/6 = 1/6
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0)):cond3b:=subs(solucion,cond3b);
                                     cond3b := 1/3 = 1/3
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2):cond3c:=subs(solucion,cond3c);
                                     cond3c := 1/6 = 1/6
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0)):cond3d:=subs(solucion,cond3d);
                                     cond3d := 1/6 = 1/6
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0)):cond3e:=subs(solucion,cond3e);
                                     cond3e := 1/6 = 1/6

```

Con el mismo valor $\alpha = 1$ se obtiene orden 3

c) orden 4

En ningún caso aparece en la familia el método de GAUSS de dos etapas.

Por lo tanto, no se alcanza nunca el orden 4

d)

función de estabilidad

```

> f:=(x,y)->lambda*y;

```

```

[                                     f := (x, y) → λ y
[ > ka1:=unapply(f(x0+(1/3)*h,y0+(1/3)*h*kp1(h)),h);
[                                     ka1 := h → λ (y0 + 1/3 h kp1(h))
[ > test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
[                                     test := {k1 = - 3 λ y0 / (-3 + λ h)}
[ > k1:=subs(test,k1);
[                                     k1 := - 3 λ y0 / (-3 + λ h)
[ >
[ > k2:=f(x0+h,y0+alpha*h*k1);
[                                     k2 := λ (y0 - 3 α h λ y0 / (-3 + λ h))
[ >
[ >
[ > y1:=expand(y0+h*((3/4)*k1+(1/4)*k2));
[                                     y1 := y0 - 9 h λ y0 / (4(-3 + λ h)) + h λ y0 / 4 - 3 α h² λ² y0 / (4(-3 + λ h))
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 := (1 - 9 z / (4(-3 + z)) + z / 4 - 3 α z² / (4(-3 + z))) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[                                     r := z → - 1/4 (12 + (-1 + 3 α) z² + 8 z) / (-3 + z)

```

que es la función de estabilidad de la familia general propuesta

e)

función de estabilidad cuando $\alpha = \frac{1}{3}$ (única posible A-estabilidad)

```

[ > r_estab:=simplify(subs({alpha=1/3},r(z)));

```

$$r_{estab} := -\frac{3+2z}{-3+z}$$

expresión con límite 2 cuando z tiende a ∞ , por lo que no puede haber A-estabilidad

f)

función de estabilidad cuando $\alpha = 1$ (orden máximo 3)

```

[ > r_orden3:=simplify(subs(solucion,r(z)));

```

$$r_{orden3} := -\frac{6+z^2+4z}{2(-3+z)}$$

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-05 (del EXAMEN EXTRAORDINARIO de 14JUL04)

Para los métodos de RUNGE-KUTTA implícitos del siguiente tablero

$$\begin{array}{c|ccc} 1 & 1 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & a & -3 & 1 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

y teniendo en cuenta que para tales métodos de 3 etapas se puede obtener hasta orden 6, dígame cuál es el orden máximo que se obtiene realmente y cuáles son los valores del parámetro a para los que se alcanza dicho orden.

(desde aquí es Ejercicio del capítulo 8)

Calcúlese la función de estabilidad para los métodos de la familia y para los métodos de orden máximo.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+h,y0+h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+(1/2)*h,y0-(1/2)*h*kp1_(h)+h*kp2_(h)),h):
[ > k3_:=unapply(f(x0,y0+a*h*kp1_(h)-3*h*kp2_(h)+h*kp3_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k3_0:=k3_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k31:=D(k3_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k21(0)):
[ > k310:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k31(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[
```


[Nunca se obtiene el orden 3

[función de estabilidad

[> restart;

[> f:=(x,y)->lambda*y;

$$f := (x, y) \rightarrow \lambda y$$

[> ka1:=unapply(f(x0+h,y0+h*kp1(h)),h);

$$ka1 := h \rightarrow \lambda (y0 + h kp1(h))$$

[> test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);

$$test := \{k1 = -\frac{\lambda y0}{-1 + \lambda h}\}$$

[> k1:=subs(test,k1);

$$k1 := -\frac{\lambda y0}{-1 + \lambda h}$$

[> ka2:=simplify(unapply(f(x0+(1/2)*h,y0-(1/2)*h*k1+h*kp2(h)),h));

$$ka2 := h \rightarrow \lambda \left(y0 + \frac{1}{2} \frac{h \lambda y0}{-1 + \lambda h} + h kp2(h) \right)$$

[> test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);

$$test := \{k2 = -\frac{\lambda y0(-2 + 3 \lambda h)}{2(-1 + \lambda h)^2}\}$$

[> k2:=subs(test,k2);

$$k2 := -\frac{\lambda y0(-2 + 3 \lambda h)}{2(-1 + \lambda h)^2}$$

[> ka3:=simplify(unapply(f(x0,y0+a*h*k1-3*h*k2+h*kp3(h)),h));

$$ka3 := h \rightarrow \lambda \left(y0 - \frac{a h \lambda y0}{-1 + \lambda h} + \frac{3}{2} \frac{h \lambda y0(-2 + 3 \lambda h)}{(-1 + \lambda h)^2} + h kp3(h) \right)$$

[> test:=solve({k3=subs(kp3(h)=k3,ka3(h))},k3);

$$test := \{k3 = \frac{\lambda y0(-2 + 10 \lambda h - 11 \lambda^2 h^2 - 2 a h \lambda + 2 a h^2 \lambda^2)}{2(-1 + \lambda h)^3}\}$$

[> k3:=subs(test,k3);

$$k3 := \frac{\lambda y0(-2 + 10 \lambda h - 11 \lambda^2 h^2 - 2 a h \lambda + 2 a h^2 \lambda^2)}{2(-1 + \lambda h)^3}$$

[> y1:=expand(y0+h*((1/6)*k1+(2/3)*k2+(1/6)*k3));

$$y1 := y0 - \frac{h \lambda y0}{6(-1 + \lambda h)} + \frac{2 h \lambda y0}{3(-1 + \lambda h)^2} - \frac{\lambda^2 y0 h^2}{(-1 + \lambda h)^2} - \frac{h \lambda y0}{6(-1 + \lambda h)^3} + \frac{5 \lambda^2 y0 h^2}{6(-1 + \lambda h)^3} - \frac{11 \lambda^3 y0 h^3}{12(-1 + \lambda h)^3} - \frac{\lambda^2 y0 a h^2}{6(-1 + \lambda h)^3} + \frac{\lambda^3 y0 a h^3}{6(-1 + \lambda h)^3}$$

[> y1:=collect(subs(lambda=z/h,y1),y0);

$$y1 := \left(1 - \frac{z}{6(-1+z)} + \frac{2z}{3(-1+z)^2} - \frac{z^2}{(-1+z)^2} - \frac{z}{6(-1+z)^3} + \frac{5z^2}{6(-1+z)^3} - \frac{11z^3}{12(-1+z)^3} - \frac{z^2 a}{6(-1+z)^3} + \frac{z^3 a}{6(-1+z)^3} \right) y0$$

[> r:=unapply(collect(simplify(y1/y0),z),z);

$$r := z \rightarrow \frac{1}{12} \frac{-12 + (-13 + 2a)z^3 + (-2 - 2a)z^2 + 24z}{(-1+z)^3}$$

[que es la función de estabilidad de la familia general propuesta

[función de estabilidad cuando a = 2 (orden 2)

[> r_2:=simplify(subs({a=2},r(z)));

$$r_2 := -\frac{4 + 3z^3 + 2z^2 - 8z}{4(-1+z)^3}$$

[>

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-06 (del EXAMEN ORDINARIO de 18FEB05)

De entre los métodos implícitos de 2 etapas de tablero

$$0 \mid \alpha \quad 0$$

$$1 \mid \alpha \quad 0$$

$$\mid \quad \frac{1}{2} \quad \frac{1}{2}$$

entre los que se encuentra, para $\alpha = 1/2$, el llamado LOBATTO IIIb, se buscarán los que sean de mayor orden.

(desde aquí es Ejercicio del capítulo 8)

Calcúlese la función de estabilidad de la familia general de los métodos

del tablero y discútase la A-estabilidad del método LOBATTO IIIb y de los otros

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > c[1]:=0:c[2]:=1:
[ > a[1,1]:=alpha:a[1,2]:=0:a[2,1]:=alpha:a[2,2]:=0:
[ > b[1]:=1/2:b[2]:=1/2:
[ > k10:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp10(h)+a[1,2]*h*kp20(h)),h):
[ > k20:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp10(h)+a[2,2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > Fi0:=expand(b[1]*k100+b[2]*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > Fi1:=expand(b[1]*k110+b[2]*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
```

```

[ xexpand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
[ xexpand(k220):
[ > Fi2:=expand(b[1]*k120+b[2]*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[ kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[ kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand(b[1]*k130+b[2]*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
[ cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
[ cond2a :=  $\frac{1}{2} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
[ ,f(x0,y0)));
[ cond2b :=  $\alpha = \frac{1}{2}$ 
[ orden 2 para  $\alpha = 1/2$ , o sea, LOBATTO IIIb
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));subs(alpha=1/2,co
[ nd3a);
[ cond3a :=  $\frac{1}{2} = \frac{1}{3}$ 
[  $\frac{1}{2} = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
[ y0)),f(x0,y0));subs(alpha=1/2,cond3b);
[ cond3b :=  $\alpha = \frac{2}{3}$ 
[  $\frac{1}{2} = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
[ 0,y0)),f(x0,y0)^2);subs(alpha=1/2,cond3c);
[ cond3c :=  $\alpha^2 = \frac{1}{3}$ 
[  $\frac{1}{4} = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
[ 0,y0)),D[1](f)(x0,y0));subs(alpha=1/2,cond3d);
[ cond3d :=  $0 = \frac{1}{3}$ 
[  $0 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0
[ )^2),f(x0,y0));subs(alpha=1/2,cond3e);
[ cond3e :=  $2\alpha^2 = \frac{1}{3}$ 
[  $\frac{1}{2} = \frac{1}{3}$ 
[ no hay orden 3 para ningun valor de  $\alpha$ 
[ # funcion de estabilidad
[ > f:=(x,y)->lambda*y;

```

```

[
> f := (x, y) → λ y
> ka1 := unapply(f(x0+c[1]*h, y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)), h);
      ka1 := h → λ (y0 + α h kp1(h))
> ka2 := unapply(f(x0+c[2]*h, y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)), h);
      ka2 := h → λ (y0 + α h kp1(h))
> test := solve({k1=subs({kp1(h)=k1, kp2(h)=k2}, ka1(h)), k2=subs({kp1(h)=k1, kp2(h)=k2}, ka2(h))}, {k1, k2});
      test := {k1 = -λ y0 / (-1 + λ α h), k2 = -λ y0 / (-1 + λ α h)}
> k1 := subs(test, k1); k2 := subs(test, k2);
      k1 := -λ y0 / (-1 + λ α h)
      k2 := -λ y0 / (-1 + λ α h)
> y1 := expand(y0+h*(b[1]*k1+b[2]*k2));
      y1 := y0 - h λ y0 / (-1 + λ α h)
> y1 := collect(subs(lambda=z/h, y1), y0);
      y1 := (1 - z / (-1 + z α)) y0
> r := unapply(collect(simplify(y1/y0), z), z);
      r := z → (-1 + (α - 1) z) / (-1 + z α)
# A-estabilidad para el método LOBATTO IIIb
> simplify(subs(alpha=1/2, r(z)));
      - (2 + z) / (-2 + z)

```

El dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método LOBATTO IIIb es A-estable

A-estabilidad para los otros métodos

En general, como hemos visto

$$r(z) = \frac{1 + (1 - \alpha)z}{1 - \alpha z}$$

Supondremos $\alpha \neq \frac{1}{2}$ y pondremos que $z = x + iy$.

Como

$$1 + (1 - \alpha)z = 1 + (1 - \alpha)x + i(1 - \alpha)y$$

y

$$1 - \alpha z = 1 - \alpha x + i\alpha y,$$

resulta

$$|1 + (1 - \alpha)z|^2 = (1 + (1 - \alpha)x)^2 + (1 - \alpha)^2 y^2$$

y

$$|1 - \alpha z|^2 = (1 - \alpha x)^2 + \alpha^2 y^2$$

por lo que la curva $|r(z)|^2 = 1$ se escribe

$$(1 + (1 - \alpha)x)^2 + (1 - \alpha)^2 y^2 = (1 - \alpha x)^2 + \alpha^2 y^2$$

o sea

$$(1 - 2\alpha)y^2 + (1 - 2\alpha)x^2 + 2x = 0$$

$$x^2 + \frac{2}{1 - 2\alpha}x + y^2 = 0$$

La curva límite de la region de estabilidad pasa por (0,0)

Supongamos primeramente que $\alpha < \frac{1}{2}$ o sea $1 - 2\alpha > 0$

La curva es

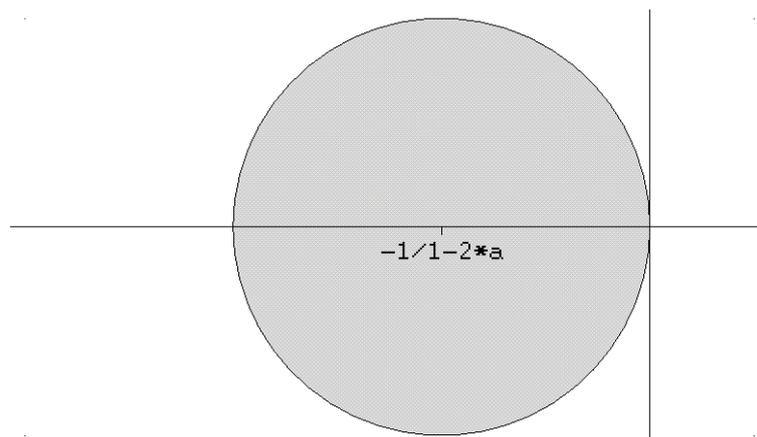
$$\left(x + \frac{1}{1 - 2\alpha}\right)^2 + y^2 = \frac{1}{(1 - 2\alpha)^2}$$

o sea, una circunferencia de centro $-\frac{1}{1 - 2\alpha}$ y radio $\frac{1}{1 - 2\alpha}$ que pasa por (0,0)

Como $z=1=(1,0)$ verifica $|r(z)| = \left|\frac{2 - \alpha}{1 - \alpha}\right| > 1$ resulta que el dominio de estabilidad

es el interior de la circunferencia

```
> with(plottools):with(plots):macro(gris=COLOR(RGB,.8,.8,.8)):exten:=point([-3,-1],symbol=point),point([-3,1],symbol=point),point([0.5,1],symbol=point),point([0.5,-1],symbol=point):noAest:=disk([-1,0],1,color=gris):display(exten,noAest,axes=normal,scaling=constrained,tickmarks=[[-1="-1/1-2*a"],[0,0]],axesfont=[COURIER,10]);
>
```



y el método no es A-estable.

Supongamos ahora que $\frac{1}{2} < \alpha$ o sea $1 - 2\alpha < 0$ y $2\alpha - 1 > 0$

La curva es

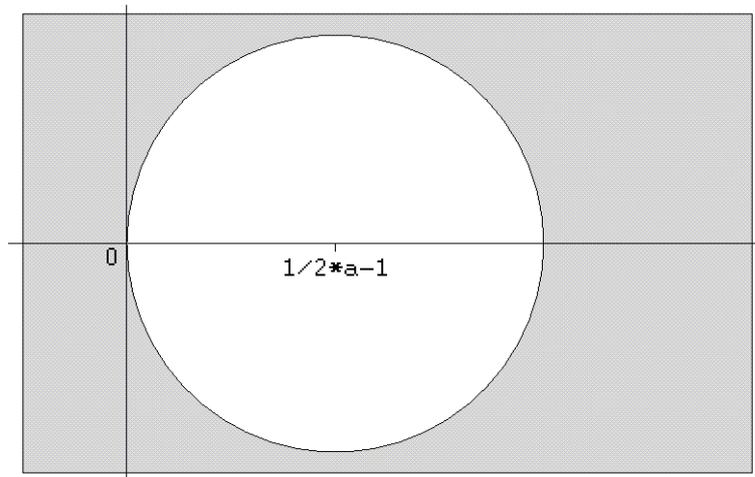
$$\left(x - \frac{1}{2\alpha - 1}\right)^2 + y^2 = \frac{1}{(2\alpha - 1)^2}$$

o sea, una circunferencia de centro $\frac{1}{2\alpha - 1}$ y radio $\frac{1}{2\alpha - 1}$ que pasa por $(0,0)$

Como $z = -1 = (-1, 0)$ verifica $|r(z)| = \left|\frac{\alpha}{1 + \alpha}\right| < 1$ resulta que el dominio de estabilidad

es el exterior de la circunferencia

```
> exten:=point([-0.5,-1.1],symbol=point),point([-0.5,1.1],symbol=point),point([3,1.1],symbol=point),point([3,-1.1],symbol=point):plano:= rectangle([-0.5,-1.1],[3,1.1],color=gris):noAest:= disk([1/2*a-1],1,color="White"):display(exten,noAest,plano,axes=normal,scaling=constrained,tickmarks=[[1="1/2*a-1"],[0,0]],axesfont=[COURIER,10])
>
```



[>
[>

[y los métodos son ahora A-estables.

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-07 (del EXAMEN EXTRAORDINARIO de 04JUL06)

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

$$\begin{array}{c|cc} 0 & & 0 \\ 2\alpha & \alpha & \alpha \\ \hline & \beta & \beta \end{array}$$

donde $0 < \alpha$ y $0 < \beta$

(desde aquí es Ejercicio del capítulo 8)

b) Calcúlese la función de estabilidad para los métodos de la familia propuesta.

También para los métodos del orden máximo alcanzable

c) Compruébese que dichos métodos de orden máximo son A-estables. Búsquese algún otro método de la familia que también sea A-estable, describiendo en este caso la región de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0,y0),h):
[ > k2_:=unapply(f(x0+2*alpha*h,y0+alpha*h*kp1_(h)+alpha*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

[

```

> Fi_0:=expand(beta*k1_0+beta*k2_0);
                                     Fi_0 := 2 β f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                     cond1a := 1 = 2 β
> solve(cond1a,{beta});
                                     { β = 1/2 }

condiciones para el orden 2
> Fi10:=expand(beta*k110+beta*k210);
                                     Fi10 := 2 β D1(f)(x0, y0) α + 2 β D2(f)(x0, y0) α f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a := 1/2 = 2 β α
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b := 1/2 = 2 β α
> solucion:=solve({cond1a,cond2a,cond2b},{alpha,beta});
                                     solucion := { β = 1/2, α = 1/2 }

condiciones para el orden 3
> Fi20:=expand(beta*k120+beta*k220);
Fi20 := 4 β D1,1(f)(x0, y0) α2 + 8 β D1,2(f)(x0, y0) α2 f(x0, y0) + 4 β α2 f(x0, y0)2 D2,2(f)(x0, y0)
      + 4 β D2(f)(x0, y0) α2 D1(f)(x0, y0) + 4 β D2(f)(x0, y0)2 α2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
                                     cond3a := 1/6 = 2 β α2
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
                                     cond3b := 1/3 = 4 β α2
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
                                     cond3c := 1/6 = 2 β α2
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
                                     cond3d := 1/6 = 2 β α2
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
                                     cond3e := 1/6 = 2 β α2
> solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});

```

Es decir, no se alcanza orden 3. El orden máximo alcanzable es 2 y se alcanza con el método dado por el tablero

$$0 \quad | \quad 0$$

$$1 \quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

```
> r2:=unapply(simplify(subs(solucion2,r(z))),z);
```

$$r2 := z \rightarrow -\frac{1}{-1+z}$$

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda h$ cuya distancia al punto -1 sea superior a 1 , que es el exterior de la circunferencia de centro -1 y radio 1 (que pasa, pues, por el origen). Esta región contiene al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-08 (del EXAMEN EXTRAORDINARIO DE 15JUL09)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{l|ll} 0 & \alpha & -\alpha \\ 2 & \alpha & 2-\alpha \end{array}$$

$$\begin{array}{l|ll} & \frac{1}{4} & \frac{3}{4} \end{array}$$

- a) Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α para los que se consigue (desde aquí es Ejercicio del capítulo 8)
- b) Calcúlese la función de estabilidad del método en el caso general.
- c) Búsqese un valor de α para el que el correspondiente método sea A-estable, probando, además, dicha A-estabilidad.

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0);
[                                     f0 := f(x0, y0)
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1);
[                                     f1 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2);
[ f2 := D1,1(f)(x0, y0) + 2 D1,2(f)(x0, y0) f(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)2 + D2(f)(x0, y0) D1(f)(x0, y0)
[       + D2(f)(x0, y0)2 f(x0, y0)
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3);
[ f3 := D1,1,1(f)(x0, y0) + 3 D1,1,2(f)(x0, y0) f(x0, y0) + 3 D1,2,2(f)(x0, y0) f(x0, y0)2 + 3 D1,2(f)(x0, y0) D1(f)(x0, y0)
[       + 5 D1,2(f)(x0, y0) D2(f)(x0, y0) f(x0, y0) + D2,2,2(f)(x0, y0) f(x0, y0)3 + 3 f(x0, y0) D2,2(f)(x0, y0) D1(f)(x0, y0)
[       + 4 D2,2(f)(x0, y0) D2(f)(x0, y0) f(x0, y0)2 + D2(f)(x0, y0) D1,1(f)(x0, y0) + D2(f)(x0, y0)2 D1(f)(x0, y0)
[       + D2(f)(x0, y0)3 f(x0, y0)
[ # desarrollo del método
[ > k10:=unapply(f(x0,y0+alpha*h*kp10(h)-alpha*h*kp20(h)),h);
[                                     k10 := h → f(x0, y0 + α h kp10(h) - α h kp20(h))
[ > k20:=unapply(f(x0+2*h,y0+alpha*h*kp10(h)+(2-alpha)*h*kp20(h)),h);
[                                     k20 := h → f(x0 + 2 h, y0 + α h kp10(h) + (2 - α) h kp20(h))
```

```

> k100:=k10(0):k100:=expand(k100);
                                k100 := f(x0, y0)
> k200:=k20(0):k200:=expand(k200);
                                k200 := f(x0, y0)
> Fi0:=expand((1/4)*k100+(3/4)*k200);
                                Fi0 := f(x0, y0)
> k11:=D(k10):
> k21:=D(k20):
> k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110);
                                k110 := 0
> k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210);
                                k210 := 2 D1(f)(x0, y0) + 2 D2(f)(x0, y0) f(x0, y0)
> Fil:=expand((1/4)*k110+(3/4)*k210);
                                Fil := 3/2 D1(f)(x0, y0) + 3/2 D2(f)(x0, y0) f(x0, y0)
> k12:=D(k11):
> k22:=D(k21):
> k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
xpand(k120);
                                k120 := -4 D2(f)(x0, y0) α D1(f)(x0, y0) - 4 D2(f)(x0, y0)2 α f(x0, y0)
> k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
xpand(k220);
k220 := 4 D1,1(f)(x0, y0) + 8 D1,2(f)(x0, y0) f(x0, y0) + 4 D2,2(f)(x0, y0) f(x0, y0)2 + 8 D2(f)(x0, y0) D1(f)(x0, y0)
+ 8 D2(f)(x0, y0)2 f(x0, y0) - 4 D2(f)(x0, y0) α D1(f)(x0, y0) - 4 D2(f)(x0, y0)2 α f(x0, y0)
> Fi2:=expand((1/4)*k120+(3/4)*k220);
Fi2 := -4 D2(f)(x0, y0) α D1(f)(x0, y0) - 4 D2(f)(x0, y0)2 α f(x0, y0) + 3 D1,1(f)(x0, y0) + 6 D1,2(f)(x0, y0) f(x0, y0)
+ 3 D2,2(f)(x0, y0) f(x0, y0)2 + 6 D2(f)(x0, y0) D1(f)(x0, y0) + 6 D2(f)(x0, y0)2 f(x0, y0)
# consistencia y orden 1
> cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
                                cond1a := 1 = 1
# condiciones para el orden 2
> cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
                                cond2a := 3/2 = 1/2
> cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
,f(x0,y0));
                                cond2b := 3/2 = 1/2
sólo se obtiene orden 1 y eso para cualquier valor de α

# funcion de estabilidad general
> restart:
> interface(labeling=false):
> f:=(x,y)->lambda*y;
                                f := (x, y) → λ y
> ka1:=unapply(f(x0,y0+alpha*h*kp1(h)-alpha*h*kp2(h)),h);
                                ka1 := h → λ (y0 + α h kp1(h) - α h kp2(h))
> ka2:=unapply(f(x0+2*h,y0+alpha*h*kp1(h)+(2-alpha)*h*kp2(h)),h);
                                ka2 := h → λ (y0 + α h kp1(h) + (2 - α) h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h)
)}),{k1,k2});
                                test := {k1 = -λ y0 (-1 + 2 λ h) / (1 - 2 λ h + 2 λ2 h2 α), k2 = λ y0 / (1 - 2 λ h + 2 λ2 h2 α)}
> k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);

```

$$k1 := -\frac{\lambda y0 (-1 + 2 \lambda h)}{1 - 2 \lambda h + 2 \lambda^2 h^2 \alpha}$$

$$k2 := \frac{\lambda y0}{1 - 2 \lambda h + 2 \lambda^2 h^2 \alpha}$$

$$-\frac{2 \lambda y0 (-2 + \lambda h)}{1 - 2 \lambda h + 2 \lambda^2 h^2 \alpha}$$

```
> y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
```

$$y1 := y0 + \frac{h \lambda y0}{1 - 2 \lambda h + 2 \lambda^2 h^2 \alpha} - \frac{\lambda^2 y0 h^2}{2 (1 - 2 \lambda h + 2 \lambda^2 h^2 \alpha)}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \frac{z}{1 - 2 z + 2 z^2 \alpha} - \frac{z^2}{2 (1 - 2 z + 2 z^2 \alpha)} \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z):r_general:=r(z);
```

$$r_general := \frac{2 + (4 \alpha - 1) z^2 - 2 z}{2 (1 - 2 z + 2 z^2 \alpha)}$$

```
# parece oportuno tomar  $\alpha = \frac{1}{4}$ 
```

```
> alpha:=1/4;num:=numer(r(z));den:=denom(r(z));
```

$$\alpha := \frac{1}{4}$$

$$num := 2 - 2 z$$

$$den := 2 - 4 z + z^2$$

```
> z:=x+I*y:assume(x::real):assume(y::real):
```

```
> numr:=Re(num);numi:=Im(num);denr:=Re(den);deni:=Im(den);
```

$$numr := 2 - 2 x\sim$$

$$numi := -2 y\sim$$

$$denr := 2 - 4 x\sim + x\sim^2 - y\sim^2$$

$$deni := -4 y\sim + 2 x\sim y\sim$$

```
> simplify(numr^2+numi^2);simplify(denr^2+deni^2);
```

$$4 - 8 x\sim + 4 x\sim^2 + 4 y\sim^2$$

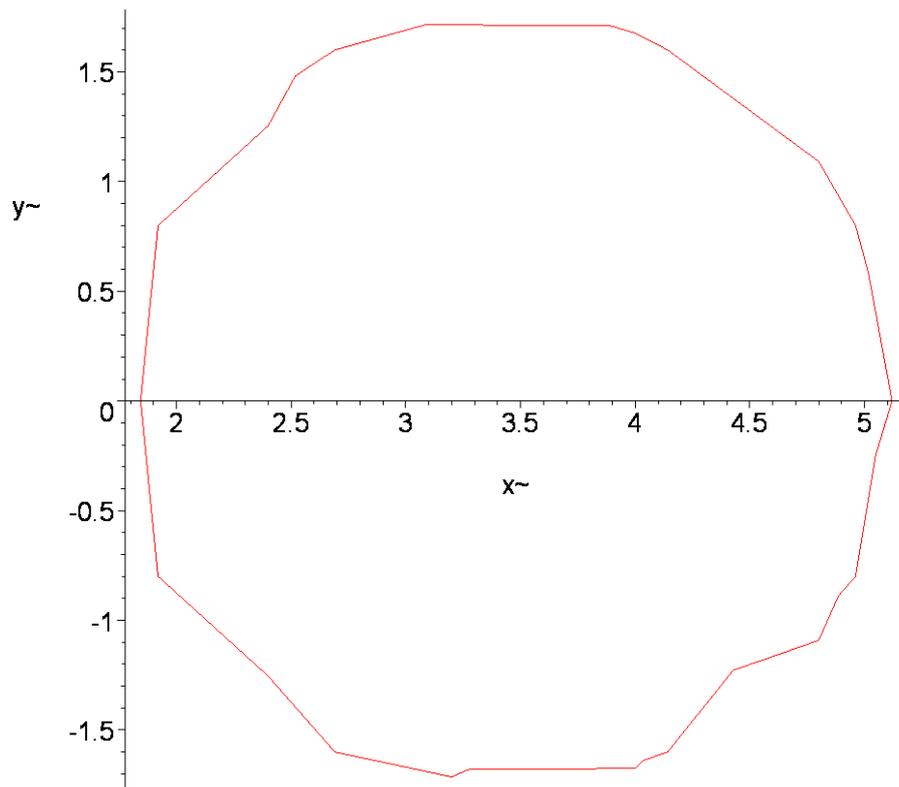
$$4 - 16 x\sim + 20 x\sim^2 + 12 y\sim^2 - 8 x\sim^3 - 8 x\sim y\sim^2 + x\sim^4 + 2 x\sim^2 y\sim^2 + y\sim^4$$

```
> borde:=simplify(numr^2+numi^2-denr^2-deni^2);
```

$$borde := 8 x\sim - 16 x\sim^2 - 8 y\sim^2 + 8 x\sim^3 + 8 x\sim y\sim^2 - x\sim^4 - 2 x\sim^2 y\sim^2 - y\sim^4$$

```
> with(plots):
```

```
> implicitplot(borde=0,x=-40..40,y=-40..40, gridrefine=2);
```



Debe hacerse notar que, cuando $x < 0$, entonces $\text{borde} < 0$, o sea, el borde de la región de estabilidad se encuentra en el semiplano $\text{Re}(z) > 0$

$> r(-1)$;

$$\frac{4}{7}$$

En el caso $\alpha = \frac{1}{4}$ la región de estabilidad contiene todo el semiplano $\text{Re}(z) < 0$,

por lo que el método es A-estable. Se trata del método

$$0 \mid \frac{1}{4} \quad -\frac{1}{4}$$

$$2 \mid \frac{1}{4} \quad \frac{5}{4}$$

$$\mid \frac{1}{4} \quad \frac{3}{4}$$

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-09 (del EXAMEN ORDINARIO de 01FEB07)

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

α		α	
1		$\frac{1}{2}$	$\frac{1}{2}$

		β	β
--	--	---------	---------

donde α y β son parámetros arbitrarios

(desde aquí es Ejercicio del capítulo 8)

b) Calcúlese la función de estabilidad para los métodos de orden máximo hallados.

¿Son A-estables dichos métodos de orden máximo?.

c) Búsquese algún método de orden 1 de la familia que sea A-estable,

describiendo en este caso la región de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+alpha*h,y0+alpha*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+h,y0+(1/2)*h*kp1_(h)+(1/2)*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

```

> Fi_0:=expand(beta*k1_0+beta*k2_0);
                                     Fi_0 := 2 β f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                     cond1a := 1 = 2 β
> solve(cond1a,{beta});
                                     { β = 1/2 }

condiciones para el orden 2
> Fi10:=expand(beta*k110+beta*k210);
                                     Fi10 := β D1(f)(x0, y0) α + β D2(f)(x0, y0) α f(x0, y0) + β D1(f)(x0, y0) + β D2(f)(x0, y0) f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a := 1/2 = β α + β
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b := 1/2 = β α + β
> solucion:=solve({cond1a,cond2a,cond2b},{alpha,beta});
                                     solucion := { β = 1/2, α = 0 }

condiciones para el orden 3
> Fi20:=expand(beta*k120+beta*k220);
Fi20 := β D1,1(f)(x0, y0) α2 + 2 β D1,2(f)(x0, y0) α2 f(x0, y0) + β α2 f(x0, y0)2 D2,2(f)(x0, y0)
+ 2 β D2(f)(x0, y0) α2 D1(f)(x0, y0) + 2 β D2(f)(x0, y0)2 α2 f(x0, y0) + β D1,1(f)(x0, y0) + 2 β D1,2(f)(x0, y0) f(x0, y0)
+ β D2,2(f)(x0, y0) f(x0, y0)2 + β D2(f)(x0, y0) D1(f)(x0, y0) α + β D2(f)(x0, y0)2 α f(x0, y0)
+ β D2(f)(x0, y0) D1(f)(x0, y0) + β D2(f)(x0, y0)2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
                                     cond3a := 1/6 = 1/2 β α2 + 1/2 β
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
                                     cond3b := 1/3 = β α2 + β
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
                                     cond3c := 1/6 = 1/2 β α2 + 1/2 β
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
                                     cond3d := 1/6 = β α2 + 1/2 β α + 1/2 β
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
                                     cond3e := 1/6 = β α2 + 1/2 β α + 1/2 β
> subs(solucion,[cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e]);
                                     [ 1 = 1, 1/2 = 1/2, 1/2 = 1/2, 1/6 = 1/6, 1/3 = 1/3, 1/2 = 1/2, 1/6 = 1/6, 1/4 = 1/4, 1/4 = 1/4 ]
> solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});

```

Es decir, no se alcanza orden 3. El orden máximo alcanzable es 2 y se alcanza con el método dado por el tablero

$$1 \quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

$$| \quad \frac{1}{2} \quad \frac{1}{2}$$

b)

función de estabilidad para el método de orden 2 máximo

Cuando α vale 0 y β vale $1/2$, o sea, en el único caso de orden máximo 2, la función de estabilidad vale

```

> f:=(x,y)->lambda*y;
                                f := (x, y) -> lambda y
> ka1:=unapply(subs(solucion,f(x0+alpha*h,y0+alpha*h*kp1(h))),h);
                                ka1 := h -> lambda y0
> ka2:=unapply(subs(solucion,f(x0+h,y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))),h);
                                ka2 := h -> lambda (y0 + 1/2 h kp1(h) + 1/2 h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                test := {k1 = lambda y0, k2 = - lambda y0 (2 + lambda h) / (-2 + lambda h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
                                k1 := lambda y0
                                k2 := - lambda y0 (2 + lambda h) / (-2 + lambda h)
> y1:=expand(subs(solucion,y0+h*(beta*k1+beta*k2)));
                                y1 := y0 + h lambda y0 / 2 - h lambda y0 / (-2 + lambda h) - lambda^2 y0 h^2 / (2 (-2 + lambda h))
> y1:=collect(subs(lambda=z/h,y1),y0);
                                y1 := (1 + z/2 - z/(-2+z) - z^2/(2(-2+z))) y0
> r:=unapply(collect(simplify(y1/y0),z),z):rz:=simplify(r(z));
                                rz := - (2+z) / (-2+z)

```

el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

c)

función de estabilidad para un método de orden exactamente 1

Tomemos ahora cualquiera de los casos para los que $\beta = \frac{1}{2}$, de manera que el orden sea 1 al menos. Vamos a ensayar con el valor $\alpha = 1$, para el que no se tiene orden 2 (máximo)

```

> restart;
> solucion1:={alpha=1,beta=1/2};
                                solucion1 := {alpha = 1, beta = 1/2}
> f:=(x,y)->lambda*y;
                                f := (x, y) -> lambda y
> ka1:=unapply(subs(solucion1,f(x0+alpha*h,y0+alpha*h*kp1(h))),h);
                                ka1 := h -> lambda (y0 + h kp1(h))

```

```

> ka2:=unapply(subs(solucion1,f(x0+h,y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))),h);
      ka2 := h → λ (y0 + 1/2 h kp1(h) + 1/2 h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -λ y0 / (-1 + λ h), k2 = -λ y0 / (-1 + λ h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -λ y0 / (-1 + λ h)
      k2 := -λ y0 / (-1 + λ h)
> y1:=expand(subs(solucion1,y0+h*(beta*k1+beta*k2)));
      y1 := y0 - h λ y0 / (-1 + λ h)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-1 + z)) y0
> r:=unapply(collect(simplify(y1/y0),z),z):rz:=simplify(r(z));
      rz := -1 / (-1 + z)

```

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda h$ cuya distancia al punto -1 sea superior a 1 , que es el exterior de la circunferencia de centro -1 y radio 1 (que pasa, pues, por el origen). Esta región contiene al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Otra opción: tomemos $\beta = \frac{1}{2}$, y $\alpha = \frac{1}{2}$, para el que tampoco se tiene orden 2 (máximo)

```

> restart;
> solucion1:={alpha=1/2,beta=1/2};
      solucion1 := {α = 1/2, β = 1/2}
> f:=(x,y)->lambda*y;
      f := (x, y) → λ y
> ka1:=unapply(subs(solucion1,f(x0+alpha*h,y0+alpha*h*kp1(h))),h);
      ka1 := h → λ (y0 + 1/2 h kp1(h))
> ka2:=unapply(subs(solucion1,f(x0+h,y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))),h);
      ka2 := h → λ (y0 + 1/2 h kp1(h) + 1/2 h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -2 λ y0 / (-2 + λ h), k2 = 4 λ y0 / (4 - 4 λ h + λ² h²)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -2 λ y0 / (-2 + λ h)
      k2 := 4 λ y0 / (4 - 4 λ h + λ² h²)
> y1:=expand(subs(solucion1,y0+h*(beta*k1+beta*k2)));
      y1 := y0 - h λ y0 / (-2 + λ h) + 2 h λ y0 / (4 - 4 λ h + λ² h²)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-2 + z) + 2 z / (4 - 4 z + z²)) y0

```

```
> r:=unapply(collect(simplify(y1/y0),z),z):rz:=factor(simplify(r(z)));
```

$$rz := \frac{4}{(-2+z)^2}$$

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda$ h cuya distancia al punto -2 sea superior a 2 , que es el exterior de la circunferencia de centro -2 y radio 2 (que pasa, pues, por el origen). Esta región también contiene al semiplano de parte real negativa, por lo que el método también es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-10 (del EXAMEN ORDINARIO DE 09FEB09)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\alpha \quad | \quad \alpha$$
$$2\beta \quad | \quad \beta \quad \beta$$

$$| \quad \frac{1}{4} \quad \frac{3}{4}$$

Búsquese el mayor orden alcanzable por los miembros de esta familia, indicando

los valores de α y de β para los que se consigue

(desde aquí es Ejercicio del capítulo 8)

Quedádonos ahora (por simplicidad) con la situación $\alpha = 0$, calcúlese la función

de estabilidad del método en estos casos

Estúdiense la A-estabilidad de estos métodos para los que $\alpha = 0$

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+alpha*h*kp10(h)),h):
[ > k20:=unapply(f(x0+2*beta*h,y0+beta*h*kp10(h)+beta*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
xpand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
xpand(k220):
[
```

```

[ > Fi2:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[   kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[   kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
[                                     cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
[                                     cond2a :=  $\frac{\alpha}{4} + \frac{3\beta}{2} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
[   ,f(x0,y0)));
[                                     cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{2} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a});
[                                     orden2 := {  $\alpha = -6\beta + 2, \beta = \beta$  }
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
[                                     cond3a :=  $\frac{\alpha^2}{4} + 3\beta^2 = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
[   y0)),f(x0,y0));
[                                     cond3b :=  $\frac{\alpha^2}{2} + 6\beta^2 = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
[   0,y0)),f(x0,y0)^2);
[                                     cond3c :=  $\frac{\alpha^2}{4} + 3\beta^2 = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
[   0,y0)),D[1](f)(x0,y0));
[                                     cond3d :=  $\frac{1}{2}\alpha^2 + \frac{3}{2}\beta\alpha + 3\beta^2 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0
[   )^2),f(x0,y0));
[                                     cond3e :=  $\frac{1}{2}\alpha^2 + \frac{3}{2}\beta\alpha + 3\beta^2 = \frac{1}{3}$ 
[ > orden3:=solve({cond2a,cond3a,cond3d});
[                                     orden3 := {  $\beta = \frac{1}{3}, \alpha = 0$  }
[ se obtiene orden 3 para los valores alpha = 0 y beta = 1/3
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
[                                     cond4a :=  $\frac{\alpha^3}{4} + 6\beta^3 = \frac{1}{4}$ 
[ > cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
[   (x0,y0)),f(x0,y0));
[                                     cond4b :=  $18\beta^3 + \frac{3\alpha^3}{4} = \frac{3}{4}$ 
[ > cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](
[   f)(x0,y0)),f(x0,y0)^2);
[                                     cond4c :=  $18\beta^3 + \frac{3\alpha^3}{4} = \frac{3}{4}$ 

```

```

> cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0));

cond4d := 18 β³ + 3/2 α³ + 9 β² α = 3/4

> cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));

cond4e := 3 α³ + 36 β³ + 9 β² α + 9/2 β α² = 5/4

> cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);

cond4f := α³/4 + 6 β³ = 1/4

> cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));

cond4g := 18 β³ + 3/2 α³ + 9 β² α = 3/4

> cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);

cond4h := 9 β² α + 9/4 β α² + 9/4 α³ + 27 β³ = 1

> cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));

cond4i := 9 β³ + 3/4 α³ + 9/4 β α² = 1/4

> cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);

cond4j := 3/2 α³ + 9 β³ + 9/2 β α² + 9/2 β² α = 1/4

> cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));

cond4k := 3/2 α³ + 9 β³ + 9/2 β α² + 9/2 β² α = 1/4

> subs(ordena3,{cond4a,cond4b,cond4c,cond4d,cond4e,cond4f,cond4g,cond4h,cond4i,cond4j,cond4k});

{1/3 = 1/4, 1 = 1, 2/3 = 3/4, 2/9 = 1/4, 3/4 = 5/4}

```

y no se puede alcanzar orden 4 , lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = 0$ y $\beta = \frac{1}{3}$, o sea

para el método

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

que suele recibir el nombre específico de RADAU I de 2 etapas

```

[ # funcion de estabilidad para  $\alpha=0$ 
[ > restart:
[ > interface(labeling=false):
[ > alpha:=0:
[ > f:=(x,y)->lambda*y;
[                                     f:=(x,y) ->  $\lambda y$ 
[ > alpha:=0;
[                                      $\alpha := 0$ 
[ > ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
[                                     ka1 :=  $h \rightarrow \lambda y0$ 
[ > ka2:=unapply(f(x0+2*beta*h,y0+beta*h*kp1(h)+beta*h*kp2(h)),h);
[                                     ka2 :=  $h \rightarrow \lambda (y0 + \beta h kp1(h) + \beta h kp2(h))$ 
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                                     test := {  $k1 = \lambda y0, k2 = -\frac{\lambda y0 (1 + \lambda \beta h)}{-1 + \lambda \beta h}$  }
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
[                                     k1 :=  $\lambda y0$ 
[                                     k2 :=  $-\frac{\lambda y0 (1 + \lambda \beta h)}{-1 + \lambda \beta h}$ 
[                                      $-\frac{2 \lambda y0 (2 + \lambda \beta h)}{-1 + \lambda \beta h}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                     y1 :=  $y0 + \frac{h \lambda y0}{4} - \frac{3 h \lambda y0}{4 (-1 + \lambda \beta h)} - \frac{3 \lambda^2 y0 \beta h^2}{4 (-1 + \lambda \beta h)}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 :=  $\left(1 + \frac{z}{4} - \frac{3z}{4(-1+z\beta)} - \frac{3z^2\beta}{4(-1+z\beta)}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_general:=r(z);
[                                     r_general :=  $-\frac{2+z^2\beta+(-2\beta+2)z}{2(-1+z\beta)}$ 

```

[# A-estabilidad para $\alpha=0$

Estos métodos (incluyendo el ya citado RADAU I de 2 etapas nunca pueden ser A-estables, ya que el numerador de $r(z)$ es de grado 2 mientras que el denominador lo es de 1

La excepción podría ser $\beta = 0$ En este último caso, la función de estabilidad vale

$$r(z)=1+z$$

por lo que tampoco se tiene A-estabilidad. Se observará que este último método concreto,

para el que $\alpha = 0$ y $\beta = 0$ es el método de tablero

$$\begin{array}{c|c}
 0 & 0 \\
 0 & 0 \quad 0 \\
 \hline
 & \frac{1}{4} \quad \frac{3}{4}
 \end{array}$$

que es de hecho el método explícito de EULER (de 1 etapa)

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-11 (del EXAMEN ORDINARIO DE 09FEB10)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} \alpha & \alpha - 1 & 1 \\ \beta & \beta & 0 \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando

los valores de α y de β para los que se consigue

(desde aquí es Ejercicio del capítulo 8)

Considérese un ejemplo de los de mayor orden y búsqese en ese caso

la función de estabilidad del método

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp10(h)+h*kp20(h)),h):
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp10(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
x
pand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
x
pand(k220):
[ > F12:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[
```

```

[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
                                     cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
                                     cond2a :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0))
  ,f(x0,y0));
                                     cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a});
                                     orden2 := { $\alpha = -3\beta + 2, \beta = \beta$ }
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
                                     cond3a :=  $\frac{\alpha^2}{4} + \frac{3\beta^2}{4} = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
  y0)),f(x0,y0));
                                     cond3b :=  $\frac{\alpha^2}{2} + \frac{3\beta^2}{2} = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
  0,y0)),f(x0,y0)^2);
                                     cond3c :=  $\frac{\alpha^2}{4} + \frac{3\beta^2}{4} = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
  0,y0)),D[1](f)(x0,y0));
                                     cond3d :=  $\frac{1}{2}\alpha^2 - \frac{1}{2}\alpha + \frac{1}{2}\beta + \frac{3}{2}\beta\alpha = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
  ^2),f(x0,y0));
                                     cond3e :=  $\frac{1}{2}\alpha^2 - \frac{1}{2}\alpha + \frac{1}{2}\beta + \frac{3}{2}\beta\alpha = \frac{1}{3}$ 
[ > orden3:=solve({cond2a,cond3a,cond3d});
                                     orden3 := { $\alpha = 0, \beta = \frac{2}{3}$ }
[ se obtiene orden 3 para los valores alpha = 0 y beta = 2/3
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
                                     cond4a :=  $\frac{\alpha^3}{4} + \frac{3\beta^3}{4} = \frac{1}{4}$ 
[ > cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
  (x0,y0)),f(x0,y0));
                                     cond4b :=  $\frac{9\beta^3}{4} + \frac{3\alpha^3}{4} = \frac{3}{4}$ 
[ > cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](
  f)(x0,y0)),f(x0,y0)^2);
                                     cond4c :=  $\frac{9\beta^3}{4} + \frac{3\alpha^3}{4} = \frac{3}{4}$ 
[ > cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](

```

```
f)(x0,y0),D[1](f)(x0,y0));
```

$$\text{cond4d} := \frac{3}{2}\alpha^3 - \frac{3}{2}\alpha^2 + \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 = \frac{3}{4}$$

```
> cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));
```

$$\text{cond4e} := 3\alpha^3 + \frac{3}{2}\beta^2 + \frac{9}{2}\alpha\beta^2 + \frac{9}{2}\beta\alpha^2 + \frac{3}{2}\beta\alpha - 3\alpha^2 = \frac{5}{4}$$

```
> cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);
```

$$\text{cond4f} := \frac{\alpha^3}{4} + \frac{3\beta^3}{4} = \frac{1}{4}$$

```
> cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));
```

$$\text{cond4g} := \frac{3}{2}\alpha^3 - \frac{3}{2}\alpha^2 + \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 = \frac{3}{4}$$

```
> cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);
```

$$\text{cond4h} := \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 + \frac{9}{4}\beta\alpha^2 + \frac{9}{4}\alpha^3 - \frac{9}{4}\alpha^2 + \frac{3}{4}\beta^2 = 1$$

```
> cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));
```

$$\text{cond4i} := \frac{3}{4}\beta^2 - \frac{3}{4}\alpha^2 + \frac{9}{4}\beta\alpha^2 + \frac{3}{4}\alpha^3 = \frac{1}{4}$$

```
> cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);
```

$$\text{cond4j} := \frac{3}{2}\alpha^3 - 3\alpha^2 + \frac{9}{2}\beta^2 - \frac{3}{2}\beta + \frac{3}{2}\alpha - \frac{3}{2}\beta\alpha + \frac{9}{2}\beta\alpha^2 = \frac{1}{4}$$

```
> cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));
```

$$\text{cond4k} := \frac{3}{2}\alpha^3 - 3\alpha^2 + \frac{9}{2}\beta^2 - \frac{3}{2}\beta + \frac{3}{2}\alpha - \frac{3}{2}\beta\alpha + \frac{9}{2}\beta\alpha^2 = \frac{1}{4}$$

```
> subs(ordena3,{cond4a,cond4b,cond4c,cond4d,cond4e,cond4f,cond4g,cond4h,cond4i,cond4j,cond4k});
```

$$\left\{ \frac{1}{3} = \frac{1}{4}, \frac{2}{3} = \frac{5}{4}, 0 = \frac{3}{4}, 1 = \frac{1}{4}, \frac{2}{3} = \frac{3}{4}, \frac{1}{3} = 1, \frac{2}{9} = \frac{1}{4} \right\}$$

y no se puede alcanzar orden 4 , lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = 0$ y $\beta = \frac{2}{3}$, o sea

para el método

$$0 \quad | \quad -1 \quad 1$$

$$\frac{2}{3} \quad | \quad \frac{2}{3} \quad 0$$

$$| \quad \frac{1}{4} \quad \frac{3}{4}$$

funcion de estabilidad para $\alpha=0$ y $\beta = \frac{2}{3}$

```
> restart;
```

```

[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
                                     f:=(x,y) -> lambda*y
[ > alpha:=0;beta:=2/3;
                                     alpha:=0
                                     beta:=2/3
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
                                     k10:=h -> lambda*(y0-h*kp1(h)+h*kp2(h))
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
                                     k20:=h -> lambda*(y0+2/3*h*kp1(h))
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
                                     ka1:=h -> lambda*(y0-h*kp1(h)+h*kp2(h))
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
                                     ka2:=h -> lambda*(y0+2/3*h*kp1(h))
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                     test:={k1=-3*lambda*y0*(1+lambda*h)/(-3-3*lambda*h+2*lambda^2*h^2),k2=-lambda*y0*(3+5*lambda*h)/(-3-3*lambda*h+2*lambda^2*h^2)}
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
                                     k1:=-3*lambda*y0*(1+lambda*h)/(-3-3*lambda*h+2*lambda^2*h^2)
                                     k2:=-lambda*y0*(3+5*lambda*h)/(-3-3*lambda*h+2*lambda^2*h^2)
                                     -6*lambda*y0*(2+3*lambda*h)/(-3-3*lambda*h+2*lambda^2*h^2)
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
                                     y1:=y0-3*h*lambda*y0/(-3-3*lambda*h+2*lambda^2*h^2)-9*lambda^2*y0*h^2/(2*(-3-3*lambda*h+2*lambda^2*h^2))
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                     y1:=(1-3*z/(-3-3*z+2*z^2)-9*z^2/(2*(-3-3*z+2*z^2)))*y0
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
                                     r_particular:=-6+12*z+5*z^2/(2*(-3-3*z+2*z^2))
[ >
[ >
[ # A-estabilidad para alpha = 0 y beta = 2/3
[ Este método no es A-estable, ya que r(z) tiende a 5/2 cuando z tiende a infinity

```

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-12 (del EXAMEN ORDINARIO DE 03JUL10)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} 0 & & 0 \\ C & A_1 & A_2 \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

a) Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de C , A_1 y A_2 para los que se consigue

(desde aquí es Ejercicio del capítulo 8)

b) Calcúlese la función de estabilidad en el caso general

c) Búsqese un ejemplo de método A-estable dentro de la familia de métodos del problema.

d) Compruébese si es o no es A-estable el método con valores $A_1 = 2/3$ y $A_2 = 2$

de los parámetros (C no influye, claro) y dibújese la región de estabilidad en este caso.

```
[ > restart:
[ > interface(labeling=false):
[ a)
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0,y0),h):
[ > k20:=unapply(f(x0+C*h,y0+A[1]*h*kp10(h)+A[2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
```

```

[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > Fi1:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=expand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=expand(k220):
[ > Fi2:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
[ 
$$cond1a := 1 = 1$$

[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
[ 
$$cond2a := \frac{3C}{4} = \frac{1}{2}$$

[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)),f(x0,y0));
[ 
$$cond2b := \frac{3}{4}A_1 + \frac{3}{4}A_2 = \frac{1}{2}$$

[ > orden2:=solve({cond2a,cond2b});
[ 
$$orden2 := \{A_2 = -A_1 + \frac{2}{3}, C = \frac{2}{3}, A_1 = A_1\}$$

[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
[ 
$$cond3a := \frac{3C^2}{4} = \frac{1}{3}$$

[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,y0)),f(x0,y0));
[ 
$$cond3b := \frac{3}{2}CA_1 + \frac{3}{2}CA_2 = \frac{2}{3}$$

[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x0,y0)),f(x0,y0)^2);
[ 
$$cond3c := \frac{3}{4}A_1^2 + \frac{3}{2}A_1A_2 + \frac{3}{4}A_2^2 = \frac{1}{3}$$

[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)),D[1](f)(x0,y0));
[ 
$$cond3d := \frac{3}{2}CA_2 = \frac{1}{3}$$

[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)^2),f(x0,y0));
[ 
$$cond3e := \frac{3}{2}A_1A_2 + \frac{3}{2}A_2^2 = \frac{1}{3}$$

[ > orden3:=solve({cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
[ 
$$orden3 := \{C = \frac{2}{3}, A_1 = \frac{1}{3}, A_2 = \frac{1}{3}\}$$

[ se obtiene orden 3 para los valores  $C = 2/3$  ,  $A_1 = 1/3$  y  $A_2 = 1/3$ 
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));

```

$$cond4a := \frac{3C^3}{4} = \frac{1}{4}$$

> cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)(x0,y0)),f(x0,y0));

$$cond4b := \frac{9}{4}C^2A_1 + \frac{9}{4}C^2A_2 = \frac{3}{4}$$

> cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2);

$$cond4c := \frac{9}{2}CA_1A_2 + \frac{9}{4}CA_1^2 + \frac{9}{4}CA_2^2 = \frac{3}{4}$$

> cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0));

$$cond4d := \frac{9}{2}C^2A_2 = \frac{3}{4}$$

> cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));

$$cond4e := 9CA_1A_2 + 9CA_2^2 = \frac{5}{4}$$

> cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);

$$cond4f := \frac{3}{4}A_2^3 + \frac{3}{4}A_1^3 + \frac{9}{4}A_1^2A_2 + \frac{9}{4}A_1A_2^2 = \frac{1}{4}$$

> cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));

$$cond4g := \frac{9}{2}CA_1A_2 + \frac{9}{2}CA_2^2 = \frac{3}{4}$$

> cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);

$$cond4h := \frac{27}{4}A_1^2A_2 + \frac{27}{2}A_1A_2^2 + \frac{27}{4}A_2^3 = 1$$

> cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));

$$cond4i := \frac{9}{4}C^2A_2 = \frac{1}{4}$$

> cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);

$$cond4j := \frac{9}{2}CA_2^2 = \frac{1}{4}$$

> cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));

$$cond4k := \frac{9}{2}A_2^3 + \frac{9}{2}A_1A_2^2 = \frac{1}{4}$$

> subs(ordena3,{cond4a,cond4b,cond4c,cond4d,cond4e,cond4f,cond4g,cond4h,cond4i,cond4j,cond4k});

$$\{1 = 1, \frac{2}{9} = \frac{1}{4} \cdot \frac{2}{3} = \frac{3}{4} \cdot \frac{4}{3} = \frac{5}{4} \cdot \frac{1}{3} = \frac{1}{4}\}$$

y no se puede alcanzar orden 4 , lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $C = 2/3$, $A_1 = 1/3$ y $A_2 = 1/3$, o sea para el método

$$\begin{array}{c|c} 0 & 0 \\ \hline \frac{2}{3} & \frac{1}{3} \quad \frac{1}{3} \end{array}$$

$$\left| \begin{array}{cc} 1 & 3 \\ 4 & 4 \end{array} \right|$$

que suele recibir el nombre específico de RADAU I de 2 etapas

b)

funcion de estabilidad

> restart;

> interface(labeling=false):

> f:=(x,y)->lambda*y;

$$f := (x, y) \rightarrow \lambda y$$

> ka1:=unapply(f(x0,y0),h);

$$ka1 := h \rightarrow \lambda y0$$

> ka2:=unapply(f(x0+C*h,y0+A[1]*h*kp1(h)+A[2]*h*kp2(h)),h);

$$ka2 := h \rightarrow \lambda (y0 + A_1 h kp1(h) + A_2 h kp2(h))$$

> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});

$$test := \{k1 = \lambda y0, k2 = -\frac{\lambda y0 (1 + \lambda A_1 h)}{-1 + \lambda A_2 h}\}$$

> k1:=subs(test,k1);k2:=subs(test,k2);

$$k1 := \lambda y0$$

$$k2 := -\frac{\lambda y0 (1 + \lambda A_1 h)}{-1 + \lambda A_2 h}$$

> y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));

$$y1 := y0 + \frac{h \lambda y0}{4} - \frac{3}{4} \frac{h \lambda y0}{-1 + \lambda A_2 h} - \frac{3}{4} \frac{\lambda^2 y0 A_1 h^2}{-1 + \lambda A_2 h}$$

> y1:=collect(subs(lambda=z/h,y1),y0);

$$y1 := \left(1 + \frac{z}{4} - \frac{3}{4} \frac{z}{-1 + z A_2} - \frac{3}{4} \frac{z^2 A_1}{-1 + z A_2}\right) y0$$

> r:=unapply(collect(simplify(y1/y0),z),z):r_general:=r(z);

$$r_general := -\frac{1}{4} \frac{4 + (-A_2 + 3 A_1) z^2 + (-4 A_2 + 4) z}{-1 + z A_2}$$

> r_nulo:=simplify(subs({A[2]=3*A[1]},r(z)));

$$r_nulo := \frac{-1 + 3 z A_1 - z}{-1 + 3 z A_1}$$

>

c)

A-estabilidad

Estos métodos (incluyendo el ya citado RADAU I de 2 etapas) nunca pueden ser A-estables,

ya que el numerador de $r(z)$ es de grado 2 mientras que el denominador lo es de 1

La excepción podría ser $A_2 = 3 A_1$ En este último caso, la función de estabilidad vale

$$r(z) = \frac{1 + (1 - 3 A_1) z}{1 - 3 A_1 z}$$

y es en principio de consecuencias dudosas.

Si hacemos entonces $A_1 = 1/3$ obtenemos como función de estabilidad

$$r(z) = \frac{1}{1-z}$$

y ya sabemos que dicho método es A-estable. Se trata del método

$$0 \mid 0$$

$$C \mid \frac{1}{3} \quad 1$$

$$\mid \frac{1}{4} \quad \frac{3}{4}$$

> `r_ejemplo1:=simplify(subs({A[1]=1/3},r_nulo));`

$$r_ejemplo1 := -\frac{1}{-1+z}$$

d)

Para $A_1 = 2/3$ y $A_2 = 2$, que entra dentro de la situación $A_2 = 3 A_1$, la función de estabilidad es

$$r(z) = \frac{z-1}{2z-1}$$

> `r_ejemplo2:=simplify(subs({A[1]=2/3,A[1]=2},r_nulo));`

$$r_ejemplo2 := \frac{-1+z}{-1+2z}$$

>

>

Pongamos $z=x+iy$. Como

$$|2z-1|^2 = (2x-1)^2 + 4y^2$$

y

$$|z-1|^2 = (x-1)^2 + y^2$$

la frontera $|r(z)| = 1$ o mejor $|r(z)|^2 = 1$ del dominio de estabilidad

es la curva de ecuación

$$(2x-1)^2 + 4y^2 = (x-1)^2 + y^2$$

que, tras operar, es la circunferencia

$$\left(x - \frac{1}{3}\right)^2 + y^2 = \frac{1}{9}$$

de centro $\frac{1}{3}$ y radio $\frac{1}{3}$, que pasa por $(0, 0)$. El valor $z = -1$ queda en

el exterior de la circunferencia, y, para él, $r(-1) = 2/3$, luego -1 pertenece

al dominio de estabilidad. El dominio de estabilidad

es el exterior de esa circunferencia, y el método es A-estable

Ejercicios de

(7) Métodos de RUNGE-KUTTA implícitos y semi-implícitos

Ejercicio 07-13 (del EXAMEN ORDINARIO DE 24ENE11)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} \alpha & \alpha - 1 & 1 \\ 1 & \beta & 0 \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

- a) Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α y de β para los que se consigue (desde aquí es Ejercicio del capítulo 8)
- b) Búsqese un método de la familia del enunciado que sea A-estable
- c) Estúdiense la A-estabilidad de los métodos de orden máximo obtenidos antes

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp10(h)+h*kp20(h)),h):
[ > k20:=unapply(f(x0+h,y0+beta*h*kp10(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10(0))=k110,D(kp20(0))=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
x
pand(k120):
[ > k220:=subs({D(kp10(0))=k110,D(kp20(0))=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
x
pand(k220):
[ > F12:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[
```

```

[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
                                     cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
                                     cond2a :=  $\frac{\alpha}{4} + \frac{3}{4} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0))
  ,f(x0,y0));
                                     cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a, cond2b});
                                     orden2 := { $\alpha = -1, \beta = 1$ }
[ > alpha:=subs(orden2,alpha);beta:=subs(orden2,beta);
                                      $\alpha := -1$ 
                                      $\beta := 1$ 
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
                                     cond3a :=  $1 = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
  y0)),f(x0,y0));
                                     cond3b :=  $2 = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
  0,y0)),f(x0,y0)^2);
                                     cond3c :=  $1 = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
  0,y0)),D[1](f)(x0,y0));
                                     cond3d :=  $0 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
  )^2,f(x0,y0));
                                     cond3e :=  $0 = \frac{1}{3}$ 

```

No se puede alcanzar orden 3 ; basta ver la primera de las condiciones,

que es imposible. El máximo orden es 2 con $\alpha = -1$ y $\beta = 1$

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = -1$ y $\beta = 1$, o sea

para el método

$$\begin{array}{c|cc}
 -1 & -2 & 1 \\
 1 & 1 & 0 \\
 \hline
 & \frac{1}{4} & \frac{3}{4}
 \end{array}$$

```

[ # funcion de estabilidad para  $\alpha=1$  y  $\beta=1$  (solo con orden 1 )
[ > restart:
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[                                     f := (x, y) → λ y
[ > alpha:=1;beta:=1;
[                                     α := 1
[                                     β := 1
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                     k10 := h → λ (y0 + h kp2(h))
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                     k20 := h → λ (y0 + h kp1(h))
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                     ka1 := h → λ (y0 + h kp2(h))
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                     ka2 := h → λ (y0 + h kp1(h))
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                                     test := {k2 = - $\frac{\lambda y0}{\lambda h - 1}$ , k1 = - $\frac{\lambda y0}{\lambda h - 1}$ }
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
[                                     k1 := - $\frac{\lambda y0}{\lambda h - 1}$ 
[                                     k2 := - $\frac{\lambda y0}{\lambda h - 1}$ 
[                                     - $\frac{4 \lambda y0}{\lambda h - 1}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                     y1 := y0 -  $\frac{h \lambda y0}{\lambda h - 1}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 :=  $\left(1 - \frac{z}{z-1}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
[                                     r_particular := - $\frac{1}{z-1}$ 
[ >
[ >
[ >
[ >
[ # para estos valores alpha = 1 y beta = 1 el método es evidentemente A-estable
[
[ # funcion de estabilidad para  $\alpha=-1$  y  $\beta=1$  (orden 2 máximo)
[ > restart:
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[                                     f := (x, y) → λ y
[ > alpha:=-1;beta:=1;

```

```

[                                      $\alpha := -1$ 
[                                      $\beta := 1$ 
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                      $k10 := h \rightarrow \lambda (y0 - 2 h kp1(h) + h kp2(h))$ 
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                      $k20 := h \rightarrow \lambda (y0 + h kp1(h))$ 
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                      $ka1 := h \rightarrow \lambda (y0 - 2 h kp1(h) + h kp2(h))$ 
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                      $ka2 := h \rightarrow \lambda (y0 + h kp1(h))$ 
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                                      $test := \{k1 = -\frac{\lambda y0 (1 + \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}, k2 = -\frac{\lambda y0 (1 + 3 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}\}$ 
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
[                                      $k1 := -\frac{\lambda y0 (1 + \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[                                      $k2 := -\frac{\lambda y0 (1 + 3 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[                                      $-\frac{2 \lambda y0 (2 + 5 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                      $y1 := y0 - \frac{h \lambda y0}{-1 - 2 \lambda h + \lambda^2 h^2} - \frac{5 \lambda^2 y0 h^2}{2 (-1 - 2 \lambda h + \lambda^2 h^2)}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                      $y1 := \left(1 - \frac{z}{-1 - 2 z + z^2} - \frac{5 z^2}{2 (-1 - 2 z + z^2)}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
[                                      $r\_particular := -\frac{2 + 6 z + 3 z^2}{2 (-1 - 2 z + z^2)}$ 
[ # para estos valores alpha = -1 y beta = 1 el método no es A-estable,
[ ya que r(z) tiende a 3/2 cuando z tiende a  $\infty$ 

```


Parte 2. Estabilidad lineal y métodos para ecuaciones 'stiff'

Estabilidad lineal

Estabilidad no lineal

08. Estabilidad lineal

- 1 La ecuación 'test'
- 2 Métodos A-estables
- 3 Ejemplos
- 4 Estabilidad lineal para sistemas

Comenzamos el estudio de un tipo de estabilidad para los sistemas y ecuaciones lineales que recibe el nombre de **estabilidad lineal**. En general los sistemas que se presentan en los fenómenos físicos, químicos, mecánicos, económicos, etc. no son nunca lineales, porque los fenómenos 'naturales' son mucho más complejos que lo que expresa la simple linealidad. Sin embargo, muchas veces la aproximación lineal es importante; a veces es un buen modelo para el fenómeno; otras veces es una aproximación que orienta el verdadero desarrollo, aunque no contenga toda su diversidad. Se suele hablar de la **primera aproximación** para referirse a la aproximación lineal de un fenómeno más complejo.

A veces esta primera aproximación describe casi exactamente un fenómeno; en ese caso poder resolverlo exacta o numéricamente deja el problema resuelto casi por completo. Otras veces la aproximación no es del todo exacta, aunque las líneas de la aproximación lineal guíen el desarrollo del fenómeno. Finalmente, hay ocasiones en que las direcciones del modelo lineal y del modelo completo son divergentes. Y, naturalmente, este último caso es el que plantea problemas más difíciles de abordar con las herramientas matemáticas.

Hablaremos un capítulo más tarde de esta disparidad entre un modelo y su versión lineal.

Lo que ahora vamos a ver es que los sistemas lineales, o sea,

$$\mathbf{y}' = \mathbf{A} \mathbf{y},$$

(autónomos, claro) con A una matriz constante y $m \times m$,

y para $m = 1$ la ecuación lineal escalar autónoma

$$y' = \lambda y$$

o no autónoma

$$y' = \lambda y + g(x),$$

a pesar de su sencillez, no siempre son 'bien tratados' por los métodos numéricos.

Hablaremos de 'stiff' (así, con esta palabra inglesa) o rígidos para referirnos a estas ecuaciones y sistemas que se integran con dificultad con buena parte de los métodos que hemos estudiado. Aunque la 'stiffness' o rigidez es, como veremos, una propiedad que no es de sí o no, sino cuantitativa, lo que nos hará hablar de ecuaciones más o menos stiff.

Comenzaremos por mostrar algunos ejemplos sencillos de lo que queremos decir.

Los ficheros elaborados en 'Maple' con los nombres **EuleryStiff1.mws**, **EuleryStiff2.mws** y **EuleryStiff3.mws** nos enseñan lo que ocurre cuando integramos con el método de EULER dos problemas bien sencillos y de solución sencilla y completamente conocida. El primero y el tercero lo hacen para el problema escalar lineal autónomo

$$y' = -110 y,$$

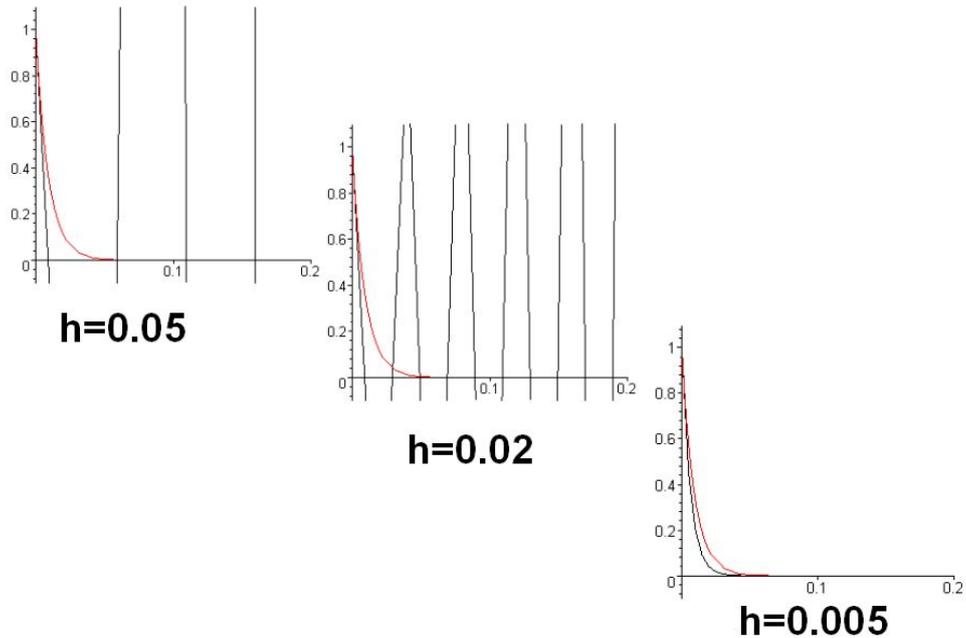
y el segundo para la ecuación lineal no autónoma

$$y' = -50 (y - \cos(x)).$$

En ambos casos, un paso de tamaño adecuado no es suficiente. La mala integración sólo se arregla cuando hacemos el paso bastante más pequeño.

Y el culpable no es, como veremos, el bajo orden del método de EULER; ocurre lo mismo para métodos de orden más elevado. Y, además, el método implícito de EULER, que es también de orden 1 no presenta estos problemas, como también se muestra.

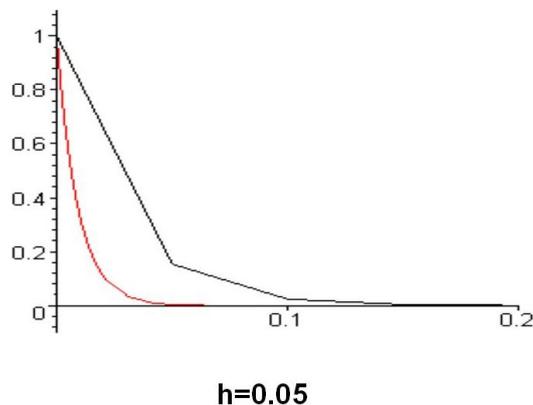
Las gráficas más relevantes son, para $y' = -110y$ con EULER



en las que se ve que la solución numérica es 'errante' hasta que un paso, en este caso de 0,005, la lleva a la proximidad de la verdadera solución.



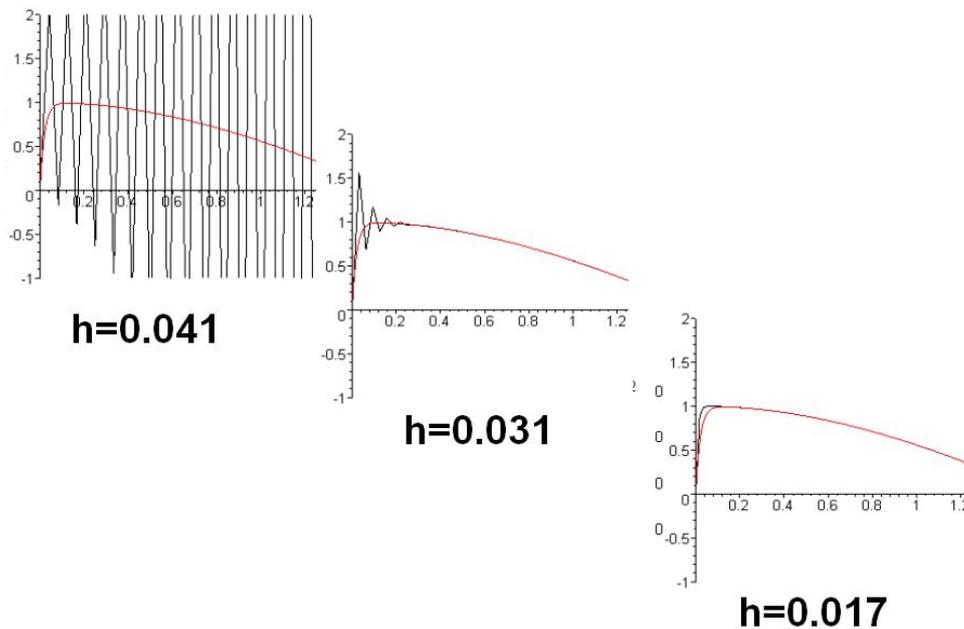
Sin embargo, el método implícito de EULER (precisamente por ser implícito) a pesar de ser del mismo orden, aproxima razonablemente la solución con el mayor paso de los anteriores.



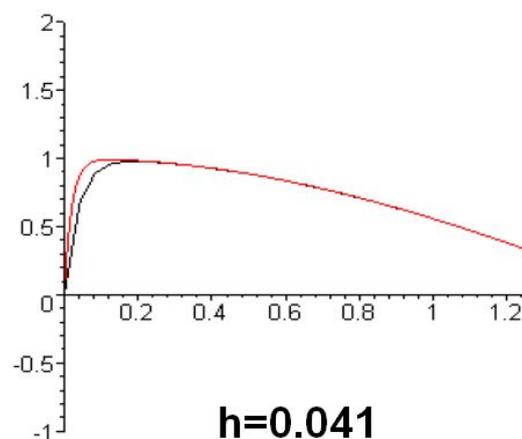
Obsérvese que la separación de la solución es razonable cuando sólo se están dando 4 pasos en el intervalo $[0, 0,2]$ mostrado.



También para $y' = -50(y - \cos(x))$ con EULER ocurre el mismo fenómeno



Y de nuevo el método implícito de EULER lo soluciona incluso con el mayor paso de los anteriores.



La ecuación 'test'

Una de las conclusiones que resultarán al final de este capítulo es que la sencilla ecuación escalar autónoma

$$y' = \lambda y$$

permite comprobar la estabilidad lineal de los métodos numéricos. Pero también explicaremos que es imprescindible tratar esta con λ complejo, que es lo que supondremos de ahora en adelante. La ecuación precedente, con $\lambda \in \mathbb{C}$ se denomina **ecuación test de DAHLQUIST**; si la hacemos acompañar (por simplicidad) de la condición inicial $y(0) = 1$, el problema resultante tiene a $y(x) = e^{\lambda x}$ como única solución en $[0, +\infty]$ o en cualquier intervalo $[0, b]$ que nos parezca oportuno.

Esta ecuación test tenía un comportamiento stiff bastante acusado cuando $\lambda = -110$, como ya hemos visto en un ejemplo. Y en realidad es así siempre que λ sea un número real negativo de módulo algo grande. Generalmente se expresa esto con

$$\lambda \ll 0,$$

de significado bastante obvio. La causa es que, entonces, la solución decae rápidamente (exponencialmente) hacia 0. Pero si λ es complejo lo que equivale a lo anterior es que

$$\Re(\lambda) \ll 0;$$

esto es así porque $y(x) = e^{\lambda x} = e^{\Re(\lambda)x} e^{i\Im(\lambda)x}$ y la parte $e^{i\Im(\lambda)x}$ es de módulo 1 siempre. El decaimiento a 0 depende entonces sólo de la otra parte.

¿Qué valores proporciona el método de EULER

$$y_{n+1} = y_n + h f(y_n),$$

para el problema dado por la ecuación test? Como $f(y) = \lambda y$, lo que resulta es

$$y_{n+1} = (1 + h \lambda) y_n,$$

o sea, que

$$y_n = (1 + h \lambda)^n$$

(recuérdese que $y_0 = y(0) = 1$), como se sigue fácilmente por recurrencia.

Cuando $\Re(\lambda) \ll 0$ y la solución exacta $y(x) = e^{\lambda x}$ decae en módulo exponencialmente a 0, la aproximación numérica sólo sigue el mismo camino cuando

$$|1 + h \lambda| < 1;$$

Por el contrario, la situación en que $|1 + h \lambda| > 1$ hace que el módulo de $y_n = (1 + h \lambda)^n$ tienda a infinito, separándose de la solución exacta de forma 'brutal'. Si $|1 + h \lambda| = 1$ la solución numérica no tiende a infinito, aunque tampoco decae a 0, sino que se mantiene de módulo 1 constantemente. Se suele añadir este caso al primero o al segundo según interese.

Examinando el valor de $1 + h \lambda$ para $\lambda \in \mathbb{C}$, con $\Re(\lambda) \ll 0$, y $h > 0$, es fácil ver que, para el funcionamiento correcto (o sea $|1 + h \lambda| < 1$) de la solución numérica, es obligado mantener h en valores pequeños. Por ejemplo, para nuestra ecuación $y' = -110y$ el ejemplo visto en **EuleryStiff1.mws** muestra que h debe estar en un intervalo $(0, h_0]$ para un valor h_0 entre 0,01 y 0,05; un examen más cuidadoso muestra que $h_0 \sim 0,018$.

Es evidente que la conveniencia de optimizar operaciones y evaluaciones de f llevan a la búsqueda de métodos que no tengan esta dificultad que el método de EULER presenta para los problemas stiff. Los métodos implícitos que hemos visto hace poco van en este sentido.

Los ejemplos de **EuleryStiff2.mws** y **EuleryStiff3.mws** dejaban ver que el método implícito de EULER, en su forma exacta, no exigía tomar h pequeño salvo por el objetivo del acercamiento a la solución exacta, sobre todo a causa de su orden un poco bajo.

En lo que sigue vamos a ver que la dualidad métodos explícitos-métodos implícitos de RUNGE-KUTTA es la clave de los comportamientos acerca de la estabilidad lineal, o sea, de los comportamientos de cara a los problemas stiff.

Aparcamos el método de EULER para probar lo que ocurre en otros casos. Fijémonos en el 'método del punto medio' que se escribía como

$$\begin{aligned} k_1 &= f(x_n, y_n), \\ k_2 &= f\left(x_n + \frac{1}{2} h, y_n + \frac{1}{2} h k_1\right) \\ y_{n+1} &= y_n + h k_2. \end{aligned}$$

Para la ecuación test $y' = \lambda y$ con $f(y) = \lambda y$ vamos teniendo

$$\begin{aligned} k_1 &= \lambda y_n, \\ k_2 &= \lambda \left(y_n + \frac{1}{2} h k_1\right) = \lambda \left(1 + \frac{1}{2} h \lambda\right) y_n, \end{aligned}$$

y, finalmente,

$$y_{n+1} = y_n + h k_2 = \left(1 + h \lambda + \frac{1}{2} h^2 \lambda^2\right) y_n,$$

por lo que

$$y_n = \left(1 + h \lambda + \frac{1}{2} h^2 \lambda^2\right)^n y_0.$$

Similar al de EULER, pero $1 + h \lambda$ ha sido reemplazado por

$$1 + h \lambda + \frac{1}{2} h^2 \lambda^2.$$

Lo más importante que el desarrollo anterior es que la expresión a considerar $(1 + h\lambda$ o $1 + h\lambda + \frac{1}{2}h^2\lambda^2$, dependiendo del caso) contiene h y λ siempre emparejadas. Para un método convendría tal vez decir lo que debe suceder con h ; pero eso no es en principio posible (ya que depende del valor de λ). Lo único que es posible es concluir lo que debe ocurrir con

$$z = h\lambda,$$

para luego, si es que conocemos el problema, o sea λ , hacer precisiones sobre h .

Mantendremos siempre esta variable $z = h\lambda$ agrupada. Esta variable, producto de $h > 0$ y de $\lambda \in \mathbb{C}$ toma valores complejos. Para los métodos vistos hemos tenido

$$y_{n+1} = r(z) y_n \quad \text{y} \quad y_n = r(z)^n y_0$$

con

$$r(z) = 1 + z$$

en el caso de EULER, y

$$r(z) = 1 + z + \frac{1}{2}z^2$$

en el caso del 'método del punto medio'.

Llamaremos **función de estabilidad** a la función $r(z)$, $z = h\lambda$ para la que

$$y_{n+1} = r(z) y_n \quad \text{y} \quad y_n = r(z)^n y_0$$

En el caso de EULER $r(z) = 1 + z$, en el 'del punto medio' $r(z) = 1 + z + \frac{1}{2}z^2$; otros métodos traerán diferentes funciones de estabilidad, no siempre polinómicas.

Dominio de estabilidad

Lo que interesa respecto de la estabilidad lineal para un método numérico es el tamaño de la región en que

$$|r(z)| < 1.$$

Cuando $z = h\lambda$ está en esta región (y eso, al conocer λ nos indicará cuánto debe valer h) la solución numérica de un problema stiff tenderá, como la exacta, a 0. Denominaremos **región de estabilidad** o **dominio de estabilidad** a este subconjunto

$$\mathcal{D} = \{z \in \mathbb{C} \mid |r(z)| < 1\}.$$

del plano complejo \mathbb{C} .

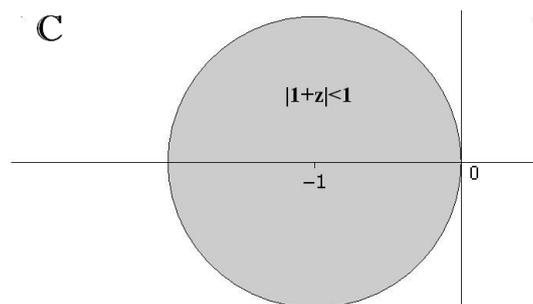
Como ya hemos dicho no suele tener importancia considerar o no incluidos en \mathcal{D} los z para los que $|r(z)| = 1$; cuando no se incluyen, se habla a veces de 'dominio estricto de estabilidad'; la opción contraria es tomar \leq en vez de $<$. Ya indicamos que esto no suele tener trascendencia en la práctica.



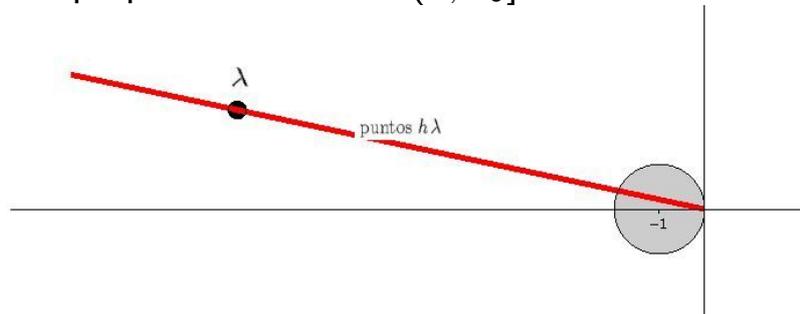
Por ejemplo, para el método de EULER, el dominio (estricto) de estabilidad es

$$\mathcal{D} = \{z \in \mathbb{C} \mid |1 + z| < 1\}.$$

Como $|1 + z| = |z - (-1)|$ representa la distancia al complejo -1 , el conjunto \mathcal{D} es el interior del círculo de centro -1 y radio 1 (cuya circunferencia pasa pues por el origen 0). Es la región señalada en el dibujo



Para esta región de estabilidad y tomando por ejemplo λ como en la figura, los puntos $h\lambda$ para $h > 0$ están sobre la semirrecta roja. Y sólo están sobre el círculo-dominio de estabilidad los $z = h\lambda$ para h en un 'pequeño' intervalo $(0, h_0]$.



Naturalmente, si queremos métodos para los que el dominio de estabilidad no establezca estas 'limitaciones' para h , necesitamos que la semirrecta en rojo y todas las análogas para cualquier λ con $\Re(\lambda) < 0$ estén dentro del dominio de estabilidad. Y esto es tanto como decir que necesitamos que todo el semiplano izquierdo de \mathbb{C} esté contenido en el dominio de estabilidad.

Métodos A-estables

Denotaremos abreviadamente este semiplano izquierdo de \mathbb{C} , o sea el formado por los z de parte real negativa, como

$$\Re(z) < 0.$$

Y llamaremos **métodos A-estables** (proviene de 'asintóticamente estables') a aquellos cuyo dominio de estabilidad contenga este semiplano, o sea

$$\Re(z) < 0 \subset \mathcal{D}.$$

Tras lo visto para el dominio de estabilidad del método de EULER, este método no es A-estable.

Tampoco lo es el 'método del punto medio', para el que $r(z) = 1 + z + \frac{1}{2}z^2$. Sin embargo, en este último caso es más trabajoso obtener una descripción del dominio de estabilidad. Entre otros procedimientos, un método de carácter gráfico permite una descripción adecuada de dicho dominio. Vamos a ver cómo se puede hacer en este caso.

La idea es ir mostrando los cortes con la frontera $|r(z)| = 1$ del dominio \mathcal{D} . Por razones operativas obvias, será mejor escribir esta frontera como

$$|r(z)|^2 = 1,$$

Las rectas con las que se corta esta frontera son las de la forma $y = cx$ (pasando por el origen). Por ejemplo, la recta $y = -x$, de pendiente -1 , está formada por los complejos

$$z = x - ix;$$

sobre ellos, la frontera del dominio vale

$$|1 + x - ix + \frac{1}{2}(x - ix)|^2 = 1,$$

$$|1 + x - ix(1 + x)|^2 = 1,$$

tras hacer algunas operaciones. Eso es lo mismo que (recuérdese cuánto vale el módulo de un número complejo)

$$(1 + x)^2 + x^2(1 + x)^2 = 1,$$

$$(1 + x)^2(1 + x^2) = 1,$$

$$x(x^3 + 2x^2 + 2x + 2) = 0,$$

que tiene raíces

$$x = 0, \quad x \sim -1,54$$

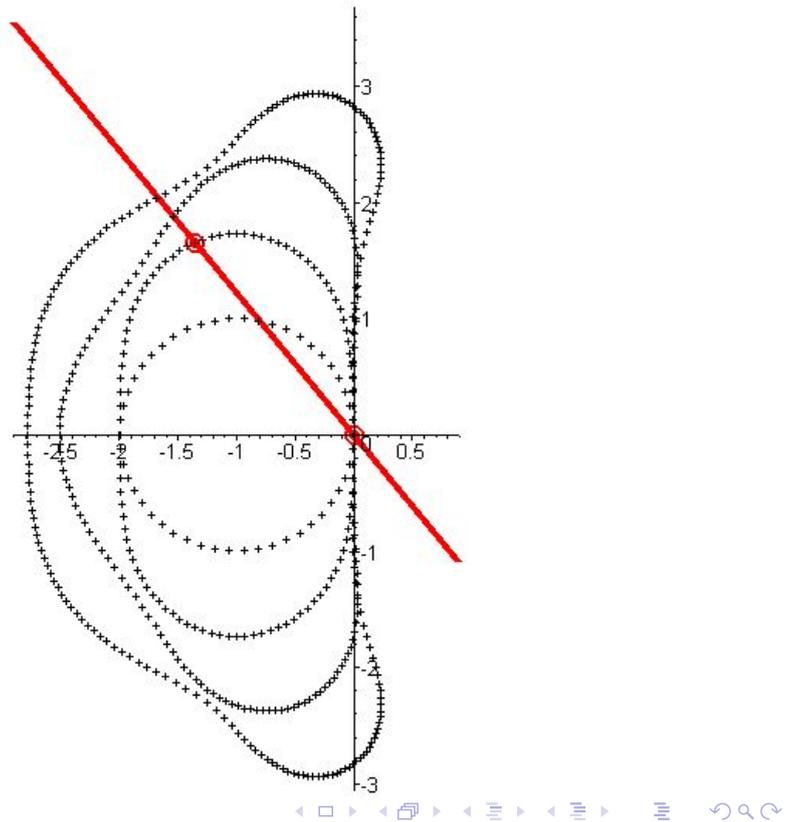
además de 2 raíces complejas que no aportan nada en este caso. Por lo tanto, la recta $z = x - ix$ aporta al contorno de \mathcal{D} los puntos $z = 0$ y $z = -1,54 + i1,54$, señalados con color rojo en la gráfica que sigue.

Gráfica que muestra las regiones de estabilidad de 4 modelos de métodos explícitos de RUNGE-KUTTA; son el método de EULER, el 'método del punto medio' (aunque la función de estabilidad $r(z)$ es la misma para todos los métodos explícitos de RUNGE-KUTTA con 2 etapas y orden 2), lo mismo para 3 etapas y orden 3 y, finalmente, lo mismo para 4 etapas y orden 4.

Todos los dominios dibujados por el programa

RungeKuttaRegion.mws con la técnica gráfica explicada.

RungeKuttaRegion.mws



08. Estabilidad lineal

La gráfica anterior permite intuir que ningún método explícito de RUNGE-KUTTA va a ser A-estable. Esto es así porque la función de estabilidad para cualquiera de estos métodos explícitos es de la forma $r(z) = p(z)$, donde $p(z)$ es algún polinomio en z . En los anteriores ejemplos, que eran todos de métodos explícitos del máximo orden para un número dado de etapas, el polinomio era

$$r(z) = 1 + z + \frac{1}{2} z^2 + \frac{1}{6} z^3 + \frac{1}{24} z^4 + \dots ,$$

o sea, los primeros del desarrollo en serie de e^z .

Cuando $r(z)$ es un polinomio (no constante), el módulo de $r(z)$ es infinito en $z = \infty$, o sea, el límite es

$$\lim_{z \rightarrow \infty} r(z) = \infty$$

(el ∞ complejo). Por lo tanto, es imposible que $|r(z)|$ se mantenga menor que 1 en todo el semiplano izquierdo de \mathbb{C} .

Bastante diferente es lo que ocurre para los métodos implícitos de RUNGE-KUTTA. En ese caso, la función de estabilidad $r(z)$ es una función racional de z , o sea,

$$r(z) = \frac{p(z)}{q(z)}$$

con $p(z)$ y $q(z)$ polinomios en z . En esa situación, cuando el grado de $p(z)$ es menor o igual al de $q(z)$, el hecho de que el método sea A-estable es, como veremos, bastante frecuente. En la situación anterior es a veces enrevesado demostrar la A-estabilidad de un método cuando el grado de ambos polinomios coincide. Estudiaremos algunas técnicas concretas para obtener conclusiones en esa situación, generalmente mediante representación analítica o gráfica de la frontera del dominio de estabilidad.

Método de GAUSS de 1 etapa

El método de GAUSS de 1 etapa tiene por tablero

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}$$

y es el único método de 1 etapa y orden 2. Su expresión completa para en caso autónomo es

$$k_1 = f\left(y_n + \frac{1}{2} h k_1\right)$$

$$y_{n+1} = y_n + h k_1.$$

Para la ecuación test $y' = \lambda y$ con $f(y) = \lambda y$ se obtiene

$$k_1 = \lambda \left(y_n + \frac{1}{2} h k_1\right) = \lambda y_n + \frac{1}{2} h \lambda k_1$$

$$\left(1 - \frac{1}{2} h \lambda\right) k_1 = \lambda y_n$$

$$k_1 = \frac{\lambda y_n}{1 - \frac{1}{2} h \lambda}, \quad \text{luego}$$

$$y_{n+1} = y_n + \frac{h\lambda}{1 - \frac{1}{2}h\lambda} y_n = \left(1 + \frac{h\lambda}{1 - \frac{1}{2}h\lambda}\right) y_n,$$

por lo que (z es $h\lambda$)

$$r(z) = 1 + \frac{z}{1 - \frac{1}{2}z} = \frac{2+z}{2-z}.$$

El dominio de estabilidad \mathcal{D} está en este caso formado por aquellos z tales que $|2+z| < |2-z|$, es decir, los z más cercanos a -2 que a 2 . Esto es justamente el semiplano $\Re(z) < 0$ de los complejos de parte real negativa.



Por lo tanto, este método de GAUSS de 1 etapa es un método A-estable.

Método implícito de EULER

El método implícito de EULER tiene por tablero

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

y tiene orden 1. Su expresión completa para en caso autónomo es

$$k_1 = f(y_n + h k_1)$$

$$y_{n+1} = y_n + h k_1.$$

Para la ecuación test $y' = \lambda y$ con $f(y) = \lambda y$ se obtiene

$$k_1 = \lambda(y_n + h k_1) = \lambda y_n + h \lambda k_1$$

$$(1 - h \lambda) k_1 = \lambda y_n$$

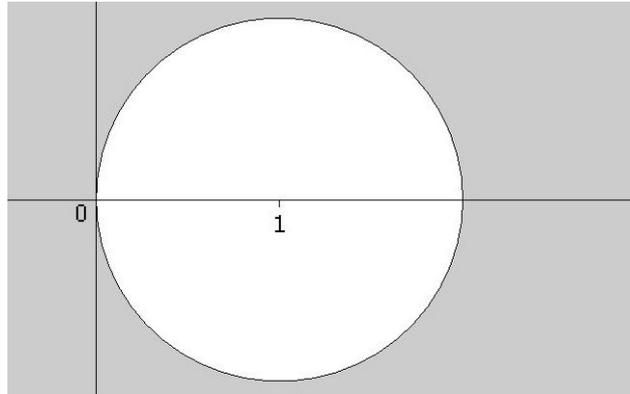
$$k_1 = \frac{\lambda}{1 - h \lambda}, \quad \text{luego}$$

$$y_{n+1} = y_n + \frac{h\lambda}{1 - h\lambda} y_n = \left(1 + \frac{h\lambda}{1 - h\lambda}\right) y_n,$$

por lo que

$$r(z) = 1 + \frac{z}{1 - z} = \frac{1}{1 - z}.$$

El dominio de estabilidad \mathcal{D} es ahora el exterior del círculo de centro 1 y radio 1 (también su circunferencia pasa por el origen). Es la región señalada en el dibujo.



Por lo tanto, este método implícito de EULER es un método A-estable.

Método de GAUSS de 2 etapas

El método de GAUSS de 2 etapas tiene por tablero

$\frac{3-\sqrt{3}}{6}$	$\frac{1}{4}$	$\frac{3-2\sqrt{3}}{12}$
$\frac{3+\sqrt{3}}{6}$	$\frac{3+2\sqrt{3}}{12}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

y es el único de 2 etapas con ese orden. Su expresión completa para en caso autónomo es

$$\begin{aligned} k_1 &= f\left(y_n + \frac{1}{4} h k_1 + \frac{3-2\sqrt{3}}{12} h k_2\right) \\ k_2 &= f\left(y_n + \frac{3+2\sqrt{3}}{12} h k_1 + \frac{1}{4} h k_2\right) \\ y_{n+1} &= y_n + h\left(\frac{1}{2} k_1 + \frac{1}{2} k_2\right) = y_n + \frac{1}{2} h(k_1 + k_2). \end{aligned}$$

Para la ecuación test $y' = \lambda y$ con $f(y) = \lambda y$ se obtiene

$$\begin{aligned}k_1 &= \lambda y_n + \frac{1}{4} h \lambda k_1 + \frac{3-2\sqrt{3}}{12} h \lambda k_2 \\k_2 &= \lambda y_n + \frac{3+2\sqrt{3}}{12} h \lambda k_1 + \frac{1}{4} h \lambda k_2\end{aligned}$$

que no es sino el sistema lineal en k_1, k_2

$$\begin{aligned}\left(1 - \frac{1}{4}\right) h \lambda k_1 - \frac{3-2\sqrt{3}}{12} h \lambda k_2 &= \lambda y_n \\-\frac{3+2\sqrt{3}}{12} h \lambda k_1 + \left(1 - \frac{1}{4}\right) h \lambda k_2 &= \lambda y_n.\end{aligned}$$

La solución de este sistema debe hacerse con algún cuidado; yo he usado el método de CRAMER, bastante eficaz para sistemas de tamaño 2. La solución es algo larga tanto para k_1 como para k_2 , pero para $k_1 + k_2$ la expresión se simplifica un poco y vale

$$k_1 + k_2 = \frac{24 \lambda}{12 - 6 h \lambda + h^2 \lambda^2} y_n.$$

y esta es justo la expresión que hay que utilizar:



$$y_{n+1} = y_n + \frac{12 h \lambda}{12 - 6 h \lambda + h^2 \lambda^2} y_n = \left(1 + \frac{12 h \lambda}{12 - 6 h \lambda + h^2 \lambda^2}\right) y_n.$$

Por lo tanto,

$$r(z) = 1 + \frac{12 z}{12 - 6 z + z^2} = \frac{12 + 6 z + z^2}{12 - 6 z + z^2}.$$

Respecto de la A-estabilidad, el grado de los polinomios deja el caso en situación dudosa, más aún porque

$$\lim_{z \rightarrow \infty} r(z) = 1.$$

Ahora bien, factorizando numerador y denominador, $12 + 6 z + z^2$ tiene raíces $-3 \pm i \sqrt{3}$, mientras que $12 - 6 z + z^2$ tiene raíces $3 \pm i \sqrt{3}$, por lo que

$$12 + 6 z + z^2 = (z + 3 - i \sqrt{3})(z + 3 + i \sqrt{3}) = (z + 3)^2 + 3$$

$$12 - 6 z + z^2 = (z - 3 - i \sqrt{3})(z - 3 + i \sqrt{3}) = (z - 3)^2 + 3$$



$$r(z) = \frac{(z+3)^2 + 3}{(z-3)^2 + 3}.$$

Como, para z real, $(z+3)^2$ es el cuadrado de la distancia de z a -3 y lo análogo $(z-3)^2$ y 3 estamos en condiciones de decir que los números reales negativos están en el dominio \mathcal{D} y los positivos en su complementario. Pero no hay en el caso complejo la misma interpretación sobre la distancia mientras no intervenga el módulo, así que todavía no podemos concluir si el método es o no A-estable.

La idea más sencilla va a ser la de construir la frontera de \mathcal{D} , de ecuación $|r(z)| = 1$ en el plano complejo o, mejor,

$$|r(z)|^2 = 1.$$

Poniendo x, y como las partes real e imaginaria de z , luego $z = x + iy$ con x, y reales, la curva frontera en el plano (x, y) que es, como hemos dicho,

$$|(z+3)^2 + 3|^2 = |(z-3)^2 + 3|^2,$$



se escribe

$$((x+3)^2 - y^2 + 3)^2 + 4y^2(x+3)^2 = ((x-3)^2 - y^2 + 3)^2 + 4y^2(x-3)^2,$$

o, simplificando,

$$24x^3 + 288x + 24xy^2 = 0$$

$$x(24x^2 + 288 + 24y^2) = 0$$

y la curva frontera del dominio se compone sólo de los puntos con $x = 0$, o sea, los $z = iy$, ya que la expresión $24x^2 + 288 + 24y^2$ no puede anularse.

Queda por determinar cuál de los dos semiplanos

$$\Re(z) < 0 \quad \text{o} \quad \Re(z) > 0,$$

separados por la curva que hemos calculado, es el dominio de estabilidad. Pues bien, como $r(-3) = 3/39$, -3 está en \mathcal{D} , y \mathcal{D} coincide exactamente con el semiplano $\Re(z) < 0$, por lo que el Método de GAUSS de 2 etapas es A-estable.



Método trapezoidal o LOBATTO IIIa

El método trapezoidal, implícito, de 2 etapas y orden 2 tiene el tablero

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

y se escribe para el caso autónomo como

$$\begin{aligned} k_1 &= f(y_n) \\ k_2 &= f\left(y_n + \frac{1}{2} h k_1 + \frac{1}{2} h k_2\right) \\ y_{n+1} &= y_n + h\left(\frac{1}{2} k_1 + \frac{1}{2} k_2\right). \end{aligned}$$

Para la ecuación test $y' = \lambda y$ con $f(y) = \lambda y$ se obtiene

$$k_1 = \lambda y_n$$



$$k_2 = \lambda y_n + \frac{1}{2} h \lambda k_1 + \frac{1}{2} h \lambda k_2 = \lambda y_n + \frac{1}{2} h \lambda^2 y_n + \frac{1}{2} h \lambda k_2$$

$$\left(1 - \frac{1}{2} h \lambda\right) k_2 = \lambda \left(1 + \frac{1}{2} h \lambda\right) y_n$$

$$k_2 = \frac{\lambda \left(1 + \frac{1}{2} h \lambda\right)}{1 - \frac{1}{2} h \lambda} y_n$$

$$y_{n+1} = y_n + \frac{1}{2} h \lambda y_n + \frac{\frac{1}{2} h \lambda \left(1 + \frac{1}{2} h \lambda\right)}{1 - \frac{1}{2} h \lambda} y_n,$$

con lo que

$$r(z) = 1 + \frac{1}{2} z + \frac{\frac{1}{2} z \left(1 + \frac{1}{2} z\right)}{1 - \frac{1}{2} z} = \frac{1 + \frac{1}{2} z}{1 - \frac{1}{2} z} = \frac{2 + z}{2 - z}$$

Ya hemos visto antes que, para esta función de estabilidad, el dominio de estabilidad \mathcal{D} está formado por aquellos z más cercanos a -2 que a 2 , o sea, es el semiplano $\Re(z) < 0$ y este método trapezoidal es un método A-estable.



Los sistemas lineales de coeficientes constantes y la ecuación test

Aunque hemos dedicado todo el esfuerzo a saber como actuaban los distintos métodos de RUNGE-KUTTA con la función test, el problema que realmente interesa a la estabilidad lineal es el de los sistemas lineales de coeficientes constantes. Saber cuándo estos sistemas son o no stiff y saber en ese caso como se comportaría un método con ellos.

Pero en absoluto hemos desviado nuestra atención al caso de la ecuación test por ser muy sencilla. Vamos a ver que el comportamiento con un sistema lineal queda perfectamente explicado en cuanto conocemos su comportamiento con la ecuación

$$y' = \lambda y.$$

Aunque hablaremos de sistemas de tamaño m , escribiremos, como otras veces, las expresiones para $m = 2$.

Supongamos que tenemos un sistema lineal de coeficientes constantes

$$y' = Ay,$$

en donde A es una matriz constante

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

No importa que A sea real o compleja (lo que vamos a decir sirve en ambos casos) pero lo más corriente es que sea real. No obstante, sus valores propios pueden ser complejos, aún en ese caso. Como reducción por semejanza para A a una matriz diagonal o 'casi' D , podemos encontrar dos casos

$$D = \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} \quad \text{o} \quad D = \begin{bmatrix} \lambda_1 & 1 \\ & \lambda_1 \end{bmatrix}$$

Cuando aparece el 1 en la forma de JORDAN, el valor propio a quien corresponde debe ser múltiple.

Haremos un alto en el primer caso. Los valores propios pueden ser iguales o diferentes, y si, en el caso real, son complejos, entonces son complejos conjugados.

Existe una matriz inversible P (compleja si los autovalores lo son) tal que

$$P^{-1} A P = D,$$

de donde obtenemos entonces que $P^{-1} A = D P^{-1}$. El sistema se puede escribir entonces como

$$P^{-1} \begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = P^{-1} A \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix} = D P^{-1} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

El cambio de variable dependiente

$$\begin{bmatrix} {}^1z \\ {}^2z \end{bmatrix} = P^{-1} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

convierte el sistema en

$$\begin{bmatrix} {}^1z' \\ {}^2z' \end{bmatrix} = D \begin{bmatrix} {}^1z \\ {}^2z \end{bmatrix} = \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} \begin{bmatrix} {}^1z \\ {}^2z \end{bmatrix}$$

que no es sino el conjunto de las dos ecuaciones 'separadas' o sea, cada una con su función incógnita)

$$\begin{aligned} {}^1z' &= \lambda_1 {}^1z, & y \\ {}^2z' &= \lambda_2 {}^2z, \end{aligned}$$

que responden al modelo de la ecuación test

$$y' = \lambda y,$$

pero donde los λ_i son los valores propios de A , que pueden ser complejos. El caso no diagonalizable

$$D = \begin{bmatrix} \lambda_1 & 1 \\ & \lambda_1 \end{bmatrix}$$

es análogo, aunque requiere alguna precisión más. En este caso, el mismo cambio de variable lleva el sistema a

$$\begin{bmatrix} {}^1z' \\ {}^2z' \end{bmatrix} = D \begin{bmatrix} {}^1z \\ {}^2z \end{bmatrix} = \begin{bmatrix} \lambda_1 & 1 \\ & \lambda_1 \end{bmatrix} \begin{bmatrix} {}^1z \\ {}^2z \end{bmatrix}$$

que es ahora el conjunto de las dos ecuaciones

$$\begin{aligned} {}^1z' &= \lambda_1 {}^1z + {}^2z, & y \\ {}^2z' &= \lambda_1 {}^2z, \end{aligned}$$

La segunda es del modelo

$$y' = \lambda y,$$

y entonces la primera es del modelo

$$y' = \lambda y + g(x),$$

o sea, la ecuación test pero no autónoma. Pero su comportamiento es semejante al del caso autónomo, como hemos tenido ocasión de comprobar en un ejemplo, el que vimos en **EuleryStiff2.mws** (en él, λ era -50 y $g(x)$ era la función $50 \cos(x)$).

En conclusión: el problema de la estabilidad lineal en el caso de sistemas lineales de coeficientes constantes, si son o no stiff y cómo se comportan los métodos numéricos con el sistema, se estudia considerando la actuación del método sobre la sencilla ecuación test.

Es ese sentido, podemos decir ahora que un sistema lineal de coeficientes constantes es stiff cuando alguno de los valores propios de la matriz del sistema es de parte real algo o muy negativa (más stiff cuanto más negativa sea). Y un método se comportará adecuadamente (es decir, sin limitaciones en el tamaño del paso) en estos casos 'problemáticos', cuando lo haga con la ecuación test. Esta ecuación se convierte por lo tanto en un referente sencillo para la estabilidad lineal.

Ejercicios de

(8) Estabilidad lineal

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-01

a) Cuál es el mayor orden posible que puede alcanzar un método de Runge-Kutta con el siguiente tablero

0		
1/(2 α)		1/(2 α)

		1 - α α

(lo que antecede es Ejercicio del capítulo 3)

b) Para α arbitrario, calcúlese la función de estabilidad de estos métodos

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/(2*alpha)*h,y0+1/(2*alpha)*h*k1_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand((1-alpha)*k1_0+alpha*k2_0);
[                                     Fi_0 := f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                     cond1a := 1 = 1
[ condiciones para el orden 2
[ > Fi10:=expand((1-alpha)*k110+alpha*k210);
[                                     Fi10 := 1/2 D1(f)(x0, y0) + 1/2 D2(f)(x0, y0) f(x0, y0)
```

```

> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
cond2a :=  $\frac{1}{2} = \frac{1}{2}$ 
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
cond2b :=  $\frac{1}{2} = \frac{1}{2}$ 
[ condiciones para el orden 3
> Fi20:=expand((1-alpha)*k120+alpha*k220);
Fi20 :=  $\frac{1}{4} \frac{D_{1,1}(f)(x0,y0)}{\alpha} + \frac{1}{2} \frac{D_{1,2}(f)(x0,y0) f(x0,y0)}{\alpha} + \frac{1}{4} \frac{f(x0,y0)^2 D_{2,2}(f)(x0,y0)}{\alpha}$ 
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
cond3a :=  $\frac{1}{6} = \frac{1}{8\alpha}$ 
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
cond3b :=  $\frac{1}{3} = \frac{1}{4\alpha}$ 
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
cond3c :=  $\frac{1}{6} = \frac{1}{8\alpha}$ 
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
cond3d :=  $\frac{1}{6} = 0$ 
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
cond3e :=  $\frac{1}{6} = 0$ 

```

b)

función de estabilidad

```

> restart;
> interface(labeling=false);
> f:=(x,y)->lambda*y;
f := (x, y) → λ y
> k1_:=f(x0,y0);
k1_ := λ y0
> k2_:=f(x0+1/(2*alpha)*h,y0+1/(2*alpha)*h*k1_);
k2_ :=  $\lambda \left( y0 + \frac{h \lambda y0}{2 \alpha} \right)$ 
> y1:=expand(y0+h*((1-alpha)*k1_+alpha*k2_));
y1 :=  $y0 + h \lambda y0 + \frac{1}{2} \lambda^2 h^2 y0$ 
> y1:=collect(subs(lambda=z/h,y1),y0);
y1 :=  $\left( 1 + z + \frac{1}{2} z^2 \right) y0$ 
> r:=unapply(collect(simplify(y1/y0),z),z);
r :=  $z \rightarrow 1 + z + \frac{1}{2} z^2$ 

```

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-02

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0			
3/2	β		
0	1	-1	

	1/3	α	1/3

- a) Para qué valores de los parámetros α y β los métodos son de orden 2 ?
- b) Existe alguna elección de α y β que proporcione un método de orden 3 ?

(lo que antecede es Ejercicio del capítulo 3)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α y β .

d) En el supuesto de que $|\alpha| < \frac{2}{3}$ y de que $\beta = 0$, represéntese la región de estabilidad del método.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+3/2*h,y0+beta*h*k1_):
[ > k3_:=f(x0,y0+h*k1_-h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
```

```
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(1/3*k1_0+alpha*k2_0+1/3*k3_0);
```

$$Fi_0 := \frac{2}{3} f(x0, y0) + \alpha f(x0, y0)$$

```
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

$$cond1a := 1 = \frac{2}{3} + \alpha$$

```
[ > sol1:=solve(cond1a,{alpha});
```

$$sol1 := \{ \alpha = \frac{1}{3} \}$$

```
[ > alpha:=subs(sol1,alpha);
```

$$\alpha := \frac{1}{3}$$

[condiciones para el orden 2

```
[ > Fi10:=expand(1/3*k110+alpha*k210+1/3*k310);
```

$$Fi10 := \frac{1}{2} D_1(f)(x0, y0) + \frac{1}{3} D_2(f)(x0, y0) \beta f(x0, y0)$$

```
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
```

$$cond2a := \frac{1}{2} = \frac{1}{2}$$

```
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = \frac{\beta}{3}$$

```
[ > sol2:=solve({cond2a,cond2b},{beta});
```

$$sol2 := \{ \beta = \frac{3}{2} \}$$

```
[ > beta:=subs(sol2,beta);
```

$$\beta := \frac{3}{2}$$

[b)

[condiciones para el orden 3

```
[ > Fi20:=expand(1/3*k120+alpha*k220+1/3*k320);
```

$$Fi20 := \frac{3}{4} D_{1,1}(f)(x0, y0) + \frac{3}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{3}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2 - D_2(f)(x0, y0) D_1(f)(x0, y0) - D_2(f)(x0, y0)^2 f(x0, y0)$$

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{3}{8}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{3}{4}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{3}{8}$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[
```

```
2](f)(x0,y0),D[1](f)(x0,y0));
```

$$\text{cond3d} := \frac{1}{6} = \frac{-1}{2}$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \frac{-1}{2}$$

c)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false):
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+3/2*h,y0+beta*h*k1_);
```

$$k2_ := \lambda (y0 + \beta h \lambda y0)$$

```
> k3_:=f(x0,y0+h*k1_-h*k2_);
```

$$k3_ := \lambda (y0 + h \lambda y0 - h \lambda (y0 + \beta h \lambda y0))$$

```
> y1:=expand(y0+h*(1/3*k1_+alpha*k2_+1/3*k3_));
```

$$y1 := y0 + \frac{2 h \lambda y0}{3} + h \alpha \lambda y0 + \alpha \lambda^2 \beta h^2 y0 - \frac{h^3 \lambda^3 \beta y0}{3}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \frac{2}{3} z + \alpha z + \alpha z^2 \beta - \frac{1}{3} z^3 \beta \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 - \frac{1}{3} z^3 \beta + \alpha z^2 \beta + \left(\frac{2}{3} + \alpha \right) z$$

d)

```
> beta:=0;
```

$$\beta := 0$$

```
> r(z);
```

$$1 + \left(\frac{2}{3} + \alpha \right) z$$

```
> region:=abs(r(z))<1;
```

$$\text{region} := \left| 1 + \left(\frac{2}{3} + \alpha \right) z \right| < 1$$

```
> div:=2/3+alpha;
```

$$\text{div} := \frac{2}{3} + \alpha$$

```
> region:=abs(collect(r(z)/div,z))<1/div;
```

$$\text{region} := \left| z + \frac{1}{\frac{2}{3} + \alpha} \right| < \frac{1}{\frac{2}{3} + \alpha}$$

```
> radio:=1/div;
```

$$\text{radio} := \frac{1}{\frac{2}{3} + \alpha}$$

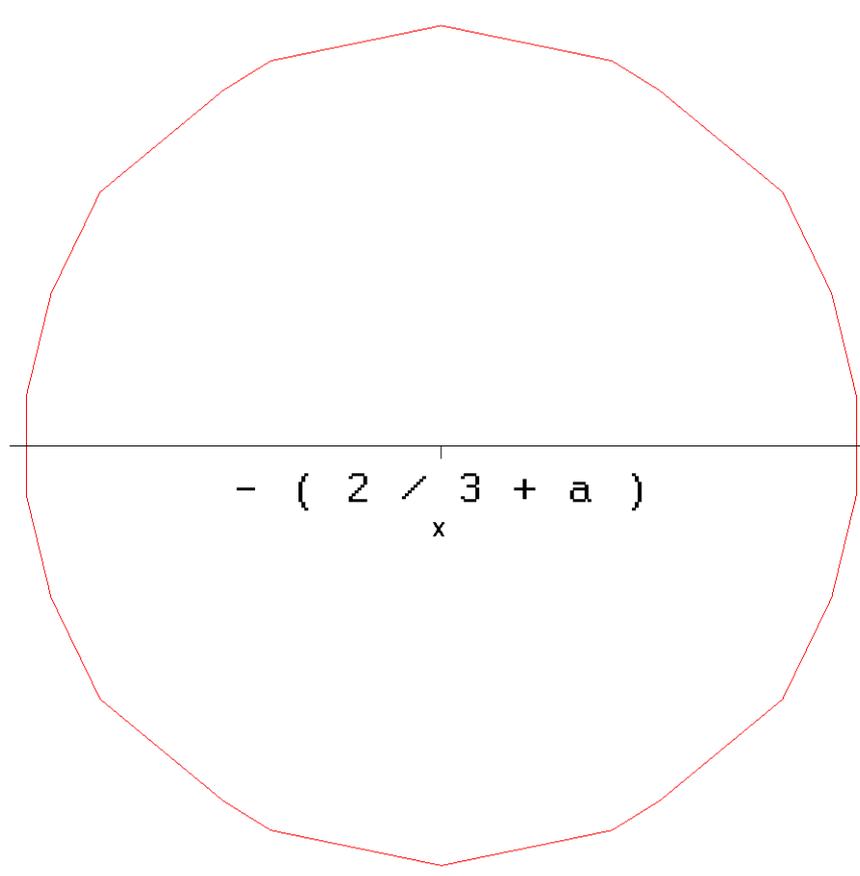
```
> with(plots):alpha:=1/3;
```

```
implicitplot((x+radio)^2+y^2=radio^2,x=-5..5,y=-3..3
```

```
,xtickmarks=[-1=`- ( 2 / 3 + a
```

```
)`],ytickmarks=0,scaling=constrained,axesfont=[COURIER,BOLD,18]);
```

$$\alpha := \frac{1}{3}$$



Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-03

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0			
1/2	1/2		
1	1- α	α	

	1/6	b_2	b_3

- a) Para qué valores de los parámetros α , b_2 y b_3 los métodos son de orden 2 ?
- b) Existen valores de los parámetros α , b_2 y b_3 para los que los métodos sean de orden 3 ?

(lo que antecede es Ejercicio del capítulo 3)

- c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α , b_2 y b_3 . Calcúlese esta misma función para los métodos de orden 2 y orden 3 que, en su caso, se hayan podido obtener.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/2*h,y0+1/2*h*k1_):
[ > k3_:=f(x0+h,y0+(1-alpha)*h*k1_+alpha*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
```

```
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(1/6*k1_0+b[2]*k2_0+b[3]*k3_0);
```

$$Fi_0 := \frac{1}{6}f(x0, y0) + b_2 f(x0, y0) + b_3 f(x0, y0)$$

```
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

$$cond1a := 1 = \frac{1}{6} + b_2 + b_3$$

[condiciones para el orden 2

```
[ > Fi10:=expand(1/6*k110+b[2]*k210+b[3]*k310);
```

$$Fi10 := \frac{1}{2}b_2 D_1(f)(x0, y0) + \frac{1}{2}b_2 D_2(f)(x0, y0) f(x0, y0) + b_3 D_1(f)(x0, y0) + b_3 D_2(f)(x0, y0) f(x0, y0)$$

```
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
```

$$cond2a := \frac{1}{2} = \frac{1}{2}b_2 + b_3$$

```
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = \frac{1}{2}b_2 + b_3$$

```
[ > sol1:=solve({cond1a,cond2a,cond2b},{alpha,b[2],b[3]});
```

$$sol1 := \{ \alpha = \alpha, b_2 = \frac{2}{3}, b_3 = \frac{1}{6} \}$$

```
[ > b[2]:=subs(sol1,b[2]);
```

$$b_2 := \frac{2}{3}$$

```
[ > b[3]:=subs(sol1,b[3]);
```

$$b_3 := \frac{1}{6}$$

[b)

[condiciones para el orden 3

```
[ > Fi20:=expand(1/6*k120+b[2]*k220+b[3]*k320);
```

$$Fi20 := \frac{1}{3}D_{1,1}(f)(x0, y0) + \frac{2}{3}D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{3}D_{2,2}(f)(x0, y0) f(x0, y0)^2 + \frac{1}{6}D_2(f)(x0, y0) \alpha D_1(f)(x0, y0) + \frac{1}{6}D_2(f)(x0, y0)^2 \alpha f(x0, y0)$$

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{1}{6}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{1}{3}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{6}$$

```
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = \frac{\alpha}{12}$$

```
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f
```

```
(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \frac{\alpha}{12}$$

```
> sol2:=solve({cond3a,cond3b,cond3c,cond3d,cond3e},{alpha});
```

```
sol2 := {alpha = 2}
```

```
> alpha:=subs(sol2,alpha);
```

```
alpha := 2
```

c)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false);
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+1/2*h,y0+1/2*h*k1_);
```

$$k2_ := \lambda \left(y0 + \frac{1}{2} h \lambda y0 \right)$$

```
> k3_:=f(x0+h,y0+(1-alpha)*h*k1_+alpha*h*k2_);
```

$$k3_ := \lambda \left(y0 + (1 - \alpha) h \lambda y0 + \alpha h \lambda \left(y0 + \frac{1}{2} h \lambda y0 \right) \right)$$

```
> y1:=expand(y0+h*(1/6*k1_+b[2]*k2_+b[3]*k3_));
```

$$y1 := y0 + \frac{h \lambda y0}{6} + h b_2 \lambda y0 + \frac{1}{2} b_2 \lambda^2 h^2 y0 + h b_3 \lambda y0 + b_3 \lambda^2 h^2 y0 + \frac{1}{2} b_3 \lambda^3 \alpha h^3 y0$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \frac{z}{6} + b_2 z + \frac{1}{2} b_2 z^2 + b_3 z + b_3 z^2 + \frac{1}{2} b_3 z^3 \alpha \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + \frac{1}{2} b_3 z^3 \alpha + \left(\frac{1}{2} b_2 + b_3 \right) z^2 + \left(\frac{1}{6} + b_2 + b_3 \right) z$$

función de estabilidad para los metodos de orden 2

```
> r2:=unapply(subs({b[2]=2/3,b[3]=1/6},r(z)),z);
```

$$r2 := z \rightarrow 1 + \frac{1}{12} z^3 \alpha + \frac{1}{2} z^2 + z$$

función de estabilidad para los metodos de orden 3

```
> r3:=unapply(subs({alpha=2},r2(z)),z);
```

$$r3 := z \rightarrow 1 + \frac{1}{6} z^3 + \frac{1}{2} z^2 + z$$

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-04

Considérese el método genérico explícito de Runge-Kutta dado por el siguiente tablero

0			
c_2	c_2		
c_3	0	c_3	
	0	0	1

- a) Cuáles deben ser los coeficientes para alcanzar al menos el orden 2 ?
- b) Es posible alcanzar el orden 3 ?

(lo que antecede es Ejercicio del capítulo 3)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+c[2]*h,y0+c[2]*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+c[3]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

- [a)
- [condiciones para el orden 1
- [> Fi_0:=expand(k3_0);

```

[                                      $Fi_0 := f(x_0, y_0)$ 
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                      $cond1a := 1 = 1$ 
[ condiciones para el orden 2
[ > Fi10:=expand(k310);
[                                      $Fi10 := D_1(f)(x_0, y_0) c_3 + D_2(f)(x_0, y_0) c_3 f(x_0, y_0)$ 
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                      $cond2a := \frac{1}{2} = c_3$ 
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
[                                      $cond2b := \frac{1}{2} = c_3$ 
[ > sol1:=solve({cond2a,cond2b},{c[3]});
[                                      $sol1 := \{c_3 = \frac{1}{2}\}$ 
[ > c[3]:=subs(sol1,c[3]);
[                                      $c_3 := \frac{1}{2}$ 

```

b)

condiciones para el orden 3

```

[ > Fi20:=expand(k320);
[  $Fi20 := \frac{1}{4} D_{1,1}(f)(x_0, y_0) + \frac{1}{2} D_{1,2}(f)(x_0, y_0) f(x_0, y_0) + \frac{1}{4} D_{2,2}(f)(x_0, y_0) f(x_0, y_0)^2 + D_2(f)(x_0, y_0) D_1(f)(x_0, y_0) c_2$ 
[  $+ D_2(f)(x_0, y_0)^2 c_2 f(x_0, y_0)$ 
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
[                                      $cond3a := \frac{1}{6} = \frac{1}{8}$ 
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
[                                      $cond3b := \frac{1}{3} = \frac{1}{4}$ 
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
[                                      $cond3c := \frac{1}{6} = \frac{1}{8}$ 
[ > cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
[                                      $cond3d := \frac{1}{6} = \frac{1}{2} c_2$ 
[ > cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
[                                      $cond3e := \frac{1}{6} = \frac{1}{2} c_2$ 

```

c)

función de estabilidad

```

[ > restart;
[ > interface(labeling=false);
[ > f:=(x,y)->lambda*y;
[                                      $f := (x, y) \rightarrow \lambda y$ 
[ > k1_:=f(x0,y0);
[                                      $k1_ := \lambda y_0$ 
[ > k2_:=f(x0+c[2]*h,y0+c[2]*h*k1_);
[                                      $k2_ := \lambda (y_0 + c_2 h \lambda y_0)$ 

```

```

[ > k3_:=f(x0+c[3]*h,y0+c[3]*h*k2_);
                                     k3_ := λ (y0 + c3 h λ (y0 + c2 h λ y0))
[ > y1:=expand(y0+h*(k3_));
                                     y1 := y0 + h λ y0 + h2 λ2 c3 y0 + h3 λ3 c3 c2 y0
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                     y1 := (1 + z + z2 c3 + z3 c3 c2) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
                                     r := z → 1 + z + z2 c3 + z3 c3 c2
[ función de estabilidad para los metodos de orden 2
[ > r2:=unapply(subs(c[3]=1/2,r(z)),z);
                                     r2 := z → 1 + z +  $\frac{1}{2}z^2 + \frac{1}{2}z^3 c_2$ 

```

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-05

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos dados por el tablero

0			
c_2		$a_{2,1}$	
c_3		$a_{3,1}$	$a_{3,2}$
		0	0
			b_3

a) Para qué valores de los parámetros $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 los métodos son de orden 2 ?

b) De los métodos anteriores, cuáles minimizan además el término del error ?

c) Existen valores de los parámetros $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 para los que los métodos sean de orden 3 ?

(lo que antecede es Ejercicio del capítulo 3)

d) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de $c_2, c_3, a_{2,1}, a_{3,1}, a_{3,2}$ y b_3 . Calcúlese esta misma función para los métodos de orden 2, los que minimizan el error y los de orden 3 que, en su caso, se hayan podido obtener.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+c[2]*h,y0+a[2,1]*h*k1_):
[ > k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ ]
```

```
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(0*k1_0+0*k2_0+b[3]*k3_0);
[                                      $Fi_0 := b_3 f(x0, y0)$ 
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                      $cond1a := 1 = b_3$ 
```

[condiciones para el orden 2

```
[ > Fi10:=expand(0*k110+0*k210+b[3]*k310);
[                                      $Fi10 := b_3 D_1(f)(x0, y0) c_3 + b_3 D_2(f)(x0, y0) a_{3,1} f(x0, y0) + b_3 D_2(f)(x0, y0) a_{3,2} f(x0, y0)$ 
[ > cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
[                                      $cond2a := \frac{1}{2} = b_3 c_3$ 
[ > cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
```

$$cond2b := \frac{1}{2} = b_3 a_{3,1} + b_3 a_{3,2}$$

```
[ > sol1:=solve({cond1a,cond2a,cond2b},{b[3],c[3],a[3,1]});
```

$$sol1 := \{ c_3 = \frac{1}{2}, b_3 = 1, a_{3,1} = \frac{1}{2} - a_{3,2} \}$$

```
[ > b[3]:=subs(sol1,b[3]);
```

$$b_3 := 1$$

```
[ > c[3]:=subs(sol1,c[3]);
```

$$c_3 := \frac{1}{2}$$

```
[ > a[3,1]:=subs(sol1,a[3,1]);
```

$$a_{3,1} := \frac{1}{2} - a_{3,2}$$

[b)

[condiciones para el orden 3 o para el error

```
[ > Fi20:=expand(0*k120+0*k220+b[3]*k320);
[  $Fi20 := \frac{1}{4} D_{1,1}(f)(x0, y0) + \frac{1}{2} D_{1,2}(f)(x0, y0) f(x0, y0) + \frac{1}{4} D_{2,2}(f)(x0, y0) f(x0, y0)^2 + 2 D_2(f)(x0, y0) a_{3,2} D_1(f)(x0, y0) c_2$ 
[  $+ 2 D_2(f)(x0, y0)^2 a_{3,2} a_{2,1} f(x0, y0)$ 
```

```
[ > cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$cond3a := \frac{1}{6} = \frac{1}{8}$$

```
[ > cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$cond3b := \frac{1}{3} = \frac{1}{4}$$

```
[ > cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$cond3c := \frac{1}{6} = \frac{1}{8}$$

```
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$cond3d := \frac{1}{6} = a_{3,2} c_2$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$cond3e := \frac{1}{6} = a_{3,2} a_{2,1}$$

c)

minimización del error

```
> sol2:=solve({cond3d,cond3e},{a[2,1],a[3,2]});
```

$$sol2 := \{a_{2,1} = c_2, a_{3,2} = \frac{1}{6} \frac{1}{c_2}\}$$

```
> a[2,1]:=subs(sol2,a[2,1]);
```

$$a_{2,1} := c_2$$

```
> a[3,2]:=subs(sol2,a[3,2]);
```

$$a_{3,2} := \frac{1}{6} \frac{1}{c_2}$$

d)

función de estabilidad

```
> restart;
```

```
> interface(labeling=false);
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+c[2]*h,y0+a[2,1]*h*k1_);
```

$$k2_ := \lambda (y0 + a_{2,1} h \lambda y0)$$

```
> k3_:=f(x0+c[3]*h,y0+a[3,1]*h*k1_+a[3,2]*h*k2_);
```

$$k3_ := \lambda (y0 + a_{3,1} h \lambda y0 + a_{3,2} h \lambda (y0 + a_{2,1} h \lambda y0))$$

```
> y1:=expand(y0+h*(0*k1_+0*k2_+b[3]*k3_));
```

$$y1 := y0 + h b_3 \lambda y0 + h^2 b_3 \lambda^2 a_{3,1} y0 + h^2 b_3 \lambda^2 a_{3,2} y0 + h^3 b_3 \lambda^3 a_{3,2} a_{2,1} y0$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := (1 + b_3 z + b_3 z^2 a_{3,1} + b_3 z^2 a_{3,2} + b_3 z^3 a_{3,2} a_{2,1}) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + b_3 z^3 a_{3,2} a_{2,1} + (b_3 a_{3,1} + b_3 a_{3,2}) z^2 + b_3 z$$

función de estabilidad para los metodos de orden 2

```
> r2:=unapply(subs({c[3]=1/2,a[3,1]=1/2-a[3,2],b[3]=1},r(z)),z);
```

$$r2 := z \rightarrow 1 + z^3 a_{3,2} a_{2,1} + \frac{1}{2} z^2 + z$$

función de estabilidad para los metodos que minimizan el error

```
> r3:=unapply(subs({a[2,1]=c[2],a[3,2]=1/6/c[2]},r2(z)),z);
```

$$r3 := z \rightarrow 1 + \frac{1}{6} z^3 + \frac{1}{2} z^2 + z$$

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-06 (del EXAMEN EXTRAORDINARIO de 22JUN11)

Considérese la familia de métodos de Runge-Kutta de 3 etapas y explícitos de tablero

0				
α		α		
α		0	α	

		β	θ	θ

De entre ellos se buscaran los que proporcionen el orden mas alto.

(lo que antecede es Ejercicio del capítulo 3)

Dígase cuál es la función de estabilidad de esos métodos de mayor orden.

```

[ > restart;
[ > interface(labeling=false):
[ > y1:=f(x,y(x));
[                                     y1 := f(x, y(x))
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x));
[                                     y2 := D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x))
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x));
[ y3 := D1,1(f)(x, y(x)) + D1,2(f)(x, y(x)) f(x, y(x)) + (D1,2(f)(x, y(x)) + D2,2(f)(x, y(x)) f(x, y(x))) f(x, y(x))
[       + D2(f)(x, y(x)) (D1(f)(x, y(x)) + D2(f)(x, y(x)) f(x, y(x)))
[ > Y1:=subs({y(x)=y0,x=x0},y1);
[                                     Y1 := f(x0, y0)
[ > Y2:=subs({y(x)=y0,x=x0},y2);
[                                     Y2 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0)
[ > Y3:=subs({y(x)=y0,x=x0},y3);
[ Y3 := D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + (D1,2(f)(x0, y0) + D2,2(f)(x0, y0) f(x0, y0)) f(x0, y0)
[       + D2(f)(x0, y0) (D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0))
[ > k1_:=f(x0,y0);
[                                     k1_ := f(x0, y0)
[ > k2_:=f(x0+alpha*h,y0+alpha*h*k1_);
[                                     k2_ := f(x0 + alpha h, y0 + alpha h f(x0, y0))
[ > k3_:=f(x0+alpha*h,y0+alpha*h*k2_);
[                                     k3_ := f(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0)))
[ > k1_0:=subs(h=0,k1_);
[                                     k1_0 := f(x0, y0)
[ > k2_0:=subs(h=0,k2_);
[                                     k2_0 := f(x0, y0)
[ > k3_0:=subs(h=0,k3_);
[                                     k3_0 := f(x0, y0)
[ > k11:=diff(k1_,h);

```

```

[
    k11 := 0
> k21:=diff(k2_,h);
    k21 := D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)
> k31:=diff(k3_,h);
k31 := D1(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha +
    D2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))
> k110:=subs(h=0,k11);
    k110 := 0
> k210:=subs(h=0,k21);
    k210 := D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0)
> k310:=subs(h=0,k31);
    k310 := D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0)
> k12:=diff(k11,h);
    k12 := 0
> k22:=diff(k21,h);
k22 := (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha
    + (D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha f(x0, y0)
> k32:=diff(k31,h);
k32 := (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha +
    D1,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))) alpha + (
    D1,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (
    alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)))) (
    alpha f(x0 + alpha h, y0 + alpha h f(x0, y0))
    + alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0))) +
    D2(f)(x0 + alpha h, y0 + alpha h f(x0 + alpha h, y0 + alpha h f(x0, y0))) (
    2 alpha (D1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) + alpha h (
    (D1,1(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha
    + (D1,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha + D2,2(f)(x0 + alpha h, y0 + alpha h f(x0, y0)) alpha f(x0, y0)) alpha f(x0, y0)))
> k120:=subs(h=0,k12);
    k120 := 0
> k220:=subs(h=0,k22);
    k220 := (D1,1(f)(x0, y0) alpha + D1,2(f)(x0, y0) alpha f(x0, y0)) alpha + (D1,2(f)(x0, y0) alpha + D2,2(f)(x0, y0) alpha f(x0, y0)) alpha f(x0, y0)
> k320:=subs(h=0,k32);
k320 := (D1,1(f)(x0, y0) alpha + D1,2(f)(x0, y0) alpha f(x0, y0)) alpha + (D1,2(f)(x0, y0) alpha + D2,2(f)(x0, y0) alpha f(x0, y0)) alpha f(x0, y0)
    + 2 D2(f)(x0, y0) alpha (D1(f)(x0, y0) alpha + D2(f)(x0, y0) alpha f(x0, y0))
a)
condiciones para el orden 1
> Fi_0:=expand(beta*k1_0+theta*k2_0+theta*k3_0);
    Fi_0 := beta f(x0, y0) + 2 theta f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
    cond1a := 1 = beta + 2 theta
condiciones para el orden 2
> Fi10:=expand(beta*k110+theta*k210+theta*k310);
    Fi10 := 2 theta D1(f)(x0, y0) alpha + 2 theta D2(f)(x0, y0) alpha f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
    cond2a := 1/2 = 2 theta alpha
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0))

```

```
), f(x0, y0));
```

$$\text{cond2b} := \frac{1}{2} = 2 \theta \alpha$$

```
> sol1:=solve({cond1a, cond2a, cond2b});
```

$$\text{sol1} := \{ \beta = 1 - 2 \theta, \alpha = \frac{1}{4 \theta}, \theta = \theta \}$$

b)

condiciones para el orden 3

```
> Fi20:=expand(beta*k120+theta*k220+theta*k320);
```

$$\text{Fi20} := 2 \theta D_{1,1}(f)(x_0, y_0) \alpha^2 + 4 \theta D_{1,2}(f)(x_0, y_0) \alpha^2 f(x_0, y_0) + 2 \theta \alpha^2 f(x_0, y_0)^2 D_{2,2}(f)(x_0, y_0) \\ + 2 \theta D_2(f)(x_0, y_0) \alpha^2 D_1(f)(x_0, y_0) + 2 \theta D_2(f)(x_0, y_0)^2 \alpha^2 f(x_0, y_0)$$

```
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
```

$$\text{cond3a} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
```

$$\text{cond3b} := \frac{1}{3} = 2 \theta \alpha^2$$

```
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
```

$$\text{cond3c} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
```

$$\text{cond3d} := \frac{1}{6} = \theta \alpha^2$$

```
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
```

$$\text{cond3e} := \frac{1}{6} = \theta \alpha^2$$

```
> sol2:=solve({cond1a, cond2a, cond2b, cond3a, cond3b, cond3c, cond3d, cond3e}, {alpha, beta, theta});
```

$$\text{sol2} := \{ \alpha = \frac{2}{3}, \theta = \frac{3}{8}, \beta = \frac{1}{4} \}$$

metodo de mayor orden, orden 3,

0 |

2/3 | 2/3

2/3 | 0 2/3

| 1/4 3/8 3/8

que suele llamarse de NYSTRÖM de tercer orden

es sabido que los métodos de 3 etapas y orden máximo 3 tienen función de estabilidad

$$r(z) = 1 + z + \frac{1}{2} z^2 + \frac{1}{6} z^3$$

o, sea, cuatro primeros términos del desarrollo de e^z

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-07 (del EXAMEN ORDINARIO de 24ENE11)

Considérese la familia de métodos de RUNGE-KUTTA de 3 etapas y explícitos dados por el tablero

0				
3/2		β		
0		α	-1	

		1/3	1/3	1/3

a) Dígase para qué valores de α y β los métodos son de orden 2

b) De entre los anteriores métodos, búsquense los que minimizan el coeficiente del término del error.

(lo que antecede es Ejercicio del capítulo 3)

c) Calcúlese la función de estabilidad de los métodos de orden 2

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+alpha*h,y0+alpha*h*k1_):
[ > k3_:=f(x0+h,y0+(1/2)*h*k1_+(1/2)*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[ > k12:=diff(k11,h):
[ > k22:=diff(k21,h):
[ > k32:=diff(k31,h):
[ > k120:=subs(h=0,k12):
[ > k220:=subs(h=0,k22):
[ > k320:=subs(h=0,k32):
```

[a)
[condiciones para el orden 1
[

```

> Fi_0:=expand(beta*k1_0+0*k2_0+(1-beta)*k3_0);
                                Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                cond1a := 1 = 1
condiciones para el orden 2
> Fi10:=expand(beta*k110+0*k210+(1-beta)*k310);
                                Fi10 := D1(f)(x0, y0) + D2(f)(x0, y0) f(x0, y0) - beta D1(f)(x0, y0) - beta D2(f)(x0, y0) f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                cond2a := 1/2 = 1 - beta
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                cond2b := 1/2 = 1 - beta
> sol2a:=solve({cond2a,cond2b},{beta});
                                sol2a := {beta = 1/2}
> beta:=subs(sol2a,beta);
                                beta := 1/2

```

b)

coeficientes del termino $\frac{1}{3}f^2 - \Phi'(0)$ para el orden 3

```

> Fi20:=expand(beta*k120+0*k220+(1-beta)*k320);
Fi20 := 1/2 D1,1(f)(x0, y0) + D1,2(f)(x0, y0) f(x0, y0) + 1/2 D2,2(f)(x0, y0) f(x0, y0)^2 + 1/2 D2(f)(x0, y0) D1(f)(x0, y0) alpha
+ 1/2 D2(f)(x0, y0)^2 alpha f(x0, y0)
> tau3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))-coeff(Fi20,D[1,1](f)(x0,y0));
                                tau3a := -1/6
> tau3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))-coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
                                tau3b := -1/3
> tau3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)-coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
                                tau3c := -1/6
> tau3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
                                tau3d := 1/3 - alpha/2
> tau3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))-coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
                                tau3e := 1/3 - alpha/2
> sol3:=solve({tau3d,tau3e},{alpha});
                                sol3 := {alpha = 2/3}

```

c)

función de estabilidad para orden 2

```

> restart:
> interface(labeling=false):

```

```

[ > beta:=1/2:
[ > f:=(x,y)->lambda*y;
                                f:=(x,y) -> lambda y
[ > k1_:=f(x0,y0);
                                k1_ := lambda y0
[ > k2_:=f(x0+alpha*h,y0+alpha*h*k1_);
                                k2_ := lambda (y0 + alpha h lambda y0)
[ > k3_:=f(x0+h,y0+(1/2)*h*k1_+(1/2)*h*k2_);
                                k3_ := lambda (y0 + (h lambda y0)/2 + (h lambda (y0 + alpha h lambda y0))/2)
[ > y1:=expand(y0+h*(beta*k1_+0*k2_+(1-beta)*k3_));
                                y1 := y0 + h lambda y0 + (h^2 lambda^2 y0)/2 + (h^3 lambda^3 alpha y0)/4
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                y1 := (1 + z + (1/2)z^2 + (1/4)z^3 alpha) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
                                r := z -> 1 + z + (1/2)z^2 + (1/4)z^3 alpha
[ >
[ >

```

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-08 (del EXAMEN EXTRAORDINARIO de 09JUL03)

Considérese la familia de métodos de RUNGE-KUTTA de 3 etapas y explícitos dados por el siguiente tablero

0			
1/2	α		
1	β	2	

	1/6	2/3	1/6

- a) Para qué valores de los parámetros α y β los métodos son de orden 2 ?
- b) Existe alguna elección de α y β que proporcione un método de orden 3 ?

(lo que antecede es Ejercicio del capítulo 3)

- c) Para la familia de métodos propuesta, calcúlese la función de estabilidad en términos de α y β .
- d) Calcúlese ahora la función de estabilidad para los valores para los que se obtiene el orden 2. Si en este caso hacemos $\beta = 1$ o $\alpha = 0$, que es lo mismo, dibújese la correspondiente región de estabilidad.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+1/2*h,y0+alpha*h*k1_):
[ > k3_:=f(x0+h,y0+h*beta*k1_+h*2*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
[
```



```
> sol3:=solve({cond3b,cond3c,cond3d,cond3e},{alpha});
```

$$\text{sol3} := \left\{ \alpha = \frac{1}{2} \right\}$$

```
> alpha:=subs(sol3,alpha);
```

$$\alpha := \frac{1}{2}$$

c)

función de estabilidad, caso general

```
> restart;
```

```
> interface(labeling=false);
```

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> k1_:=f(x0,y0);
```

$$k1_ := \lambda y0$$

```
> k2_:=f(x0+1/2*h,y0+alpha*h*k1_);
```

$$k2_ := \lambda (y0 + \alpha h \lambda y0)$$

```
> k3_:=f(x0+h,y0+h*beta*k1_+h*2*k2_);
```

$$k3_ := \lambda (y0 + h \beta \lambda y0 + 2 h \lambda (y0 + \alpha h \lambda y0))$$

```
> y1:=expand(y0+h*(1/6*k1_+2/3*k2_+1/6*k3_));
```

$$y1 := y0 + h \lambda y0 + \frac{2 h^2 \lambda^2 \alpha y0}{3} + \frac{h^2 \beta \lambda^2 y0}{6} + \frac{h^2 \lambda^2 y0}{3} + \frac{h^3 \lambda^3 \alpha y0}{3}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + z + \frac{2}{3} z^2 \alpha + \frac{1}{6} \beta z^2 + \frac{1}{3} z^2 + \frac{1}{3} z^3 \alpha \right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow 1 + \frac{1}{3} z^3 \alpha + \left(\frac{2\alpha}{3} + \frac{\beta}{6} + \frac{1}{3} \right) z^2 + z$$

d)

función de estabilidad, orden 2

```
> beta := 1-4*alpha;
```

$$\beta := 1 - 4 \alpha$$

```
> r(z);
```

$$1 + \frac{1}{3} z^3 \alpha + \frac{1}{2} z^2 + z$$

región de estabilidad para $\alpha = 0$

```
> alpha:=0;
```

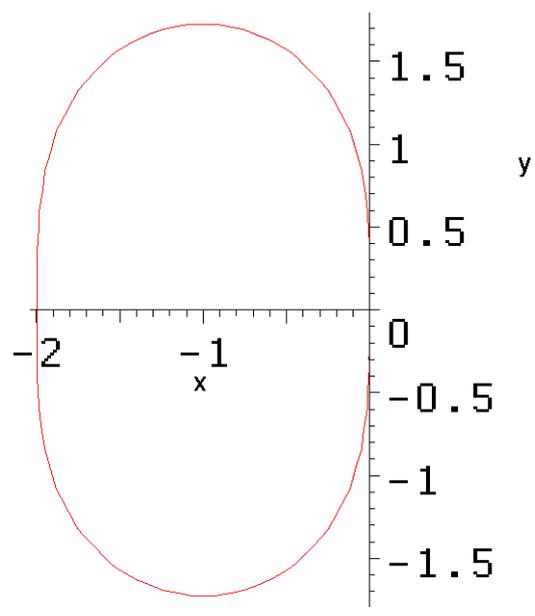
$$\alpha := 0$$

```
> r(z);
```

$$1 + \frac{1}{2} z^2 + z$$

la región de estabilidad es la figura ya estudiada:

```
> z:=x+I*y:with(plots):implicitplot(abs(r(z))=1,x=-3..1,y=-3..3,scaling=constrained,axes  
font=[COURIER,BOLD,18]);
```



Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-09 (del EXAMEN ORDINARIO de 07FEB03)

Considérese el método genérico explícito de Runge-Kutta dado por el siguiente tablero

0			
$\frac{1}{2}$	a_{21}		
1	0	a_{32}	
	0	0	1

a) Dígase cual es el orden máximo que se puede alcanzar y cuáles deben ser los coeficientes para alcanzar dicho orden

b) De entre los métodos de orden máximo, dígase cuáles son los que minimizan el error de truncación

(lo que antecede es Ejercicio del capítulo 3)

c) Para la familia de métodos propuesta, calcúlese la función de estabilidad. Calcúlese la misma función para los métodos de orden máximo y para los que minimizan el error

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=f(x0,y0):
[ > k2_:=f(x0+(1/2)*h,y0+a[2,1]*h*k1_):
[ > k3_:=f(x0+h,y0+a[3,2]*h*k2_):
[ > k1_0:=subs(h=0,k1_):
[ > k2_0:=subs(h=0,k2_):
[ > k3_0:=subs(h=0,k3_):
[ > k11:=diff(k1_,h):
[ > k21:=diff(k2_,h):
[ > k31:=diff(k3_,h):
[ > k110:=subs(h=0,k11):
[ > k210:=subs(h=0,k21):
[ > k310:=subs(h=0,k31):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(k3_0);
```


Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-10

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

0		0	
c_2		a_{21}	a_{22}
		b_1	b_2

b) Compruébese qué se obtiene cuando se fuerza el que $c_2 = 1$, o sea, que las abscisas implicadas sean 0 y 1

(lo que antecede es Ejercicio del capítulo 7)

c) Para los métodos de orden máximo obtenidos, calcúlese la función de estabilidad

d) Calcúlese la función de estabilidad para los métodos del apartado b)

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0,y0),h):
[ > k2_:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1_(h)+a[2,2]*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
[                                     Fi_0 := b1 f(x0, y0) + b2 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

```

[
cond1a := 1 = b1 + b2
> solve(cond1a);
{ b2 = 1 - b1, b1 = b1 }
condiciones para el orden 2
> Fi10:=expand(b[1]*k110+b[2]*k210);
Fi10 := b2 D1(f)(x0, y0) c2 + b2 D2(f)(x0, y0) a2,1 f(x0, y0) + b2 D2(f)(x0, y0) a2,2 f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
cond2a := 1/2 = b2 c2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
cond2b := 1/2 = b2 a2,1 + b2 a2,2
> solve({cond1a,cond2a,cond2b},{b[1],c[2],a[2,1]});
{ c2 = 1/2 * 1/b2, b1 = 1 - b2, a2,1 = -1/2 * (-1 + 2 b2 a2,2) / b2 }
condiciones para el orden 3
> Fi20:=expand(b[1]*k120+b[2]*k220);
Fi20 := b2 D1,1(f)(x0, y0) c2^2 + 2 b2 c2 D1,2(f)(x0, y0) a2,1 f(x0, y0) + 2 b2 c2 D1,2(f)(x0, y0) a2,2 f(x0, y0)
+ b2 D2,2(f)(x0, y0) a2,1^2 f(x0, y0)^2 + 2 b2 D2,2(f)(x0, y0) a2,1 f(x0, y0)^2 a2,2 + b2 D2,2(f)(x0, y0) a2,2^2 f(x0, y0)^2
+ 2 b2 D2(f)(x0, y0) a2,2 D1(f)(x0, y0) c2 + 2 b2 D2(f)(x0, y0)^2 a2,2 a2,1 f(x0, y0) + 2 b2 D2(f)(x0, y0)^2 a2,2^2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
cond3a := 1/6 = 1/2 b2 c2^2
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
cond3b := 1/3 = b2 c2 a2,1 + b2 c2 a2,2
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
cond3c := 1/6 = 1/2 b2 a2,1^2 + b2 a2,1 a2,2 + 1/2 b2 a2,2^2
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
cond3d := 1/6 = b2 c2 a2,2
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
cond3e := 1/6 = b2 a2,1 a2,2 + b2 a2,2^2
> solucion:=solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});
solucion := { c2 = 2/3, b1 = 1/4, b2 = 3/4, a2,1 = 1/3, a2,2 = 1/3 }

```

Es decir, existe una única solución de orden 3. Se trata del método dado por el tablero

$$\begin{array}{c|c}
0 & 0 \\
\hline
\frac{2}{3} & \frac{1}{3} \quad \frac{1}{3}
\end{array}$$

$$\left| \begin{array}{cc} 1 & 3 \\ 4 & 4 \end{array} \right.$$

b)

forzando a que $c_2 = 1$

```
> solucion1:=solve({cond1a,cond2a,cond2b,c[2]=1},{c[2],a[2,1],b[1],b[2]});
```

$$solucion1 := \{c_2 = 1, b_1 = \frac{1}{2}, b_2 = \frac{1}{2}, a_{2,1} = 1 - a_{2,2}\}$$

Se obtiene la familia de orden 2 dada por el tablero

$$\begin{array}{c|cc} 0 & 0 & \\ \hline 1 & 1 - a_{22} & a_{22} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

uno de cuyos métodos es

```
> solucion2:=subs(a[2,2]=1/2,solucion1) union {a[2,2]=1/2};
```

$$solucion2 := \{c_2 = 1, b_1 = \frac{1}{2}, b_2 = \frac{1}{2}, a_{2,2} = \frac{1}{2}, a_{2,1} = \frac{1}{2}\}$$

el conocido como LOBATTO IIIA de 2 etapas

$$\begin{array}{c|cc} 0 & 0 & \\ \hline 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

c)

función de estabilidad

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> ka1:=unapply(f(x0,y0),h);
```

$$ka1 := h \rightarrow \lambda y0$$

```
> ka2:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)),h);
```

$$ka2 := h \rightarrow \lambda (y0 + a_{2,1} h kp1(h) + a_{2,2} h kp2(h))$$

```
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
```

$$test := \{k1 = \lambda y0, k2 = -\frac{\lambda y0 (1 + \lambda a_{2,1} h)}{-1 + \lambda a_{2,2} h}\}$$

```
> k1:=subs(test,k1);k2:=subs(test,k2);
```

$$k1 := \lambda y0$$

$$k2 := -\frac{\lambda y0 (1 + \lambda a_{2,1} h)}{-1 + \lambda a_{2,2} h}$$

```
> y1:=expand(y0+h*(b[1]*k1+b[2]*k2));
```

$$y1 := y0 + h b_1 \lambda y0 - \frac{h b_2 \lambda y0}{-1 + \lambda a_{2,2} h} - \frac{b_2 \lambda^2 y0 a_{2,1} h^2}{-1 + \lambda a_{2,2} h}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + b_1 z - \frac{b_2 z}{-1 + z a_{2,2}} - \frac{b_2 z^2 a_{2,1}}{-1 + z a_{2,2}} \right) y0$$

> `r:=unapply(collect(simplify(y1/y0),z),z);`

$$r := z \rightarrow -\frac{1 + (-b_1 a_{2,2} + b_2 a_{2,1}) z^2 + (-a_{2,2} + b_1 + b_2) z}{-1 + z a_{2,2}}$$

que es la función de estabilidad de la familia general propuesta, mientras que

> `r1:=unapply(simplify(subs(solucion,r(z))),z);`

$$r1 := z \rightarrow -\frac{1}{2} \frac{6 + z^2 + 4z}{-3 + z}$$

lo es de la solución de orden 3 obtenida. Ahora, el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r1(z)$ sea inferior a 1 no será todo el semiplano de parte real negativa, ya que la función de estabilidad tiende a infinito cuando z tiende a infinito en cualquier dirección. O sea, el método no es A-estable

d)

función de estabilidad para métodos de b)

> `r2:=unapply(simplify(subs(solucion1,r(z))),z);`

$$r2 := z \rightarrow \frac{1}{2} \frac{-2 + 2 z^2 a_{2,2} - z^2 - 2 z + 2 z a_{2,2}}{-1 + z a_{2,2}}$$

y, a priori, nada podemos decir de la A-estabilidad. Sin embargo, para el método LOBATTO IIIA de 2 etapas

> `r3:=unapply(simplify(subs(solucion2,r(z))),z);`

$$r3 := z \rightarrow -\frac{2 + z}{-2 + z}$$

>

el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r3(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2. Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-11

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

1		1		
c_2		0		a_{22}

		b_1		b_2

b) Pruébese que, entre los métodos obtenidos, se encuentra el método de GAUSS de 1 sola etapa

(lo que antecede es Ejercicio del capítulo 7)

c) Para los métodos obtenidos, calcúlese la función de estabilidad

d) Calcúlese la función de estabilidad para el citado método de GAUSS, y compruébese que es A-estable

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+h,y0+h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+c[2]*h,y0+a[2,2]*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
[ a)
[ condiciones para el orden 1
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
```


$$\frac{1}{2} \mid 0 \quad \frac{1}{2}$$

$$\mid 0 \quad 1$$

que no hace intervenir la primera etapa y que coincide, por lo tanto con

$$\frac{1}{2} \mid \frac{1}{2}$$

$$\mid 1$$

que no es sino el método de Gauss de 1 etapa

c)

función de estabilidad

> `f:=(x,y)->lambda*y;`

$$f := (x, y) \rightarrow \lambda y$$

> `ka1:=unapply(f(x0+h,y0+h*kp1(h)),h);`

$$ka1 := h \rightarrow \lambda (y0 + h kp1(h))$$

> `ka2:=unapply(f(x0+c[2]*h,y0+a[2,2]*h*kp2(h)),h);`

$$ka2 := h \rightarrow \lambda (y0 + a_{2,2} h kp2(h))$$

> `test:=solve({k1=subs(kp1(h)=k1,ka1(h)),k2=subs(kp2(h)=k2,ka2(h))},{k1,k2});`

$$test := \left\{ k2 = -\frac{\lambda y0}{-1 + \lambda a_{2,2} h}, k1 = -\frac{\lambda y0}{-1 + \lambda h} \right\}$$

> `k1:=subs(test,k1);k2:=subs(test,k2);`

$$k1 := -\frac{\lambda y0}{-1 + \lambda h}$$

$$k2 := -\frac{\lambda y0}{-1 + \lambda a_{2,2} h}$$

> `y1:=expand(y0+h*(b[1]*k1+b[2]*k2));`

$$y1 := y0 - \frac{h b_1 \lambda y0}{-1 + \lambda h} - \frac{h b_2 \lambda y0}{-1 + \lambda a_{2,2} h}$$

> `y1:=collect(subs(lambda=z/h,y1),y0);`

$$y1 := \left(1 - \frac{b_1 z}{-1 + z} - \frac{b_2 z}{-1 + z a_{2,2}} \right) y0$$

> `r:=unapply(collect(simplify(y1/y0),z),z);`

$$r := z \rightarrow -\frac{-1 + (-a_{2,2} + b_1 a_{2,2} + b_2) z^2 + (a_{2,2} + 1 - b_1 - b_2) z}{(-1 + z) (-1 + z a_{2,2})}$$

que es la función de estabilidad de la familia general propuesta

> `r2:=unapply(simplify(subs(solucion,r(z))),z);`

$$r2 := z \rightarrow -\frac{-2 b_2 + b_2 z^2 - z + 2 b_2 z}{(-1 + z) (-2 b_2 - z + 2 b_2 z)}$$

que es la función de estabilidad para los métodos de orden 2

d)

Para el método de GAUSS de 1 etapa, obtenido en el conjunto-solucion solucion1 tenemos como función de estabilidad

> `rg:=unapply(simplify(subs(solucion1,r(z))),z);`

$$rg := z \rightarrow -\frac{z + 2}{-2 + z}$$

El dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r1(z)$

sea inferior a 1 , estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

[>

O sea

$$r(z) = \frac{4}{(z-2)^2}$$

El contorno de la región de estabilidad es

$$4 = |z-2|^2$$

o sea

$$2 = |z-2|$$

circunferencia de centro 2 y radio 2. Como además el punto

$$z = -1$$

está en el dominio de estabilidad ya que $r(-1) = \frac{4}{9}$, resulta que

el dominio de estabilidad es el exterior de esta circunferencia y

el método es A-estable.

[>

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-13

Búsqese el método de GAUSS de 2 etapas (es decir, el método implícito de RUNGE-KUTTA con dos etapas y orden 4)

$$c_1 \mid a_{11} \quad a_{12}$$

$$c_2 \mid a_{21} \quad a_{22}$$

$$\mid b_1 \quad b_2$$

(lo que antecede es Ejercicio del capítulo 7)

Calcúlese la función de estabilidad del método y compruébese que es A-estable

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp10(h)+a[1,2]*h*kp20(h)),h):
[ > k20:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp10(h)+a[2,2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand(b[1]*k100+b[2]*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand(b[1]*k110+b[2]*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
x
pand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
x
pand(k220):
[ > F12:=expand(b[1]*k120+b[2]*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ ]
```

```

[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand(b[1]*k130+b[2]*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
  cond1a := b1 + b2 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
  cond2a := b1 c1 + b2 c2 =  $\frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0),
  f(x0,y0)));
  cond2b := b1 a1,1 + b1 a1,2 + b2 a2,1 + b2 a2,2 =  $\frac{1}{2}$ 
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
  cond3a := b1 c12 + b2 c22 =  $\frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
  y0)),f(x0,y0));
  cond3b := 2 b1 c1 a1,1 + 2 b1 c1 a1,2 + 2 b2 c2 a2,1 + 2 b2 c2 a2,2 =  $\frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
  0,y0)),f(x0,y0)^2);
  cond3c := b1 a1,12 + 2 b1 a1,1 a1,2 + b1 a1,22 + b2 a2,12 + 2 b2 a2,1 a2,2 + b2 a2,22 =  $\frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
  0,y0)),D[1](f)(x0,y0));
  cond3d := 2 b1 c1 a1,1 + 2 b1 a1,2 c2 + 2 b2 a2,1 c1 + 2 b2 c2 a2,2 =  $\frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
  )^2,f(x0,y0));
  cond3e :=
  2 b1 a1,12 + 2 b1 a1,1 a1,2 + 2 b1 a1,2 a2,1 + 2 b1 a1,2 a2,2 + 2 b2 a2,1 a1,1 + 2 b2 a1,2 a2,1 + 2 b2 a2,1 a2,2 + 2 b2 a2,22 =  $\frac{1}{3}$ 
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
  cond4a := b2 c23 + b1 c13 =  $\frac{1}{4}$ 
[ > cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
  (x0,y0)),f(x0,y0));
  cond4b := 3 b2 c22 a2,2 + 3 b1 c12 a1,1 + 3 b1 c12 a1,2 + 3 b2 c22 a2,1 =  $\frac{3}{4}$ 
[ > cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](f)
  (x0,y0)),f(x0,y0)^2);
  cond4c := 3 b2 c2 a2,22 + 6 b1 c1 a1,1 a1,2 + 3 b1 c1 a1,12 + 3 b1 c1 a1,22 + 3 b2 c2 a2,12 + 6 b2 c2 a2,1 a2,2 =  $\frac{3}{4}$ 
[ > cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](f)
  (x0,y0)),D[1](f)(x0,y0));
  cond4d := 6 b1 c12 a1,1 + 6 b1 c1 a1,2 c2 + 6 b2 c22 a2,2 + 6 b2 c2 a2,1 c1 =  $\frac{3}{4}$ 
[ > cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(
  coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));
  cond4e := 6 b2 c1 a1,2 a2,1 + 6 b1 a1,2 c2 a2,2 + 6 b1 a1,2 c2 a2,1 + 6 b1 c1 a1,2 a2,2 + 6 b1 c1 a1,2 a2,1 + 12 b1 c1 a1,1 a1,2
  + 6 b2 a2,1 c1 a1,1 + 12 b2 c2 a2,1 a2,2 + 6 b2 c2 a2,1 a1,1 + 6 b2 a1,2 c2 a2,1 + 12 b1 c1 a1,12 + 12 b2 c2 a2,22 =  $\frac{5}{4}$ 

```

> **cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);**

$$\text{cond4f} := 3 b_2 a_{2,1} a_{2,2}^2 + b_1 a_{1,2}^3 + b_1 a_{1,1}^3 + b_2 a_{2,1}^3 + b_2 a_{2,2}^3 + 3 b_2 a_{2,1}^2 a_{2,2} + 3 b_1 a_{1,1}^2 a_{1,2} + 3 b_1 a_{1,1} a_{1,2}^2 = \frac{1}{4}$$

> **cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));**

cond4g :=

$$6 b_2 c_2 a_{2,1} a_{2,2} + 6 b_2 c_2 a_{2,2}^2 + 6 b_2 a_{2,1}^2 c_1 + 6 b_1 c_1 a_{1,1}^2 + 6 b_1 c_1 a_{1,1} a_{1,2} + 6 b_1 a_{1,2} c_2 a_{1,1} + 6 b_1 a_{1,2}^2 c_2 + 6 b_2 a_{2,1} c_1 a_{2,2} = \frac{3}{4}$$

> **cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);**

$$\begin{aligned} \text{cond4h} := & 18 b_1 a_{1,1}^2 a_{1,2} + 9 b_1 a_{1,1} a_{1,2}^2 + 6 b_1 a_{1,2}^2 a_{2,1} + 6 b_1 a_{1,2}^2 a_{2,2} + 6 b_1 a_{1,2} a_{2,1} a_{1,1} + 6 b_1 a_{1,2} a_{2,2} a_{1,1} \\ & + 3 b_1 a_{1,2} a_{2,1}^2 + 6 b_2 a_{1,2} a_{2,1} a_{2,2} + 9 b_2 a_{2,2}^3 + 9 b_1 a_{1,1}^3 + 3 b_1 a_{1,2} a_{2,2}^2 + 18 b_2 a_{2,1} a_{2,2}^2 + 3 b_2 a_{2,1} a_{1,1}^2 + 6 b_2 a_{2,1}^2 a_{1,1} \\ & + 6 b_2 a_{2,1} a_{1,1} a_{2,2} + 9 b_2 a_{2,1}^2 a_{2,2} + 6 b_2 a_{1,2} a_{2,1} a_{1,1} + 3 b_2 a_{1,2}^2 a_{2,1} + 6 b_2 a_{1,2} a_{2,1}^2 + 6 b_1 a_{1,2} a_{2,1} a_{2,2} = 1 \end{aligned}$$

> **cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));**

$$\text{cond4i} := 3 b_1 a_{1,2} c_2^2 + 3 b_1 c_1^2 a_{1,1} + 3 b_2 a_{2,1} c_1^2 + 3 b_2 c_2^2 a_{2,2} = \frac{1}{4}$$

> **cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);**

$$\begin{aligned} \text{cond4j} := & 6 b_2 a_{1,2} c_2 a_{2,1} + 6 b_1 c_1 a_{1,2} a_{2,1} + 6 b_2 a_{2,1} c_1 a_{1,1} + 6 b_2 a_{2,1} c_1 a_{2,2} + 6 b_1 c_1 a_{1,1}^2 + 6 b_1 a_{1,2} c_2 a_{1,1} \\ & + 6 b_1 a_{1,2} c_2 a_{2,2} + 6 b_2 c_2 a_{2,2}^2 = \frac{1}{4} \end{aligned}$$

> **cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));**

$$\begin{aligned} \text{cond4k} := & 6 b_1 a_{1,1}^3 + 6 b_2 a_{2,2}^3 + 6 b_2 a_{1,2} a_{2,1} a_{1,1} + 12 b_2 a_{1,2} a_{2,1} a_{2,2} + 6 b_2 a_{2,1} a_{1,1}^2 + 6 b_2 a_{1,2} a_{2,1}^2 + 6 b_2 a_{2,1} a_{2,2}^2 \\ & + 6 b_2 a_{2,1} a_{1,1} a_{2,2} + 6 b_1 a_{1,1}^2 a_{1,2} + 6 b_1 a_{1,2}^2 a_{2,1} + 6 b_1 a_{1,2} a_{2,2}^2 + 12 b_1 a_{1,2} a_{2,1} a_{1,1} + 6 b_1 a_{1,2} a_{2,2} a_{1,1} \\ & + 6 b_1 a_{1,2} a_{2,1} a_{2,2} = \frac{1}{4} \end{aligned}$$

soluciones para el método

> **metodo:=allvalues(solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e,cond4a,cond4b,cond4c,cond4d,cond4f,c[1]=a[1,1]+a[1,2],c[2]=a[2,1]+a[2,2]}))[2];**

$$\text{metodo} := \left\{ b_1 = \frac{1}{2}, b_2 = \frac{1}{2}, a_{1,1} = \frac{1}{4}, a_{2,2} = \frac{1}{4}, c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}, c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}, a_{1,2} = \frac{1}{4} - \frac{\sqrt{3}}{6}, a_{2,1} = \frac{1}{4} + \frac{\sqrt{3}}{6} \right\}$$

> **c[1]:=subs(metodo,c[1]);c[2]:=subs(metodo,c[2]);a[1,1]:=subs(metodo,a[1,1]);a[1,2]:=subs(metodo,a[1,2]);a[2,1]:=subs(metodo,a[2,1]);a[2,2]:=subs(metodo,a[2,2]);b[1]:=subs(metodo,b[1]);b[2]:=subs(metodo,b[2]);**

$$c_1 := \frac{1}{2} - \frac{\sqrt{3}}{6}$$

$$c_2 := \frac{1}{2} + \frac{\sqrt{3}}{6}$$

$$a_{1,1} := \frac{1}{4}$$

$$a_{1,2} := \frac{1}{4} - \frac{\sqrt{3}}{6}$$

$$a_{2,1} := \frac{1}{4} + \frac{\sqrt{3}}{6}$$

$$a_{2,2} := \frac{1}{4}$$

$$b_1 := \frac{1}{2}$$

$$b_2 := \frac{1}{2}$$

Significa esto que el método de GAUSS de 2 etapas es

$$\begin{array}{c|cc} \frac{3-\sqrt{3}}{6} & \frac{1}{4} & \frac{3-2\sqrt{3}}{12} \\ \frac{3+\sqrt{3}}{6} & \frac{3+2\sqrt{3}}{12} & \frac{1}{4} \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{2} & \frac{1}{2} \end{array}$$

función de estabilidad

> f := (x, y) -> lambda * y;

$$f := (x, y) \rightarrow \lambda y$$

> ka1 := unapply(f(x0+c[1]*h, y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)), h);

$$ka1 := h \rightarrow \lambda \left(y0 + \frac{1}{4} h kp1(h) + \left(\frac{1}{4} - \frac{\sqrt{3}}{6} \right) h kp2(h) \right)$$

> ka2 := unapply(f(x0+c[2]*h, y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)), h);

$$ka2 := h \rightarrow \lambda \left(y0 + \left(\frac{1}{4} + \frac{\sqrt{3}}{6} \right) h kp1(h) + \frac{1}{4} h kp2(h) \right)$$

> test := solve({k1=subs({kp1(h)=k1, kp2(h)=k2}, ka1(h)), k2=subs({kp1(h)=k1, kp2(h)=k2}, ka2(h))}, {k1, k2});

$$test := \{k2 = \frac{2\sqrt{3}\lambda y0(\lambda h + 2\sqrt{3})}{-6\lambda h + 12 + \lambda^2 h^2}, k1 = -\frac{2\sqrt{3}\lambda y0(-2\sqrt{3} + \lambda h)}{-6\lambda h + 12 + \lambda^2 h^2}\}$$

> k1 := subs(test, k1); k2 := subs(test, k2);

$$k1 := -\frac{2\sqrt{3}\lambda y0(-2\sqrt{3} + \lambda h)}{-6\lambda h + 12 + \lambda^2 h^2}$$

$$k2 := \frac{2\sqrt{3}\lambda y0(\lambda h + 2\sqrt{3})}{-6\lambda h + 12 + \lambda^2 h^2}$$

> y1 := expand(y0+h*(b[1]*k1+b[2]*k2));

$$y1 := y0 + \frac{12 h \lambda y0}{-6 \lambda h + 12 + \lambda^2 h^2}$$

> y1 := collect(subs(lambda=z/h, y1), y0);

$$y1 := \left(1 + \frac{12 z}{-6 z + 12 + z^2} \right) y0$$

> r := unapply(collect(simplify(y1/y0), z), z);

$$r := z \rightarrow \frac{6 z + 12 + z^2}{-6 z + 12 + z^2}$$

que es la función de estabilidad del método de GAUSS de 2 etapas.

Como las raíces del numerador de r son $-3 + i\sqrt{3}$ y las del

denominador son $3 + i\sqrt{3}$, tenemos que

$$\frac{(z+3-i\sqrt{3})(z+3+i\sqrt{3})}{(z-3-i\sqrt{3})(z-3+i\sqrt{3})} = \frac{(z+3)^2+3}{(z-3)^2+3}$$

Para z real, $(z + 3)^2$ es el cuadrado de la distancia de z a -3 y lo análogo con $(z - 3)^2$ y 3

Los números reales negativos están en el dominio D y los positivos en su complementario.

Pero no hay en el caso complejo la misma interpretación sobre la distancia mientras

no intervenga el módulo, así que todavía no podemos concluir si el método es o no A-estable.

Vamos a construir la frontera de D de ecuación $|r(z)|^2 = 1$ en el plano complejo, o lo que

es lo mismo

$$|(z + 3)^2 + 3|^2 = |(z - 3)^2 + 3|^2$$

poniendo $z=x+iy$, lo anterior se escribe

$$((x + 3)^2 - y^2 + 3)^2 + 4 y^2 (x + 3)^2 = ((x - 3)^2 - y^2 + 3)^2 + 4 y^2 (x - 3)^2$$

o simplificando

$$24 x^3 + 288 x + 24 x y^2 = 0$$

$$x (24 x^2 + 288 + 24 y^2) = 0$$

y la curva frontera del dominio se compone sólo de los puntos con $x = 0$, o sea, los $z = iy$,

ya que la expresión $24 x^2 + 288 + 24 y^2$ no puede anularse.

Queda por determinar cuál de los dos semiplanos

$$\operatorname{Re}(z) < 0 \text{ o } \operatorname{Re}(z) > 0$$

separados por la curva que hemos calculado, es el dominio de estabilidad. Pues bien, como

$$r(-3) = 3/39,$$

-3 está en D , y D coincide exactamente con el semiplano $\operatorname{Re}(z) < 0$, por lo que el

Método de Gauss de 2 etapas es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-14 (del EXAMEN EXTRAORDINARIO de 01JUL05)

Estúdiense si el método implícito de RUNGE-KUTTA del tablero

$$\frac{1}{4} \quad | \quad \frac{7}{24} \quad - \frac{1}{24}$$

$$\frac{3}{4} \quad | \quad \frac{13}{24} \quad \frac{5}{24}$$

$$| \quad \frac{1}{2} \quad \frac{1}{2}$$

es o no A-estable.

```

[ > restart;
[ > interface(labeling=false):
[ función de estabilidad
[ > f:=(x,y)->lambda*y;
[                               f:=(x,y) -> lambda*y
[ > ka1:=unapply(f(x0+(1/4)*h,y0+(7/24)*h*kp1(h)+(-1/24)*h*kp2(h)),h);
[                               ka1 := h -> lambda * (y0 + (7/24)*h*kp1(h) - (1/24)*h*kp2(h))
[ > ka2:=unapply(f(x0+(3/4)*h,y0+(13/24)*h*kp1(h)+(5/24)*h*kp2(h)),h);
[                               ka2 := h -> lambda * (y0 + (13/24)*h*kp1(h) + (5/24)*h*kp2(h))
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                               test := {k1 = -3 * (lambda*y0*(lambda*h-4) / (lambda^2*h^2+12-6*lambda*h), k2 = 3 * (lambda*y0*(4+lambda*h) / (lambda^2*h^2+12-6*lambda*h)}
[ > k1:=subs(test,k1);k2:=subs(test,k2);
[                               k1 := -3 * (lambda*y0*(lambda*h-4) / (lambda^2*h^2+12-6*lambda*h)
[                               k2 := 3 * (lambda*y0*(4+lambda*h) / (lambda^2*h^2+12-6*lambda*h)
[ > y1:=expand(y0+h*((1/2)*k1+(1/2)*k2));
[                               y1 := y0 + (12*h*lambda*y0 / (lambda^2*h^2+12-6*lambda*h)
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                               y1 := (1 + (12*z / (z^2+12-6*z))) * y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[                               r := z -> (z^2+12+6*z / (z^2+12-6*z)

```

es la función de estabilidad del método

A-estabilidad del método

```

> solve(z^2+6*z+12);
                                     -3+√3 I, -3-√3 I
> solve(z^2-6*z+12);
                                     3+√3 I, 3-√3 I
> r:=unapply(((z+3-I*sqrt(3))*(z+3+I*sqrt(3)))/((z-3-I*sqrt(3))*(z-3+I*sqrt(3))),z);
                                     r:=z → (z+3-√3 I)(z+3+√3 I)
                                               (z-3-√3 I)(z-3+√3 I)
> r:=unapply(((z+3)^2+3)/((z-3)^2+3),z);
                                     r:=z → (z+3)^2+3
                                               (z-3)^2+3

```

Para z real, $(z + 3)^2$ es el cuadrado de la distancia de z a -3 y lo análogo con $(z - 3)^2$ y 3

Los números reales negativos están en el dominio D y los positivos en su complementario.

Pero no hay en el caso complejo la misma interpretación sobre la distancia mientras no intervenga el módulo, así que todavía no podemos concluir si el método es o no A-estable.

Vamos a construir la frontera de D de ecuación $|r(z)|^2 = 1$ en el plano complejo, o lo que es lo mismo

$$|(z + 3)^2 + 3|^2 = |(z - 3)^2 + 3|^2$$

poniendo $z=x+i y$, lo anterior se escribe

$$((x + 3)^2 - y^2 + 3)^2 + 4 y^2 (x + 3)^2 = ((x - 3)^2 - y^2 + 3)^2 + 4 y^2 (x - 3)^2$$

o simplificando

$$24 x^3 + 288 x + 24 x y^2 = 0$$

$$x (24 x^2 + 288 + 24 y^2) = 0$$

y la curva frontera del dominio se compone sólo de los puntos con $x = 0$, o sea, los $z = i y$,

ya que la expresión $24 x^2 + 288 + 24 y^2$ no puede anularse.

Queda por determinar cuál de los dos semiplanos

$$\operatorname{Re}(z) < 0 \text{ o } \operatorname{Re}(z) > 0$$

separados por la curva que hemos calculado, es el dominio de estabilidad. Pues bien, como

$$r(-3) = 3/39,$$

-3 está en D , y D coincide exactamente con el semiplano $\operatorname{Re}(z) < 0$, por lo que el método del enunciado es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-15

Estúdiese si el método diagonalmente implícito de RUNGE-KUTTA del tablero

$$\begin{array}{c|cc}
 \alpha & \alpha & 0 \\
 1 & 1-\alpha & \alpha \\
 \hline
 & \frac{1}{2(1-\alpha)} & \frac{1-2\alpha}{2(1-\alpha)}
 \end{array}$$

es o no A-estable. (Está hecho para $\alpha = 1$)

```

> restart;
> interface(labeling=false):
función de estabilidad
> f:=(x,y)->lambda*y;alpha:=1;
                                f := (x, y) -> lambda y
                                alpha := 1
> ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
                                ka1 := h -> lambda (y0 + h kp1(h))
> ka2:=unapply(f(x0+2*h,y0+(1-alpha)*h*kp1(h)+alpha*h*kp2(h)),h);
                                ka2 := h -> lambda (y0 + h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                test := { k2 = -frac(lambda y0, -1 + lambda h), k1 = -frac(lambda y0, -1 + lambda h) }
> k1:=subs(test,k1);k2:=subs(test,k2);
                                k1 := -frac(lambda y0, -1 + lambda h)
                                k2 := -frac(lambda y0, -1 + lambda h)
> y1:=expand(y0+h*((1/2)*k1+(1/2)*k2));
                                y1 := y0 -frac(h lambda y0, -1 + lambda h)
> y1:=collect(subs(lambda=z/h,y1),y0);
                                y1 := (1 -frac(z, -1 + z)) y0
> r:=unapply(collect(simplify(y1/y0),z),z);
                                r := z -> -frac(1, -1 + z)

```

es la función de estabilidad del método

A-estabilidad del método

El dominio de estabilidad está compuesto por los $z = \lambda h$ tales que el módulo de $z-1$ sea superior a 1, o sea, por los z tales que su distancia al punto 1 sea superior a 1. Eso es el exterior de la circunferencia de centro 1 y de radio 1, lo que significa que contiene al semiplano de parte

└ real negativa; por lo tanto, el método es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-16 (del EXAMEN ORDINARIO de 09FEB04)

Para los métodos de RUNGE-KUTTA implícitos del siguiente tablero

$$\begin{array}{c|cc} \frac{1}{3} & & \frac{1}{3} \\ \hline 1 & \alpha & 0 \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

- dígase qué valores de α proporcionan métodos de orden 2 ;
- dígase si existen valores de α que proporcionen métodos de orden 3 ;
- dígase si existen valores de α que proporcionen métodos de orden 4 .

(lo que antecede es Ejercicio del capítulo 7)

- Calcúlese la función de estabilidad para los métodos de la familia.
- Discútase sobre la A-estabilidad de los métodos de la familia.
- Calcúlese la función de estabilidad para los métodos obtenidos en b) y c).

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+(1/3)*h,y0+(1/3)*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+h,y0+alpha*h*kp1_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

[

```

> Fi_0:=expand((3/4)*k1_0+(1/4)*k2_0);
                                     Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                     cond1a := 1 = 1

```

El orden es siempre de 1 al menos

condiciones para el orden 2

```

> Fi10:=expand((3/4)*k110+(1/4)*k210);
                                     Fi10 := 1/2 D1(f)(x0, y0) + 1/4 D2(f)(x0, y0) f(x0, y0) + 1/4 D2(f)(x0, y0) alpha f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a := 1/2 = 1/2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b := 1/2 = 1/4 + alpha/4
> solucion:=solve({cond2a,cond2b});
                                     solucion := {alpha = 1}

```

El valor $\alpha = 1$ es el único que proporciona orden 2

b) condiciones para el orden 3

```

> Fi20:=expand((3/4)*k120+(1/4)*k220);
                                     Fi20 := 1/3 D1,1(f)(x0, y0) + 1/6 D1,2(f)(x0, y0) f(x0, y0) + 1/12 D2,2(f)(x0, y0) f(x0, y0)^2 + 1/6 D2(f)(x0, y0) D1(f)(x0, y0)
                                     + 1/6 D2(f)(x0, y0)^2 f(x0, y0) + 1/2 D1,2(f)(x0, y0) alpha f(x0, y0) + 1/4 alpha^2 f(x0, y0)^2 D2,2(f)(x0, y0)
                                     + 1/6 D2(f)(x0, y0) alpha D1(f)(x0, y0) + 1/6 D2(f)(x0, y0)^2 alpha f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0)):cond3a:=subs(solucion,cond3a);
                                     cond3a := 1/6 = 1/6
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0)):cond3b:=subs(solucion,cond3b);
                                     cond3b := 1/3 = 1/3
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2):cond3c:=subs(solucion,cond3c);
                                     cond3c := 1/6 = 1/6
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0)):cond3d:=subs(solucion,cond3d);
                                     cond3d := 1/6 = 1/6
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0)):cond3e:=subs(solucion,cond3e);
                                     cond3e := 1/6 = 1/6

```

Con el mismo valor $\alpha = 1$ se obtiene orden 3

c) orden 4

En ningún caso aparece en la familia el método de GAUSS de dos etapas.

Por lo tanto, no se alcanza nunca el orden 4

d)

función de estabilidad

```

> f:=(x,y)->lambda*y;

```

```

[                                     f := (x, y) → λ y
[ > k1:=unapply(f(x0+(1/3)*h,y0+(1/3)*h*kp1(h)),h);
[                                     ka1 := h → λ (y0 + 1/3 h kp1(h))
[ > test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
[                                     test := {k1 = - 3 λ y0 / (-3 + λ h)}
[ > k1:=subs(test,k1);
[                                     k1 := - 3 λ y0 / (-3 + λ h)
[ >
[ > k2:=f(x0+h,y0+alpha*h*k1);
[                                     k2 := λ (y0 - 3 α h λ y0 / (-3 + λ h))
[ >
[ >
[ > y1:=expand(y0+h*((3/4)*k1+(1/4)*k2));
[                                     y1 := y0 - 9 h λ y0 / (4(-3 + λ h)) + h λ y0 / 4 - 3 α h² λ² y0 / (4(-3 + λ h))
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 := (1 - 9 z / (4(-3 + z)) + z / 4 - 3 α z² / (4(-3 + z))) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[                                     r := z → - 1/4 (12 + (-1 + 3 α) z² + 8 z) / (-3 + z)

```

que es la función de estabilidad de la familia general propuesta

e)

función de estabilidad cuando $\alpha = \frac{1}{3}$ (única posible A-estabilidad)

```

[ > r_estab:=simplify(subs({alpha=1/3},r(z)));

```

$$r_{estab} := -\frac{3+2z}{-3+z}$$

expresión con límite 2 cuando z tiende a ∞ , por lo que no puede haber A-estabilidad

f)

función de estabilidad cuando $\alpha = 1$ (orden máximo 3)

```

[ > r_orden3:=simplify(subs(solucion,r(z)));

```

$$r_{orden3} := -\frac{6+z^2+4z}{2(-3+z)}$$

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-17 (del EXAMEN EXTRAORDINARIO de 14JUL04)

Para los métodos de RUNGE-KUTTA implícitos del siguiente tablero

$$\begin{array}{c|ccc} 1 & 1 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 1 & 0 \\ 0 & a & -3 & 1 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

y teniendo en cuenta que para tales métodos de 3 etapas se puede obtener hasta orden 6, dígame cuál es el orden máximo que se obtiene realmente y cuáles son los valores del parámetro a para los que se alcanza dicho orden.

(lo que antecede es Ejercicio del capítulo 7)

Calcúlese la función de estabilidad para los métodos de la familia y para los métodos de orden máximo.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+h,y0+h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+(1/2)*h,y0-(1/2)*h*kp1_(h)+h*kp2_(h)),h):
[ > k3_:=unapply(f(x0,y0+a*h*kp1_(h)-3*h*kp2_(h)+h*kp3_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k3_0:=k3_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k31:=D(k3_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k21(0)):
[ > k310:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0,kp3_(0)=k3_0},k31(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[
```


[Nunca se obtiene el orden 3

[función de estabilidad

[> restart;

[> f:=(x,y)->lambda*y;

$$f := (x, y) \rightarrow \lambda y$$

[> ka1:=unapply(f(x0+h,y0+h*kp1(h)),h);

$$ka1 := h \rightarrow \lambda (y0 + h kp1(h))$$

[> test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);

$$test := \{k1 = -\frac{\lambda y0}{-1 + \lambda h}\}$$

[> k1:=subs(test,k1);

$$k1 := -\frac{\lambda y0}{-1 + \lambda h}$$

[> ka2:=simplify(unapply(f(x0+(1/2)*h,y0-(1/2)*h*k1+h*kp2(h)),h));

$$ka2 := h \rightarrow \lambda \left(y0 + \frac{1}{2} \frac{h \lambda y0}{-1 + \lambda h} + h kp2(h) \right)$$

[> test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);

$$test := \{k2 = -\frac{\lambda y0(-2 + 3 \lambda h)}{2(-1 + \lambda h)^2}\}$$

[> k2:=subs(test,k2);

$$k2 := -\frac{\lambda y0(-2 + 3 \lambda h)}{2(-1 + \lambda h)^2}$$

[> ka3:=simplify(unapply(f(x0,y0+a*h*k1-3*h*k2+h*kp3(h)),h));

$$ka3 := h \rightarrow \lambda \left(y0 - \frac{a h \lambda y0}{-1 + \lambda h} + \frac{3}{2} \frac{h \lambda y0(-2 + 3 \lambda h)}{(-1 + \lambda h)^2} + h kp3(h) \right)$$

[> test:=solve({k3=subs(kp3(h)=k3,ka3(h))},k3);

$$test := \{k3 = \frac{\lambda y0(-2 + 10 \lambda h - 11 \lambda^2 h^2 - 2 a h \lambda + 2 a h^2 \lambda^2)}{2(-1 + \lambda h)^3}\}$$

[> k3:=subs(test,k3);

$$k3 := \frac{\lambda y0(-2 + 10 \lambda h - 11 \lambda^2 h^2 - 2 a h \lambda + 2 a h^2 \lambda^2)}{2(-1 + \lambda h)^3}$$

[> y1:=expand(y0+h*((1/6)*k1+(2/3)*k2+(1/6)*k3));

$$y1 := y0 - \frac{h \lambda y0}{6(-1 + \lambda h)} + \frac{2 h \lambda y0}{3(-1 + \lambda h)^2} - \frac{\lambda^2 y0 h^2}{(-1 + \lambda h)^2} - \frac{h \lambda y0}{6(-1 + \lambda h)^3} + \frac{5 \lambda^2 y0 h^2}{6(-1 + \lambda h)^3} - \frac{11 \lambda^3 y0 h^3}{12(-1 + \lambda h)^3} - \frac{\lambda^2 y0 a h^2}{6(-1 + \lambda h)^3} + \frac{\lambda^3 y0 a h^3}{6(-1 + \lambda h)^3}$$

[> y1:=collect(subs(lambda=z/h,y1),y0);

y1 :=

$$\left(1 - \frac{z}{6(-1+z)} + \frac{2z}{3(-1+z)^2} - \frac{z^2}{(-1+z)^2} - \frac{z}{6(-1+z)^3} + \frac{5z^2}{6(-1+z)^3} - \frac{11z^3}{12(-1+z)^3} - \frac{z^2 a}{6(-1+z)^3} + \frac{z^3 a}{6(-1+z)^3} \right) y0$$

[> r:=unapply(collect(simplify(y1/y0),z),z);

$$r := z \rightarrow \frac{1}{12} \frac{-12 + (-13 + 2a)z^3 + (-2 - 2a)z^2 + 24z}{(-1+z)^3}$$

[que es la función de estabilidad de la familia general propuesta

[función de estabilidad cuando a = 2 (orden 2)

[> r_2:=simplify(subs({a=2},r(z)));

$$r_2 := -\frac{4 + 3z^3 + 2z^2 - 8z}{4(-1+z)^3}$$

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-18 (del EXAMEN ORDINARIO de 18FEB05)

De entre los métodos implícitos de 2 etapas de tablero

$$0 \mid \alpha \quad 0$$

$$1 \mid \alpha \quad 0$$

$$\mid \frac{1}{2} \quad \frac{1}{2}$$

entre los que se encuentra, para $\alpha = 1/2$, el llamado LOBATTO IIIb, se busquen los que sean de mayor orden.

(desde aquí es Ejercicio del capítulo 8)

Calcúlese la función de estabilidad de la familia general de los métodos

del tablero y discútase la A-estabilidad del método LOBATTO IIIb y de los otros

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > c[1]:=0:c[2]:=1:
[ > a[1,1]:=alpha:a[1,2]:=0:a[2,1]:=alpha:a[2,2]:=0:
[ > b[1]:=1/2:b[2]:=1/2:
[ > k10:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp10(h)+a[1,2]*h*kp20(h)),h):
[ > k20:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp10(h)+a[2,2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > Fi0:=expand(b[1]*k100+b[2]*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > Fi1:=expand(b[1]*k110+b[2]*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
[ xpand(k120):
[ ]
```

```

[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
xpend(k220):
[ > Fi2:=expand(b[1]*k120+b[2]*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand(b[1]*k130+b[2]*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
cond2a :=  $\frac{1}{2} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
,f(x0,y0)));
cond2b :=  $\alpha = \frac{1}{2}$ 
[ orden 2 para  $\alpha = 1/2$ , o sea, LOBATTO IIIb
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));subs(alpha=1/2,co
nd3a);
cond3a :=  $\frac{1}{2} = \frac{1}{3}$ 
 $\frac{1}{2} = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
y0)),f(x0,y0));subs(alpha=1/2,cond3b);
cond3b :=  $\alpha = \frac{2}{3}$ 
 $\frac{1}{2} = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
0,y0)),f(x0,y0)^2);subs(alpha=1/2,cond3c);
cond3c :=  $\alpha^2 = \frac{1}{3}$ 
 $\frac{1}{4} = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
0,y0)),D[1](f)(x0,y0));subs(alpha=1/2,cond3d);
cond3d :=  $0 = \frac{1}{3}$ 
 $0 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
^2),f(x0,y0));subs(alpha=1/2,cond3e);
cond3e :=  $2\alpha^2 = \frac{1}{3}$ 
 $\frac{1}{2} = \frac{1}{3}$ 
[ no hay orden 3 para ningun valor de  $\alpha$ 
[ # funcion de estabilidad
[ > f:=(x,y)->lambda*y;
f := (x, y) →  $\lambda y$ 

```

```

> ka1:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)),h);
      ka1 := h → λ (y0 + α h kp1(h))
> ka2:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)),h);
      ka2 := h → λ (y0 + α h kp1(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -λ y0 / (-1 + λ α h), k2 = -λ y0 / (-1 + λ α h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -λ y0 / (-1 + λ α h)
      k2 := -λ y0 / (-1 + λ α h)
> y1:=expand(y0+h*(b[1]*k1+b[2]*k2));
      y1 := y0 - h λ y0 / (-1 + λ α h)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-1 + z α)) y0
> r:=unapply(collect(simplify(y1/y0),z),z);
      r := z → (-1 + (α - 1) z) / (-1 + z α)
# A-estabilidad para el método LOBATTO IIIb
> simplify(subs(alpha=1/2,r(z)));
      - (2 + z) / (-2 + z)

```

El dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método LOBATTO IIIb es A-estable

A-estabilidad para los otros métodos

En general, como hemos visto

$$r(z) = \frac{1 + (1 - \alpha)z}{1 - \alpha z}$$

Supondremos $\alpha \neq \frac{1}{2}$ y pondremos que $z=x+iy$.

Como

$$1 + (1 - \alpha)z = 1 + (1 - \alpha)x + i(1 - \alpha)y$$

y

$$1 - \alpha z = 1 - \alpha x + i\alpha y,$$

resulta

$$|1 + (1 - \alpha)z|^2 = (1 + (1 - \alpha)x)^2 + (1 - \alpha)^2 y^2$$

y

$$|1 - \alpha z|^2 = (1 - \alpha x)^2 + \alpha^2 y^2$$

por lo que la curva $|r(z)|^2 = 1$ se escribe

$$(1 + (1 - \alpha)x)^2 + (1 - \alpha)^2 y^2 = (1 - \alpha x)^2 + \alpha^2 y^2$$

o sea

$$(1 - 2\alpha)y^2 + (1 - 2\alpha)x^2 + 2x = 0$$

$$x^2 + \frac{2}{1 - 2\alpha}x + y^2 = 0$$

La curva límite de la region de estabilidad pasa por (0,0)

Supongamos primeramente que $\alpha < \frac{1}{2}$ o sea $1 - 2\alpha > 0$

La curva es

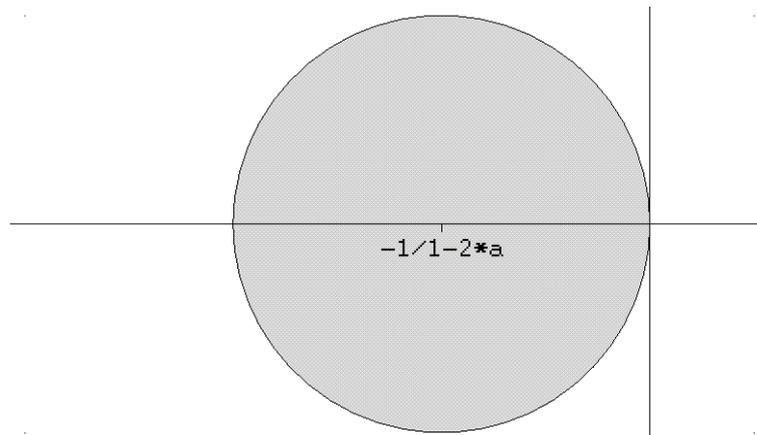
$$\left(x + \frac{1}{1 - 2\alpha}\right)^2 + y^2 = \frac{1}{(1 - 2\alpha)^2}$$

o sea, una circunferencia de centro $-\frac{1}{1 - 2\alpha}$ y radio $\frac{1}{1 - 2\alpha}$ que pasa por (0,0)

Como $z=1=(1,0)$ verifica $|r(z)| = \left|\frac{2 - \alpha}{1 - \alpha}\right| > 1$ resulta que el dominio de estabilidad

es el interior de la circunferencia

```
> with(plottools):with(plots):macro(gris=COLOR(RGB,.8,.8,.8)):exten:=point([-3,-1],symbol=point),point([-3,1],symbol=point),point([0.5,1],symbol=point),point([0.5,-1],symbol=point):noAest:=disk([-1,0],1,color=gris):display(exten,noAest,axes=normal,scaling=constrained,tickmarks=[[ -1="-1/1-2*a"],[0,0]],axesfont=[COURIER,10]);>
```



y el método no es A-estable.

Supongamos ahora que $\frac{1}{2} < \alpha$ o sea $1 - 2\alpha < 0$ y $2\alpha - 1 > 0$

La curva es

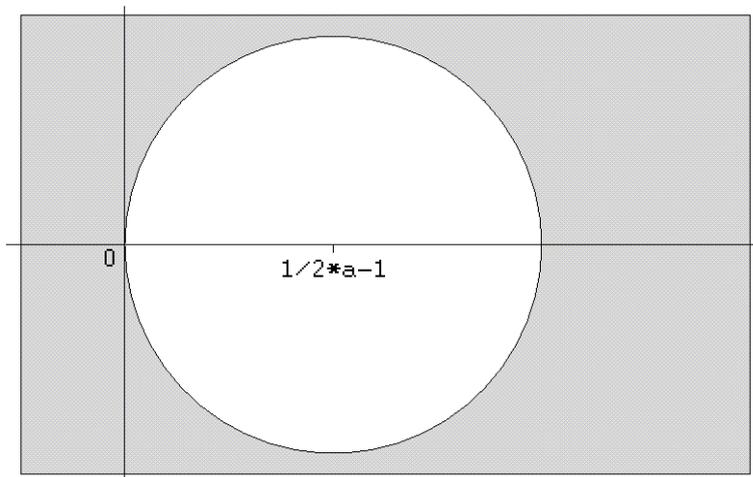
$$\left(x - \frac{1}{2\alpha - 1}\right)^2 + y^2 = \frac{1}{(2\alpha - 1)^2}$$

o sea, una circunferencia de centro $\frac{1}{2\alpha - 1}$ y radio $\frac{1}{2\alpha - 1}$ que pasa por $(0,0)$

Como $z = -1 = (-1,0)$ verifica $|r(z)| = \left|\frac{\alpha}{1 + \alpha}\right| < 1$ resulta que el dominio de estabilidad

es el exterior de la circunferencia

```
> exten:=point([-0.5,-1.1],symbol=point),point([-0.5,1.1],symbol=point),point([3,1.1],symbol=point),point([3,-1.1],symbol=point):plano:= rectangle([-0.5,-1.1],[3,1.1],color=gris):noAest:= disk([1,0],1,color="White"):display(exten,noAest,plano,axes=normal,scaling=constrained,tickmarks=[[1="1/2*a-1"],[0,0]],axesfont=[COURIER,10]);
>
```



[y los métodos son ahora A-estables.

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-19 (del EXAMEN ESP DE JUL05)

Dígase si el método de RUNGE-KUTTA dado por el tablero

$$0 \quad | \quad 2 \quad 1$$

$$1 \quad | \quad 0 \quad 1$$

$$\quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

es o no A-estable

```
> restart;
> interface(labeling=false):
# funcion de estabilidad
> f:=(x,y)->lambda*y;
                                     f:=(x,y) -> lambda*y
> c[1]:=0:c[2]:=1:a[1,1]:=2:a[1,2]:=1:a[2,1]:=0:a[2,2]:=1:b[1]:=1/2:b[2]:=1/2:
> ka1:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)),h);
                                     ka1:=h -> lambda*(y0+2*h*kp1(h)+h*kp2(h))
> ka2:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)),h);
                                     ka2:=h -> lambda*(y0+h*kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                     test:={k1=frac(lambda*y0,1-3*lambda*h+2*lambda^2*h^2),k2=-frac(lambda*y0,-1+lambda*h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
                                     k1:=frac(lambda*y0,1-3*lambda*h+2*lambda^2*h^2)
                                     k2:=-frac(lambda*y0,-1+lambda*h)
> y1:=expand(y0+h*(b[1]*k1+b[2]*k2));
                                     y1:=y0+frac(1/2*h*lambda*y0,1-3*lambda*h+2*lambda^2*h^2)-frac(1/2*h*lambda*y0,-1+lambda*h)
> y1:=collect(subs(lambda=z/h,y1),y0);
                                     y1:=left(1+frac(1/2*z,1-3*z+2*z^2)-frac(1/2*z,-1+z)right)*y0
> r:=unapply(collect(simplify(y1/y0),z),z);
                                     r:=z -> frac(-1+z,2*z-1)
```

Supongamos que $z=x+i$ y con $x \leq 0$.

Como

$$|2z - 1|^2 = (2x - 1)^2 + 4y^2$$

y

$$|z - 1|^2 = (x - 1)^2 + y^2$$

entonces tanto $2x - 1$ como $x - 1$ son negativos y

$$2x - 1 \leq x - 1$$

o sea,

$$(2x - 1)^2 \geq (x - 1)^2$$

y

$$(2x - 1)^2 + 4y^2 \geq (x - 1)^2 + y^2$$

o sea

$$|r(z)| \leq 1$$

y el semiplano de parte real negativa está contenido en el

dominio de estabilidad. El método es A-estable.

De hecho, la frontera $|r(z)| = 1$ del dominio de estabilidad

es la curva de ecuación

$$(2x - 1)^2 + 4y^2 = (x - 1)^2 + y^2$$

que, tras operar, es la circunferencia

$$\left(x - \frac{1}{3}\right)^2 + y^2 = \frac{1}{9}$$

de centro $\frac{1}{3}$ y radio $\frac{1}{3}$, que pasa por $(0, 0)$, y el dominio de estabilidad

es el exterior de esa circunferencia.

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-20 (del EXAMEN ORDINARIO de 01FEB06)

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

0		0	
β		$1 - \alpha$	α

		b_1	b_2

b) Para los métodos de la familia, calcúlese la función de estabilidad. Hágase lo mismo para los métodos de orden máximo obtenidos.

c) Discútese si estos últimos métodos son o no A-estables.

```
[ > restart;
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0,y0),h):
[ > k2_:=unapply(f(x0+beta*h,y0+(1-alpha)*h*kp1_(h)+alpha*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(b[1]*k1_0+b[2]*k2_0);
[                                     Fi_0 := b1 f(x0, y0) + b2 f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                     cond1a := 1 = b1 + b2
[ > solve(cond1a);
[                                     {b2 = 1 - b1, b1 = b1}
```

[condiciones para el orden 2

[

```

> Fi10:=expand(b[1]*k110+b[2]*k210);
      
$$Fi10 := b_2 D_1(f)(x0, y0) \beta + b_2 D_2(f)(x0, y0) f(x0, y0)$$

> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
      
$$cond2a := \frac{1}{2} = b_2 \beta$$

> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
      
$$cond2b := \frac{1}{2} = b_2$$

> solucion1:=solve({cond1a,cond2a,cond2b});
      
$$solucion1 := \{ \beta = 1, b_1 = \frac{1}{2}, b_2 = \frac{1}{2} \}$$


```

condiciones para el orden 3

```

> Fi20:=expand(b[1]*k120+b[2]*k220);
      
$$Fi20 := b_2 D_{1,1}(f)(x0, y0) \beta^2 + 2 b_2 \beta D_{1,2}(f)(x0, y0) f(x0, y0) + b_2 D_{2,2}(f)(x0, y0) f(x0, y0)^2$$

      
$$+ 2 b_2 D_2(f)(x0, y0) \alpha D_1(f)(x0, y0) \beta + 2 b_2 D_2(f)(x0, y0)^2 \alpha f(x0, y0)$$

> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
      
$$cond3a := \frac{1}{6} = \frac{1}{2} b_2 \beta^2$$

> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
      
$$cond3b := \frac{1}{3} = b_2 \beta$$

> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
      
$$cond3c := \frac{1}{6} = \frac{1}{2} b_2$$

> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
      
$$cond3d := \frac{1}{6} = b_2 \alpha \beta$$

> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
      
$$cond3e := \frac{1}{6} = b_2 \alpha$$

> solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});

```

Es decir, no hay métodos de orden 3. Existe una familia infinita de métodos de orden 2, la que viene dada por el tablero

0		0	
1		1 - α	α

		$\frac{1}{2}$	$\frac{1}{2}$

uno de cuyos métodos es

```

> solucion2:=subs(alpha=1/2,solucion1) union {alpha=1/2};
      
$$solucion2 := \{ \beta = 1, b_1 = \frac{1}{2}, b_2 = \frac{1}{2}, \alpha = \frac{1}{2} \}$$


```

el conocido como LOBATTO IIIA de 2 etapas

0		0
----------	--	----------

$$1 \quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

$$| \quad \frac{1}{2} \quad \frac{1}{2}$$

b)

función de estabilidad

```
> f := (x, y) -> lambda * y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> ka1 := unapply(f(x0, y0), h);
```

$$ka1 := h \rightarrow \lambda y0$$

```
> ka2 := unapply(f(x0 + beta * h, y0 + (1 - alpha) * h * kp1(h) + alpha * h * kp2(h)), h);
```

$$ka2 := h \rightarrow \lambda (y0 + (1 - \alpha) h kp1(h) + \alpha h kp2(h))$$

```
> test := solve({k1 = subs({kp1(h) = k1, kp2(h) = k2}, ka1(h)), k2 = subs({kp1(h) = k1, kp2(h) = k2}, ka2(h))}, {k1, k2});
```

$$test := \{k1 = \lambda y0, k2 = \frac{\lambda y0 (-1 - \lambda h + \lambda \alpha h)}{-1 + \lambda \alpha h}\}$$

```
> k1 := subs(test, k1); k2 := subs(test, k2);
```

$$k1 := \lambda y0$$

$$k2 := \frac{\lambda y0 (-1 - \lambda h + \lambda \alpha h)}{-1 + \lambda \alpha h}$$

```
> y1 := expand(y0 + h * (b[1] * k1 + b[2] * k2));
```

$$y1 := y0 + h b_1 \lambda y0 - \frac{h b_2 \lambda y0}{-1 + \lambda \alpha h} - \frac{b_2 \lambda^2 y0 h^2}{-1 + \lambda \alpha h} + \frac{b_2 \lambda^2 y0 \alpha h^2}{-1 + \lambda \alpha h}$$

```
> y1 := collect(subs(lambda = z/h, y1), y0);
```

$$y1 := \left(1 + b_1 z - \frac{b_2 z}{-1 + z \alpha} - \frac{b_2 z^2}{-1 + z \alpha} + \frac{b_2 z^2 \alpha}{-1 + z \alpha}\right) y0$$

```
> r := unapply(collect(simplify(y1/y0), z), z);
```

$$r := z \rightarrow \frac{-1 + (b_1 \alpha - b_2 + b_2 \alpha) z^2 + (\alpha - b_1 - b_2) z}{-1 + z \alpha}$$

que es la función de estabilidad de la familia general propuesta, mientras que

```
> r1 := unapply(simplify(subs(solucion1, r(z))), z);
```

$$r1 := z \rightarrow \frac{1 - 2 + 2 z^2 \alpha - z^2 + 2 z \alpha - 2 z}{2(-1 + z \alpha)}$$

lo es de las soluciones de orden 2 obtenidas, y

```
> r2 := unapply(simplify(subs(solucion2, r(z))), z);
```

$$r2 := z \rightarrow -\frac{2 + z}{-2 + z}$$

lo es del citado método LOBATTO IIIA de 2 etapas

c)

función de estabilidad para los métodos del mayor orden

Para la familia de los métodos de orden 2 obtenidos, el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r1(z)$ sea inferior a 1 no será en general todo el semiplano de parte real negativa, ya que la función de estabilidad tiende a infinito cuando z tiende a infinito en cualquier dirección. O sea, los métodos no serán A-estables

Claro que, si $2\alpha - 1 = 0$, o sea $\alpha = 1/2$, numerador y denominador son polinomios de igual grado y la cosa cambia. Porque, en ese caso, el método es justamente el

ya citado LOBATTO IIIA de 2 etapas, cuya función de estabilidad ya se ha exhibido. En ese caso, el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r_2(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método LOBATTO IIIA de 2 etapas es A-estable, y es el único que lo es de entre los de orden 2, máximo posible.

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-21 (del EXAMEN EXTRAORDINARIO de 04JUL06)

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

0		0	
2α		α	α

		β	β

donde $0 < \alpha$ y $0 < \beta$

(lo que antecede es Ejercicio del capítulo 7)

b) Calcúlese la función de estabilidad para los métodos de la familia propuesta.

También para los métodos del orden máximo alcanzable

c) Compruébese que dichos métodos de orden máximo son A-estables. Búsquese algún otro método de la familia que también sea A-estable, describiendo en este caso la región de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0,y0),h):
[ > k2_:=unapply(f(x0+2*alpha*h,y0+alpha*h*kp1_(h)+alpha*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

[

```

> Fi_0:=expand(beta*k1_0+beta*k2_0);
                                Fi_0 := 2 β f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                cond1a := 1 = 2 β
> solve(cond1a,{beta});
                                { β = 1/2 }

condiciones para el orden 2
> Fi10:=expand(beta*k110+beta*k210);
                                Fi10 := 2 β D1(f)(x0, y0) α + 2 β D2(f)(x0, y0) α f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                cond2a := 1/2 = 2 β α
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                cond2b := 1/2 = 2 β α
> solucion:=solve({cond1a,cond2a,cond2b},{alpha,beta});
                                solucion := { β = 1/2, α = 1/2 }

condiciones para el orden 3
> Fi20:=expand(beta*k120+beta*k220);
                                Fi20 := 4 β D1,1(f)(x0, y0) α2 + 8 β D1,2(f)(x0, y0) α2 f(x0, y0) + 4 β α2 f(x0, y0)2 D2,2(f)(x0, y0)
                                + 4 β D2(f)(x0, y0) α2 D1(f)(x0, y0) + 4 β D2(f)(x0, y0)2 α2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
                                cond3a := 1/6 = 2 β α2
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
                                cond3b := 1/3 = 4 β α2
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
                                cond3c := 1/6 = 2 β α2
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
                                cond3d := 1/6 = 2 β α2
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
                                cond3e := 1/6 = 2 β α2
> solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});

```

Es decir, no se alcanza orden 3. El orden máximo alcanzable es 2 y se alcanza con el método dado por el tablero

$$0 \quad | \quad 0$$

$$1 \quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

$$\left| \frac{1}{2} \quad \frac{1}{2} \right|$$

b)

función de estabilidad

```
> f:=(x,y)->lambda*y;
```

$$f := (x, y) \rightarrow \lambda y$$

```
> ka1:=unapply(f(x0,y0),h);
```

$$ka1 := h \rightarrow \lambda y0$$

```
> ka2:=unapply(f(x0+2*alpha*h,y0+alpha*h*kp1(h)+alpha*h*kp2(h)),h);
```

$$ka2 := h \rightarrow \lambda (y0 + \alpha h kp1(h) + \alpha h kp2(h))$$

```
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
```

$$test := \{k1 = \lambda y0, k2 = -\frac{\lambda y0 (1 + \lambda \alpha h)}{-1 + \lambda \alpha h}\}$$

```
> k1:=subs(test,k1);k2:=subs(test,k2);
```

$$k1 := \lambda y0$$

$$k2 := -\frac{\lambda y0 (1 + \lambda \alpha h)}{-1 + \lambda \alpha h}$$

```
> y1:=expand(y0+h*(beta*k1+beta*k2));
```

$$y1 := y0 + h \beta \lambda y0 - \frac{h \beta \lambda y0}{-1 + \lambda \alpha h} - \frac{\beta \lambda^2 y0 \alpha h^2}{-1 + \lambda \alpha h}$$

```
> y1:=collect(subs(lambda=z/h,y1),y0);
```

$$y1 := \left(1 + \beta z - \frac{\beta z}{-1 + z \alpha} - \frac{\beta z^2 \alpha}{-1 + z \alpha}\right) y0$$

```
> r:=unapply(collect(simplify(y1/y0),z),z);
```

$$r := z \rightarrow \frac{-1 + (\alpha - 2 \beta) z}{-1 + \alpha z}$$

que es la función de estabilidad de la familia general propuesta.

Cuando α y β valen $1/2$, o sea, en el único caso de orden máximo 2, la función de estabilidad vale

```
> r1:=unapply(simplify(subs(solucion,r(z))),z);
```

$$r1 := z \rightarrow -\frac{2+z}{-2+z}$$

c)

La A-estabilidad

Cuando α y β valen $1/2$, o sea, en el único caso de orden máximo 2, la función de estabilidad vale

```
> r1:=unapply(simplify(subs(solucion,r(z))),z);
```

$$r1 := z \rightarrow -\frac{2+z}{-2+z}$$

el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r3(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2 . Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Tomemos ahora cualquiera de los casos para los que $\alpha = 2 \beta$, lo que reduce a 1 el numerador de la función de estabilidad. Por ejemplo, hagamos $\alpha = 1$ y $\beta = \frac{1}{2}$. Para ese caso, la función de estabilidad vale

```
> solucion2:={alpha=1,beta=1/2};
```

$$solucion2 := \{\beta = \frac{1}{2}, \alpha = 1\}$$

```
> r2:=unapply(simplify(subs(solucion2,r(z))),z);
```

$$r2 := z \rightarrow -\frac{1}{-1+z}$$

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda h$ cuya distancia al punto -1 sea superior a 1 , que es el exterior de la circunferencia de centro -1 y radio 1 (que pasa, pues, por el origen). Esta región contiene al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-22 (del EXAMEN ORDINARIO de 01FEB07)

a) Búsquense los métodos con el mayor orden posible para la familia de RUNGE-KUTTA implícitos del siguiente tablero

α		α	
1		$\frac{1}{2}$	$\frac{1}{2}$

	β	β
--	---------	---------

donde α y β son parámetros arbitrarios

(lo que antecede es Ejercicio del capítulo 7)

b) Calcúlese la función de estabilidad para los métodos de orden máximo hallados.

¿Son A-estables dichos métodos de orden máximo?.

c) Búsquese algún método de orden 1 de la familia que sea A-estable,

describiendo en este caso la región de estabilidad

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+alpha*h,y0+alpha*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+h,y0+(1/2)*h*kp1_(h)+(1/2)*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_)(0)=k110,D(kp2_)(0)=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

[a)

[condiciones para el orden 1

```
[ > Fi_0:=expand(beta*k1_0+beta*k2_0);
```

```

[
    Fi_0 := 2 β f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
    cond1a := 1 = 2 β
> solve(cond1a,{beta});
    {β = 1/2}
[condiciones para el orden 2
> Fi10:=expand(beta*k110+beta*k210);
    Fi10 := β D1(f)(x0, y0) α + β D2(f)(x0, y0) α f(x0, y0) + β D1(f)(x0, y0) + β D2(f)(x0, y0) f(x0, y0)
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
    cond2a := 1/2 = β α + β
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
    cond2b := 1/2 = β α + β
> solucion:=solve({cond1a,cond2a,cond2b},{alpha,beta});
    solucion := {β = 1/2, α = 0}
[condiciones para el orden 3
> Fi20:=expand(beta*k120+beta*k220);
    Fi20 := β D1,1(f)(x0, y0) α2 + 2 β D1,2(f)(x0, y0) α2 f(x0, y0) + β α2 f(x0, y0)2 D2,2(f)(x0, y0)
    + 2 β D2(f)(x0, y0) α2 D1(f)(x0, y0) + 2 β D2(f)(x0, y0)2 α2 f(x0, y0) + β D1,1(f)(x0, y0) + 2 β D1,2(f)(x0, y0) f(x0, y0)
    + β D2,2(f)(x0, y0) f(x0, y0)2 + β D2(f)(x0, y0) D1(f)(x0, y0) α + β D2(f)(x0, y0)2 α f(x0, y0)
    + β D2(f)(x0, y0) D1(f)(x0, y0) + β D2(f)(x0, y0)2 f(x0, y0)
> cond3a:=(1/6)*coeff(Y3,D[1,1](f)(x0,y0))=(1/2)*coeff(Fi20,D[1,1](f)(x0,y0));
    cond3a := 1/6 = 1/2 β α2 + 1/2 β
> cond3b:=(1/6)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
    cond3b := 1/3 = β α2 + β
> cond3c:=(1/6)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=(1/2)*coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
    cond3c := 1/6 = 1/2 β α2 + 1/2 β
> cond3d:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
    cond3d := 1/6 = β α2 + 1/2 β α + 1/2 β
> cond3e:=(1/6)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=(1/2)*coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
    cond3e := 1/6 = β α2 + 1/2 β α + 1/2 β
> subs(solucion,[cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e]);
    [1 = 1, 1/2 = 1/2, 1/2 = 1/2, 1/6 = 1/2, 1/3 = 1/2, 1/6 = 1/2, 1/6 = 1/2, 1/6 = 1/2]
> solve({cond1a,cond2a,cond2b,cond3a,cond3b,cond3c,cond3d,cond3e});

```

Es decir, no se alcanza orden 3. El orden máximo alcanzable es 2 y se alcanza con el método dado por el tablero

$$1 \quad | \quad \frac{1}{2} \quad \frac{1}{2}$$

$$| \quad \frac{1}{2} \quad \frac{1}{2}$$

b)

función de estabilidad para el método de orden 2 máximo

Cuando α vale 0 y β vale $1/2$, o sea, en el único caso de orden máximo 2, la función de estabilidad vale

> `f := (x, y) -> lambda*y;`

$$f := (x, y) \rightarrow \lambda y$$

> `ka1 := unapply(subs(solucion, f(x0+alpha*h, y0+alpha*h*kp1(h))), h);`

$$ka1 := h \rightarrow \lambda y0$$

> `ka2 := unapply(subs(solucion, f(x0+h, y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))), h);`

$$ka2 := h \rightarrow \lambda \left(y0 + \frac{1}{2} h kp1(h) + \frac{1}{2} h kp2(h) \right)$$

> `test := solve({k1=subs({kp1(h)=k1, kp2(h)=k2}, ka1(h)), k2=subs({kp1(h)=k1, kp2(h)=k2}, ka2(h))}, {k1, k2});`

$$test := \{ k1 = \lambda y0, k2 = -\frac{\lambda y0 (2 + \lambda h)}{-2 + \lambda h} \}$$

> `k1 := subs(test, k1); k2 := subs(test, k2);`

$$k1 := \lambda y0$$

$$k2 := -\frac{\lambda y0 (2 + \lambda h)}{-2 + \lambda h}$$

> `y1 := expand(subs(solucion, y0+h*(beta*k1+beta*k2)));`

$$y1 := y0 + \frac{h \lambda y0}{2} - \frac{h \lambda y0}{-2 + \lambda h} - \frac{\lambda^2 y0 h^2}{2(-2 + \lambda h)}$$

> `y1 := collect(subs(lambda=z/h, y1), y0);`

$$y1 := \left(1 + \frac{z}{2} - \frac{z}{-2 + z} - \frac{z^2}{2(-2 + z)} \right) y0$$

> `r := unapply(collect(simplify(y1/y0), z), z):rz := simplify(r(z));`

$$rz := -\frac{2 + z}{-2 + z}$$

el dominio de estabilidad, compuesto por los $z = \lambda h$ tales que el módulo de $r(z)$ sea inferior a 1, estará formado por los valores de z que disten menos del punto -2 que del punto 2. Esto corresponde al semiplano de parte real negativa, por lo que el método estudiado es A-estable

c)

función de estabilidad para un método de orden exactamente 1

Tomemos ahora cualquiera de los casos para los que $\beta = \frac{1}{2}$, de manera que el orden sea 1 al menos. Vamos a ensayar con el valor $\alpha = 1$, para el que no se tiene orden 2 (máximo)

> `restart;`

> `solucion1 := {alpha=1, beta=1/2};`

$$solucion1 := \{ \alpha = 1, \beta = \frac{1}{2} \}$$

> `f := (x, y) -> lambda*y;`

$$f := (x, y) \rightarrow \lambda y$$

> `ka1 := unapply(subs(solucion1, f(x0+alpha*h, y0+alpha*h*kp1(h))), h);`

$$ka1 := h \rightarrow \lambda (y0 + h kp1(h))$$

```

> ka2:=unapply(subs(solucion1,f(x0+h,y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))),h);
      ka2 := h → λ (y0 + 1/2 h kp1(h) + 1/2 h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -λ y0 / (-1 + λ h), k2 = -λ y0 / (-1 + λ h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -λ y0 / (-1 + λ h)
      k2 := -λ y0 / (-1 + λ h)
> y1:=expand(subs(solucion1,y0+h*(beta*k1+beta*k2)));
      y1 := y0 - h λ y0 / (-1 + λ h)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-1 + z)) y0
> r:=unapply(collect(simplify(y1/y0),z),z):rz:=simplify(r(z));
      rz := -1 / (-1 + z)

```

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda h$ cuya distancia al punto -1 sea superior a 1 , que es el exterior de la circunferencia de centro -1 y radio 1 (que pasa, pues, por el origen). Esta región contiene al semiplano de parte real negativa, por lo que el método estudiado es A-estable

Otra opción: tomemos $\beta = \frac{1}{2}$, y $\alpha = \frac{1}{2}$, para el que tampoco se tiene orden 2 (máximo)

```

> restart:
> solucion1:={alpha=1/2,beta=1/2};
      solucion1 := {α = 1/2, β = 1/2}
> f:=(x,y)->lambda*y;
      f := (x, y) → λ y
> ka1:=unapply(subs(solucion1,f(x0+alpha*h,y0+alpha*h*kp1(h))),h);
      ka1 := h → λ (y0 + 1/2 h kp1(h))
> ka2:=unapply(subs(solucion1,f(x0+h,y0+(1/2)*h*kp1(h)+(1/2)*h*kp2(h))),h);
      ka2 := h → λ (y0 + 1/2 h kp1(h) + 1/2 h kp2(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -2 λ y0 / (-2 + λ h), k2 = 4 λ y0 / (4 - 4 λ h + λ² h²)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -2 λ y0 / (-2 + λ h)
      k2 := 4 λ y0 / (4 - 4 λ h + λ² h²)
> y1:=expand(subs(solucion1,y0+h*(beta*k1+beta*k2)));
      y1 := y0 - h λ y0 / (-2 + λ h) + 2 h λ y0 / (4 - 4 λ h + λ² h²)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-2 + z) + 2 z / (4 - 4 z + z²)) y0

```

```
> r:=unapply(collect(simplify(y1/y0),z),z):rz:=factor(simplify(r(z)));
```

$$rz := \frac{4}{(-2+z)^2}$$

Ahora, el dominio de estabilidad está compuesto por los $z = \lambda$ h cuya distancia al punto -2 sea superior a 2 , que es el exterior de la circunferencia de centro -2 y radio 2 (que pasa, pues, por el origen). Esta región también contiene al semiplano de parte real negativa, por lo que el método también es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-23 (del EXAMEN EXTRAORDINARIO DE 15JUL09)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{l|l} 0 & \alpha & -\alpha \\ 2 & \alpha & 2-\alpha \end{array}$$

$$\begin{array}{l|l} & \frac{1}{4} & \frac{3}{4} \end{array}$$

a) Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α para los que se consigue

(lo que antecede es Ejercicio del capítulo 7)

b) Calcúlese la función de estabilidad del método en el caso general.

c) Búsqese un valor de α para el que el correspondiente método sea A-estable, probando, además, dicha A-estabilidad.

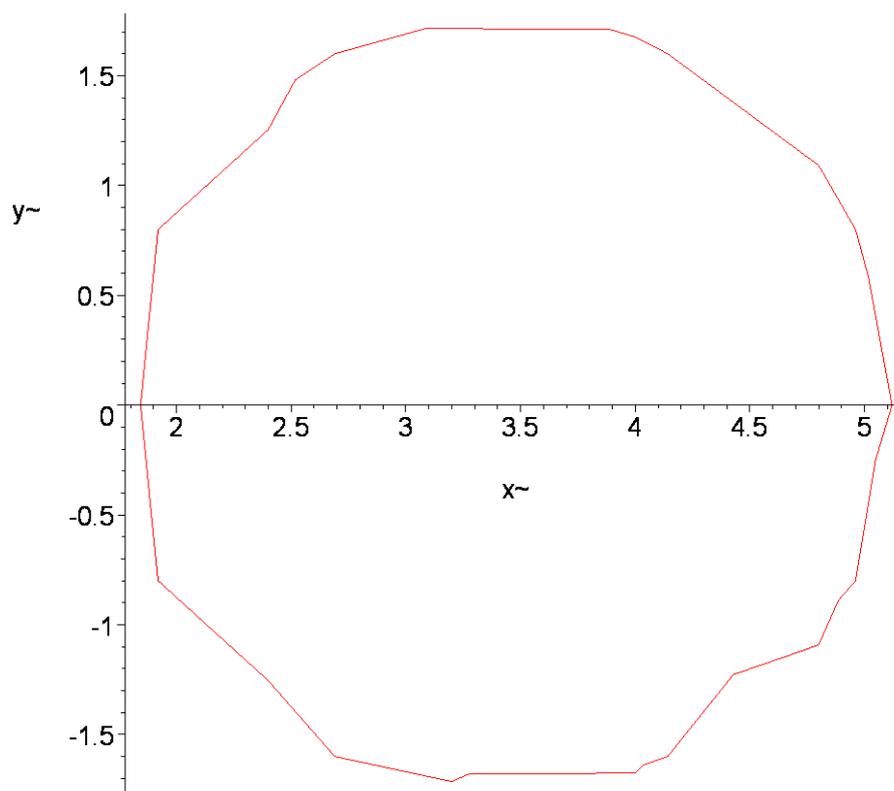
```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0,y0+alpha*h*kp10(h)-alpha*h*kp20(h)),h):
[ > k20:=unapply(f(x0+2*h,y0+alpha*h*kp10(h)+(2-alpha)*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ ]
```



```

den := 2 - 4 z + z^2
> z:=x+I*y:assume(x::real):assume(y::real):
> numr:=Re(num);numi:=Im(num);denr:=Re(den);deni:=Im(den);
numr := 2 - 2 x~
numi := -2 y~
denr := 2 - 4 x~ + x~^2 - y~^2
deni := -4 y~ + 2 x~ y~
> simplify(numr^2+numi^2);simplify(denr^2+deni^2);
4 - 8 x~ + 4 x~^2 + 4 y~^2
4 - 16 x~ + 20 x~^2 + 12 y~^2 - 8 x~^3 - 8 x~ y~^2 + x~^4 + 2 x~^2 y~^2 + y~^4
> borde:=simplify(numr^2+numi^2-denr^2-deni^2);
borde := 8 x~ - 16 x~^2 - 8 y~^2 + 8 x~^3 + 8 x~ y~^2 - x~^4 - 2 x~^2 y~^2 - y~^4
> with(plots):
> implicitplot(borde=0,x=-40..40,y=-40..40, gridrefine=2);

```



Debe hacerse notar que, cuando $x < 0$, entonces $\text{borde} < 0$, o sea, el borde de la región de estabilidad se encuentra en el semiplano $\text{Re}(z) > 0$

```
> r(-1);
```

$$\frac{4}{7}$$

```
>
```

En el caso $\alpha = \frac{1}{4}$ la región de estabilidad contiene todo el semiplano $\text{Re}(z) < 0$,

por lo que el método es A-estable. Se trata del método

$$0 \mid \frac{1}{4} \quad - \frac{1}{4}$$

$$2 \mid \frac{1}{4} \quad \frac{5}{4}$$

$$\mid \frac{1}{4} \quad \frac{3}{4}$$

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-24 (del EXAMEN ORDINARIO DE 09FEB09)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\alpha \quad | \quad \alpha$$

$$2\beta \quad | \quad \beta \quad \beta$$

$$| \quad \frac{1}{4} \quad \frac{3}{4}$$

Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α y de β para los que se consigue

(lo que antecede es Ejercicio del capítulo 7)

Quedádonos ahora (por simplicidad) con la situación $\alpha = 0$, calcúlese la función de estabilidad del método en estos casos

Estúdiense la A-estabilidad de estos métodos para los que $\alpha = 0$

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+alpha*h*kp10(h)),h):
[ > k20:=unapply(f(x0+2*beta*h,y0+beta*h*kp10(h)+beta*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ ]
```

```

[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
  xpanse(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
  xpanse(k220):
[ > Fi2:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
  cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
  cond2a :=  $\frac{\alpha}{4} + \frac{3\beta}{2} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
  ,f(x0,y0)));
  cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{2} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a});
  orden2 := { $\alpha = -6\beta + 2, \beta = \beta$ }
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
  cond3a :=  $\frac{\alpha^2}{4} + 3\beta^2 = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
  y0)),f(x0,y0));
  cond3b :=  $\frac{\alpha^2}{2} + 6\beta^2 = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
  0,y0)),f(x0,y0)^2);
  cond3c :=  $\frac{\alpha^2}{4} + 3\beta^2 = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
  0,y0)),D[1](f)(x0,y0));
  cond3d :=  $\frac{1}{2}\alpha^2 + \frac{3}{2}\beta\alpha + 3\beta^2 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
  )^2,f(x0,y0));
  cond3e :=  $\frac{1}{2}\alpha^2 + \frac{3}{2}\beta\alpha + 3\beta^2 = \frac{1}{3}$ 
[ > orden3:=solve({cond2a,cond3a,cond3d});
  orden3 := { $\beta = \frac{1}{3}, \alpha = 0$ }
[ se obtiene orden 3 para los valores alpha = 0 y beta = 1/3
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
  cond4a :=  $\frac{\alpha^3}{4} + 6\beta^3 = \frac{1}{4}$ 
[ > cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
  (x0,y0)),f(x0,y0));

```

$$\text{cond4b} := 18\beta^3 + \frac{3\alpha^3}{4} = \frac{3}{4}$$

> `cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2);`

$$\text{cond4c} := 18\beta^3 + \frac{3\alpha^3}{4} = \frac{3}{4}$$

> `cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0));`

$$\text{cond4d} := 18\beta^3 + \frac{3}{2}\alpha^3 + 9\beta^2\alpha = \frac{3}{4}$$

> `cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));`

$$\text{cond4e} := 3\alpha^3 + 36\beta^3 + 9\beta^2\alpha + \frac{9}{2}\beta\alpha^2 = \frac{5}{4}$$

> `cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);`

$$\text{cond4f} := \frac{\alpha^3}{4} + 6\beta^3 = \frac{1}{4}$$

> `cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));`

$$\text{cond4g} := 18\beta^3 + \frac{3}{2}\alpha^3 + 9\beta^2\alpha = \frac{3}{4}$$

> `cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);`

$$\text{cond4h} := 9\beta^2\alpha + \frac{9}{4}\beta\alpha^2 + \frac{9}{4}\alpha^3 + 27\beta^3 = 1$$

> `cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));`

$$\text{cond4i} := 9\beta^3 + \frac{3}{4}\alpha^3 + \frac{9}{4}\beta\alpha^2 = \frac{1}{4}$$

> `cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);`

$$\text{cond4j} := \frac{3}{2}\alpha^3 + 9\beta^3 + \frac{9}{2}\beta\alpha^2 + \frac{9}{2}\beta^2\alpha = \frac{1}{4}$$

> `cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));`

$$\text{cond4k} := \frac{3}{2}\alpha^3 + 9\beta^3 + \frac{9}{2}\beta\alpha^2 + \frac{9}{2}\beta^2\alpha = \frac{1}{4}$$

> `subs(orden3,{cond4a,cond4b,cond4c,cond4d,cond4e,cond4f,cond4g,cond4h,cond4i,cond4j,cond4k});`

$$\left\{ \frac{1}{3} = \frac{1}{4}, 1 = 1, \frac{2}{3} = \frac{3}{4}, \frac{2}{9} = \frac{1}{4}, \frac{4}{3} = \frac{5}{4} \right\}$$

y no se puede alcanzar orden 4 , lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = 0$ y $\beta = \frac{1}{3}$, o sea

para el método

$$\begin{array}{c|cc} 0 & & 0 \\ \hline \frac{2}{3} & \frac{1}{3} & \frac{1}{3} \end{array}$$

$$\left| \begin{array}{cc} 1 & 3 \\ 4 & 4 \end{array} \right|$$

que suele recibir el nombre específico de RADAU I de 2 etapas

```
# funcion de estabilidad para  $\alpha=0$ 
[ > restart:
[ > interface(labeling=false):
[ > alpha:=0:
[ > f:=(x,y)->lambda*y;
                                f:=(x,y) ->  $\lambda y$ 
[ > alpha:=0;
                                 $\alpha := 0$ 
[ > ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
                                ka1 :=  $h \rightarrow \lambda y0$ 
[ > ka2:=unapply(f(x0+2*beta*h,y0+beta*h*kp1(h)+beta*h*kp2(h)),h);
                                ka2 :=  $h \rightarrow \lambda (y0 + \beta h kp1(h) + \beta h kp2(h))$ 
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                test := {k1 =  $\lambda y0$ , k2 =  $-\frac{\lambda y0(1 + \lambda \beta h)}{-1 + \lambda \beta h}$ }
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
                                k1 :=  $\lambda y0$ 
                                k2 :=  $-\frac{\lambda y0(1 + \lambda \beta h)}{-1 + \lambda \beta h}$ 
                                 $-\frac{2 \lambda y0(2 + \lambda \beta h)}{-1 + \lambda \beta h}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
                                y1 :=  $y0 + \frac{h \lambda y0}{4} - \frac{3 h \lambda y0}{4(-1 + \lambda \beta h)} - \frac{3 \lambda^2 y0 \beta h^2}{4(-1 + \lambda \beta h)}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                y1 :=  $\left(1 + \frac{z}{4} - \frac{3z}{4(-1+z\beta)} - \frac{3z^2\beta}{4(-1+z\beta)}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_general:=r(z);
                                r_general :=  $-\frac{2+z^2\beta+(-2\beta+2)z}{2(-1+z\beta)}$ 
```

A-estabilidad para $\alpha=0$

Estos métodos (incluyendo el ya citado RADAU I de 2 etapas nunca pueden ser A-estables,

ya que el numerador de $r(z)$ es de grado 2 mientras que el denominador lo es de 1

La excepción podría ser $\beta = 0$ En este último caso, la función de estabilidad vale

$$r(z)=1+z$$

por lo que tampoco se tiene A-estabilidad. Se observará que este último método concreto,

para el que $\alpha = 0$ y $\beta = 0$ es el método de tablero

$$\begin{array}{cc} 0 & | & 0 \\ 0 & | & 0 & 0 \end{array}$$

$$\left| \begin{array}{cc} 1 & 3 \\ 4 & 4 \end{array} \right|$$

que es de hecho el método explícito de EULER (de 1 etapa)

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-25 (del EXAMEN ORDINARIO DE 09FEB10)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} \alpha & \alpha - 1 & 1 \\ \beta & \beta & 0 \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α y de β para los que se consigue

(lo que antecede es Ejercicio del capítulo 7)

Considérese un ejemplo de los de mayor orden y búsqese en ese caso

la función de estabilidad del método

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp10(h)+h*kp20(h)),h):
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp10(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
```

```

[ xexpand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
[ xexpand(k220):
[ > Fi2:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[ kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
[ kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
[ cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
[ cond2a :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
[ ,f(x0,y0)));
[ cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a});
[ orden2 := { $\alpha = -3\beta + 2, \beta = \beta$ }
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
[ cond3a :=  $\frac{\alpha^2}{4} + \frac{3\beta^2}{4} = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
[ y0)),f(x0,y0));
[ cond3b :=  $\frac{\alpha^2}{2} + \frac{3\beta^2}{2} = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
[ 0,y0)),f(x0,y0)^2);
[ cond3c :=  $\frac{\alpha^2}{4} + \frac{3\beta^2}{4} = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
[ 0,y0)),D[1](f)(x0,y0));
[ cond3d :=  $\frac{1}{2}\alpha^2 - \frac{1}{2}\alpha + \frac{1}{2}\beta + \frac{3}{2}\beta\alpha = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
[ ^2),f(x0,y0));
[ cond3e :=  $\frac{1}{2}\alpha^2 - \frac{1}{2}\alpha + \frac{1}{2}\beta + \frac{3}{2}\beta\alpha = \frac{1}{3}$ 
[ > orden3:=solve({cond2a,cond3a,cond3d});
[ orden3 := { $\alpha = 0, \beta = \frac{2}{3}$ }
[ se obtiene orden 3 para los valores alpha = 0 y beta = 2/3
[ # condiciones para el orden 4
[ > cond4a:=coeff(Fi3,D[1,1,1](f)(x0,y0))=coeff((1/4)*f3,D[1,1,1](f)(x0,y0));
[ cond4a :=  $\frac{\alpha^3}{4} + \frac{3\beta^3}{4} = \frac{1}{4}$ 
[ > cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)
[ (x0,y0)),f(x0,y0));
[ cond4b :=  $\frac{9\beta^3}{4} + \frac{3\alpha^3}{4} = \frac{3}{4}$ 
[ > cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](

```

$f(x_0, y_0), f(x_0, y_0)^2);$

$$cond4c := \frac{9\beta^3}{4} + \frac{3\alpha^3}{4} = \frac{3}{4}$$

> $cond4d := \text{coeff}(\text{coeff}(\text{coeff}(Fi3, D[1, 2])(f(x_0, y_0)), D[1](f(x_0, y_0))), D[1](f(x_0, y_0))) = \text{coeff}(\text{coeff}((1/4)*f3, D[1, 2])(f(x_0, y_0)), D[1](f(x_0, y_0)));$

$$cond4d := \frac{3}{2}\alpha^3 - \frac{3}{2}\alpha^2 + \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 = \frac{3}{4}$$

> $cond4e := \text{coeff}(\text{coeff}(\text{coeff}(Fi3, D[1, 2])(f(x_0, y_0)), D[2](f(x_0, y_0))), f(x_0, y_0)) = \text{coeff}(\text{coeff}(\text{coeff}((1/4)*f3, D[1, 2])(f(x_0, y_0)), D[2](f(x_0, y_0))), f(x_0, y_0));$

$$cond4e := 3\alpha^3 + \frac{3}{2}\beta^2 + \frac{9}{2}\alpha\beta^2 + \frac{9}{2}\beta\alpha^2 + \frac{3}{2}\beta\alpha - 3\alpha^2 = \frac{5}{4}$$

> $cond4f := \text{coeff}(\text{coeff}(Fi3, D[2, 2, 2])(f(x_0, y_0)), f(x_0, y_0)^3) = \text{coeff}(\text{coeff}((1/4)*f3, D[2, 2, 2])(f(x_0, y_0)), f(x_0, y_0)^3);$

$$cond4f := \frac{\alpha^3}{4} + \frac{3\beta^3}{4} = \frac{1}{4}$$

> $cond4g := \text{coeff}(\text{coeff}(\text{coeff}(Fi3, D[2, 2])(f(x_0, y_0)), D[1](f(x_0, y_0))), f(x_0, y_0)) = \text{coeff}(\text{coeff}(\text{coeff}((1/4)*f3, D[2, 2])(f(x_0, y_0)), D[1](f(x_0, y_0))), f(x_0, y_0));$

$$cond4g := \frac{3}{2}\alpha^3 - \frac{3}{2}\alpha^2 + \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 = \frac{3}{4}$$

> $cond4h := \text{coeff}(\text{coeff}(\text{coeff}(Fi3, D[2, 2])(f(x_0, y_0)), D[2](f(x_0, y_0))), f(x_0, y_0)^2) = \text{coeff}(\text{coeff}(\text{coeff}((1/4)*f3, D[2, 2])(f(x_0, y_0)), D[2](f(x_0, y_0))), f(x_0, y_0)^2);$

$$cond4h := \frac{3}{2}\beta\alpha + \frac{9}{2}\alpha\beta^2 + \frac{9}{4}\beta\alpha^2 + \frac{9}{4}\alpha^3 - \frac{9}{4}\alpha^2 + \frac{3}{4}\beta^2 = 1$$

> $cond4i := \text{coeff}(\text{coeff}(Fi3, D[1, 1])(f(x_0, y_0)), D[2](f(x_0, y_0))) = \text{coeff}(\text{coeff}((1/4)*f3, D[1, 1])(f(x_0, y_0)), D[2](f(x_0, y_0)));$

$$cond4i := \frac{3}{4}\beta^2 - \frac{3}{4}\alpha^2 + \frac{9}{4}\beta\alpha^2 + \frac{3}{4}\alpha^3 = \frac{1}{4}$$

> $cond4j := \text{coeff}(\text{coeff}(Fi3, D[1])(f(x_0, y_0)), D[2](f(x_0, y_0)^2)) = \text{coeff}(\text{coeff}((1/4)*f3, D[1])(f(x_0, y_0)), D[2](f(x_0, y_0)^2));$

$$cond4j := \frac{3}{2}\alpha^3 - 3\alpha^2 + \frac{9}{2}\beta^2 - \frac{3}{2}\beta + \frac{3}{2}\alpha - \frac{3}{2}\beta\alpha + \frac{9}{2}\beta\alpha^2 = \frac{1}{4}$$

> $cond4k := \text{coeff}(\text{coeff}(Fi3, D[2])(f(x_0, y_0))^3, f(x_0, y_0)) = \text{coeff}(\text{coeff}((1/4)*f3, D[2])(f(x_0, y_0))^3, f(x_0, y_0));$

$$cond4k := \frac{3}{2}\alpha^3 - 3\alpha^2 + \frac{9}{2}\beta^2 - \frac{3}{2}\beta + \frac{3}{2}\alpha - \frac{3}{2}\beta\alpha + \frac{9}{2}\beta\alpha^2 = \frac{1}{4}$$

> $\text{subs}(\text{orden3}, \{cond4a, cond4b, cond4c, cond4d, cond4e, cond4f, cond4g, cond4h, cond4i, cond4j, cond4k\});$

$$\left\{ \frac{1}{3} = \frac{1}{4}, \frac{2}{3} = \frac{5}{4}, 0 = \frac{3}{4}, 1 = \frac{1}{4}, \frac{2}{3} = \frac{3}{4}, \frac{1}{3} = 1, \frac{2}{9} = \frac{1}{4} \right\}$$

y no se puede alcanzar orden 4, lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = 0$ y $\beta = \frac{2}{3}$, o sea

para el método

$$0 \quad | \quad -1 \quad 1$$

$$\frac{2}{3} \quad | \quad \frac{2}{3} \quad 0$$

$$| \quad \frac{1}{4} \quad \frac{3}{4}$$

```

# funcion de estabilidad para  $\alpha=0$  y  $\beta=\frac{2}{3}$ 
[ > restart:
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
                                f:=(x,y) -> lambda*y
[ > alpha:=0;beta:=2/3;
                                alpha:=0
                                beta:=2/3
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
                                k10 := h -> lambda (y0 - h kp1(h) + h kp2(h))
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
                                k20 := h -> lambda (y0 + 2/3 h kp1(h))
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
                                ka1 := h -> lambda (y0 - h kp1(h) + h kp2(h))
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
                                ka2 := h -> lambda (y0 + 2/3 h kp1(h))
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
                                test := {k1 = - 3 lambda y0 (1 + lambda h) / (-3 - 3 lambda h + 2 lambda^2 h^2), k2 = - lambda y0 (3 + 5 lambda h) / (-3 - 3 lambda h + 2 lambda^2 h^2)}
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
                                k1 := - 3 lambda y0 (1 + lambda h) / (-3 - 3 lambda h + 2 lambda^2 h^2)
                                k2 := - lambda y0 (3 + 5 lambda h) / (-3 - 3 lambda h + 2 lambda^2 h^2)
                                - 6 lambda y0 (2 + 3 lambda h) / (-3 - 3 lambda h + 2 lambda^2 h^2)
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
                                y1 := y0 - 3 h lambda y0 / (-3 - 3 lambda h + 2 lambda^2 h^2) - 9 lambda^2 y0 h^2 / (2 (-3 - 3 lambda h + 2 lambda^2 h^2))
[ > y1:=collect(subs(lambda=z/h,y1),y0);
                                y1 := (1 - 3 z / (-3 - 3 z + 2 z^2) - 9 z^2 / (2 (-3 - 3 z + 2 z^2))) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
                                r_particular := - (6 + 12 z + 5 z^2) / (2 (-3 - 3 z + 2 z^2))
[ >
[ >
[ # A-estabilidad para alpha = 0 y beta = 2/3
[ Este método no es A-estable, ya que r(z) tiende a -5/4 cuando z tiende a infinity

```

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-26 (del EXAMEN ORDINARIO DE 03JUL10)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} 0 & & 0 \\ C & A_1 & A_2 \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

a) Búsqese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de C , A_1 y A_2 para los que se consigue

(hasta aquí es Ejercicio del capítulo 7)

b) Calcúlese la función de estabilidad en el caso general

c) Búsqese un ejemplo de método A-estable dentro de la familia de métodos del problema.

d) Compruébese si es o no es A-estable el método con valores $A_1 = 2/3$ y $A_2 = 2$

de los parámetros (C no influye, claro) y dibújese la región de estabilidad en este caso.

```
[ > restart:
[ > interface(labeling=false):
[ a)
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0,y0),h):
[ > k20:=unapply(f(x0+C*h,y0+A[1]*h*kp10(h)+A[2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
```


$$cond4a := \frac{3C^3}{4} = \frac{1}{4}$$

> cond4b:=coeff(coeff(Fi3,D[1,1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/4)*f3,D[1,1,2](f)(x0,y0)),f(x0,y0));

$$cond4b := \frac{9}{4}C^2A_1 + \frac{9}{4}C^2A_2 = \frac{3}{4}$$

> cond4c:=coeff(coeff(Fi3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1,2,2](f)(x0,y0)),f(x0,y0)^2);

$$cond4c := \frac{9}{2}CA_1A_2 + \frac{9}{4}CA_1^2 + \frac{9}{4}CA_2^2 = \frac{3}{4}$$

> cond4d:=coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[1](f)(x0,y0));

$$cond4d := \frac{9}{2}C^2A_2 = \frac{3}{4}$$

> cond4e:=coeff(coeff(coeff(Fi3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[1,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0));

$$cond4e := 9CA_1A_2 + 9CA_2^2 = \frac{5}{4}$$

> cond4f:=coeff(coeff(Fi3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3)=coeff(coeff((1/4)*f3,D[2,2,2](f)(x0,y0)),f(x0,y0)^3);

$$cond4f := \frac{3}{4}A_2^3 + \frac{3}{4}A_1^3 + \frac{9}{4}A_1^2A_2 + \frac{9}{4}A_1A_2^2 = \frac{1}{4}$$

> cond4g:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0))=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[1](f)(x0,y0)),f(x0,y0));

$$cond4g := \frac{9}{2}CA_1A_2 + \frac{9}{2}CA_2^2 = \frac{3}{4}$$

> cond4h:=coeff(coeff(coeff(Fi3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(coeff((1/4)*f3,D[2,2](f)(x0,y0)),D[2](f)(x0,y0)),f(x0,y0)^2);

$$cond4h := \frac{27}{4}A_1^2A_2 + \frac{27}{2}A_1A_2^2 + \frac{27}{4}A_2^3 = 1$$

> cond4i:=coeff(coeff(Fi3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0))=coeff(coeff((1/4)*f3,D[1,1](f)(x0,y0)),D[2](f)(x0,y0));

$$cond4i := \frac{9}{4}C^2A_2 = \frac{1}{4}$$

> cond4j:=coeff(coeff(Fi3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2)=coeff(coeff((1/4)*f3,D[1](f)(x0,y0)),D[2](f)(x0,y0)^2);

$$cond4j := \frac{9}{2}CA_2^2 = \frac{1}{4}$$

> cond4k:=coeff(coeff(Fi3,D[2](f)(x0,y0)^3),f(x0,y0))=coeff(coeff((1/4)*f3,D[2](f)(x0,y0)^3),f(x0,y0));

$$cond4k := \frac{9}{2}A_2^3 + \frac{9}{2}A_1A_2^2 = \frac{1}{4}$$

> subs(ordena3,{cond4a,cond4b,cond4c,cond4d,cond4e,cond4f,cond4g,cond4h,cond4i,cond4j,cond4k});

$$\{1 = 1, \frac{2}{9} = \frac{1}{4} \cdot \frac{2}{3} = \frac{3}{4} \cdot \frac{4}{3} = \frac{5}{4} \cdot \frac{1}{3} = \frac{1}{4}\}$$

y no se puede alcanzar orden 4 , lo que por otra parte era obvio ya que el método de GAUSS de 2 etapas no está entre los de la familia propuesta

Se obtiene orden máximo 3 de una sola manera, para los valores $C = 2/3$, $A_1 = 1/3$ y $A_2 = 1/3$, o sea para el método

$$\begin{array}{c|c} 0 & 0 \\ \hline \frac{2}{3} & \frac{1}{3} \quad \frac{1}{3} \end{array}$$

$$\left| \begin{array}{cc} 1 & 3 \\ 4 & 4 \end{array} \right|$$

que suele recibir el nombre específico de RADAU I de 2 etapas

b)

funcion de estabilidad

> restart;

> interface(labeling=false):

> f:=(x,y)->lambda*y;

$$f := (x, y) \rightarrow \lambda y$$

> ka1:=unapply(f(x0,y0),h);

$$ka1 := h \rightarrow \lambda y0$$

> ka2:=unapply(f(x0+C*h,y0+A[1]*h*kp1(h)+A[2]*h*kp2(h)),h);

$$ka2 := h \rightarrow \lambda (y0 + A_1 h kp1(h) + A_2 h kp2(h))$$

> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});

$$test := \{k1 = \lambda y0, k2 = -\frac{\lambda y0 (1 + \lambda A_1 h)}{-1 + \lambda A_2 h}\}$$

> k1:=subs(test,k1);k2:=subs(test,k2);

$$k1 := \lambda y0$$

$$k2 := -\frac{\lambda y0 (1 + \lambda A_1 h)}{-1 + \lambda A_2 h}$$

> y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));

$$y1 := y0 + \frac{h \lambda y0}{4} - \frac{3}{4} \frac{h \lambda y0}{-1 + \lambda A_2 h} - \frac{3}{4} \frac{\lambda^2 y0 A_1 h^2}{-1 + \lambda A_2 h}$$

> y1:=collect(subs(lambda=z/h,y1),y0);

$$y1 := \left(1 + \frac{z}{4} - \frac{3}{4} \frac{z}{-1 + z A_2} - \frac{3}{4} \frac{z^2 A_1}{-1 + z A_2}\right) y0$$

> r:=unapply(collect(simplify(y1/y0),z),z):r_general:=r(z);

$$r_general := -\frac{1}{4} \frac{4 + (-A_2 + 3 A_1) z^2 + (-4 A_2 + 4) z}{-1 + z A_2}$$

> r_nulo:=simplify(subs({A[2]=3*A[1]},r(z)));

$$r_nulo := \frac{-1 + 3 z A_1 - z}{-1 + 3 z A_1}$$

>

c)

A-estabilidad

Estos métodos (incluyendo el ya citado RADAU I de 2 etapas) nunca pueden ser A-estables,

ya que el numerador de $r(z)$ es de grado 2 mientras que el denominador lo es de 1

La excepción podría ser $A_2 = 3 A_1$ En este último caso, la función de estabilidad vale

$$r(z) = \frac{1 + (1 - 3 A_1) z}{1 - 3 A_1 z}$$

y es en principio de consecuencias dudosas.

Si hacemos entonces $A_1 = 1/3$ obtenemos como función de estabilidad

$$r(z) = \frac{1}{1-z}$$

y ya sabemos que dicho método es A-estable. Se trata del método

$$0 \mid 0$$

$$C \mid \frac{1}{3} \quad 1$$

$$\mid \frac{1}{4} \quad \frac{3}{4}$$

> `r_ejemplo1:=simplify(subs({A[1]=1/3},r_nulo));`

$$r_ejemplo1 := -\frac{1}{-1+z}$$

d)

Para $A_1 = 2/3$ y $A_2 = 2$, que entra dentro de la situación $A_2 = 3 A_1$, la función de estabilidad es

$$r(z) = \frac{z-1}{2z-1}$$

> `r_ejemplo2:=simplify(subs({A[1]=2/3,A[1]=2},r_nulo));`

$$r_ejemplo2 := \frac{-1+z}{-1+2z}$$

>

>

Pongamos $z=x+iy$. Como

$$|2z-1|^2 = (2x-1)^2 + 4y^2$$

y

$$|z-1|^2 = (x-1)^2 + y^2$$

la frontera $|r(z)| = 1$ o mejor $|r(z)|^2 = 1$ del dominio de estabilidad

es la curva de ecuación

$$(2x-1)^2 + 4y^2 = (x-1)^2 + y^2$$

que, tras operar, es la circunferencia

$$\left(x - \frac{1}{3}\right)^2 + y^2 = \frac{1}{9}$$

de centro $\frac{1}{3}$ y radio $\frac{1}{3}$, que pasa por $(0, 0)$. El valor $z = -1$ queda en

el exterior de la circunferencia, y, para él, $r(-1) = 2/3$, luego -1 pertenece

al dominio de estabilidad. El dominio de estabilidad

es el exterior de esa circunferencia, y el método es A-estable

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-27 (del EXAMEN ORDINARIO DE 19ENE12)

De entre los métodos implícitos de 2 etapas de tablero

$$0 \mid \alpha \quad 0$$

$$1 \mid \alpha \quad 0$$

$$\mid \frac{1}{2} \quad \frac{1}{2}$$

se buscaran los que sean de mayor orden.

(hasta aquí es Ejercicio del capítulo 7)

Calcúlese la función de estabilidad de la familia general de los métodos del tablero y, para $\alpha = 2$, dibújese el dominio de estabilidad y discútase

la A-estabilidad del método

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > c[1]:=0:c[2]:=1:
[ > a[1,1]:=alpha:a[1,2]:=0:a[2,1]:=alpha:a[2,2]:=0:
[ > b[1]:=1/2:b[2]:=1/2:
[ > k10:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp10(h)+a[1,2]*h*kp20(h)),h):
[ > k20:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp10(h)+a[2,2]*h*kp20(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > Fi0:=expand(b[1]*k100+b[2]*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > Fi1:=expand(b[1]*k110+b[2]*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
[ xpand(k120):
[ ]
```

```

[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
xpend(k220):
[ > Fi2:=expand(b[1]*k120+b[2]*k220):
[ > k13:=D(k12):
[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand(b[1]*k130+b[2]*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
cond2a :=  $\frac{1}{2} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0)
,f(x0,y0)));
cond2b :=  $\alpha = \frac{1}{2}$ 
[ orden 2 para  $\alpha = 1/2$  (o sea, LOBATTO IIIb)
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));subs(alpha=1/2,co
nd3a);
cond3a :=  $\frac{1}{2} = \frac{1}{3}$ 
 $\frac{1}{2} = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
y0)),f(x0,y0));subs(alpha=1/2,cond3b);
cond3b :=  $\alpha = \frac{2}{3}$ 
 $\frac{1}{2} = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
0,y0)),f(x0,y0)^2);subs(alpha=1/2,cond3c);
cond3c :=  $\alpha^2 = \frac{1}{3}$ 
 $\frac{1}{4} = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
0,y0)),D[1](f)(x0,y0));subs(alpha=1/2,cond3d);
cond3d :=  $0 = \frac{1}{3}$ 
 $0 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
^2),f(x0,y0));subs(alpha=1/2,cond3e);
cond3e :=  $2\alpha^2 = \frac{1}{3}$ 
 $\frac{1}{2} = \frac{1}{3}$ 
[ no hay orden 3 para ningun valor de  $\alpha$ 
[ # funcion de estabilidad
[ > f:=(x,y)->lambda*y;
f := (x, y) →  $\lambda y$ 

```

```

> ka1:=unapply(f(x0+c[1]*h,y0+a[1,1]*h*kp1(h)+a[1,2]*h*kp2(h)),h);
      ka1 := h → λ (y0 + α h kp1(h))
> ka2:=unapply(f(x0+c[2]*h,y0+a[2,1]*h*kp1(h)+a[2,2]*h*kp2(h)),h);
      ka2 := h → λ (y0 + α h kp1(h))
> test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
      test := {k1 = -λ y0 / (-1 + λ α h), k2 = -λ y0 / (-1 + λ α h)}
> k1:=subs(test,k1);k2:=subs(test,k2);
      k1 := -λ y0 / (-1 + λ α h)
      k2 := -λ y0 / (-1 + λ α h)
> y1:=expand(y0+h*(b[1]*k1+b[2]*k2));
      y1 := y0 - h λ y0 / (-1 + λ α h)
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (-1 + z α)) y0
> r:=unapply(collect(simplify(y1/y0),z),z);
      r := z → (-1 + (α - 1) z) / (-1 + z α)

```

A-estabilidad para $\alpha = 2$

En general, como hemos visto

$$r(z) = \frac{1 + (1 - \alpha)z}{1 - \alpha z}$$

y, para $\alpha = 2$,

$$r(z) = \frac{1 - z}{1 - 2z}$$

Pondremos que $z=x+iy$.

Como

$$1 - z = 1 - x - iy$$

y

$$1 - 2z = 1 - 2x - 2iy,$$

resulta

$$|1 - z|^2 = (1 - x)^2 + y^2$$

y

$$|1 - 2z|^2 = (1 - 2x)^2 + 4y^2$$

por lo que la curva $|r(z)|^2 = 1$ se escribe

$$(1 - x)^2 + y^2 = (1 - 2x)^2 + 4y^2$$

o sea

$$3x^2 - 2x + 3y^2 = 0$$

$$x^2 - \frac{2}{3}x + y^2 = 0$$

La curva es

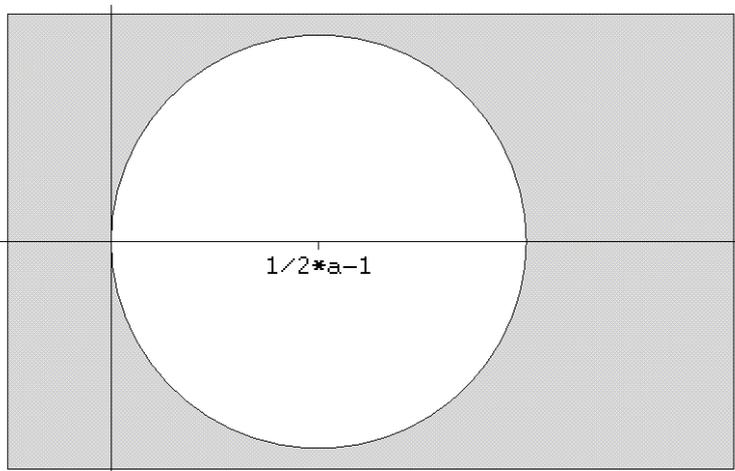
$$\left(x - \frac{1}{3}\right)^2 + y^2 = \frac{1}{9}$$

o sea, una circunferencia de centro $\frac{1}{3} = \left(\frac{1}{3}, 0\right)$ y radio $\frac{1}{3}$ que pasa por $(0,0)$

Como $z=-1=(-1,0)$ verifica $|r(z)| = \left|\frac{2}{3}\right| < 1$ resulta que el dominio de estabilidad

es el exterior de la circunferencia y que el dominio de estabilidad contiene (sobradamente)

al semiplano de parte real negativa; por lo tanto, el método es A-estable



Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-28 (del EXAMEN ORDINARIO DE 24ENE11)

Considérese la familia de métodos implícitos de RUNGE-KUTTA con dos etapas

$$\begin{array}{c|cc} \alpha & \alpha - 1 & 1 \\ 1 & \beta & 0 \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

- Búsquese el mayor orden alcanzable por los miembros de esta familia, indicando los valores de α y de β para los que se consigue (hasta aquí es Ejercicio del capítulo 7)
- Búsquese un método de la familia del enunciado que sea A-estable
- Estúdiense la A-estabilidad de los métodos de orden máximo obtenidos antes

```
[ > restart:
[ > interface(labeling=false):
[ # desarrollo de la verdadera solucion
[ > #y:=unapply(y(x),x):
[ > #f:=unapply(f(x,y),x,y):
[ > f0:=f(x,y(x)):
[ > f1:=subs(diff(y(x),x)=f(x,y(x)),diff(f0,x)):
[ > f2:=subs(diff(y(x),x)=f(x,y(x)),diff(f1,x)):
[ > f3:=subs(diff(y(x),x)=f(x,y(x)),diff(f2,x)):
[ > f0:=subs({y(x)=y0,x=x0},f0):f0:=expand(f0):
[ > f1:=subs({y(x)=y0,x=x0},f1):f1:=expand(f1):
[ > f2:=subs({y(x)=y0,x=x0},f2):f2:=expand(f2):
[ > f3:=subs({y(x)=y0,x=x0},f3):f3:=expand(f3):
[ # desarrollo del método
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp10(h)+h*kp20(h)),h):
[ > k20:=unapply(f(x0+h,y0+beta*h*kp10(h)),h):
[ > k100:=k10(0):k100:=expand(k100):
[ > k200:=k20(0):k200:=expand(k200):
[ > F10:=expand((1/4)*k100+(3/4)*k200):
[ > k11:=D(k10):
[ > k21:=D(k20):
[ > k110:=subs({kp10(0)=k100,kp20(0)=k200},k11(0)):k110:=expand(k110):
[ > k210:=subs({kp10(0)=k100,kp20(0)=k200},k21(0)):k210:=expand(k210):
[ > F11:=expand((1/4)*k110+(3/4)*k210):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k12(0)):k120:=e
x
pand(k120):
[ > k220:=subs({D(kp10)(0)=k110,D(kp20)(0)=k210,kp10(0)=k100,kp20(0)=k200},k22(0)):k220:=e
x
pand(k220):
[ > F12:=expand((1/4)*k120+(3/4)*k220):
[ > k13:=D(k12):
[
```

```

[ > k23:=D(k22):
[ > k130:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k13(0)):k130:=expand(k130):
[ > k230:=subs({(D@@2)(kp10)(0)=k120,(D@@2)(kp20)(0)=k220,D(kp10)(0)=k110,D(kp20)(0)=k210,
  kp10(0)=k100,kp20(0)=k200},k23(0)):k230:=expand(k230):
[ > Fi3:=expand((1/4)*k130+(3/4)*k230):
[ # consistencia y orden 1
[ > cond1a:=coeff(Fi0,f(x0,y0))=coeff(f0,f(x0,y0));
                                     cond1a := 1 = 1
[ # condiciones para el orden 2
[ > cond2a:=coeff(Fi1,D[1](f)(x0,y0))=coeff((1/2)*f1,D[1](f)(x0,y0));
                                     cond2a :=  $\frac{\alpha}{4} + \frac{3}{4} = \frac{1}{2}$ 
[ > cond2b:=coeff(coeff(Fi1,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/2)*f1,D[2](f)(x0,y0))
  ,f(x0,y0));
                                     cond2b :=  $\frac{\alpha}{4} + \frac{3\beta}{4} = \frac{1}{2}$ 
[ > orden2:=solve({cond2a, cond2b});
                                     orden2 := { $\alpha = -1, \beta = 1$ }
[ > alpha:=subs(orden2,alpha);beta:=subs(orden2,beta);
                                      $\alpha := -1$ 
                                      $\beta := 1$ 
[ # condiciones para el orden 3
[ > cond3a:=coeff(Fi2,D[1,1](f)(x0,y0))=coeff((1/3)*f2,D[1,1](f)(x0,y0));
                                     cond3a :=  $1 = \frac{1}{3}$ 
[ > cond3b:=coeff(coeff(Fi2,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff((1/3)*f2,D[1,2](f)(x0,
  y0)),f(x0,y0));
                                     cond3b :=  $2 = \frac{2}{3}$ 
[ > cond3c:=coeff(coeff(Fi2,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff((1/3)*f2,D[2,2](f)(x
  0,y0)),f(x0,y0)^2);
                                     cond3c :=  $1 = \frac{1}{3}$ 
[ > cond3d:=coeff(coeff(Fi2,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x
  0,y0)),D[1](f)(x0,y0));
                                     cond3d :=  $0 = \frac{1}{3}$ 
[ > cond3e:=coeff(coeff(Fi2,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff((1/3)*f2,D[2](f)(x0,y0)
  )^2,f(x0,y0));
                                     cond3e :=  $0 = \frac{1}{3}$ 

```

No se puede alcanzar orden 3 ; basta ver la primera de las condiciones,

que es imposible. El máximo orden es 2 con $\alpha = -1$ y $\beta = 1$

Se obtiene orden máximo 3 de una sola manera, para los valores $\alpha = -1$ y $\beta = 1$, o sea

para el método

$$\begin{array}{c|cc}
 -1 & -2 & 1 \\
 1 & 1 & 0 \\
 \hline
 & \frac{1}{4} & \frac{3}{4}
 \end{array}$$

```

[ # funcion de estabilidad para  $\alpha=1$  y  $\beta=1$  (solo con orden 1 )
[ > restart:
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[                                     f := (x, y) → λ y
[ > alpha:=1;beta:=1;
[                                     α := 1
[                                     β := 1
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                     k10 := h → λ (y0 + h kp2(h))
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                     k20 := h → λ (y0 + h kp1(h))
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                     ka1 := h → λ (y0 + h kp2(h))
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                     ka2 := h → λ (y0 + h kp1(h))
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                                     test := {k2 = - $\frac{\lambda y0}{\lambda h - 1}$ , k1 = - $\frac{\lambda y0}{\lambda h - 1}$ }
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
[                                     k1 := - $\frac{\lambda y0}{\lambda h - 1}$ 
[                                     k2 := - $\frac{\lambda y0}{\lambda h - 1}$ 
[                                     - $\frac{4 \lambda y0}{\lambda h - 1}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                     y1 := y0 -  $\frac{h \lambda y0}{\lambda h - 1}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 :=  $\left(1 - \frac{z}{z-1}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
[                                     r_particular := - $\frac{1}{z-1}$ 
[ >
[ >
[ >
[ >
[ # para estos valores alpha = 1 y beta = 1 el método es evidentemente A-estable
[
[ # funcion de estabilidad para  $\alpha=-1$  y  $\beta=1$  (orden 2 máximo)
[ > restart:
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[                                     f := (x, y) → λ y
[ > alpha:=-1;beta:=1;

```

```

[                                      $\alpha := -1$ 
[                                      $\beta := 1$ 
[ >
[ >
[ > k10:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                      $k10 := h \rightarrow \lambda (y0 - 2 h kp1(h) + h kp2(h))$ 
[ > k20:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                      $k20 := h \rightarrow \lambda (y0 + h kp1(h))$ 
[ >
[ >
[ >
[ > ka1:=unapply(f(x0+alpha*h,y0+(alpha-1)*h*kp1(h)+h*kp2(h)),h);
[                                      $ka1 := h \rightarrow \lambda (y0 - 2 h kp1(h) + h kp2(h))$ 
[ > ka2:=unapply(f(x0+beta*h,y0+beta*h*kp1(h)),h);
[                                      $ka2 := h \rightarrow \lambda (y0 + h kp1(h))$ 
[ > test:=solve({k1=subs({kp1(h)=k1,kp2(h)=k2},ka1(h)),k2=subs({kp1(h)=k1,kp2(h)=k2},ka2(h))},{k1,k2});
[                                      $test := \{k1 = -\frac{\lambda y0 (1 + \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}, k2 = -\frac{\lambda y0 (1 + 3 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}\}$ 
[ > k1:=subs(test,k1);k2:=subs(test,k2);simplify(k1+3*k2);
[                                      $k1 := -\frac{\lambda y0 (1 + \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[                                      $k2 := -\frac{\lambda y0 (1 + 3 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[                                      $-\frac{2 \lambda y0 (2 + 5 \lambda h)}{-1 - 2 \lambda h + \lambda^2 h^2}$ 
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                      $y1 := y0 - \frac{h \lambda y0}{-1 - 2 \lambda h + \lambda^2 h^2} - \frac{5 \lambda^2 y0 h^2}{2 (-1 - 2 \lambda h + \lambda^2 h^2)}$ 
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                      $y1 := \left(1 - \frac{z}{-1 - 2 z + z^2} - \frac{5 z^2}{2 (-1 - 2 z + z^2)}\right) y0$ 
[ > r:=unapply(collect(simplify(y1/y0),z),z):r_particular:=r(z);
[                                      $r\_particular := -\frac{2 + 6 z + 3 z^2}{2 (-1 - 2 z + z^2)}$ 
[ # para estos valores alpha = -1 y beta = 1 el método no es A-estable,
[ ya que r(z) tiende a 3/2 cuando z tiende a  $\infty$ 

```

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-29 (del EXAMEN EXTRAORDINARIO de 07JUL12)

Para la familia de métodos implícitos de RUNGE-KUTTA de dos etapas de tablero

α		α	0
$\frac{2}{3}$		$\frac{1}{3}$	$\frac{1}{3}$

		$\frac{1}{4}$	$\frac{3}{4}$

calcúlese el orden máximo de consistencia y dígase si los casos de orden máximo son o no A-estables. Para el método con $\alpha = 2/3$, dígase qué orden posee y si es o no A-estable

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+alpha*h,y0+alpha*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+2/3*h,y0+1/3*h*kp1_(h)+1/3*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
[ condiciones para el orden 1
[ > Fi_0:=expand((1/4)*k1_0+(3/4)*k2_0);
[                                     Fi_0 := f(x0, y0)
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[                                     cond1a := 1 = 1
[ El orden es de 1 al menos
[ condiciones para el orden 2
[ > Fi10:=expand((1/4)*k110+(3/4)*k210);
[                                     Fi10 := 1/4 D1(f)(x0, y0) + 1/4 D2(f)(x0, y0) + f(x0, y0) + 1/2 D1(f)(x0, y0) + 1/2 D2(f)(x0, y0) f(x0, y0)
```

```

> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
cond2a := 1/2 = alpha/4 + 1/2
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
cond2b := 1/2 = alpha/4 + 1/2
> orden2:=solve({cond2a,cond2b},alpha);
orden2 := {alpha = 0}
> alpha:=subs(orden2,alpha);
alpha := 0

```

El orden es de 2 al menos para $\alpha = 0$

condiciones para el orden 3

```

> Fi20:=expand((1/4)*k120+(3/4)*k220);
Fi20 := 1/3 D1,1(f)(x0,y0) + 2/3 D1,2(f)(x0,y0) f(x0,y0) + 1/3 D2,2(f)(x0,y0) f(x0,y0)^2 + 1/3 D2(f)(x0,y0) D1(f)(x0,y0)
+ 1/3 D2(f)(x0,y0)^2 f(x0,y0)
> cond3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))=coeff(Fi20,D[1,1](f)(x0,y0));
cond3a := 1/3 = 1/3
> cond3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));
cond3b := 2/3 = 2/3
> cond3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);
cond3c := 1/3 = 1/3
> cond3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));
cond3d := 1/3 = 1/3
> cond3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));
cond3e := 1/3 = 1/3

```

El orden es de 3 al menos para $\alpha = 0$

orden 4

El método con $\alpha=0$ no es el método de GAUSS de dos etapas.

Por lo tanto, no se alcanza el orden 4

O sea, el orden máximo de la familia es 3 para $\alpha=0$

función de estabilidad para α general

```

> restart;
> interface(labeling=false);
> f:=(x,y)->lambda*y;
f := (x, y) -> lambda y
> ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
ka1 := h -> lambda (y0 + alpha h kp1(h))
> test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
test := {k1 = - lambda y0 / (-1 + lambda alpha h)}
> k1:=subs(test,k1);
k1 := - lambda y0 / (-1 + lambda alpha h)

```

```

> ka2:=unapply(f(x0+(2/3)*h,y0+(1/3)*h*k1+(1/3)*h*kp2(h)),h);
      ka2 := h → λ (y0 - 1/3 * h λ y0 / (-1 + λ α h) + 1/3 * h kp2(h))
> test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);
      test := {k2 = - λ y0 (-3 + 3 λ α h - λ h) / ((-1 + λ α h) (-3 + λ h))}
> k2:=subs(test,k2);
      k2 := - λ y0 (-3 + 3 λ α h - λ h) / ((-1 + λ α h) (-3 + λ h))
> y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
      y1 := y0 - h λ y0 / (4(-1 + λ α h)) + 9 h λ y0 / (4(-1 + λ α h)(-3 + λ h)) - 9 λ² y0 α h² / (4(-1 + λ α h)(-3 + λ h)) + 3 λ² y0 h² / (4(-1 + λ α h)(-3 + λ h))
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 - z / (4(-1 + z α)) + 9 z / (4(-1 + z α)(-3 + z)) - 9 z² α / (4(-1 + z α)(-3 + z)) + 3 z² / (4(-1 + z α)(-3 + z))) y0
> r:=unapply(collect(simplify(y1/y0),z),z);
      r := z → -1/4 * (-12 + (5 α - 2) z² + (-8 + 12 α) z) / ((-1 + z α)(-3 + z))

```

función de estabilidad para $\alpha = 0$ (orden maximo)

```

> restart;
> alpha:=0;
      α := 0
> interface(labeling=false);
> f:=(x,y)->lambda*y;
      f := (x, y) → λ y
> ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
      ka1 := h → λ y0
> test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
      test := {k1 = λ y0}
> k1:=subs(test,k1);
      k1 := λ y0
> ka2:=unapply(f(x0+(2/3)*h,y0+(1/3)*h*k1+(1/3)*h*kp2(h)),h);
      ka2 := h → λ (y0 + 1/3 * h λ y0 + 1/3 * h kp2(h))
> test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);
      test := {k2 = - λ y0 (3 + λ h) / (-3 + λ h)}
> k2:=subs(test,k2);
      k2 := - λ y0 (3 + λ h) / (-3 + λ h)
> y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
      y1 := y0 + h λ y0 / 4 - 9 h λ y0 / (4(-3 + λ h)) - 3 λ² y0 h² / (4(-3 + λ h))
> y1:=collect(subs(lambda=z/h,y1),y0);
      y1 := (1 + z/4 - 9 z / (4(-3 + z)) - 3 z² / (4(-3 + z))) y0
> r:=unapply(collect(simplify(y1/y0),z),z);
      r := z → -1/2 * (6 + 4 z + z²) / (-3 + z)

```

Lo que hace que el método para $\alpha = 0$ (orden maximo) no sea A-estable.

función de estabilidad para $\alpha = 2/3$

```

> restart;
> alpha:=2/3;

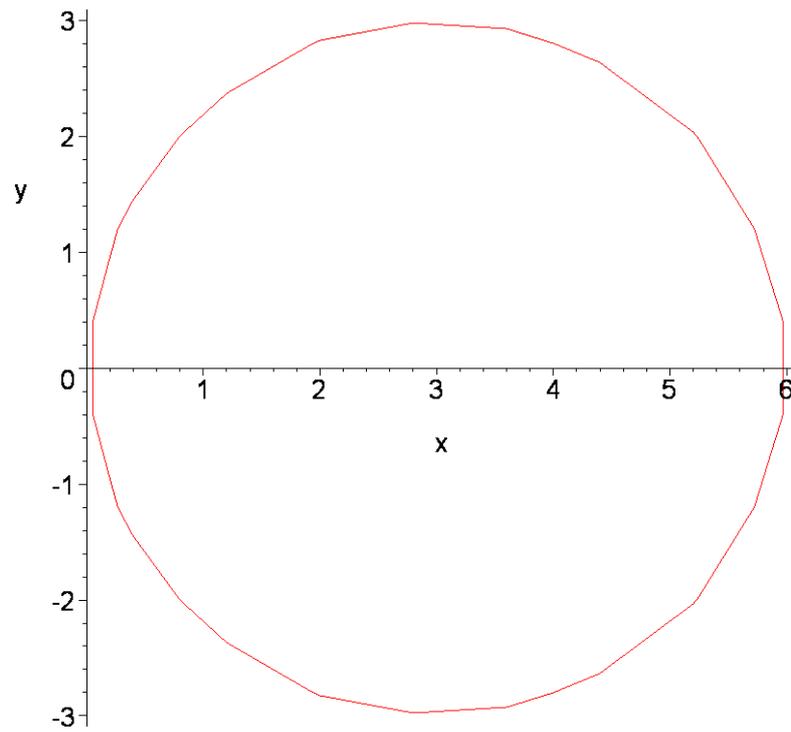
```

$$\alpha := \frac{2}{3}$$

```

[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[ > ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
[ > ka1 := h -> lambda * (y0 + (2/3) * h * kp1(h))
[ > test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
[ > test := {k1 = - (3 * lambda * y0) / (-3 + 2 * lambda * h)}
[ > k1:=subs(test,k1);
[ > k1 := - (3 * lambda * y0) / (-3 + 2 * lambda * h)
[ > ka2:=unapply(f(x0+(2/3)*h,y0+(1/3)*h*k1+(1/3)*h*kp2(h)),h);
[ > ka2 := h -> lambda * (y0 - (h * lambda * y0) / (-3 + 2 * lambda * h) + (1/3) * h * kp2(h))
[ > test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);
[ > test := {k2 = - (3 * lambda * y0) / (-3 + 2 * lambda * h)}
[ > k2:=subs(test,k2);
[ > k2 := - (3 * lambda * y0) / (-3 + 2 * lambda * h)
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[ > y1 := y0 - (3 * h * lambda * y0) / (-3 + 2 * lambda * h)
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[ > y1 := (1 - (3 * z) / (-3 + 2 * z)) * y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[ > r := z -> - (3 + z) / (-3 + 2 * z)
[ > curva:=(3+x)^2+y^2-(-3+2*x)^2-4*y^2=0;
[ > curva := (3 + x)^2 - 3 * y^2 - (-3 + 2 * x)^2 = 0
[ > simplify(curva);
[ > 18 * x - 3 * x^2 - 3 * y^2 = 0
[ > with(plots):
[ > implicitplot(curva,x=-10..10,y=-10..10);

```



[> `r(-1);`

$\frac{2}{5}$

[O sea, la region de estabilidad es el exterior de
 [la circunferencia de centro 3 y radio 3 y, para $\alpha = 2/3$,
 [el metodo es A-estable

[>

[>

Ejercicios de

(8) Estabilidad lineal

Ejercicio 08-30 (del EXAMEN ORDINARIO de 01FEB13)

De entre los métodos implícitos de RUNGE-KUTTA de dos etapas de tablero

$$\begin{array}{c|cc} \alpha & \alpha & 0 \\ \beta & \delta & 3\delta \end{array}$$

$$\begin{array}{c|cc} & \frac{1}{4} & \frac{3}{4} \end{array}$$

se buscarán los que sean de mayor orden y se dirá para qué valores de α , β y δ se alcanzan.

Se responderá a la misma pregunta en el caso $\alpha = 0$ (naturalmente con β y δ arbitrarios), caso del que nos ocupamos en lo que sigue.

En esta última situación $\alpha = 0$, se calculará la función de estabilidad de la familia de los métodos del tablero y se buscará en ella un método

A-estable y otro que no lo sea, razonando por qué es así.

```
[ > restart:
[ > interface(labeling=false):
[ > y1:=f(x,y(x)):
[ > y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
[ > y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
[ > Y1:=subs({y(x)=y0,x=x0},y1):
[ > Y2:=subs({y(x)=y0,x=x0},y2):
[ > Y3:=subs({y(x)=y0,x=x0},y3):
[ > k1_:=unapply(f(x0+alpha*h,y0+alpha*h*kp1_(h)),h):
[ > k2_:=unapply(f(x0+beta*h,y0+delta*h*kp1_(h)+3*delta*h*kp2_(h)),h):
[ > k1_0:=k1_(0):
[ > k2_0:=k2_(0):
[ > k11:=D(k1_):
[ > k21:=D(k2_):
[ > k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
[ > k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
[ > k12:=D(k11):
[ > k22:=D(k21):
[ > k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
[ > k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
[ condiciones para el orden 1
[ > Fi_0:=expand((1/4)*k1_0+(3/4)*k2_0);
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
[ > cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
```

cond1a := 1 = 1

El orden es de 1 al menos

condiciones para el orden 2

> Fi10:=expand((1/4)*k110+(3/4)*k210);

$$Fi10 := \frac{1}{4} D_1(f)(x0, y0) \alpha + \frac{1}{4} D_2(f)(x0, y0) \alpha f(x0, y0) + \frac{3}{4} D_1(f)(x0, y0) \beta + 3 D_2(f)(x0, y0) \delta f(x0, y0)$$

> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));

$$cond2a := \frac{1}{2} = \frac{\alpha}{4} + \frac{3\beta}{4}$$

> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));

$$cond2b := \frac{1}{2} = \frac{\alpha}{4} + 3\delta$$

> orden2:=solve({cond2a,cond2b},{alpha,beta});

$$orden2 := \{ \alpha = 2 - 12\delta, \beta = 4\delta \}$$

> alpha:=subs(orden2,alpha);beta:=subs(orden2,beta);

$$\alpha := 2 - 12\delta$$

$$\beta := 4\delta$$

El orden es de al menos 2 para $\alpha = 2 - 12\delta$ y $\beta = 4\delta$

condiciones para el orden 3

> Fi20:=expand((1/4)*k120+(3/4)*k220);

$$Fi20 := D_{1,1}(f)(x0, y0) - 12 D_{1,1}(f)(x0, y0) \delta + 48 D_{1,1}(f)(x0, y0) \delta^2 + 2 D_{1,2}(f)(x0, y0) f(x0, y0) - 24 D_{1,2}(f)(x0, y0) \delta f(x0, y0) + 96 D_{1,2}(f)(x0, y0) \delta^2 f(x0, y0) + D_{2,2}(f)(x0, y0) f(x0, y0)^2 - 12 D_{2,2}(f)(x0, y0) \delta f(x0, y0)^2 + 48 D_{2,2}(f)(x0, y0) \delta^2 f(x0, y0)^2 + 2 D_2(f)(x0, y0) D_1(f)(x0, y0) - 21 D_2(f)(x0, y0) D_1(f)(x0, y0) \delta + 2 D_2(f)(x0, y0)^2 f(x0, y0) - 21 D_2(f)(x0, y0)^2 \delta f(x0, y0) + 72 D_2(f)(x0, y0) D_1(f)(x0, y0) \delta^2 + 72 D_2(f)(x0, y0)^2 \delta^2 f(x0, y0)$$

> cond3a:=(1/3)*coeff(Y3,D[1,1](f)(x0,y0))=coeff(Fi20,D[1,1](f)(x0,y0));

$$cond3a := \frac{1}{3} = 1 - 12\delta + 48\delta^2$$

> cond3b:=(1/3)*coeff(coeff(Y3,D[1,2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi20,D[1,2](f)(x0,y0)),f(x0,y0));

$$cond3b := \frac{2}{3} = 2 - 24\delta + 96\delta^2$$

> cond3c:=(1/3)*coeff(coeff(Y3,D[2,2](f)(x0,y0)),f(x0,y0)^2)=coeff(coeff(Fi20,D[2,2](f)(x0,y0)),f(x0,y0)^2);

$$cond3c := \frac{1}{3} = 1 - 12\delta + 48\delta^2$$

> cond3d:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)),D[1](f)(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)),D[1](f)(x0,y0));

$$cond3d := \frac{1}{3} = 2 - 21\delta + 72\delta^2$$

> cond3e:=(1/3)*coeff(coeff(Y3,D[2](f)(x0,y0)^2),f(x0,y0))=coeff(coeff(Fi20,D[2](f)(x0,y0)^2),f(x0,y0));

$$cond3e := \frac{1}{3} = 2 - 21\delta + 72\delta^2$$

> solve({cond3a,cond3b,cond3c});solve({cond3d,cond3e});orden3:=solve({cond3a,cond3b,cond3c,cond3d,cond3e});

$$\{ \delta = \frac{1}{12}, \{ \delta = \frac{1}{6} \}$$

$$\{ \delta = \frac{7}{48} - \frac{1}{144} I \sqrt{39}, \{ \delta = \frac{7}{48} + \frac{1}{144} I \sqrt{39} \}$$

orden3 :=

No se alcanza orden e 3

O sea el mayor orden que se alcanza es 2 en la forma indicada

Orden en el caso a=0

```
> restart;
> interface(labeling=false):
> alpha:=0;
                                     alpha := 0
> y1:=f(x,y(x)):
> y2:=subs(diff(y(x),x)=y1,diff(y1,x)):
> y3:=subs(diff(y(x),x)=y1,diff(y2,x)):
> Y1:=subs({y(x)=y0,x=x0},y1):
> Y2:=subs({y(x)=y0,x=x0},y2):
> Y3:=subs({y(x)=y0,x=x0},y3):
> k1_:=unapply(f(x0+alpha*h,y0+alpha*h*kp1_(h)),h):
> k2_:=unapply(f(x0+beta*h,y0+delta*h*kp1_(h)+3*delta*h*kp2_(h)),h):
> k1_0:=k1_(0):
> k2_0:=k2_(0):
> k11:=D(k1_):
> k21:=D(k2_):
> k110:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k11(0)):
> k210:=subs({kp1_(0)=k1_0,kp2_(0)=k2_0},k21(0)):
> k12:=D(k11):
> k22:=D(k21):
> k120:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k12(0)):
> k220:=subs({D(kp1_(0))=k110,D(kp2_(0))=k210,kp1_(0)=k1_0,kp2_(0)=k2_0},k22(0)):
```

condiciones para el orden 1

```
> Fi_0:=expand((1/4)*k1_0+(3/4)*k2_0);
                                     Fi_0 := f(x0, y0)
> cond1a:=coeff(Y1,f(x0,y0))=coeff(Fi_0,f(x0,y0));
                                     cond1a := 1 = 1
```

El orden es de 1 al menos

condiciones para el orden 2

```
> Fi10:=expand((1/4)*k110+(3/4)*k210);
                                     Fi10 :=  $\frac{3}{4} D_1(f)(x0, y0) \beta + 3 D_2(f)(x0, y0) \delta f(x0, y0)$ 
> cond2a:=(1/2)*coeff(Y2,D[1](f)(x0,y0))=coeff(Fi10,D[1](f)(x0,y0));
                                     cond2a :=  $\frac{1}{2} = \frac{3 \beta}{4}$ 
> cond2b:=(1/2)*coeff(coeff(Y2,D[2](f)(x0,y0)),f(x0,y0))=coeff(coeff(Fi10,D[2](f)(x0,y0)),f(x0,y0));
                                     cond2b :=  $\frac{1}{2} = 3 \delta$ 
> orden2:=solve({cond2a,cond2b});
                                     orden2 :=  $\{\beta = \frac{2}{3}, \delta = \frac{1}{6}\}$ 
> delta:=subs(orden2,delta);beta:=subs(orden2,beta);
                                     delta :=  $\frac{1}{6}$ 
                                     beta :=  $\frac{2}{3}$ 
```

El orden maximo sigue siendo 2 ahora con b = 2/3 y d = 1/6

Naturalmente, es sencillo ver que con a = 0 y estos valores,

se alcanza orden 2, que será también el máximo en este caso

[función de estabilidad en el caso $\alpha = 0$

```
[ > restart;
[ > alpha:=0;
[                                     alpha := 0
[ > interface(labeling=false):
[ > f:=(x,y)->lambda*y;
[                                     f := (x, y) -> lambda y
[ > ka1:=unapply(f(x0+alpha*h,y0+alpha*h*kp1(h)),h);
[                                     ka1 := h -> lambda y0
[ > test:=solve({k1=subs(kp1(h)=k1,ka1(h))},k1);
[                                     test := {k1 = lambda y0}
[ > k1:=subs(test,k1);
[                                     k1 := lambda y0
[ > ka2:=unapply(f(x0+beta*h,y0+delta*h*k1+3*delta*h*kp2(h)),h);
[                                     ka2 := h -> lambda (y0 + delta h lambda y0 + 3 delta h kp2(h))
[ > test:=solve({k2=subs(kp2(h)=k2,ka2(h))},k2);
[                                     test := {k2 = -frac(lambda y0 (1 + lambda delta h), -1 + 3 lambda delta h)}
[ > k2:=subs(test,k2);
[                                     k2 := -frac(lambda y0 (1 + lambda delta h), -1 + 3 lambda delta h)
[ > y1:=expand(y0+h*((1/4)*k1+(3/4)*k2));
[                                     y1 := y0 + h lambda y0 / 4 - 3 h lambda y0 / (4 (-1 + 3 lambda delta h)) - 3 lambda^2 y0 delta h^2 / (4 (-1 + 3 lambda delta h))
[ > y1:=collect(subs(lambda=z/h,y1),y0);
[                                     y1 := (1 + z/4 - 3 z / (4 (-1 + 3 z delta)) - 3 z^2 delta / (4 (-1 + 3 z delta))) y0
[ > r:=unapply(collect(simplify(y1/y0),z),z);
[                                     r := z -> (-1 + (3 delta - 1) z) / (-1 + 3 z delta)
[ > r := z -> (-1+(3*delta-1)*z)/(-1+3*z*delta);
```

[Caso A-estable

[función de estabilidad para $\alpha = 0$ y $\delta = \frac{1}{3}$

```
[ > r1(z):=subs({delta=1/3},r(z));
[                                     r1(z) := -frac(1, -1+z)
```

[que proporciona un caso A-estable bien conocido

[Caso no A-estable

[función de estabilidad para $\alpha = 0$ y $\delta = 0$

```
[ > r1(z):=subs({delta=0},r(z));
[                                     r1(z) := 1+z
```

[La función de estabilidad, un polinomio, hace que el método no pueda ser A-estable

[>

09. Estabilidad no lineal

- 1 Relación entre estabilidad no lineal y estabilidad lineal
- 2 Ejemplos

La estabilidad no lineal, al contrario de lo que acabamos de ver (la lineal) es un problema que no admite muchas afirmaciones cerradas. En casi todas las consideraciones, los contraejemplos acompañan a los ejemplos de lo que se consideraría 'situación normal'.

Por eso nos vamos a limitar a exponer algunas ideas y ejemplos que permitan alcanzar algo de la complejidad de este tema. El punto de partida general es la consideración de un sistema autónomo no lineal

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}),$$

o en términos más concretos (para $m = 2$ como otras veces)

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{bmatrix} {}^1f({}^1y, {}^2y) \\ {}^2f({}^1y, {}^2y) \end{bmatrix}$$

Una idea que 'acerca' este sistema a uno lineal y de coeficientes constantes es ensayar la 'primera aproximación' de \mathbf{f} . Se trata de aproximar \mathbf{f} por una parte de su desarrollo en serie: hasta el término en la derivada de \mathbf{f} . Para ello se elige un vector arbitrario \mathbf{y}_0 (posiblemente el valor de la solución $\mathbf{y}_0 = \mathbf{y}(x_0)$ para un x_0 determinado) y se hace la aproximación

$$\mathbf{f}(\mathbf{y}) = \mathbf{f}(\mathbf{y}_0) + \mathbf{f}_y(\mathbf{y}_0)(\mathbf{y} - \mathbf{y}_0).$$

Aquí el símbolo $\mathbf{f}_y(\mathbf{y}_0)$ representa la matriz jacobiana

$$\begin{pmatrix} {}^1f_1({}^1y_0, {}^2y_0) & {}^1f_2({}^1y_0, {}^2y_0) \\ {}^2f_1({}^1y_0, {}^2y_0) & {}^2f_2({}^1y_0, {}^2y_0) \end{pmatrix}$$

de la función vectorial \mathbf{f} respecto de su variable, también vectorial, \mathbf{y} , evaluada en \mathbf{y}_0 , matriz que aparece multiplicada por el vector $\mathbf{y} - \mathbf{y}_0$. Luego el sistema aproximado es

$$\mathbf{y}' = \mathbf{f}_y(\mathbf{y}_0) \mathbf{y} - \mathbf{f}_y(\mathbf{y}_0) \mathbf{y}_0 + \mathbf{f}(\mathbf{y}_0),$$

o, detallado en sus componentes escalares,

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} {}^1f_1({}^1y_0, {}^2y_0) & {}^1f_2({}^1y_0, {}^2y_0) \\ {}^2f_1({}^1y_0, {}^2y_0) & {}^2f_2({}^1y_0, {}^2y_0) \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix} \\ - \begin{pmatrix} {}^1f_1({}^1y_0, {}^2y_0) & {}^1f_2({}^1y_0, {}^2y_0) \\ {}^2f_1({}^1y_0, {}^2y_0) & {}^2f_2({}^1y_0, {}^2y_0) \end{pmatrix} \begin{bmatrix} {}^1y_0 \\ {}^2y_0 \end{bmatrix} + \begin{bmatrix} {}^1f({}^1y_0, {}^2y_0) \\ {}^2f({}^1y_0, {}^2y_0) \end{bmatrix}$$

Este sistema es lineal y de coeficientes constantes, pero no es homogéneo. Su comportamiento de cara a la estabilidad lineal coincide con el comportamiento de

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \mathbf{f}_y(\mathbf{y}_0)(\mathbf{y}) = \begin{pmatrix} {}^1f_1({}^1y_0, {}^2y_0) & {}^1f_2({}^1y_0, {}^2y_0) \\ {}^2f_1({}^1y_0, {}^2y_0) & {}^2f_2({}^1y_0, {}^2y_0) \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

que es homogéneo y de coeficientes constantes y sabemos como se estudia.

Pero lo que no es necesariamente igual es el comportamiento de la estabilidad para el sistema original y del sistema linealizado. Sigue un ejemplo que trata de aclararlo.



Ejemplo de aproximación

Consideremos el sistema bidimensional

$$\begin{cases} {}^1y' = {}^1y {}^2y \\ {}^2y' = {}^1y - {}^2y \end{cases}$$

Aquí el vector \mathbf{y} es

$$\mathbf{y} = \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

La función \mathbf{f} es

$$\mathbf{f}({}^1y, {}^2y) = \begin{bmatrix} {}^1y {}^2y \\ {}^1y - {}^2y \end{bmatrix}$$

Consideremos el punto $({}^1y_0, {}^2y_0) = (0, 0)$, que es un punto crítico (o sea, en el que \mathbf{f} vale 0). La matriz jacobiana vale

$$\mathbf{f}_y(\mathbf{y}) = \begin{pmatrix} {}^2y & {}^1y \\ 1 & -1 \end{pmatrix}$$



y en $\mathbf{0}$ vale

$$\mathbf{f}_y(\mathbf{0}) = \begin{pmatrix} 0 & 0 \\ 1 & -1 \end{pmatrix}$$

Entonces la aproximación lineal de \mathbf{f} alrededor de $\mathbf{0}$ es

$$\begin{aligned} \begin{bmatrix} {}^1f({}^1y, {}^2y) \\ {}^2f({}^1y, {}^2y) \end{bmatrix} &= \mathbf{f}(\mathbf{y}) \sim \mathbf{f}(\mathbf{0}) + \mathbf{f}_y(\mathbf{0})(\mathbf{y} - \mathbf{0}) \\ &= \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{pmatrix} 0 & 0 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{bmatrix} 0 \\ {}^1y - {}^2y \end{bmatrix} \end{aligned}$$

El sistema lineal aproximación lineal del original (alrededor de $({}^1y_0, {}^2y_0) = (0, 0)$), es

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix} = \begin{bmatrix} 0 \\ {}^1y - {}^2y \end{bmatrix}$$

con autovalores 0 y -1 .



Si, en lugar del punto $({}^1y_0, {}^2y_0) = (0, 0)$, tomamos $({}^1y_0, {}^2y_0) = (-2, 1)$ (ahora no se trata de un punto crítico de \mathbf{f}) entonces

$$\mathbf{f}_y(\mathbf{y}_0) = \begin{pmatrix} 1 & -2 \\ 1 & -1 \end{pmatrix}$$

y la aproximación lineal de \mathbf{f} alrededor de \mathbf{y}_0 es

$$\begin{aligned} \begin{bmatrix} {}^1f({}^1y, {}^2y) \\ {}^2f({}^1y, {}^2y) \end{bmatrix} &= \mathbf{f}(\mathbf{y}) \sim \mathbf{f}(\mathbf{y}_0) + \mathbf{f}_y(\mathbf{y}_0)(\mathbf{y} - \mathbf{y}_0) \\ &= \begin{bmatrix} -2 \\ -3 \end{bmatrix} + \begin{pmatrix} 1 & -2 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} {}^1y + 2 \\ {}^2y - 1 \end{bmatrix} = \begin{bmatrix} {}^1y - 2{}^2y + 2 \\ {}^1y - {}^2y \end{bmatrix} \end{aligned}$$

El sistema lineal aproximación lineal del original (alrededor de $({}^1y_0, {}^2y_0) = (-2, 1)$), es

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{bmatrix} {}^1y - 2{}^2y + 2 \\ {}^1y - {}^2y \end{bmatrix} = \begin{pmatrix} 1 & -2 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix} + \begin{bmatrix} 2 \\ 0 \end{bmatrix}$$

con autovalores i y $-i$ en este caso.



Ahora tomamos $(^1y_0 \ ^2y_0) = (-1, 1)$ (tampoco es un punto crítico de \mathbf{f}); entonces

$$\mathbf{f}_y(\mathbf{y}_0) = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix}$$

y la aproximación lineal de \mathbf{f} alrededor de \mathbf{y}_0 es

$$\begin{aligned} \begin{bmatrix} ^1f(^1y, ^2y) \\ ^2f(^1y, ^2y) \end{bmatrix} &= \mathbf{f}(\mathbf{y}) \sim \mathbf{f}(\mathbf{y}_0) + \mathbf{f}_y(\mathbf{y}_0)(\mathbf{y} - \mathbf{y}_0) \\ &= \begin{bmatrix} -1 \\ -2 \end{bmatrix} + \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} ^1y + 1 \\ ^2y - 1 \end{bmatrix} = \begin{bmatrix} ^1y - ^2y + 1 \\ ^1y - ^2y \end{bmatrix} \end{aligned}$$

El sistema lineal aproximación lineal del original (alrededor de $(^1y_0 \ ^2y_0) = (1, 1)$), es

$$\begin{bmatrix} ^1y' \\ ^2y' \end{bmatrix} = \begin{bmatrix} ^1y - ^2y + 1 \\ ^1y - ^2y \end{bmatrix} = \begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \begin{bmatrix} ^1y \\ ^2y \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

con un doble autovalor 0 en este caso.



Las tres opciones presentadas hacen naturalmente cambiar los autovalores de la matriz del sistema lineal (prácticamente la variedad sería ilimitada) con lo que, como antes hemos insinuado, la estabilidad lineal es diferente en cada caso y la estabilidad del sistema original no se obtiene evidentemente de esta manera.

Ejemplos con el jacobiano 'congelado'

La idea anterior se hace más evidente en un ejemplo que veremos a continuación. El problema concreto del que nos ocuparemos será el de un sistema lineal, pero de coeficientes variables (con x) y que, por lo tanto no es autónomo. Sea

$$\mathbf{y}' = A(x) \mathbf{y},$$

dicho sistema, o sea,

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} a_{11}(x) & a_{12}(x) \\ a_{21}(x) & a_{22}(x) \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

Ahora el sistema posee sus propios autovalores digamos 'generales' o 'variables', con lo que queremos decir que en principio dependen de la variable independiente x .

Puede parecer en principio que el comportamiento de su estabilidad en un punto x_0 (o sea, en un entorno de x_0) no varía cuando se substituye la matriz $A(x)$ por la matriz constante $A(x_0)$ 'congelada' en el punto x_0 . El ejemplo pone en evidencia que esto no es así. Es otra forma de ver lo que antes hemos dicho: la estabilidad no lineal no puede estar ligada al comportamiento de un sistema lineal de coeficientes constantes que aproxime el sistema original.

Ejemplo

Consideremos el sistema de la forma

$$\mathbf{y}' = A(x) \mathbf{y}$$

dado por

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} 0 & \frac{1}{1+x^2} \\ -\frac{1}{1+x^2} & 0 \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix} = \frac{1}{1+x^2} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

Los autovalores de $A(x)$ son

$$\lambda_1, \lambda_2 = \pm \frac{i}{1+x^2}$$

como se comprueba sin dificultad. Su carácter de imaginarios puros indicaría que sus soluciones serían oscilantes (o sea, periódicas).

De hecho, cualquier sistema lineal de coeficientes constantes obtenidos tomando $A(x_0)$ en lugar de $A(x)$, tiene soluciones con este comportamiento, ya que sus valores propios son también imaginarios puros.

Por ejemplo, eligiendo $x_0 = 0$, el sistema 'congelado' en 0 de coeficientes constantes, $\mathbf{y}' = A(0) \mathbf{y}$ es

$$\begin{bmatrix} {}^1y' \\ {}^2y' \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{bmatrix} {}^1y \\ {}^2y \end{bmatrix}$$

con matriz constante de autovalores

$$\lambda_1, \lambda_2 = \pm i.$$

Este sistema de coeficientes constantes tiene por matriz fundamental de soluciones

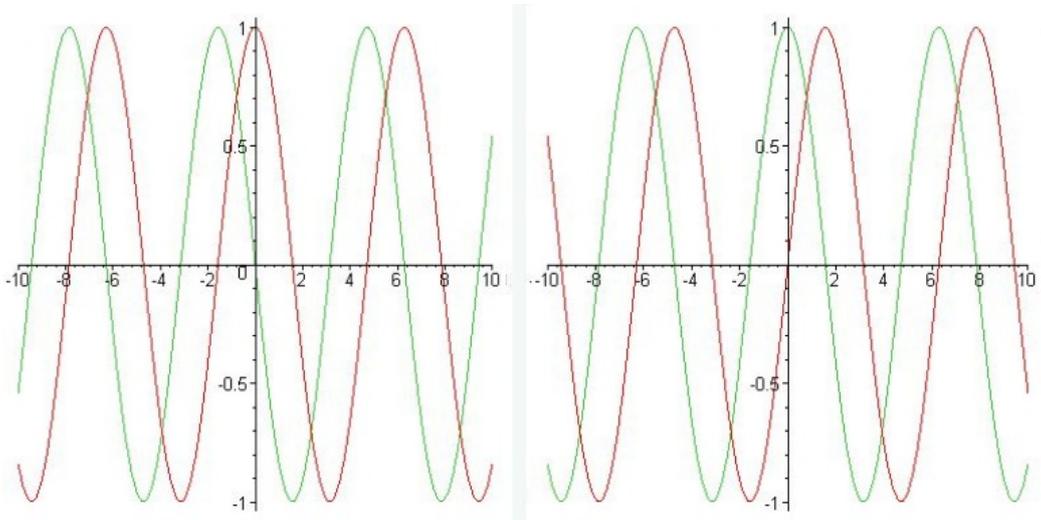
$$e^{x \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}} = \begin{pmatrix} \cos x & \operatorname{sen} x \\ -\operatorname{sen} x & \cos x \end{pmatrix}$$

O sea, su solución general es

$$\mathbf{y}(x) = C_1 \begin{bmatrix} \cos x \\ -\operatorname{sen} x \end{bmatrix} + C_2 \begin{bmatrix} \operatorname{sen} x \\ \cos x \end{bmatrix}$$

y cualquiera de dichas soluciones es periódica con período 2π .

La figura adjunta muestra a la izquierda las componentes $^1y(x)$ e $^2y(x)$ para el caso $C_1 = 1$ y $C_2 = 0$ (primera en rojo y segunda en verde). A la derecha lo hace para $C_1 = 0$ y $C_2 = 1$ (primera en rojo y segunda en verde). Como hemos dicho, se trata de soluciones periódicas.

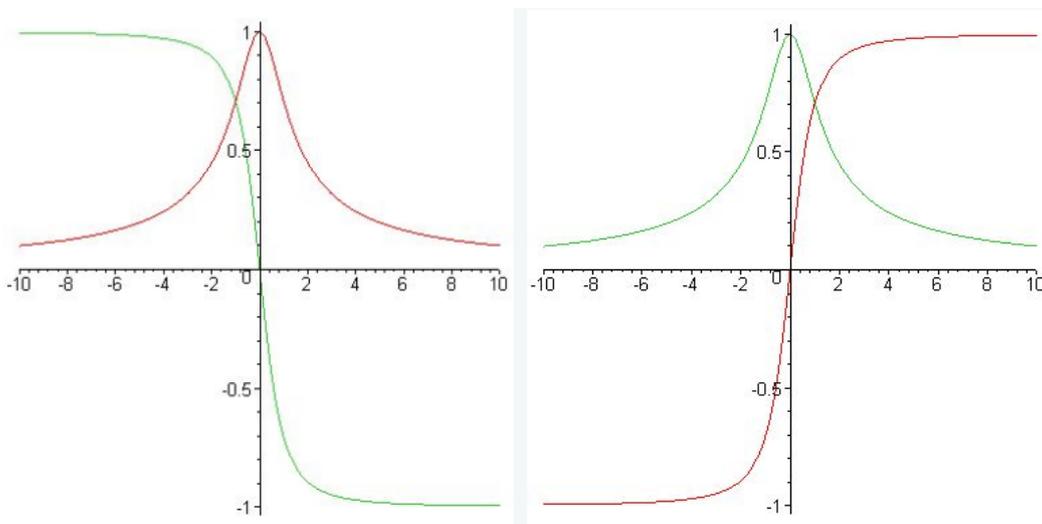


Sin embargo, es fácil comprobar que la solución general del sistema original es

$$\mathbf{y}(x) = C_1 \frac{1}{\sqrt{1+x^2}} \begin{bmatrix} 1 \\ -x \end{bmatrix} + C_2 \frac{1}{\sqrt{1+x^2}} \begin{bmatrix} x \\ 1 \end{bmatrix}$$

y que no tiene nada de oscilante.

Por ejemplo la figura adjunta dibuja a la izquierda las componentes $^1y(x)$ e $^2y(x)$ para el caso $C_1 = 1$ y $C_2 = 0$ (primera en rojo y segunda en verde). A la derecha lo hace para $C_1 = 0$ y $C_2 = 1$ (primera en rojo y segunda en verde).



Conviene además tener en cuenta que cualquier solución es combinación lineal, como hemos dicho, de estas dos.

Este ejemplo (y otros tres sobre el mismo tema) han sido extraídos del libro J.D. LAMBERT: 'Numerical Methods for Ordinary Differential Systems: The Initial Value Problem', que aborda este tema en su capítulo 7, y proporciona los ejemplos a los que aludimos en la sección 7,1

No es raro, y a veces debe hacerse, relacionar la estabilidad no lineal de un sistema con la estabilidad lineal de una aproximación lineal del mismo. Muchas veces se puede hacer una previsión acertada de la estabilidad. Pero conviene ser cauto porque, como hemos visto, a veces no es correcta la extrapolación de los resultados al sistema no lineal.

Las cuestiones relativas a la estabilidad no lineal no son tratadas en bastantes de los libros que se ocupan de nuestro tema. E incluso hay muchos problemas en relación con la estabilidad no lineal que se encuentra aún 'abiertos', o sea, que son estudiados o deben serlo todavía, no resultando en absoluto sencillos.

Ejercicios de

(9) Estabilidad no lineal

Parte 3. Métodos en diferencias para las E.D.P.

Las ecuaciones en derivadas parciales; ecuaciones de segundo orden

Generalidades sobre los métodos en diferencias

Las ecuaciones elípticas

Las ecuaciones parabólicas

Las ecuaciones hiperbólicas

El método de líneas (M.O.L.)

10. Las Ecuaciones en Derivadas Parciales; Ecuaciones de segundo orden

- 1 Introducción: EDPs y condiciones de contorno
- 2 Diferencias finitas y Elementos finitos
- 3 Tipos de EDPs lineales de segundo orden
- 4 Fórmulas para la aproximación de las derivadas

Las E.D.P. o EDPs

Por ejemplo, las expresiones

$$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0$$

o también

$$\frac{\partial u(x, t)}{\partial t} = \alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2}$$

y

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

son ecuaciones en derivadas parciales, expresión abreviada a menudo como E.D.P. o EDPs. Nótese que la función incógnita $u(x, y)$ o $u(x, t)$ es una función de varias variables y que lo que interviene en las ecuaciones son varias de las derivadas parciales de u respecto de sus variables,



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

en los ejemplos citados.

El orden de la ecuación, o sea, el mayor orden de derivación que aparece en los términos de la ecuación en derivadas parciales es de 2 en los dos primeros casos y de 1 en el segundo. Por lo tanto, los ejemplos nos traen dos EDPs de segundo orden y una última de primer orden.

Es usual (y así lo haremos con mucha frecuencia) denotar como

$$u_{xx}(x, y), u_{yy}(x, y), u_t(x, t), u_x(x, y) \text{ y } u_y(x, y)$$

o por

$$u_{xx}, u_{yy}, u_t, u_x \text{ y } u_y$$

las derivadas parciales que aparecen. Como se ve, es más corto de escritura y es perfectamente descriptivo de lo que se quiere, además de ocupar una sola línea en lugar de dos.



Es verdad que, de manera general, es posible definir una ecuación en derivadas parciales con, por ejemplo, dos variables independientes como una relación de la forma

$$F(x, y, u, u_x, u_y, u_{xx}, u_{yy}, u_{xy}, u_{xxx}, u_{xxy} \dots) = 0$$

aunque posiblemente los ejemplos hayan sido suficientes para garantizar una buena descripción de su naturaleza.

La mayor parte de las veces, ecuaciones como estas que hemos propuesto en los ejemplos poseen una infinidad de soluciones diferentes. El complemento adecuado a las EDPs son las llamadas **condiciones iniciales** o **condiciones de contorno**. Para ir fijando ideas vamos a considerar solamente el caso de funciones incógnitas de un par de variables. No abordaremos explícitamente aquí otros casos. Es decir, la función incógnita u será $u(x, y)$ o $u(x, t)$.

El dominio o recinto en el que se verifica la correspondiente ecuación y en el que estudiaremos su solución, obtenida numéricamente, es alguna región del plano x, y o x, t , descrito globalmente, muy a menudo de manera geométrica o por su curva frontera, y, en el caso en que este dominio D sea un rectángulo, descrito mediante el recorrido en un intervalo finito o infinito de las variables x, y o x, t .

Para ser intuitivos en las explicaciones, usaremos para las variables los nombres de x o y cuando estas sean 'espaciales', o sea, cuando representen una dimensión física del problema. Sin embargo, usaremos el nombre de t cuando la variable tenga en el problema una interpretación temporal. En este caso, el intervalo de variación de t será siempre un intervalo de la forma $[t_0, \infty)$ y en la práctica $[0, \infty)$ tras el adecuado cambio de variable.

Volvamos sobre uno de los ejemplos del principio, la llamada **ecuación de LAPLACE**

$$\frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0$$

que también se escribe

$$\Delta u(x, y) = 0,$$

mediante el operador de LAPLACE

$$\Delta u(x, y) = u_{xx}(x, y) + u_{yy}(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2}.$$

Considerando esta ecuación en el disco unidad del plano x, y

$$D = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 < 1\}$$

de curva frontera la circunferencia unidad

$$C = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\},$$

hay infinitas soluciones; sin complicarnos indebidamente,

cualquier función de la forma

$$u(x, y) \equiv \text{cte.}$$

sirve y, de manera más complicada, las soluciones son exactamente la llamadas 'funciones armónicas' sobre D .

Ahora bien, si añadimos, por ejemplo, la condición de frontera (sobre la frontera C del dominio D)

$$u(x, y) \equiv 0, \quad (x, y) \in C \quad \text{osea} \quad x^2 + y^2 = 1$$

obtenemos lo que se llama el **problema de DIRICHLET** en el disco unidad.

Ahora, la función

$$u(x, y) \equiv 0$$

es la única solución del problema

$$\Delta u(x, y) = 0, \quad (x, y) \in D, \quad u(x, y) \equiv 0, \quad (x, y) \in C.$$

Buscando algo menos trivial, si (ρ, θ) son las coordenadas polares de (x, y) y añadimos, por ejemplo, la condición de frontera (sobre la frontera $C, \rho = 1$ del dominio D)

$$u(1, \theta) = \text{sen } \theta, \theta \in [0, 2\pi],$$

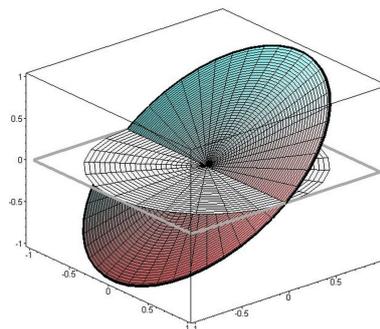
la función

$$u(\rho, \theta) = \rho \text{ sen } \theta, \text{ obien } u(x, y) = y$$

es la única solución del problema

$$\Delta u(x, y) = 0, \quad (x, y) \in D, \quad u(1, \theta) = \text{sen } \theta, \quad (1, \theta) \in C,$$

solución no constante pero que es 'plana', como muestra el siguiente dibujo



Solución de una ecuación elíptica (LAPLACE en el disco unidad para $u(1, \theta) = \text{sen } \theta$)



Es inmediato comprobar que $u(x, y) = y$ es solución de la ecuación $\Delta u(x, y) = 0$, aunque siempre es útil poder hacer la misma comprobación en coordenadas polares.

Con esta finalidad vamos a escribir el laplaciano $\Delta u(x, y)$ en polares, o sea escribir $\Delta u(\rho, \theta)$ cuando

$$\begin{aligned} x &= \rho \cos \theta, & y &= \rho \text{ sen } \theta, \\ y &= \rho \text{ sen } \theta, & x &= \rho \cos \theta, \end{aligned}$$

y donde

$$u(\rho, \theta) = u(x(\rho, \theta), y(\rho, \theta)) = u(\rho \cos \theta, \rho \text{ sen } \theta).$$

De las expresiones de $x(\rho, \theta)$ e $y(\rho, \theta)$ es inmediato que

$$\begin{aligned} x_\rho &= \cos \theta, & x_\theta &= -\rho \text{ sen } \theta, \\ y_\rho &= \text{sen } \theta, & y_\theta &= \rho \cos \theta; \end{aligned}$$



Derivando las expresiones de $x(\rho, \theta)$ e $y(\rho, \theta)$ respecto de la propia x , obtenemos

$$\begin{aligned} 1 &= \rho_x \cos \theta - \rho \operatorname{sen} \theta \theta_x, \\ 0 &= \rho_x \operatorname{sen} \theta + \rho \cos \theta \theta_x; \end{aligned}$$

de este sistema es sencillo obtener los valores de ρ_x y θ_x ; son

$$\rho_x = \cos \theta, \quad \theta_x = -\operatorname{sen} \theta / \rho.$$

Lo mismo, pero ahora derivando respecto de y proporciona

$$\begin{aligned} 0 &= \rho_y \cos \theta - \rho \operatorname{sen} \theta \theta_y, \\ 1 &= \rho_y \operatorname{sen} \theta + \rho \cos \theta \theta_y; \end{aligned}$$

y, finalmente,

$$\rho_y = \operatorname{sen} \theta, \quad \theta_y = \cos \theta / \rho.$$

La regla de la cadena y las igualdades obtenidas nos proporcionan

$$u_x = u_\rho \rho_x + u_\theta \theta_x = u_\rho \cos \theta - u_\theta \operatorname{sen} \theta / \rho$$

y, a continuación,

$$\begin{aligned} u_{xx} &= (u_{\rho\rho} \cos \theta - u_{\rho\theta} \operatorname{sen} \theta / \rho + u_\theta \operatorname{sen} \theta / \rho^2) \rho_x + \\ &\quad + (u_{\rho\theta} \cos \theta - u_\rho \operatorname{sen} \theta - u_{\theta\theta} \operatorname{sen} \theta / \rho - u_\theta \cos \theta / \rho) \theta_x \\ &= u_{\rho\rho} \cos^2 \theta - u_{\rho\theta} \operatorname{sen} \theta \cos \theta / \rho + u_\theta \operatorname{sen} \theta \cos \theta / \rho^2 - \\ &\quad - u_{\rho\theta} \operatorname{sen} \theta \cos \theta / \rho + u_\rho \operatorname{sen}^2 \theta / \rho + u_{\theta\theta} \operatorname{sen}^2 \theta / \rho^2 + u_\theta \operatorname{sen} \theta \cos \theta / \rho^2, \end{aligned}$$

e igualmente

$$u_y = u_\rho \rho_y + u_\theta \theta_y = u_\rho \operatorname{sen} \theta + u_\theta \cos \theta / \rho$$

y, a continuación,

$$\begin{aligned} u_{yy} &= (u_{\rho\rho} \operatorname{sen} \theta + u_{\rho\theta} \cos \theta / \rho - u_\theta \cos \theta / \rho^2) \rho_y + \\ &\quad + (u_{\rho\theta} \operatorname{sen} \theta + u_\rho \cos \theta + u_{\theta\theta} \cos \theta / \rho - u_\theta \operatorname{sen} \theta / \rho) \theta_y \\ &= u_{\rho\rho} \operatorname{sen}^2 \theta + u_{\rho\theta} \operatorname{sen} \theta \cos \theta / \rho - u_\theta \operatorname{sen} \theta \cos \theta / \rho^2 + \\ &\quad + u_{\rho\theta} \operatorname{sen} \theta \cos \theta / \rho + u_\rho \cos^2 \theta / \rho + u_{\theta\theta} \cos^2 \theta / \rho^2 - u_\theta \operatorname{sen} \theta \cos \theta / \rho^2. \end{aligned}$$

Por fin, operaciones ya muy sencillas nos llevan a

$$\Delta u = u_{xx} + u_{yy} = u_{\rho\rho} + \frac{1}{\rho} u_{\rho} + \frac{1}{\rho^2} u_{\theta\theta},$$

expresión que se conoce como 'el laplaciano en coordenadas polares'.

Ahora no presenta dificultad alguna la comprobación de que $u(\rho, \theta) = \rho \operatorname{sen} \theta$ es, como ya hemos afirmado, solución de la ecuación de LAPLACE en el disco unidad y, por lo tanto, solución del problema de DIRICHLET formado añadiendo la condición de frontera

$$u(1, \theta) = \operatorname{sen} \theta, \theta \in [0, 2\pi].$$

Cambiando esta condición por otras sencillas como

$$u(1, \theta) = \operatorname{sen} 3\theta, \theta \in [0, 2\pi],$$

la solución que resulta ya no es obviamente plana, pues la función que define el contorno no está sobre un plano. La solución de este problema se obtiene tras algunos cálculos. En primer lugar, nos resultarán útiles dos fórmulas:

$$\begin{aligned} \operatorname{sen}(3\theta) &= \Im[e^{i3\theta}] = \Im[(e^{i\theta})^3] = \Im[(\cos \theta + i \operatorname{sen} \theta)^3] = \\ &= 3 \cos^2 \theta \operatorname{sen} \theta - \operatorname{sen}^3 \theta \end{aligned}$$

$$\begin{aligned} \operatorname{cos}(3\theta) &= \Re[e^{i3\theta}] = \Re[(e^{i\theta})^3] = \Re[(\cos \theta + i \operatorname{sen} \theta)^3] = \\ &= \cos^3 \theta - 3 \cos \theta \operatorname{sen}^2 \theta = 4 \cos^3 \theta - 3 \cos \theta \end{aligned}$$

Viendo la expresión de $\operatorname{sen}(3\theta)$ y recordando que $x = \rho \cos \theta$ y $y = \rho \operatorname{sen} \theta$ conviene ensayar con la función

$$u(x, y) = 3x^2y - y^3$$

para la que

$$u(1, \theta) = 3 \cos^2 \theta \sin \theta - \sin^3 \theta = \sin(3\theta),$$

que es la condición de frontera de problema; como, por otra parte, $u_{xx}(x, y) = 6y$ y $u_{yy}(x, y) = -6y$, resulta que $\Delta u(x, y) = 0$ y la función $u(x, y) = 3x^2y - y^3$ es la solución de nuestro problema de DIRICHLET con frontera $\sin(3\theta)$.

La conocida como fórmula de POISSON resuelve también el problema de DIRICHLET anterior; en ese caso la solución en coordenadas polares del problema de contorno viene dada por

$$u(\rho, \theta) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sin 3t (1 + 2 \sum_{n=1}^{\infty} \rho^n \cos n(\theta - t)) dt$$

con aportaciones nulas de todos los términos, excepto el correspondiente a $n = 3$, que vale

$$\begin{aligned} u(\rho, \theta) &= \frac{\rho^3}{2\pi} \left(\frac{1}{6} \cos 3\theta + 8\pi \sin \theta \cos^2 \theta - 2\pi \sin \theta - \frac{2}{3} \cos^3 \theta + \frac{1}{2} \cos \theta \right) = \\ &= \frac{\rho^3}{2\pi} (8\pi \sin \theta \cos^2 \theta - 2\pi \sin \theta) = \rho^3 (4 \sin \theta \cos^2 \theta - \sin \theta) \\ &= \rho^3 (3 \sin \theta \cos^2 \theta - \sin^3 \theta) \end{aligned}$$

o sea, como antes, $u(x, y) = 3x^2y - y^3$.

Presentaremos enseguida un dibujo de esta solución.

Los problemas de EDPs que se plantean usualmente (por ejemplo, los de la Física matemática que iremos abordando) se presentan acompañados de condiciones que hacen que la solución sea única, unicidad que se corresponde con la idea intuitiva que tenemos sobre las soluciones de estos problemas.

Los métodos que vamos a estudiar están pensados justamente para estas ecuaciones de la Física matemática y las condiciones usuales que las acompañan. El papel de estas condiciones sobre la frontera es muy a menudo proporcionar procedimientos de 'arranque' para los métodos en diferencias.

Diferencias finitas

Para una ecuación en derivadas parciales, o mejor, para un problema formado por la ecuación sobre un dominio D del plano y condición o condiciones de contorno sobre el contorno C de D , que consigan asegurar la existencia de una única solución $u(x, y)$ en D , el propósito es elaborar métodos que sirvan para aproximar numéricamente la solución $u(x, y)$. Hay dos ideas básicas para llevar a cabo esta tarea.

Los métodos en **Diferencias finitas** proporcionan valores aproximados

$$u_{nm} \sim u(x_n, y_m)$$

que aproximen los verdaderos valores $u(x_n, y_m)$ de la solución u en nodos (x_n, y_m) repartidos en el dominio D .

La idea es muy similar a la empleada en el tratamiento numérico de las EDOs, aunque ahora la red de nodos pasa a ser bidimensional.

Elementos finitos

Otra alternativa es aproximar la solución $u(x, y)$ por alguna función 'sencilla', $\Phi(x, y)$, parecida a u . Por ejemplo, $\Phi(x, y)$ podría ser un 'polinomio a trozos' en las variables x e y . A pesar de su enorme interés, esta técnica, conocida genéricamente como método de **Elementos finitos**, no será objeto de estudio en este curso.

Tipos de EDPs lineales de segundo orden

Las EDPs, incluso desde el punto de vista tan concreto de su resolución numérica, forman un universo tan amplio que es necesario reducirse a una parcela más pequeña.

Ya hemos comentado que vamos a limitarnos al tratamiento del caso bidimensional.

Pero, además, nos reduciremos a la exposición de ciertos métodos en diferencias para ecuaciones en derivadas parciales que sean lineales y de segundo orden. Aunque esto parece demasiado particular, el caso comprende los ejemplos más notables que se presentan en la Física matemática, en donde la Ley de NEWTON limita al segundo orden las derivadas que intervienen.

De manera general, una de estas ecuaciones es de la forma

$$\alpha_1 u + \alpha_2 u_x + \alpha_3 u_y + \alpha_4 u_{xx} + \alpha_5 u_{yy} + \alpha_6 u_{xy} = \alpha_7,$$

donde las α_i son funciones de x e y , o, muy a menudo, simples constantes.

De los tres ejemplos citados al comienzo del tema, son de este tipo las dos primeras. La tercera es de primer orden.

Transformaciones bien conocidas de las variables x e y llevan a reducir la ecuación general de una curva cónica

$$\alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 y^2 + \alpha_6 xy = 0$$

dejándola en una forma reconocible como elipse, hipérbola o parábola. En el caso de nuestras ecuaciones, un cambio adecuado de variables, generado mediante técnicas similares, lleva nuestra ecuación a alguna de las formas más sencillas, siendo los casos más usuales los que vamos describiendo a continuación.

Como ya hemos anunciado, nunca nos ocuparemos de una ecuación ella sola, sino del problema formado por la ecuación y por condiciones de contorno adecuadas para asegurar una solución única del problema, solución que ya hemos dicho también que es la que aproximaremos numéricamente.

Problemas elípticos

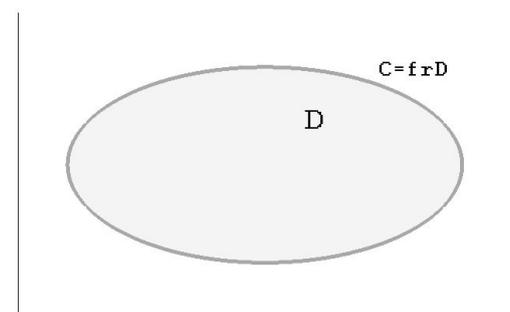
Estudiaremos el problema

$$\alpha(x, y) \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = f(x, y),$$

$$(x, y) \in D, \quad \alpha(x, y) > 0 \text{ si } (x, y) \in D$$

$$u(x, y) = g(x, y) \text{ si } (x, y) \in C = \partial D,$$

siendo D un dominio acotado y 'regular' del plano y $C = \partial D$ la curva frontera de dicho dominio.



En un ejemplo tratado antes, D era el disco unidad $x^2 + y^2 < 1$ del plano y C era la circunferencia $x^2 + y^2 = 1$ frontera de dicho disco. Si D , C y la función $g(x, y)$ son 'regulares', la condición de contorno (como se ve es una condición sobre el comportamiento de la solución $u(x, y)$ en la curva 'contorno' del dominio D) sirve para garantizar la unicidad de la solución $u(x, y)$ en el dominio D . Gracias a esta unicidad de la solución, nos es posible diseñar métodos de aproximar la misma numéricamente. Nosotros abordaremos sólo el caso en que la ecuación es

$$\Delta u(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = f(x, y),$$

que, además, es la más clásica, la llamada **ecuación de POISSON**,

completada con una condición sobre el contorno con la que forma el llamado **problema de DIRICHLET**. Esta ecuación explica el trazado de una membrana tensa (en la que la fuerza gravitatoria es pues despreciable) con borde sujeto a una posición forzada

$$u(x, y) = g(x, y) \text{ si } (x, y) \in C = \partial D;$$

por ello se suele usar el nombre de 'ecuación de las membranas tensas' para esta ecuación.

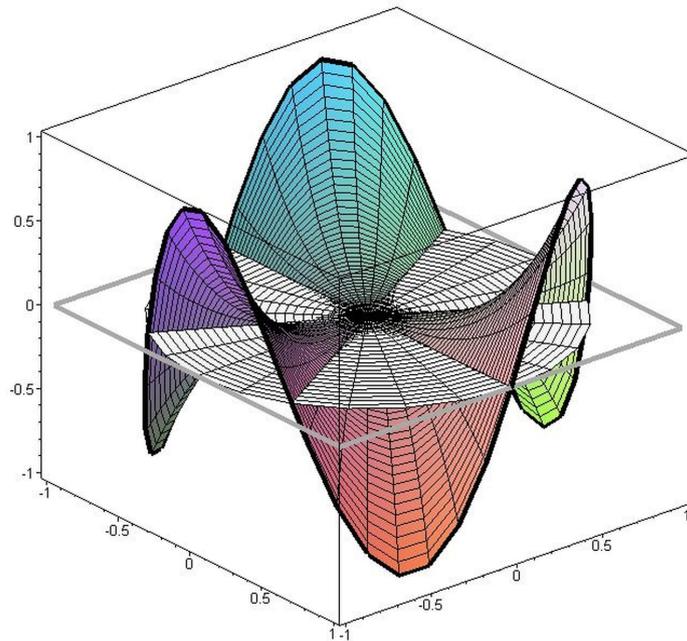
Ya hemos estudiado varios ejemplos de soluciones triviales o sencillas y citado también un ejemplo concreto no trivial, el

$$\Delta u(x, y) = 0, \quad (x, y) \in D, \quad u(1, \theta) = \text{sen } 3\theta, \quad (1, \theta) \in C,$$

con condición de contorno escrita en coordenadas polares. Como dijimos, existe una única solución de este problema, que es

$$u(x, y) = 3x^2y - y^3,$$

única solución en el disco unidad $D = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 < 1\}$ verificando $u(1, \theta) = \sin 3\theta$. Dibujada resulta ser:



Solución de una ecuación elíptica (LAPLACE en el disco unidad para $u(1, \theta) = \sin 3\theta$)



Problemas parabólicos

En su correspondiente capítulo, estudiaremos el problema

$$\alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2} = \frac{\partial u(x, t)}{\partial t}, \quad x \in [0, l], t \in [0, +\infty),$$

o sea, el dominio D para la ecuación va a ser el rectángulo 'semiinfinito'

$$D = [0, l] \times [0, +\infty),$$

infinito en lo que a la variable t respecta. En los problemas habituales en los que interviene una ecuación como esta, t suele representar el tiempo, de ahí la notación empleada y la extensión del dominio D en esta coordenada.



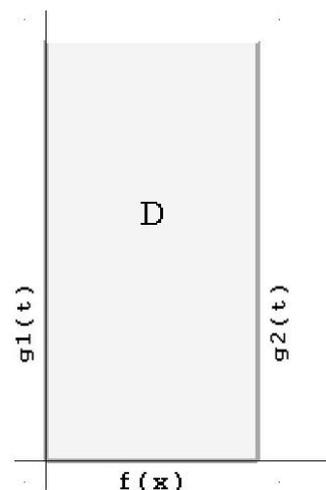
Generalmente se acompaña esta ecuación de condiciones en la frontera; la llamada 'condición inicial'

$$u(x, 0) = f(x), x \in [0, l],$$

llamada inicial porque se refiere al comportamiento de la solución $u(x, t)$ en el instante inicial $t = 0$, y las condiciones 'de contorno' (sobre el contorno físico tanto en $x = 0$ como en $x = l$)

$$u(0, t) = g_1(t), \quad u(l, t) = g_2(t), t \in [0, +\infty).$$

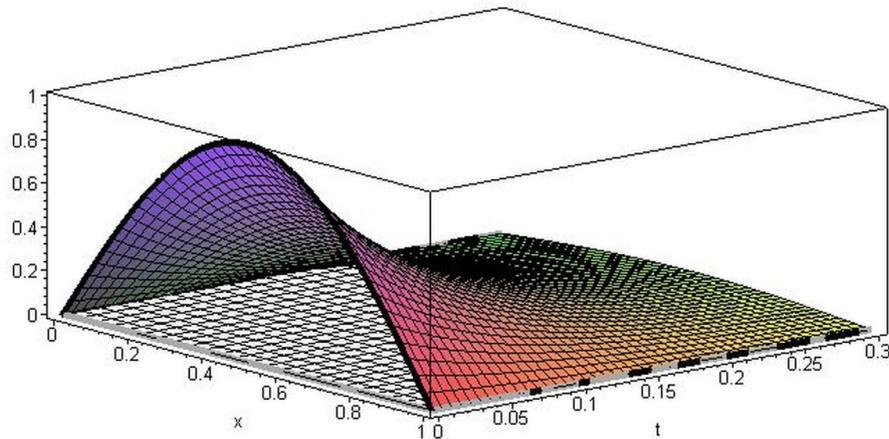
El dibujo del dominio es como



El ejemplo físico más típico para este problema es la difusión del calor (distribuido como $u(x, 0) = f(x)$ en el instante inicial) en una barra lineal de longitud l , con extremos mantenidos a temperaturas $u(0, t) = g_1(t)$ y $u(l, t) = g_2(t)$ en el instante t . Por eso se suele llamar 'ecuación de difusión del calor' a esta ecuación.

El dibujo siguiente da una idea de la solución de uno de estos problemas de difusión del calor, en este caso para una condición inicial $u(x, 0) = \sin \pi x$ en $[0, 1]$ y condiciones homogéneas de contorno, o sea, $u(0, t) = 0$ y $u(l, t) = 0$ para todo valor de t .

Solución de una ecuación del calor con $u(x,0)=\sin \pi x$ y contorno homogéneo



Problemas hiperbólicos

Estudiaremos el problema

$$\frac{\partial^2 u(x, t)}{\partial t^2} - \alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2} = f(x, t), \quad x \in [0, l], t \in [0, +\infty),$$

o sea, con dominio D para la ecuación el rectángulo 'semiinfinito'

$$D = [0, l] \times [0, +\infty),$$

infinito en lo que a la variable t respecta. En los problemas habituales en los que interviene una ecuación como esta, también t suele representar el tiempo, de ahí la notación empleada y la extensión del dominio D en esta coordenada.

Se presenta generalmente acompañada de condiciones 'de contorno' (sobre el contorno físico tanto en $x = 0$ como en $x = l$)

$$u(0, t) = g_1(t), \quad u(l, t) = g_2(t), \quad t \in [0, +\infty)$$

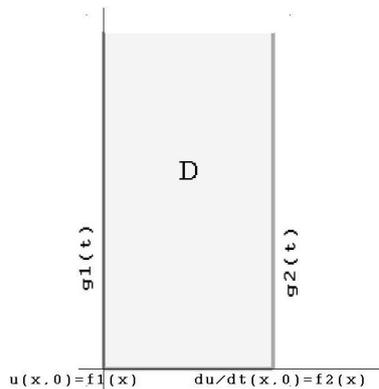
(igual que en el caso parabólico),



pero también de 'dos' condiciones iniciales (o sea, para $t = 0$); son

$$u(x, 0) = f_1(x), \quad y \quad \frac{\partial u(x, 0)}{\partial t} = f_2(x), \quad x \in [0, 1],$$

El caso más típico responde a la oscilación o vibración de una cuerda tensa (se puede pensar en una cuerda de un instrumento musical) con posición inicial $f_1(x)$ y velocidad inicial $f_2(x)$, lo que hace que suela usarse el nombre de 'ecuación de la cuerda vibrante' para referirse a este problema. El dominio es como antes

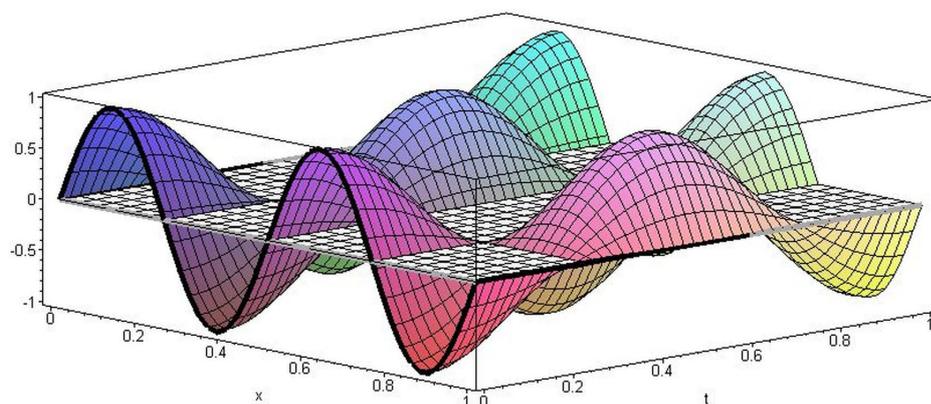


aunque hay que añadir una condición inicial.



El dibujo siguiente da una idea de la solución de uno de estos problemas de la cuerda vibrante, en este caso para condiciones iniciales $u(x, 0) = \text{sen } 4\pi x$ y $\frac{\partial u(x, 0)}{\partial t} = 0$ en $[0, 1]$ y condiciones homogéneas de contorno, o sea, $u(0, t) = 0$ y $u(1, t) = 0$ para todo valor de t .

Solución de una ecuación de la cuerda vibrante con $u(x, 0) = \text{sen } 4\pi x$ $du/dt(x, 0) = 0$



Cuando creamos los métodos en diferencias para las EDPs, el procedimiento será siempre análogo: reemplazar cada derivada $u_x(x, y) = \partial u(x, y) / \partial x$, $u_{xx}(x, y) = \partial^2 u(x, y) / \partial x^2$ o similares por expresiones que hagan intervenir solamente valores de $u(x_n, y_m)$ de la función incógnita u en puntos de la red de nodos.

Esto lo llevaremos a cabo convirtiendo las derivadas parciales en derivadas ordinarias, fijando el valor de una de las variables en la correspondiente coordenada de la red de nodos. Por ejemplo, convirtiendo $u_x(x_n, y_m) = \partial u(x_n, y_m) / \partial x$ en la derivada ordinaria en x_n de la función que lleva x en $u_m(x, y_m)$, o sea, fijando y_m . Convertidas así las derivadas parciales en ordinarias, sólo nos quedará usar alguna de la fórmulas que aproximan $f'(x)$ o $f''(x)$ mediante valores de la función f de una variable, dependiendo del grado de aproximación que nos interese y del orden de la derivada.

Tres van a ser las fórmulas de aproximación que usemos en la exposición teórica. Son

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(\xi) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h),$$

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} - \frac{h^2}{6} f'''(\xi) = \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2),$$

y para la derivada segunda,

$$\begin{aligned} f''(x) &= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} - \frac{h^2}{12} f^{iv}(\xi) \\ &= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2), \end{aligned}$$

Ejercicios de

**(10) Las Ecuaciones en Derivadas Parciales;
Ecuaciones de segundo orden**

Ejercicios de

(10) Las Ecuaciones en Derivadas Parciales;

Ecuaciones de segundo orden

Ejercicio 10-01

Recordemos que se escribe

$$\Delta u = \frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2}$$

en dos dimensiones, y análogamente en más dimensiones.

Se considera la ecuación elíptica

$$\Delta u = f, \text{ en } D$$

$$u = 0, \text{ en } \Gamma.$$

Considérese una solución cualquiera g de

$$\Delta u = f, \text{ en } D,$$

el problema

$$\Delta u = 0, \text{ en } D$$

$$u = -g, \text{ en } \Gamma,$$

y la solución, h , de este problema. Pruébese

que, entonces

$$u = g + h$$

es solución del problema original.

```
[ > restart;
[ > interface(labeling=false);
[ Como u = g + h y
[ > Delta(g):=f;
[                                     Δ(g) := f
[ > Delta(h):=0;
[                                     Δ(h) := 0
[ resulta
[ > Delta(u):=Delta(g)+Delta(h);
[                                     Δ(u) := f
[ y, como en Γ,
[ > h:=-g;
[                                     h := -g
[ resulta también en Γ
[ > u:=g+h;
[                                     u := 0
```


11. Las Ecuaciones elípticas

- 1 El problema elíptico que vamos a tratar
- 2 El método de las diferencias centradas
- 3 Orden para los nodos y ecuaciones
- 4 Error de truncación y orden de un método

El problema de DIRICHLET

Abordaremos sólo el caso en que la ecuación es

$$\Delta u(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = f(x, y),$$

que, además, es la más clásica, la llamada **ecuación de LAPLACE** (llamada también **ecuación de POISSON** cuando f no es idénticamente nula), extendida a un dominio $R = [a, b] \times [c, d]$ rectangular. Llamaremos S a la línea rectangular frontera de R . Añadiremos una condición sobre el contorno S de la forma

$$u(x, y) = g(x, y) \text{ si } (x, y) \in S = \partial R.$$

En el capítulo anterior pusimos como ejemplo del caso elíptico el **problema de DIRICHLET** en el disco unidad. Era

$$\Delta u(\rho, \theta) = 0, \text{ en } D, \quad u(1, \theta) = g(\theta), \quad (1, \theta) \in C,$$

Es un problema sobre un círculo, pero, para las coordenadas polares el dominio es, por ejemplo, el rectángulo

$$D : \rho \in [0, 1], \quad \theta \in [0, 2\pi],$$

aunque haya que preocuparse de la periodicidad en θ de la solución y de que el valor en la frontera $\rho = 0$ sea constante si queremos dar un sentido a dicha solución.

Consideramos en R el retículo definido por los nodos

$$x_0, x_1, \dots, x_N, \quad x_n = a + nh, \quad h = \frac{b-a}{N},$$

$$y_0, y_1, \dots, y_M, \quad y_m = c + mk, \quad k = \frac{d-c}{M},$$

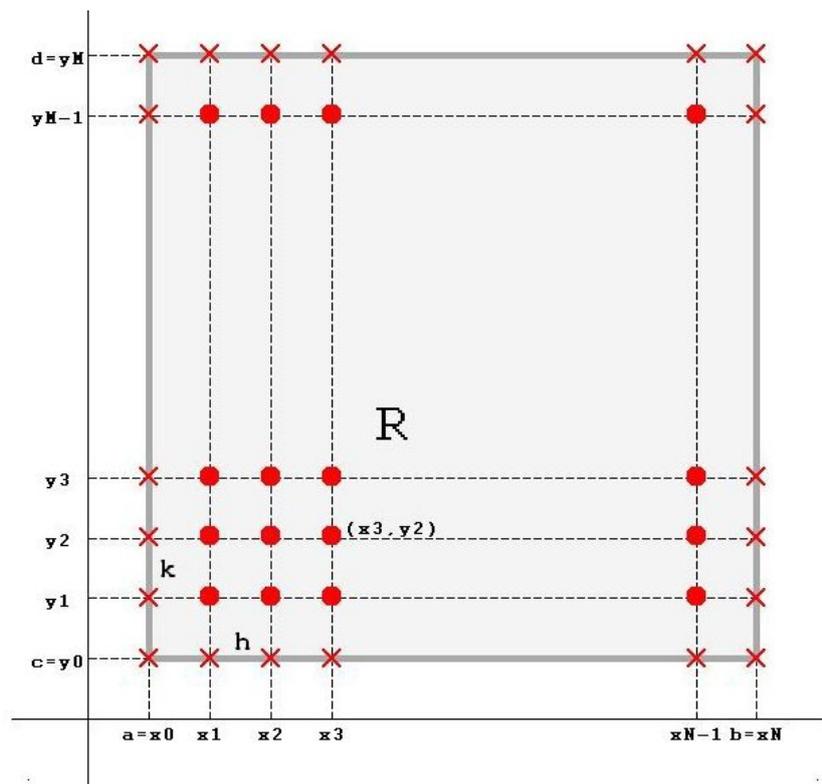
donde hemos dividido $[a, b]$ en N subintervalos iguales de longitud h y $[c, d]$ en M subintervalos iguales de longitud k . Los métodos en diferencias aspiran a calcular valores

$$u_{nm} \sim u(x_n, y_m)$$

que aproximen los de la verdadera solución $u(x, y)$ en los nodos bidimensionales (x_n, y_m) .

Justamente vamos a construir uno de estos métodos, el llamado método **de las diferencias centradas**.

Observemos en primer lugar el dominio rectangular R con sus nodos (x_n, y_m) , $n = 0, 1, \dots, N-1, N$, $m = 0, 1, \dots, M-1, M$



Marcados con un círculo rojo están los llamados 'nodos interiores' del rectángulo. Son

$$(N - 1) \times (M - 1)$$

Los restantes, marcados con un aspa, son los 'nodos del contorno'.

Como en total hay

$$(N + 1) \times (M + 1)$$

nodos, resulta que habrá

$$2N + 2M$$

nodos exteriores.

Cada nodo interior va a dar origen a una de las ecuaciones en diferencias que van a formar el método numérico. Los nodos exteriores poseen otra utilidad, que es la de proporcionar, gracias a la condición de contorno en S , procedimientos de arranque del método.

Si, fijado y_m , consideramos la función de una variable

$$f(x) = u(x, y_m),$$

resulta que las derivadas respecto a x de u en (x, y_m) son justamente las derivadas ordinarias de $f(x)$. En particular

$$\frac{\partial^2 u(x, y_m)}{\partial x^2} = f''(x) \quad \text{y} \quad \frac{\partial^2 u(x_n, y_m)}{\partial x^2} = f''(x_n)$$

Hacemos uso de la fórmula de aproximación

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} - \frac{h^2}{12} f^{(iv)}(\xi)$$

para la citada f (una diferente para cada y_m , claro), $x = x_n$ y h el h del retículo y recordando que $x_{n+1} = x_n + h$ y $x_{n-1} = x_n - h$, obtenemos

$$\frac{\partial^2 u(x_n, y_m)}{\partial x^2} = \frac{u(x_{n+1}, y_m) - 2u(x_n, y_m) + u(x_{n-1}, y_m)}{h^2} - \frac{h^2}{12} \frac{\partial^4 u(\xi_n, y_m)}{\partial x^4}$$

Un procedimiento similar, ahora con $f(y) = u(x_n, y)$, y la fórmula de aproximación de $f''(x)$ usada con $x = y_m$ y h el k del retículo, lleva a

$$\frac{\partial^2 u(x_n, y_m)}{\partial y^2} = \frac{u(x_n, y_{m+1}) - 2u(x_n, y_m) + u(x_n, y_{m-1}))}{k^2} - \frac{k^2}{12} \frac{\partial^4 u(x_n, \eta_m)}{\partial y^4}.$$

Aquí ξ_n y η_m son valores no conocidos de x y de y , de los que sólo sabemos que se encuentran en los intervalos $[x_{n-1}, x_{n+1}]$ e $[y_{m-1}, y_{m+1}]$ respectivamente.

Substituyendo ambas expresiones en la ecuación original escrita para (x_n, y_m) obtenemos

$$\begin{aligned} \frac{u(x_{n+1}, y_m) - 2u(x_n, y_m) + u(x_{n-1}, y_m))}{h^2} + \frac{u(x_n, y_{m+1}) - 2u(x_n, y_m) + u(x_n, y_{m-1}))}{k^2} = \\ = f(x_n, y_m) + \frac{h^2}{12} \frac{\partial^4 u(\xi_n, y_m)}{\partial x^4} + \frac{k^2}{12} \frac{\partial^4 u(x_n, \eta_m)}{\partial y^4}. \end{aligned}$$



11. Las Ecuaciones elípticas

La ecuación en diferencias que describe el método resulta de substituir en esa expresión los valores exactos $u(x_n, y_m)$ por los aproximados u_{nm} y 'olvidar' los términos desconocidos en (ξ_n, y_m) y (x_n, η_m) que, por otra parte, poseen un valor 'pequeño' ya que son, respectivamente, $\mathcal{O}(h^2)$ y $\mathcal{O}(k^2)$. Esta ecuación en diferencias queda

$$\frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} + \frac{u_{nm+1} - 2u_{nm} + u_{nm-1}}{k^2} = f_{nm},$$

donde hemos escrito f_{nm} para representar simplemente $f_{nm} = f(x_n, y_m)$ con estética acorde con el resto de la ecuación o también el valor 'computacionalmente posible' que represente a $f(x_n, y_m)$.



11. Las Ecuaciones elípticas

Lo primero es mejorar un poco la presentación de la ecuación en diferencias que caracteriza el método de las diferencias centradas.

Ponemos

$$\lambda = \frac{h}{k};$$

es lo usual en el caso de métodos para el caso elíptico.

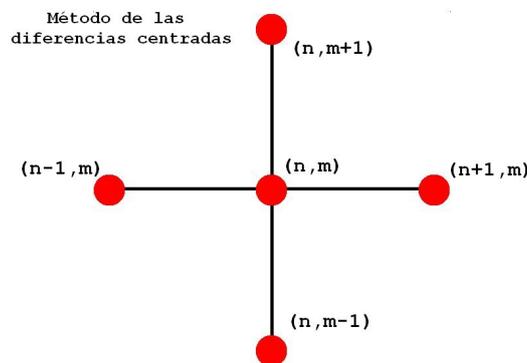
Reemplazando este λ en la ecuación que teníamos y tras unas sencillas operaciones, resulta

$$2(\lambda^2 + 1)u_{nm} - (u_{n+1m} + u_{n-1m}) - \lambda^2(u_{nm+1} + u_{nm-1}) = -h^2 f_{nm}$$

para la ecuación centrada en (x_n, y_m) .

Es una ecuación lineal, con sólo 5 incógnitas en cada ecuación; el conjunto de estas ecuaciones forma un sistema lineal que hay que resolver para obtener los valores u_{nm} que buscamos.

¿Qué incógnitas intervienen en cada ecuación de este método de las diferencias centradas? Es costumbre exhibir este reparto de incógnitas de una manera gráfica, colocando los puntos correspondientes a las aproximaciones que intervienen en la ecuación. En este caso tenemos:



Por la semejanza estética de este dibujo con los que se emplean en descripciones de la estructura química, se suele dar el nombre de **molécula computacional** a este esquema. En este caso, el punto que genera la ecuación es, justamente, el central de esta molécula.

La ecuación precedente tiene sentido cuando (x_n, y_m) en un nodo interior al retículo. Esto hace que haya $(N - 1) \times (M - 1)$ ecuaciones. Sin embargo figuran $(N + 1) \times (M + 1)$ incógnitas, lo que hace peligrar aparentemente la unicidad de la solución al superar las incógnitas el número de ecuaciones del sistema lineal. Aquí viene en nuestra ayuda la condición sobre el contorno. En efecto, u_{00} , por ejemplo, debe parecerse a $u(x_0, y_0)$. Pero este valor está determinado por la condición de contorno como $u(x_0, y_0) = g(x_0, y_0)$. Entonces la igualdad $u_{00} = g_{00}$ nos fija el valor de la aproximación u_{00} ; aquí $g_{00} = g(x_0, y_0)$ o, más exactamente, el valor 'computacionalmente posible' de $g(x_0, y_0)$. Lo mismo para las otras aproximaciones en nodos del contorno. Es decir



11. Las Ecuaciones elípticas

$$\begin{aligned} u_{0m} &= g_{0m}, & m &= 0, \dots, M, \\ u_{Nm} &= g_{Nm}, & m &= 0, \dots, M, \\ u_{n0} &= g_{n0}, & n &= 0, \dots, N, \\ u_{nM} &= g_{nM}, & n &= 0, \dots, N, \end{aligned}$$

recordando que $g_{nm} = g(x_n, y_m)$.

Las igualdades anteriores son en número de $2N + 2M + 4$ y proporcionan valores para $2N + 2M$ incógnitas correspondientes a aproximaciones en nodos del contorno. Es evidente que hay 4 repetidas, pero es la única manera de expresar con sencillez estas asignaciones.

Ahora el sistema formado por las $(N - 1) \times (M - 1)$ ecuaciones de antes, las centrales del método, poseen sólo $(N - 1) \times (M - 1)$ incógnitas, las u_{nm} para $n = 1, \dots, N - 1$ y $m = 1, \dots, M - 1$, y es de esperar una única solución del sistema.



11. Las Ecuaciones elípticas

Un ejemplo:

Acompañamos lo anterior con un ejemplo muy sencillo realizado en 'Maple' en el programa **Cap11_ejemplo1.mws** para el problema de contorno

$$\Delta u(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = 0, \quad (x, y) \in [0, 1] \times [0, 1],$$

en el cuadrado $[0, 1] \times [0, 1]$, con condición (o condiciones) de contorno en el cuadrado frontera

$$\begin{aligned} u(0, y) &= 0, & u(1, y) &= y, \\ u(x, 0) &= 0, & u(x, 1) &= x, \end{aligned}$$

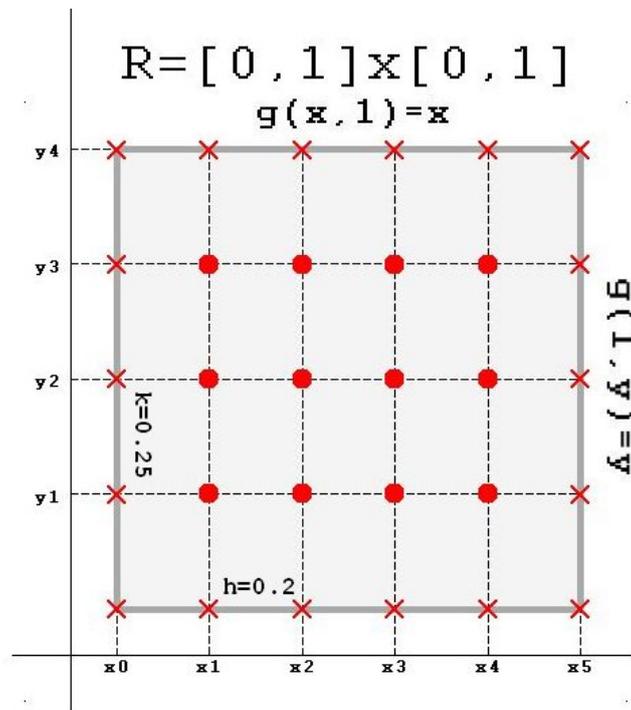
Es inmediato comprobar que este problema tiene como única solución la función

$$u(x, y) = x y.$$

Para su evaluación numérica construimos un retículo poniendo $N = 5$ (para x) y $M = 4$ (para y), luego $h = 1/5 = 0,2$ y $k = 1/4 = 0,25$. Resulta así



11. Las Ecuaciones elípticas



Tenemos en total $6 \times 5 = 30$ nodos, de los que $4 \times 3 = 12$ son interiores y $2 \cdot 5 + 2 \cdot 4 = 18$ están en el contorno.



11. Las Ecuaciones elípticas

En este caso, $\lambda = h/k = 0,2/0,25 = 0,8$ y $2(\lambda^2 + 1) = 3,28$. La ecuación tipo es ahora

$$3,28u_{nm} - (u_{n+1m} + u_{n-1m}) - 0,64(u_{nm+1} + u_{nm-1}) = 0$$

aunque aún sustituimos algunas de las u_{nm} por constantes, como dicta la condición de contorno.

Tenemos que

$$u_{0m} = 0, m = 0, 1, 2, 3, 4,$$

$$u_{n0} = 0, n = 0, 1, 2, 3, 4, 5,$$

$$u_{50} = 0, u_{51} = 0,25, u_{52} = 0,5, u_{53} = 0,75 \text{ y } u_{54} = 1., ,$$

$$u_{04} = 0, u_{14} = 0,2, u_{24} = 0,4, u_{34} = 0,6, u_{44} = 0,8 \text{ y } u_{54} = 1., ,$$

para las 18 incógnitas que así dejan de serlo.

Cada una de las 12 ecuaciones lineales principales posee 5 incógnitas a lo sumo. Pero las que corresponden a las líneas 'casi-exteiores' tienen sólo 4 incógnitas y hay 4 ecuaciones con sólo 3 incógnitas.

En este ejemplo hay que acabar resolviendo un sistema lineal de 12 ecuaciones (cada una con 5 incógnitas a lo sumo) y 12 incógnitas, las $u_{11}, u_{21}, u_{31}, u_{41}, u_{12}, u_{22}, u_{32}, u_{42}, u_{13}, u_{23}, u_{33}$ y u_{43} . Antes ya hemos visto cómo queda la ecuación tipo de este sistema. Naturalmente, resolver con un ordenador un sistema 12×12 no presenta dificultad alguna. Pero es que este ejemplo es solamente un ejemplo y no puede ser considerado 'realista'. Abandonamos momentáneamente esta descripción para seguir con el hilo del razonamiento para el caso general.

Haciendo un ejercicio de imaginación 'realista' (aunque pueda parecer contradictorio) pensemos que hemos dividido nuestro rectángulo R con $N = 100$ para la x y también con $M = 100$ para la y . Esto proporcionaría un sistema lineal de (aproximadamente) 10000 ecuaciones e igual número de incógnitas, desde $u_{1,1}$ hasta $u_{99,99}$. Sin más consideraciones, estaríamos fuera de la capacidad de resolución en tiempo real para no importa qué ordenador.

Afortunadamente, la mayor parte de los coeficientes valen 0, todos salvo 5 por ecuación a lo sumo. Esto mejora el aspecto del sistema. Pero aún es mejor si conseguimos reducir las entradas no nulas a alguna banda 'estrecha' alrededor de la diagonal del sistema. Pues bien, hay una ordenación de los nodos y ecuaciones que asegura para la matriz del sistema una estructura banda de anchura $2N - 1$.

En nuestro supuesto hipotético esto significaría una banda de anchura unos 200, lo que en relación al ancho original de 10000 es una ahorro trascendental. Así, tal sistema pasa a ser resoluble en poco tiempo incluso para ordenadores modestos.

La relación que lleva los índices

$$(n, m), \quad n = 1, \dots, N - 1, \quad m = 1, \dots, M - 1$$

a un índice único

$$l, \quad l = 1, \dots, (N - 1)(M - 1)$$

es

$$l = n + (M - 1 - m)(N - 1).$$

Esto proporciona una biyección entre la matriz de los índices dobles (n, m) y la sucesión $1, 2, \dots, (N - 1)(M - 1)$. Ordenando de esta manera las ecuaciones y las incógnitas es fácil ver que se logra el objetivo de no colocar coeficientes no nulos fuera de una banda de anchura $2N - 1$.

Así, $l = 1$ correspondería a $(n, m) = (1, M - 1)$, mientras que el último índice $l = (N - 1)(M - 1)$ correspondería a $(n, m) = (N - 1, 1)$. El orden de asignación va de arriba a abajo y de izquierda a derecha en la representación usual del dominio rectangular.

Es más difícil caracterizar la aplicación recíproca, aunque es importante a la hora de devolver a las incógnitas ya calculadas su notación inicial, que es la que permite colocar los valores en el retículo. Lo que hay que hacer es dividir (división entera por defecto) $l - 1$ entre $N - 1$; esto proporcionará un cociente q y un resto r ; entonces se obtiene (n, m) correspondiente a l como:

$$n = r + 1$$

(valor que está entre 1 y $N - 1$), mientras que

$$m = M - (q + 1)$$

(que está entre 1 y $M - 1$).

Volvamos nuevamente al ejemplo que ya tratamos.

De nuevo el ejemplo:

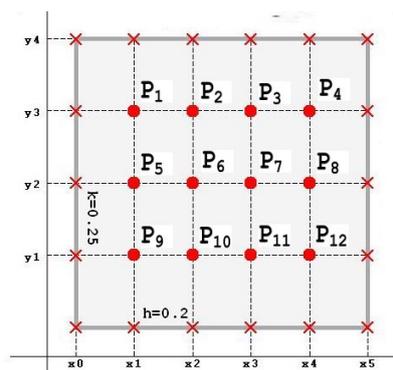
En nuestro ejemplo hay que comenzar por ordenar adecuadamente las 12 ecuaciones correspondientes a los 12 índices

$$(1, 3) \quad (2, 3) \quad (3, 3) \quad (4, 3)$$

$$(1, 2) \quad (2, 2) \quad (3, 2) \quad (4, 2)$$

$$(1, 1) \quad (2, 1) \quad (3, 1) \quad (4, 1)$$

Resulta el orden $(1, 3), (2, 3), (3, 3), (4, 3), (1, 2), (2, 2), (3, 2), (4, 2), (1, 1), (2, 1), (3, 1), (4, 1)$, orden en el que debemos colocar las 12 ecuaciones y las 12 incógnitas y que apreciamos en el gráfico.



Por ejemplo, la ecuación general del método, organizada alrededor del punto (x_x, y_m) es

$$3,28u_{nm} - (u_{n+1m} + u_{n-1m}) - 0,64(u_{nm+1} + u_{nm-1}) = 0$$

y, en particular, la del punto (x_1, y_3) es

$$3,28u_{13} - (u_{23} + u_{03}) - 0,64(u_{14} + u_{12}) = 0.$$

Como $u_{03} = 0$ y $u_{14} = 0,2$, lo que queda es

$$3,28u_{13} - u_{23} - 0,64u_{12} = 0,64 \cdot 0,2 = 0,128,$$

y, escrito para las incógnitas reenumeradas,

$$3,28u_1 - u_2 - 0,64u_5 = 0,128,$$

Esta sería la ecuación que ocupa el primer lugar, ya que a $(1, 3)$ le corresponde $l = 1$.

Así se sigue con las 11 ecuaciones restantes hasta obtener

$$3,28 u_1 - u_2 - 0,64 u_5 = 0,128$$

$$3,28 u_2 - u_3 - u_1 - 0,64 u_6 = 0,256$$

$$3,28 u_3 - u_4 - u_2 - 0,64 u_7 = 0,384$$

$$3,28 u_4 - u_3 - 0,64 u_8 = 1,262$$

$$3,28 u_5 - u_6 - 0,64 u_1 - 0,64 u_9 = 0.$$

$$3,28 u_6 - u_7 - u_5 - 0,64 u_2 - 0,64 u_{10} = 0.$$

$$3,28 u_7 - u_8 - u_6 - 0,64 u_3 - 0,64 u_{11} = 0.$$

$$3,28 u_8 - u_7 - 0,64 u_4 - 0,64 u_{12} = 0,5$$

$$3,28 u_9 - u_{10} - 0,64 u_5 = 0.$$

$$3,28 u_{10} - u_{11} - u_9 - 0,64 u_6 = 0.$$

$$3,28 u_{11} - u_{12} - u_{10} - 0,64 u_7 = 0.$$

$$3,28 u_{12} - u_{11} - 0,64 u_8 = 0,25$$

Para la matriz de este sistema, se ha logrado la estructura banda prevista de anchura $2N - 1 = 9$; la figura que sigue marca los coeficientes no nulos para dejar claro este hecho.

	01	02	03	04	05	06	07	08	09	10	11	12
01	•	•			•							
02	•	•	•			•						
03		•	•	•			•					
04			•	•				•				
05	•				•	•			•			
06		•			•	•	•			•		
07			•			•	•	•			•	
08				•			•	•				•
09					•				•	•		
10						•			•	•	•	
11							•			•	•	•
12								•			•	•

Anchura de banda = $2N-1 = 9$

Resolvemos ahora el sistema. Como es pequeño para cualquier ordenador, no nos fijaremos en una técnica especial (para tamaño muy grande habría que utilizar una buena técnica iterativa). Si que conviene hacer notar que la matriz del sistema es de diagonal dominante por filas, o sea, 3,28 es mayor que la suma de los restantes coeficientes (en valor absoluto) de la fila. Esto garantiza que la matriz es no sólo inversible, sino perfectamente preparada para el uso de técnicas iterativas.

La solución de la que hablábamos es

$$u_1 = 0,15, u_2 = 0,3, u_3 = 0,45, u_4 = 0,6, u_5 = 0,1, u_6 = 0,2$$

$$u_7 = 0,3, u_8 = 0,4, u_9 = 0,05, u_{10} = 0,1, u_{11} = 0,15, u_{12} = 0,2$$

Claro que así no describe bien la solución; lo mejor es reconvertir la notación de un solo subíndice l a dos (n, m) . Resulta

$$\text{así } u_{13} = 0,15, u_{23} = 0,3, u_{33} = 0,45, u_{43} = 0,6$$

$$u_{12} = 0,1, u_{22} = 0,2, u_{32} = 0,3, u_{42} = 0,4$$

$$u_{11} = 0,05, u_{21} = 0,1, u_{31} = 0,15, u_{41} = 0,2$$

Naturalmente, a estos valores hay que añadir los ya asignados sobre el contorno del rectángulo

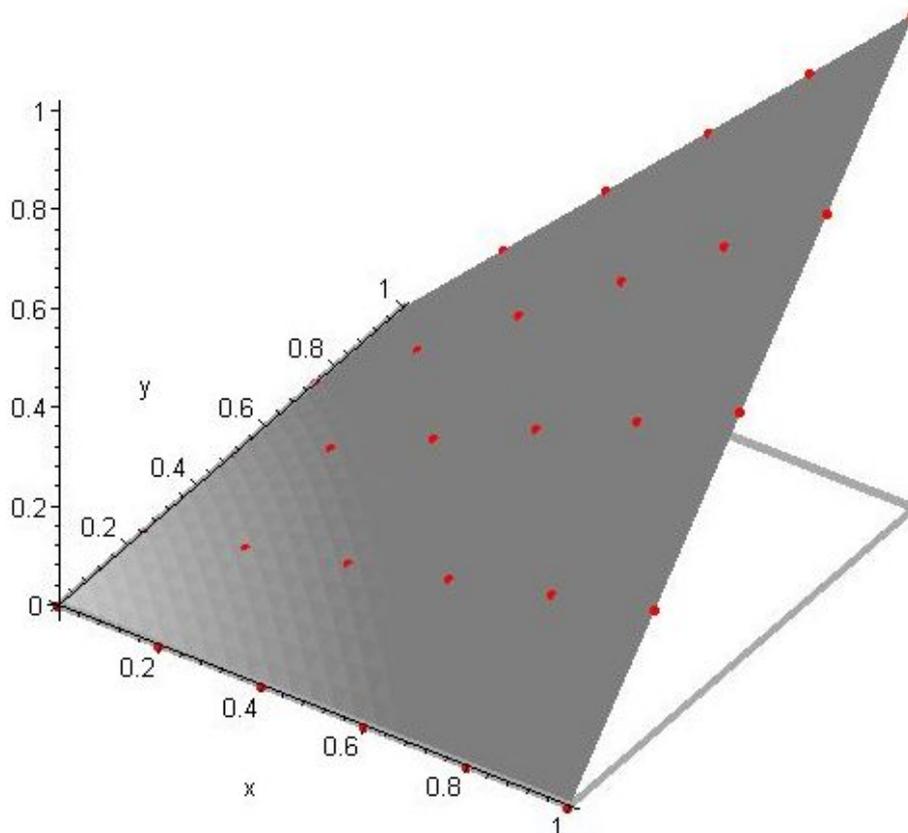
$$u_{00} = 0, u_{01} = 0, u_{02} = 0, u_{03} = 0, u_{04} = 0,$$

$$u_{10} = 0, u_{20} = 0, u_{30} = 0, u_{40} = 0, u_{50} = 0,$$

$$u_{51} = 0,25, u_{52} = 0,5, u_{53} = 0,75, u_{54} = 1.,$$

$$u_{14} = 0,2, u_{24} = 0,4, u_{34} = 0,6, u_{44} = 0,8$$

Ahora es sencillo representar la solución numérica y compararla con la solución exacta. Lo hacemos primero en una figura, con la solución exacta en $R = [0, 1] \times [0, 1]$ acompañada de unos puntos en rojo que marcan los valores aproximados calculados. En ella parece que estos valores están sobre la solución.



Podemos también hacer una comparación analítica de los u_{nm} obtenidos con los verdaderos valores $u(x_n, y_m) = x_n \cdot y_m$. Es obvio que son exactamente iguales.

Parece sorprendente tan exacta aproximación cuando el retículo empleado es bastante 'insuficiente'. Pero el hecho se entiende si recordamos que el error de truncación del método, $\mathcal{O}(h^2) + \mathcal{O}(k^2)$, es más detalladamente

$$\frac{h^2}{12} \frac{\partial^4 u(\xi_n, y_m)}{\partial x^4} + \frac{k^2}{12} \frac{\partial^4 u(x_n, \eta_m)}{\partial y^4}$$

y que el desconocimiento de ξ_n y de η_m carece de importancia, ya que al ser nulas las cuartas derivadas parciales de $u(x, y) = xy$, el valor de los términos despreciados es 0 en esta ocasión (eso no ocurriría evidentemente con otros problemas con solución diferente).

Búsqueda del orden de truncación

Veamos un primer ejemplo de obtención del orden de un método cuando no explicitamos las aproximaciones a las derivadas, o sea, cuando nos proporcionan a priori el método.

Por ejemplo, para el problema elíptico

$$\Delta u(x, y) = \frac{\partial^2 u(x, y)}{\partial x^2} + \frac{\partial^2 u(x, y)}{\partial y^2} = f(x, y),$$

en el dominio rectangular $R = [a, b] \times [c, d]$ y con una condición sobre el contorno $u(x, y) = g(x, y)$, consideramos el retículo habitual

$$\begin{aligned} x_0, x_1, \dots, x_N, \quad x_n &= a + nh, \quad h = (b - a)/N, \\ y_0, y_1, \dots, y_M, \quad y_n &= c + mk, \quad k = (d - c)/M, \end{aligned}$$

y las aproximaciones $u_{nm} \sim u(x_n, y_m)$ a la solución.

para el que pensamos usar el método

$$2(\lambda^2 + 1)u_{nm} - (u_{n+1m} + u_{n-1m}) - \lambda^2(u_{nm+1} + u_{nm-1}) = -h^2 f_{nm},$$

donde $\lambda = h/k$.

Sin duda hemos reconocido ya el método de las diferencias centradas, aunque procederemos como si fuese un método totalmente desconocido.

Igualamos a 0 la expresión del método, o sea, lo escribimos como

$$P(h, k)(n, m) = 0,$$

en una forma adecuada; no debemos olvidar que podemos multiplicar o dividir por expresiones conservando la igualdad con 0. Lo de 'adecuada' hace mención a tratar de deshacer los cambios realizados para obtener la versión abreviada con λ .

En nuestro caso, dividiremos la ecuación del método por h^2 y luego agruparemos todo en el primer miembro. Como

$$\frac{2(\lambda^2 + 1)}{h^2} = \frac{2}{h^2} + \frac{2}{k^2} \quad \text{y} \quad \frac{\lambda^2}{h^2} = \frac{1}{k^2},$$

lo que obtenemos es

$$-\frac{2u_{nm}}{h^2} - \frac{2u_{nm}}{k^2} + \frac{u_{n+1m} + u_{n-1m}}{h^2} + \frac{u_{nm+1} + u_{nm-1}}{k^2} - f_{nm} = 0$$

y, agrupando los denominadores,

$$\frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} + \frac{u_{nm+1} - 2u_{nm} + u_{nm-1}}{k^2} - f_{nm} = 0,$$

o sea, hemos puesto

$$P(h, k)(n, m) = \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} + \frac{u_{nm+1} - 2u_{nm} + u_{nm-1}}{k^2} - f_{nm}.$$

Cambiamos en $P(h, k)(n, m)$ las aproximaciones u_{nm} por los valores reales $u(x_n, y_m)$ de la solución $u(x, y)$ del problema (solución desconocida, claro) y al resultado lo llamamos $P(h, k)(x_n, y_m)$. El error de truncación es justamente esta expresión y su valor en términos de infinitésimos de potencias de h y de k se obtiene desarrollando valores como

$$(x_{n+1}, y_m) = (x_n + h, y_m)$$

y otros similares en potencias de h y de k alrededor de (x_n, y_m) , o sea, de $h = 0, k = 0$

Tras el desarrollo de TAYLOR y las operaciones de simplificación, obtendremos un resultado de una forma parecida a

$$P(h, k)(x_n, y_m) = \mathcal{O}(h^p) + \mathcal{O}(k^q)$$

que es lo que consideraremos el orden de truncación del método.

Una observación; en el presente ejemplo, como ya sabemos cómo hemos realizado la construcción del método, somos conscientes de que el orden de truncación resultante será $\mathcal{O}(h^2) + \mathcal{O}(k^2)$. Esto es sólo porque el ejemplo ya es conocido.

Comenzamos poniendo

$$P(h, k)(x_n, y_m) = \frac{u(x_{n+1}, y_m) - 2u(x_n, y_m) + u(x_{n-1}, y_m))}{h^2} + \\ + \frac{u(x_n, y_{m+1}) - 2u(x_n, y_m) + u(x_n, y_{m-1}))}{k^2} - f(x_n, y_m).$$

y proseguimos con el desarrollo empleando la notación breve u_x , u_{xx} , etc. para las derivadas parciales. Además, suprimiremos la expresión del punto (x_n, y_m) (que aparece muchas veces) para acortar las fórmulas. Tenemos

$$u(x_{n+1}, y_m) = u(x_n + h, y_m) = u + h u_x + \frac{h^2}{2} u_{xx} + \frac{h^3}{6} u_{xxx} + \frac{h^4}{24} u_{xxxx}(\xi_n, y_m),$$

o sea

$$u(x_{n+1}, y_m) = u(x_n + h, y_m) = u + h u_x + \frac{h^2}{2} u_{xx} + \frac{h^3}{6} u_{xxx} + \mathcal{O}(h^4).$$

¿Hasta dónde se prosigue en el desarrollo? La experiencia y algunas pruebas lo acaban aconsejando. A continuación

$$-2 u(x_n, y_m) = -2 u \quad y$$

y

$$u(x_{n-1}, y_m) = u(x_n - h, y_m) = u - h u_x + \frac{h^2}{2} u_{xx} - \frac{h^3}{6} u_{xxx} + \mathcal{O}(h^4).$$

Ya tenemos entonces que el primer término (todo él, con la división por h^2 incluida) de $P(h, k)(x_n, y_m)$ vale

$$(1) = u_{xx}(x_n, y_m) + \mathcal{O}(h^2).$$



Es prácticamente idéntico probar que

$$(2) = u_{yy}(x_n, y_m) + \mathcal{O}(k^2)$$

para el segundo término de $P(h, k)(x_n, y_m)$.

Juntando todo los cálculos se obtiene

$$P(h, k)(x_n, y_m) = u_{xx}(x_n, y_m) + \mathcal{O}(h^2) + u_{yy}(x_n, y_m) + \mathcal{O}(k^2) - f(x_n, y_m).$$

Pero el hecho de que $u(x, y)$ sea la solución del problema nos asegura que

$$u_{xx}(x_n, y_m) + u_{yy}(x_n, y_m) - f(x_n, y_m) = 0,$$

y finalmente

$$P(h, k)(x_n, y_m) = \mathcal{O}(h^2) + \mathcal{O}(k^2),$$

que es lo que esperábamos encontrar.

Así pues $\mathcal{O}(h^2) + \mathcal{O}(k^2)$ es el orden del método

$$2(\lambda^2 + 1)u_{nm} - (u_{n+1m} + u_{n-1m}) - \lambda^2(u_{nm+1} + u_{nm-1}) = -h^2 f_{nm},$$

donde $\lambda = h/k$.



Ejercicios de

(11) Las Ecuaciones elípticas

Ejercicios de

(11) Las Ecuaciones elípticas

Ejercicio 11-01 (del EXAMEN ORDINARIO DE 19ENE12)

Para el problema elíptico

$$\frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2} = f(x,y)$$

en el dominio $[a,b] \times [c,d]$, con una condición de contorno adecuada,

y con el retículo habitual

$$x_n = a + h n, \quad h = \frac{b-a}{N},$$

$$y_m = c + k m, \quad k = \frac{d-c}{M},$$

ponemos como es habitual $u_{nm} = u(x_n, y_m)$ y consideramos el método en diferencias,

dado por

$$6(4\lambda^2 + 5)u_{nm} + (u_{n+2m} - 16u_{n+1m} - 16u_{n-1m} + u_{n-2m}) + \\ - 12\lambda^2(u_{nm+1} + u_{nm-1}) = -12h^2 f_{nm}$$

Dígase qué orden posee el error de truncación de dicho método.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(11) Las Ecuaciones elípticas

Ejercicio 11-02 (del EXAMEN EXTRAORDINARIO de 07JUL12)

Para el problema elíptico

$$\frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2} = f(x,y)$$

en el dominio $[a,b] \times [c,d]$, con una condición de contorno adecuada,

y con el retículo habitual

$$x_n = a + h n, \quad h = \frac{b-a}{N},$$

$$y_m = c + k m, \quad k = \frac{d-c}{M},$$

ponemos como es habitual $u_{nm} = u(x_n, y_m)$ y consideramos el método en diferencias,

dado por

$$2(\lambda^2 + 4) u_{nm} - 4(u_{n+1m} + u_{n-1m}) - \lambda^2(u_{nm+2} + u_{nm-2}) = -4h^2 f_{nm}$$

Dígase qué orden posee el error de truncación de dicho método.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(11) Las Ecuaciones elípticas

Ejercicio 11-03

Para el problema elíptico

$$\frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2} = f(x,y)$$

en el dominio $[a,b] \times [c,d]$, con una condición de contorno adecuada,

y con el retículo habitual

$$x_n = a + h n, \quad h = \frac{b-a}{N},$$

$$y_m = c + k m, \quad k = \frac{d-c}{M},$$

pero ahora supondremos que, en este retículo, $h = k$, lo que es esencial para lo que sigue.

Se considera poniendo $u_{nm} = u(x_n, y_m)$ el método en diferencias, llamado de

nueve puntos, dado por

$$\begin{aligned} & \frac{1}{6} (u_{n+1m+1} + u_{n+1m-1} + u_{n-1m+1} + u_{n-1m-1}) + \\ & + \frac{2}{3} (u_{n+1m} + u_{n-1m} + u_{nm+1} + u_{nm-1}) - \\ & - \frac{10}{3} u_{nm} = \frac{h^2}{12} (f_{n+1m} + f_{n-1m} + f_{nm+1} + f_{nm-1} + 8f_{nm}) \end{aligned}$$

Dígase qué orden posee el error de truncación de dicho método en un retículo en el que,

como hemos dicho, $h = k$.

```
[ > restart:
[ > interface(labeling=false):
```


12. Las Ecuaciones parabólicas

- 1 El problema parabólico y el método de las diferencias hacia adelante
- 2 Estabilidad en el caso parabólico
- 3 Un método incondicionalmente estable
- 4 Error de truncación y orden de un método
- 5 Condición necesaria de estabilidad; el procedimiento de FOURIER
- 6 El método de CRANK-NICOLSON

El problema parabólico

Abordaremos ahora un tipo de problemas que suelen ser dependientes del tiempo, y que supondremos ya en la forma

$$\frac{\partial u(x, t)}{\partial t} = \alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2}, \quad x \in [0, l], t \in [0, +\infty),$$

ecuación en un dominio D que es el rectángulo 'semiinfinito'

$$D = [0, l] \times [0, +\infty).$$

El problema se completa con la condición inicial

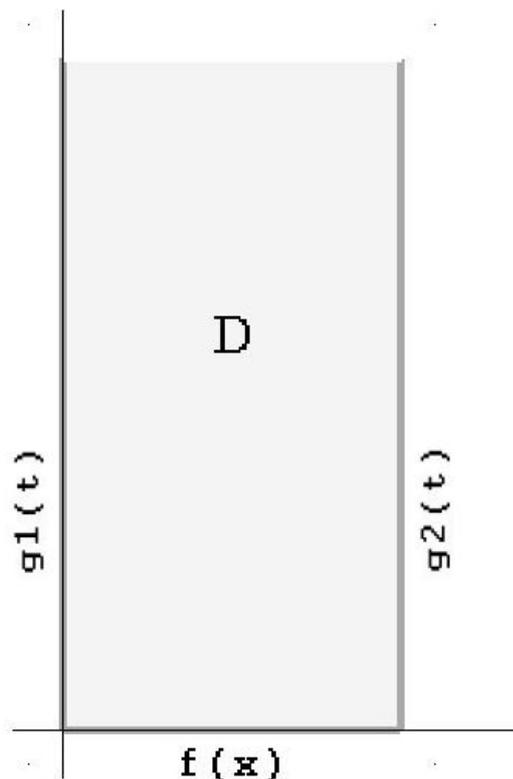
$$u(x, 0) = f(x), \quad x \in [0, l],$$

y las condiciones de contorno (sobre el contorno físico tanto en $x = 0$ como en $x = l$)

$$u(0, t) = g_1(t), \quad u(l, t) = g_2(t), \quad t \in [0, +\infty).$$



Recordemos con una figura el diseño del dominio:



El método de las diferencias hacia adelante

Consideramos en D el retículo definido como explicamos. Para la x ,

$$x_0, x_1, \dots, x_N, \quad x_n = n h, \quad h = \frac{l}{N},$$

Para t no es posible, sin embargo, hacer una división para calcular la longitud del subintervalo. Lo que se hace es tomar una longitud $k > 0$ arbitraria y fabricar con ella los t_m

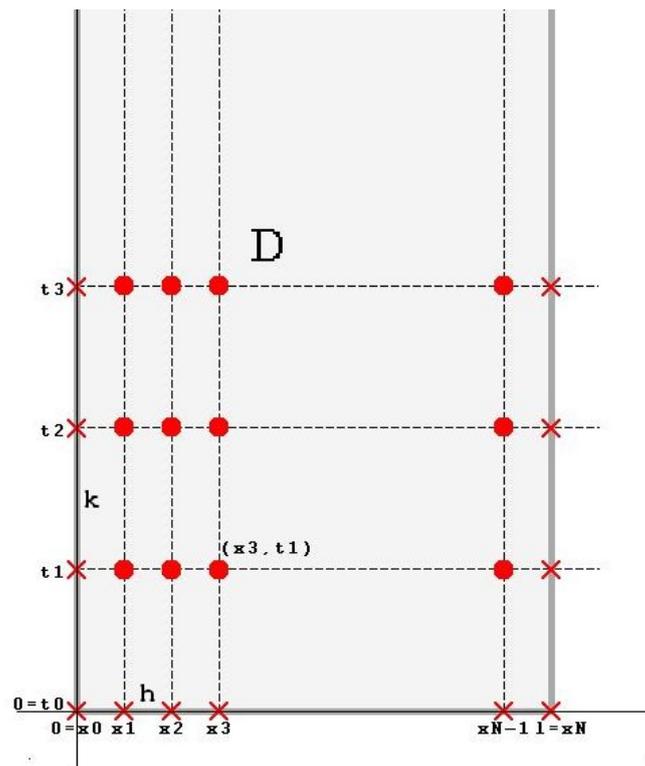
$$t_0, t_1, t_2, \dots, \quad t_m = m k, \quad k > 0.$$

Ahora hay un retículo bidimensional formado en principio por una cantidad infinita de nodos (x_n, t_m) en los que intentaremos aproximar la solución del problema por valores calculados

$$u_{nm} \sim u(x_n, t_m).$$

Comenzaremos por construir un método, el llamado método **de las diferencias hacia adelante**.

Observemos en primer lugar el dominio D con sus nodos (x_n, t_m) , $n = 0, 1, \dots, N - 1, N$, $m = 0, 1, 2, \dots$



Marcados con un círculo rojo están los llamados 'nodos interiores' del rectángulo. Son aquellos con

$$n = 1, \dots, N - 1 \quad \text{y} \quad m = 1, 2, \dots$$

Los restantes, marcados con un aspa, son los 'nodos del contorno'. Son aquellos con

$$m = 0 \quad \text{y con} \quad n = 0 \quad \text{o} \quad n = N.$$

Cada nodo interior va a dar origen a una de las ecuaciones en diferencias que van a formar el método numérico. Los nodos exteriores sirven, como antes, para proporcionar procedimientos de arranque del método.

El nodo (x_n, t_m) permite establecer una ecuación escribiendo la expresión

$$\frac{\partial u(x_n, t_m)}{\partial t} - \alpha^2 \frac{\partial^2 u(x_n, t_m)}{\partial x^2} = 0$$

de manera que sólo intervengan valores de la solución u y no de sus derivadas.

Este proceso nos es ya conocido. Fijado x_n , se considera la función de una variable $f(t) = u(x_n, t)$, para la que

$$\frac{\partial u(x_n, t)}{\partial t} = f'(t) \quad \text{y} \quad \frac{\partial u(x_n, t_m)}{\partial t} = f'(t_m)$$

En este caso (construcción del método de las diferencias hacia adelante), hacemos uso de la fórmula de aproximación

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(\xi) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h)$$

para la citada f (una diferente para cada x_n), $x = t_m$ y h el k del retículo y recordando que $t_{m+1} = t_m + k$, obtenemos

$$\frac{\partial u(x_n, t_m)}{\partial t} = \frac{u(x_n, t_{m+1}) - u(x_n, t_m)}{k} - \frac{k}{2} \frac{\partial^2 u(x_n, \eta_m)}{\partial t^2}.$$

Para la segunda derivada respecto de x repetimos exactamente el proceso seguido en el anterior capítulo, obteniendo

$$\frac{\partial^2 u(x_n, t_m)}{\partial x^2} = \frac{u(x_{n+1}, t_m) - 2u(x_n, t_m) + u(x_{n-1}, t_m))}{h^2} - \frac{h^2}{12} \frac{\partial^4 u(\xi_n, t_m)}{\partial x^4}.$$

Reuniendo ambos resultados

$$\begin{aligned} \frac{u(x_n, t_{m+1}) - u(x_n, t_m)}{k} - \alpha^2 \frac{u(x_{n+1}, t_m) - 2u(x_n, t_m) + u(x_{n-1}, t_m))}{h^2} &= \\ &= \frac{k}{2} \frac{\partial^2 u(x_n, \eta_m)}{\partial t^2} + \alpha^2 \frac{h^2}{12} \frac{\partial^4 u(\xi_n, t_m)}{\partial x^4}. \end{aligned}$$

Ahora es cuestión de definir el método de las diferencias hacia adelante como el dado por la ecuación

$$\frac{u_{nm+1} - u_{nm}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0,$$

lo que nos lleva a un error de truncación

$$\mathcal{O}(h^2) + \mathcal{O}(k).$$



Otra vez mejoramos la presentación de la ecuación en diferencias que caracteriza el método de las diferencias hacia adelante, poniendo

$$\lambda = \frac{\alpha^2 k}{h^2};$$

(suele ser lo habitual en el caso parabólico). Reemplazando este λ en la ecuación que teníamos, resulta

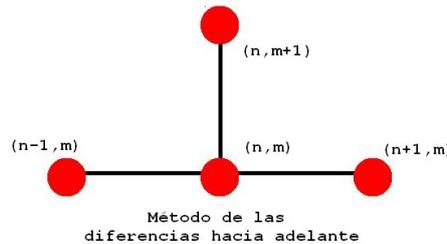
$$u_{nm+1} = (1 - 2\lambda)u_{nm} + \lambda(u_{n+1m} + u_{n-1m})$$

para la ecuación centrada en (x_n, t_m) .

Es una ecuación, claro, pero vamos a ver que se puede usar como fórmula explícita sin tener que resolver ningún sistema lineal.



La **molécula computacional** de este método es



siendo el punto de abajo y del centro el que ha servido para crear la ecuación del método.

En lo que respecta a la aportación de la condición inicial y las condiciones de contorno a la asignación de valores a las aproximaciones que no deben ser incógnitas, tendremos ahora

$$\begin{aligned} u_{n0} &= f_n, & n &= 0, \dots, N, \\ u_{0m} &= g_{1m}, & m &= 0, 1, 2, \dots, \\ u_{Nm} &= g_{2m}, & m &= 0, 1, 2, \dots, \end{aligned}$$

donde hemos puesto $f_n = f(x_n)$, $g_{1m} = g_1(t_m)$ y $g_{2m} = g_2(t_m)$.



Ya hemos hablado de utilizar la ecuación general del método como fórmula explícita. La idea es la siguiente: cada tres valores consecutivos u_{n-10} , u_{n0} , u_{n+10} permiten usar la fórmula para calcular u_{n1} . O sea, el uso ordenado de los valores $u_{n0} = f_n$, $n = 0, \dots, N$ dados por la condición inicial permite calcular los de

$$u_{n1}, \quad n = 1, \dots, N - 1$$

(o sea, las aproximaciones para puntos interiores del siguiente 'piso', el de $m=1$). Valores que se completan con $u_{01} = g_{11}$ y con $u_{N1} = g_{1N}$ (condiciones de contorno). De esta manera se pueden ir construyendo progresivamente todas las aproximaciones u_{nm} 'piso a piso' hasta llegar al valor de t_m que nos interese.



Desde varios puntos de vista es más simple y eficaz usar una notación matricial. Para ello definimos el vector $\mathbf{u}^{(m)}$, $m = 0, 1, 2, \dots$ como

$$\mathbf{u}^{(m)} = (u_{1m}, u_{2m}, \dots, u_{N-1m}),$$

que es el vector que agrupa las aproximaciones para el valor de t_m , pero únicamente en los nodos interiores.

Gracias a esta notación, vamos a presentar también este método (y también otros) en su versión matricial, lo que nos permitirá seguir más claramente el funcionamiento del método y, sobre todo, la expansión o disipación numérica de los errores que se introduzcan.

Además, nos hace falta situarnos en el caso en que las condiciones de contorno son 'homogéneas', esto es, idénticamente nulas.

En lo que sigue, supondremos para la formulación matricial y el posterior análisis de la estabilidad que las condiciones de contorno son homogéneas

$$u(0, t) \equiv 0, \quad u(l, t) \equiv 0, \quad t \in [0, +\infty),$$

o sea que $g_1(t) \equiv 0$ y $g_2(t) \equiv 0$. Aunque con este supuesto el tratamiento pierde generalidad, esta suposición nos coloca en una descripción matricial de método bastante sencilla.

Es verdad que un tratamiento práctico de la estabilidad confirma que son ciertos los resultados que se desprenden de este estudio del caso de condiciones homogéneas de contorno.

Pues bien, en este caso homogéneo, la notación vectorial introducida permite escribir matricialmente la ecuación general del método

$$u_{nm+1} = (1 - 2\lambda)u_{nm} + \lambda(u_{n+1m} + u_{n-1m})$$

en la forma

$$\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)},$$

donde A es la matriz

$$A = \begin{bmatrix} 1 - 2\lambda & \lambda & 0 & \cdots \\ \lambda & 1 - 2\lambda & \lambda & \cdots \\ 0 & \lambda & 1 - 2\lambda & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

y los vectores $\mathbf{u}^{(m+1)}$ y $\mathbf{u}^{(m)}$ se multiplican como vectores columna (o sea, lo usual aunque por comodidad se presenten en horizontal). Lo que decimos es que

$$\begin{bmatrix} u_{1m+1} \\ u_{2m+1} \\ \vdots \\ u_{N-1m+1} \end{bmatrix} = \begin{bmatrix} 1 - 2\lambda & \lambda & 0 & \cdots \\ \lambda & 1 - 2\lambda & \lambda & \cdots \\ 0 & \lambda & 1 - 2\lambda & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} u_{1m} \\ u_{2m} \\ \vdots \\ u_{N-1m} \end{bmatrix}.$$

y este proceso de asignaciones, escrito en forma matricial, avanza a partir del vector

$$\mathbf{u}^{(0)} = (u_{10}, u_{20}, \dots, u_{N-10}) = (f_1, f_2, \dots, f_{N-1}),$$

conocido a partir de la condición inicial.

No es fácil describir el aspecto de la matriz A en la forma en que la hemos escrito (o sea, ocupando poco espacio). Conviene decir además que su tamaño es $(N - 1) \times (N - 1)$ y que es tridiagonal, o sea, que es 0 todo lo que hay fuera de las tres diagonales centrales. En la diagonal aparece el valor $1 - 2\lambda$, y en la superdiagonal y la subdiagonal aparece el valor λ también repetido.

Llegado el momento de estudiar algunos aspectos de esta matriz A , como por ejemplo sus valores propios, nos encontramos para esta matriz con dos escollos. El primero es su tamaño 'inconcreto'; nos referimos a que es $(N - 1) \times (N - 1)$ y N puede tomar cualquier valor entero. El segundo es el hecho de que los elementos de A no son número explícitos, sino que interviene un parámetro λ que hay que arrastrar en los cálculos teóricos (en los numéricos, λ tiene algún valor concreto).

Ejemplo 1

Examinemos los resultados que se obtienen en un ejemplo que se encuentra en el programa realizado en 'Maple'

Cap12_ejemplo1.mws para el problema parabólico

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0,$$

en el dominio $x \in [0, 1]$, $t \in [0, +\infty)$, con condición inicial $u(x, 0) = \text{sen}(\pi x)$, $x \in [0, 1]$, y condiciones homogéneas de contorno.

Este problema tiene como única solución la función

$$u(x, y) = e^{-\pi^2 t} \text{sen}(\pi x).$$

Para su evaluación numérica construimos un retículo dividiendo $[0, 1]$ para x en $N = 10$ subintervalos, tomando $h = 1/N = 0,1$ y luego $x_n = 0 + n h$, lo que hace

$x_0 = 0, x_1 = 0,1, x_2 = 0,2, x_3 = 0,3, x_4 = 0,4, x_5 = 0,5,$
 $x_6 = 0,6, x_7 = 0,7, x_8 = 0,8, x_9 = 0,9, x_{10} = 1.$

Para t , comenzamos tomando $k = 0,0005 = 5 \cdot 10^{-4}$, lo que hace
 $t_0 = 0, t_1 = 0,0005, t_2 = 0,001, \dots$

Con estos valores, $\lambda = \alpha^2 k/h^2 = 0,05$ y la matriz A es

$$A = \begin{bmatrix} 0,9 & 0,05 & 0 & \dots \\ 0,05 & 0,9 & 0,05 & \dots \\ 0 & 0,05 & 0,9 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Integramos el problema hasta $t_{1000} = 0,5$.

Para el contorno de x tenemos que $u_{0m} = 0$ y $u_{10m} = 0$ para todo m (condiciones homogéneas).

Los valores de arranque son los $u_{n0} = f_n = \text{sen}(\pi 0,1 n)$, o sea
 $u_{10} = 0,3091, u_{20} = 0,5879, u_{30} = 0,8091, u_{40} = 0,9512, u_{50} = 1, \dots,$
 $u_{60} = 0,9510, u_{70} = 0,8091, u_{80} = 0,5872, u_{90} = 0,3085$

o sea,

$$\mathbf{u}^{(0)} = (0,3091, 0,5879, 0,8091, 0,9512, 1, \dots, 0,9510, 0,8091, 0,5872, 0,3085).$$

Ahora repetimos a partir de $\mathbf{u}^{(0)}$ 1000 veces la iteración
 $\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)}$, $m = 1, 2, \dots, 1000$ para conseguir $\mathbf{u}^{(1000)}$ que da
los valores aproximados de la solución en $t_{1000} = 0,5$, que es

$$\mathbf{u}^{(1000)} = (0,002289, 0,004352, 0,005988, 0,007039, 0,007401, \dots, 0,007040, 0,005989, 0,004352, 0,002289),$$

además, claro, de los valores nulos en los extremos. Así pues

$$\begin{aligned}
 u(0, 0, 5) &\sim u_{0\ 1000} = 0. & u(0, 1, 0, 5) &\sim u_{1\ 1000} = 0,002289 \\
 u(0, 2, 0, 5) &\sim u_{2\ 1000} = 0,004352 & u(0, 3, 0, 5) &\sim u_{3\ 1000} = 0,005988 \\
 u(0, 4, 0, 5) &\sim u_{4\ 1000} = 0,007039 & u(0, 5, 0, 5) &\sim u_{5\ 1000} = 0,007401 \\
 u(0, 6, 0, 5) &\sim u_{6\ 1000} = 0,007040 & u(0, 7, 0, 5) &\sim u_{7\ 1000} = 0,005989 \\
 u(0, 8, 0, 5) &\sim u_{8\ 1000} = 0,004352 & u(0, 9, 0, 5) &\sim u_{9\ 1000} = 0,002289 \\
 u(1, 0, 5) &\sim u_{10\ 1000} = 0. ,
 \end{aligned}$$

Al conocer la solución exacta, vamos a comparar estos valores aproximados (obtenidos con 4 cifras decimales significativas con redondeo) con los verdaderos valores de la solución. Por no alargarnos, el más alejado es

$u_{5\ 1000} = 0,007401$ para $u(0,5, 0,5) = 0,007183$ con error = 0,000218, o sea, es un error razonable para las cifras con las que se ha trabajado. Recuérdese que la expresión $\mathcal{O}(h^2) + \mathcal{O}(k)$ caracteriza este error y que, en este caso

$$h^2 + k = 0,1^2 + 0,0005 = 0,01005.$$



Pero, como segundo caso, el ejemplo emplea $k = 0,01$, lo que ahora hace $t_0 = 0.$, $t_1 = 0,01$, $t_2 = 0,02, \dots$

Con estos valores, $\lambda = \alpha^2 k / h^2 = 1.$ y la matriz A es

$$A = \begin{bmatrix} -1. & 1. & 0 & \dots \\ 1. & -1. & 1. & \dots \\ 0 & 1. & -1. & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Integramos el problema hasta el mismo valor de t , que ahora es $t_{50} = 0,5$. No varían ni las condiciones de contorno (por valer siempre 0) ni las iniciales, o sea, contamos con el mismo vector $\mathbf{u}^{(0)}$ para arrancar. La integración numérica consiste ahora en realizar 50 iteraciones de $\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)}$, $m = 1, 2, \dots, 50$ a partir de $\mathbf{u}^{(0)}$ y con la última matriz A que hemos calculado para este caso. Conseguimos así el vector

$$\begin{aligned}
 \mathbf{u}^{(50)} = & (1,598 \cdot 10^{18}, -3,051 \cdot 10^{18}, 4,224 \cdot 10^{18}, -5,002 \cdot 10^{18}, 5,302 \cdot 10^{18} \\
 & , -5,084 \cdot 10^{18}, 4,356 \cdot 10^{18}, -3,183 \cdot 10^{18}, 1,679 \cdot 10^{18}),
 \end{aligned}$$



que contiene las aproximaciones para $t_{50} = 0,5$ como antes. Es evidente que ahora son muy diferentes de los valores de la verdadera solución $u(x, y) = e^{-\pi^2 t} \text{sen}(\pi x)$ del problema. De hecho, el error en el mismo valor que antes es ahora

$$u_{5,50} - u(0,5, 0,5) = 5,302 \cdot 10^{18},$$

y en los otros nodos los errores son similares.

La primera idea puede ser culpar de este hecho al error de truncación, pero ya hemos dicho que era $\mathcal{O}(h^2) + \mathcal{O}(k)$ y que, en el primer caso caso

$$h^2 + k = 0,1^2 + 0,0005 = 0,01005,$$

mientras que este segundo

$$h^2 + k = 0,1^2 + 0,01 = 0,02.$$

Esto justifica, a lo sumo, errores de tamaño doble en el segundo caso, pero no lo que sucede.

Lo que de hecho sucede es un problema que recibe el nombre de 'inestabilidad'; veamos a qué nos referimos con estabilidad o inestabilidad en este caso. La estabilidad, entendida como buen comportamiento en relación con la transmisión de los errores, es lo que 'falla' en el segundo caso. Aunque matemáticamente considerados los datos y las operaciones son exactos, de hecho ni datos ni operaciones lo son si se emplea un ordenador. Es verdad que se producen errores muy pequeños; por ejemplo, el valor de un dato no se puede tal vez representar exactamente en nuestro ordenador, pero el número que sustituye al original es verdaderamente 'muy parecido'. El error relativo que se comete no supera 10^{-6} en precisión simple y hasta 10^{-12} o 10^{-20} en precisiones doble o cuádruple.

Pero lo importante es cómo se trasmite y tal vez se amplifica cada uno de estos errores en las operaciones sucesivas.

Así, el error $\varepsilon^{(0)} = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{N-1})$ en $\mathbf{u}^{(0)} = (u_{10}, u_{20}, \dots, u_{N-10})$, hace que el arranque del método se haga con $\tilde{\mathbf{u}}^{(0)} = \mathbf{u}^{(0)} + \varepsilon^{(0)}$ en el lugar de $\mathbf{u}^{(0)}$. Ignorando el efecto de otros errores, el valor obtenido tras la primera iteración es

$$\tilde{\mathbf{u}}^{(1)} = A\tilde{\mathbf{u}}^{(0)} = A(\mathbf{u}^{(0)} + \varepsilon^{(0)}) = \mathbf{u}^{(1)} + A\varepsilon^{(0)},$$

o sea, se transmite tras la primera iteración un error de $\varepsilon^{(1)} = A\varepsilon^{(0)}$. Es bastante sencillo ver por inducción que el error transmitido por $\varepsilon^{(0)}$ tras la iteración número m es

$$\varepsilon^{(m)} = A^m \varepsilon^{(0)}.$$

Este vector, cuando m crece ¿tenderá hacia infinito (en alguna norma)?, o ¿tenderá hacia 0?, o ¿se mantendrá acotado? Es sabido que las alternativas no dependen de $\varepsilon^{(0)}$, sino de la matriz A .

Así, cuando sucede una de las dos cosas equivalentes siguientes:

- (i) $\|A\| < 1$ para alguna norma de tipo matricial, o
- (ii) $\rho(A) < 1$ ($\rho(A)$ es el 'radio espectral' de A),

entonces, sin importar cuál sea el vector $\varepsilon^{(0)}$, la sucesión $\varepsilon^{(m)} = A^m \varepsilon^{(0)}$ converge a 0 cuando m crece hacia infinito.

Recordemos que el **radio espectral de A** es el mayor de los módulos de los valores propios de la matriz A . Y que garantizar que este radio permite asegurar que cualquier error va a terminar por desaparecer con las sucesivas iteraciones. En definitiva, nos permite garantizar la **estabilidad del método**. En otras palabras, $\rho(A) < 1$ es una **condición suficiente de estabilidad**

En la práctica, lo que haremos es probar que los valores propios de A poseen todos un módulo < 1 (esto equivale a $\rho(A) < 1$).

Como veremos, el radio espectral de A depende de λ . Por lo tanto, habrá en general valores de λ para los que un método sea estable y otros para los que sea inestable.

Cuando buscamos condiciones sobre λ que aseguren que $\rho(A) < 1$ y, por lo tanto, que el método sea estable, estamos usando lo que suele llamarse **procedimiento matricial** para el estudio de la estabilidad. Como hemos dicho, este procedimiento puede encontrar condiciones suficientes (sobre λ) para la estabilidad de un método.

Vamos a estudiar la estabilidad del método de las diferencias hacia adelante. Pero nos encontramos con que no es sencillo estudiar los valores propios de la matriz A . Esto se debe a que A es de tamaño indeterminado $(N - 1)$ y a que incluye parámetros y no sólo números. Las cosas mejoran cuando relacionamos A con la matriz

$$B = \begin{bmatrix} 2 & -1 & 0 & \cdots \\ -1 & 2 & -1 & \cdots \\ 0 & -1 & 2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

que ya no contiene el parámetro λ .

En primer lugar, es claro que

$$A = I - \lambda B \quad (I \text{ es la matriz unidad } (N - 1) \times (N - 1)).$$

Por otra parte es sencillo probar (aunque no inmediato) que los valores propios de B son

$$4 \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N - 1.$$

De hecho, sólo usaremos el hecho de que los valores propios de B son números reales (la matriz B es simétrica), distintos, y situados en el intervalo $(0, 4)$; este resultado se obtiene aplicando el conocido como 'teorema de los círculos de GERSCHGORIN'. Esto proporciona para la matriz A los valores propios

$$\mu_n = 1 - 4\lambda \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N - 1.$$

Para que la matriz A del método de las diferencias hacia adelante verifique $\rho(A) < 1$, basta con que, para $n = 1, \dots, N - 1$,

$$\left| 1 - 4\lambda \left(\operatorname{sen} \frac{n\pi}{2N} \right)^2 \right| < 1,$$

y para ello basta que

$$0 < \lambda \left(\operatorname{sen} \frac{n\pi}{2N} \right)^2 < \frac{1}{2},$$

y para ello que

$$\lambda \left(\operatorname{sen} \frac{n\pi}{2N} \right)^2 < \frac{1}{2},$$

pues la otra desigualdad es siempre cierta.

Finalmente,

$$\lambda < \frac{1}{2}$$

es una condición suficiente para lo anterior. De hecho es la mejor desigualdad que es posible obtener ya que, para N arbitrario, el valor del sen que aparece puede ser tan próximo a 1 como queramos.

En definitiva, el procedimiento matricial nos dice que cualquier valor de $\lambda < 1/2$ produce que el método de las diferencias hacia adelante sea estable. Nada dice de los valores $\lambda > 1/2$ aunque veremos enseguida que en un ejemplo que ya hemos tratado uno de estos valores hace que dicho método se comporte de manera inestable.

En el ejemplo que hemos seguido se tenía $\lambda = 0,05 < 1/2$ en el primero de los casos tratados, y la estabilidad estaba asegurada. Sin embargo, en el segundo caso, $\lambda = 1 > 1/2$ y comprobamos experimentalmente el comportamiento inestable.

Hay otros métodos que no poseen este problema de estabilidad; vamos enseguida a estudiar uno de ellos, el llamado 'de las diferencias hacia atrás'.

Como $\lambda = \alpha^2 k/h^2$, una desigualdad como la anterior para λ exige que el valor de k sea mucho más pequeño que h , que figura elevado al cuadrado. En el ejemplo que vamos siguiendo, en el que $h = 0,1$, podemos garantizar la estabilidad haciendo que

$$k < h^2/2 \alpha^2 = 0,1^2/2 = 0,005$$

Un valor como este lleva a aumentar mucho el número de nodos en la variable t (la vertical), haciendo muy costosa la integración numérica en operaciones (y en los errores que estas propagan). Por eso es de interés plantearse alternativas para las que la estabilidad no dependa del pequeño tamaño de k , haciendo que sólo haya que preocuparse de la pequeñez de h y de k en relación al error de truncación.

Pasamos a ese método alternativo.

Un método incondicionalmente estable

Para construir el **método de las diferencias hacia atrás** procedemos como en el caso anterior, expuesto al comienzo del capítulo. Pero ahora, la aproximación de la derivada en t se hace usando la técnica que calcula

$$\frac{\partial u(x_n, t_m)}{\partial t} = f'(t_m)$$

con

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h)$$

para la citada f , para $x = t_m$ y ahora para h el $-k$ del retículo en lugar de k como entonces. Se obtiene así

$$\frac{\partial u(x_n, t_m)}{\partial t} = \frac{u(x_n, t_m) - u(x_n, t_{m-1})}{k} + \frac{k}{2} \frac{\partial^2 u(x_n, \eta_m)}{\partial t^2}.$$

Esto nos lleva ahora a la ecuación en diferencias

$$\frac{u_{nm} - u_{nm-1}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0,$$

con error de truncación

$$\mathcal{O}(h^2) + \mathcal{O}(k),$$

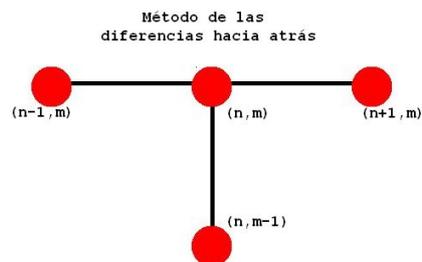
como antes. Para

$$\lambda = \frac{\alpha^2 k}{h^2};$$

las ecuaciones principales son ahora

$$u_{nm-1} = (1 + 2\lambda)u_{nm} - \lambda(u_{n+1m} + u_{n-1m}),$$

y la molécula computacional



La ecuación permite ahora calcular explícitamente u_{nm-1} en función de u_{nm} , u_{n+1m} y u_{n-1m} . Por ello se califica este método como método **de las diferencias hacia atrás**.

Pero no hay unos valores de arranque, al menos en apariencia. Lo ideal sería empezar con valores conocidos correspondientes al 'piso superior' del dominio, cosa que no existe. Los datos de arranque no han cambiado: son las condiciones de contorno (homogéneas en nuestra discusión) y, sobre todo, los valores que proporciona la condición inicial, que se agrupan en

$$\mathbf{u}^{(0)} = (u_{10}, u_{20}, \dots, u_{N-10}) = (f_1, f_2, \dots, f_{N-1}).$$

Pero ahora es sencillo ver que la ecuación

$$u_{nm-1} = (1 + 2\lambda)u_{nm} - \lambda(u_{n+1m} + u_{n-1m}),$$

se escribe matricialmente como $\tilde{A} \mathbf{u}^{(m)} = \mathbf{u}^{(m-1)}$, o mejor

$$\tilde{A} \mathbf{u}^{(m+1)} = \mathbf{u}^{(m)},$$

donde \tilde{A} es la matriz

$$\tilde{A} = \begin{bmatrix} 1 + 2\lambda & -\lambda & 0 & \dots \\ -\lambda & 1 + 2\lambda & -\lambda & \dots \\ 0 & -\lambda & 1 + 2\lambda & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

y este proceso de asignaciones, escrito en forma matricial, avanza a partir del vector

$$\mathbf{u}^{(0)} = (u_{10}, u_{20}, \dots, u_{N-10}) = (f_1, f_2, \dots, f_{N-1}),$$

conocido a partir de la condición inicial.

Si queremos saber cómo se lleva de manera efectiva el proceso de iteración, hay que comenzar por decir que lo que se tiene es

$$\mathbf{u}^{(m+1)} = \tilde{A}^{-1} \mathbf{u}^{(m)}.$$

Y lo que realmente hay que hacer no es el cálculo de la inversa \tilde{A}^{-1} , que dispararía el coste y los errores, sino calcular $\mathbf{u}^{(m+1)}$ a partir de $\mathbf{u}^{(m)}$ resolviendo el sistema lineal

$$\tilde{A} \mathbf{z} = \mathbf{u}^{(m)},$$

y asignando la solución \mathbf{z} a $\mathbf{u}^{(m+1)}$.

Empleamos el procedimiento matricial para conocer algo de la estabilidad de este método que, matricialmente, se describe como

$$\mathbf{u}^{(m+1)} = \tilde{A}^{-1} \mathbf{u}^{(m)}.$$

La matriz \tilde{A} se relaciona con la matriz B de la que ya hemos hablado mediante

$$\tilde{A} = I + \lambda B.$$

Como los valores propios de B eran

$$4 \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N-1.$$

los de \tilde{A} serán entonces

$$1 + 4\lambda \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N-1,$$

y los de \tilde{A}^{-1}

$$\tilde{\mu}_n = \frac{1}{1 + 4\lambda \left(\sin \frac{n\pi}{2N} \right)^2}, \quad n = 1, \dots, N-1.$$

El hecho de que λ sea > 0 deja claro que los $\tilde{\mu}_n$ son reales, positivos y < 1 , por lo que, independientemente del valor que tenga λ , el radio espectral de \tilde{A}^{-1} será $\rho(\tilde{A}^{-1}) < 1$ y el método estable (para cualquier valor de λ). Se expresa este hecho diciendo que el método de las diferencias hacia atrás es **incondicionalmente estable**.

Naturalmente que para este método desaparece entonces toda restricción sobre el tamaño de k en lo que respecta a la estabilidad. Sin embargo, el orden $\mathcal{O}(h^2) + \mathcal{O}(k)$ del error de truncación hace que convenga mantener k en un tamaño no muy grande y del tamaño aproximado de h^2 .

Ejemplo 2

El ejemplo que encontramos realizado en 'Maple' con el nombre **Cap12_ejemplo2.mws** para el mismo problema parabólico, aborda los mismos casos del **Cap12_ejemplo1.mws**. Con la correspondiente matriz \tilde{A} en cada caso, las 1000 y 50 iteraciones son repeticiones de la resolución del sistema

$$\mathbf{u}^{(m+1)} = \tilde{A}^{-1} \mathbf{u}^{(m)} .$$

a partir del vector $\mathbf{u}^{(0)}$ con las condiciones iniciales.

Realizando el trabajo con 4 cifras decimales significativas, el primer caso (con $\lambda = 0,05$) produce en $u(0,5, 0,5)$ el error

$u_{5\ 1000} = 0,005415$ para $u(0,5, 0,5) = 0,007183$ con error = 0,001768 ,
y en el segundo caso (con $\lambda = 1$) el error

$$u_{5\ 50} - u(0,5, 0,5) = 0,002226 ,$$

todo perfectamente acorde con el error de truncación.

El coste de resolver un sistema (además grande, pues es de tamaño $(N - 1) \times (N - 1)$) de este último método es 'infinitamente' superior al de la asignación simple del método de las diferencias hacia adelante. Por ello, el uso de este último método y otros similares con estabilidad incondicional está limitado a casos en los que un valor de k demasiado pequeño es un inconveniente importante.

Búsqueda del orden de truncación

Veamos otro ejemplo en el que se obtiene el orden de truncación de un método cuando se nos proporciona simplemente el método, sin explicitar las aproximaciones que se han hecho para las derivadas. Consideremos el problema parabólico en $D = [0, l] \times [0, +\infty)$

$$\begin{aligned}\frac{\partial u}{\partial t} &= \alpha^2 \frac{\partial^2 u}{\partial x^2} \\ u(x, 0) &= f(x) \\ u(0, t) &= g_1(t), \quad u(l, t) = g_2(t)\end{aligned}$$

Nos presentan el método

$$\frac{u_{nm+1} - u_{nm}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0$$

(que reconoceremos como el de las diferencias hacia adelante),

para el retículo usual

$$\begin{aligned}x_0, x_1, \dots, x_N, \quad x_n &= n h, \quad h = l/N, \\ t_0, t_1, t_2, \dots, \quad t_m &= m k, \quad k > 0,\end{aligned}$$

y las aproximaciones $u_{nm} \sim u(x_n, t_m)$ a la solución.

$P(h, k)(n, m) = 0$ es el método cuando ponemos

$$P(h, k)(n, m) = \frac{u_{nm+1} - u_{nm}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}.$$

La correspondiente expresión para una solución $u(x, t)$ de la ecuación es

$$\begin{aligned}P(h, k)(x_n, t_m) &= \\ &= \frac{u(x_n, t_{m+1}) - u(x_n, t_m)}{k} - \alpha^2 \frac{u(x_{n+1}, t_m) - 2u(x_n, t_m) + u(x_{n-1}, t_m)}{h^2} =\end{aligned}$$

$$= \frac{u(x_n, t_m + k) - u(x_n, t_m)}{k} - \alpha^2 \frac{u(x_n + h, t_m) - 2u(x_n, t_m) + u(x_n - h, t_m)}{h^2}.$$

Desarrollamos ahora términos como $u(x_n, t_m + k)$ y similares en potencias de h y de k alrededor de (x_n, t_m) , o sea, de $h = 0, k = 0$. Usando las notaciones resumidas ya avanzadas en el anterior capítulo, vamos obteniendo

$$\begin{aligned} u(x_n, t_m + k) &= u + k u_t + \frac{k^2}{2} u_{tt}(x_n, \eta_m) = u + k u_t + \mathcal{O}(k^2) \\ u(x_n, t_m) &= u, \end{aligned}$$

luego

$$\frac{u(x_n, t_m + k) - u(x_n, t_m)}{k} = u_t + \mathcal{O}(k).$$

Repetiendo paso por paso el proceso similar del capítulo precedente, obtenemos

$$\frac{u(x_n + h, t_m) - 2u(x_n, t_m) + u(x_n - h, t_m)}{h^2} = u_{xx} + \mathcal{O}(h^2).$$

Reuniendo todo lo anterior,

$$P(h, k)(x_n, t_m) = u_t(x_n, t_m) + \mathcal{O}(k) - \alpha^2 u_{xx}(x_n, t_m) + \mathcal{O}(h^2),$$

y teniendo en cuenta que $u(x, t)$ verifica

$$u_t - \alpha^2 u_{xx} = 0,$$

finalizamos con

$$\mathcal{O}(k) + \mathcal{O}(h^2)$$

como orden del error de truncación de este método.

El procedimiento de FOURIER

Para el problema parabólico en $D = [0, l] \times [0, +\infty)$

$$\begin{aligned}\frac{\partial u}{\partial t} &= \alpha^2 \frac{\partial^2 u}{\partial x^2} \\ u(x, 0) &= f(x) \\ u(0, t) &= 0, \quad u(l, t) = 0\end{aligned}$$

y el método de las diferencias hacia adelante

$$u_{nm+1} = (1 - 2\lambda)u_{nm} + \lambda(u_{n+1m} + u_{n-1m})$$

con $\lambda = \alpha^2 k/h^2$, hemos visto que $\lambda < 1/2$ es una condición suficiente para que tal método sea estable. La herramienta usada fue el llamado procedimiento matricial.

Vamos a presentar otra vía, llamada **procedimiento de FOURIER** (también se dice **procedimiento de VON NEUMANN**) que proporciona condiciones necesarias para la estabilidad. Para entenderlo mejor, condiciones sobre el valor de λ que se verifican necesariamente cuando el método es estable.

La finalidad de abordar también este procedimiento es doble. Por una parte, permite rematar lo que era una condición suficiente de estabilidad y convertirla en una condición necesaria y suficiente. Pero también disponer de un procedimiento aplicable cuando el matricial no lo sea.

Las componentes elementales de la descomposición de FOURIER

$$e^{i b x} e^{d t} \text{ o mejor } e^{i n b h} e^{m d k} \text{ para } x = x_n = n h \text{ y } t = t_m = m k$$

(h y k las habituales del retículo, $n = 0, \dots, N$ y $m = 0, 1, 2, \dots$) son las que van a guiar el procedimiento. Consideraremos las sucesiones elementales de FOURIER

$$v_{nm} = e^{i n b h} e^{m d k}$$

que verifiquen la ecuación del método (en este caso del método de las diferencias hacia adelante).

Imponer que v_{nm} sea solución del método, lo que se lleva a cabo sustituyendo v_{nm} en la ecuación de ese método, llevará a una relación (complicada como veremos, pero en general aprovechable) entre los parámetros b y d .

Cuando el método es estable y v_{nm} solución del método, necesariamente la sucesión v_{nm} va a tender a 0 (o, menos estrictamente, estar acotada) cuando $m \rightarrow \infty$. Como $e^{i n b h}$ es un complejo de módulo 1, esto implicará que

$$|e^{d k}| < 1$$

(≤ 1 si se opta por la opción menos estricta); y eso para todos los valores de d que figuren en las soluciones del método. Como veremos en este y otros casos, esto acaba imponiendo alguna condición sobre los valores de λ que será la que nos proporciona la condición necesaria de estabilidad que buscamos.

Veamos qué es lo que ocurre con el método de las diferencias hacia adelante.

Que v_{nm} sea solución del método significa que

$$e^{i nbh} e^{(m+1)dk} = (1-2\lambda)e^{i nbh} e^{mdk} + \lambda(e^{i(n+1)bh} e^{mdk} + e^{i(n-1)bh} e^{mdk}).$$

La división por los factores no nulos $e^{i nbh}$ y e^{mdk} , reduce la condición a

$$e^{dk} = 1 - 2\lambda + \lambda(e^{i bh} + e^{-i bh}).$$

Lo primero que hacemos es hacer aflorar que se trata de una ecuación real (no lo parece) observando que

$$e^{i bh} + e^{-i bh} = 2 \cos bh,$$

con lo que queda

$$e^{dk} = 1 - 2\lambda + 2\lambda \cos bh.$$

Además, la identidad ya conocida

$$\cos 2a = 1 - 2 \operatorname{sen}^2 a,$$



que aplicamos para $a = bh/2$ transforma nuestra expresión en

$$e^{dk} = 1 - 4\lambda \operatorname{sen}^2 \frac{bh}{2}.$$

Esta es la expresión final para nuestra relación entre b y d que, como se ve, hace intervenir, en principio, el valor de λ . Dos puntos interesan bastante: e^{dk} aparece despejado y, además, el seno, que aparece al cuadrado, toma exclusivamente valores positivos.

Ya hemos dicho que la estabilidad del método implica

$$|e^{dk}| < 1;$$

esto equivale a

$$\left| 1 - 4\lambda \operatorname{sen}^2 \frac{bh}{2} \right| < 1$$



y entonces a

$$-1 < 1 - 4\lambda \operatorname{sen}^2 \frac{bh}{2} < 1$$

y a

$$-2 < -4\lambda \operatorname{sen}^2 \frac{bh}{2}$$

(obsérvese que la otra desigualdad es siempre cierta). Por lo tanto, $|e^{dk}| < 1$ equivale a

$$\lambda \operatorname{sen}^2 \frac{bh}{2} < \frac{1}{2},$$

y que esto sea cierto para b arbitrario acaba imponiendo que

$$\lambda < \frac{1}{2}.$$

De manera que la condición suficiente de estabilidad para el método de las diferencias hacia adelante resulta ser también necesaria.

Por lo tanto, ahora tenemos la seguridad de que, en nuestro ejemplo 1, no sólo los valores $\lambda < 1/2$ conducen a la estabilidad (es lo que ya sabíamos), sino que los valores $\lambda > 1/2$ llevan a un comportamiento inestable.

El valor $\lambda = 1/2$, no incluido en el razonamiento anterior se incorpora a uno u otro campo según seamos más o menos estrictos al definir la estabilidad (según exijamos que el efecto de un error tienda a 0 al iterar el método o sólo que quede acotado sin tender a infinito, lo que es menos estricto). Una u otra alternativa no tienen excesiva importancia, pues lo importante es la delimitación del intervalo en que λ (y consecuentemente h y k) debe moverse para que exista estabilidad en los cálculos.

El método de CRANK-NICOLSON

Hemos construido un método ('diferencias hacia adelante') de carácter totalmente explícito, pero con el problema de que sólo para ciertos valores de

$$\lambda = \frac{\alpha^2 k}{h^2};$$

exactamente para

$$\lambda < \frac{1}{2}.$$

presentaba un comportamiento estable en los cálculos.

La primera solución ha sido reemplazarlo por otro método ('diferencias hacia atrás') que no tiene ningún problema de estabilidad, aunque a cambio era un método implícito, precisando de la resolución de un sistema lineal para cada avance del tiempo (sistema muy grande para cantidades 'realistas' de nodos en la variable espacial x).

Pero ambos llevaban asociados un orden de truncación del tipo

$$\mathcal{O}(h^2) + \mathcal{O}(k),$$

o sea orden 1 en el tiempo, lo que suele ser poco en los casos prácticos en los que una precisión adecuada obliga a distanciar muy poco los nodos de t , o sea a tomar k muy, muy pequeño. Pues bien, existe un método que, con la dificultad de ser implícito, no tiene problemas de estabilidad y además posee un orden de truncación

$$\mathcal{O}(h^2) + \mathcal{O}(k^2),$$

permitiendo entonces disminuciones menos importantes del paso k manteniendo una buena precisión de los resultados. El método de CRANK-NICOLSON que vamos a tratar a continuación.

Para el problema parabólico en $D = [0, l] \times [0, +\infty)$

$$\begin{aligned}\frac{\partial u}{\partial t} &= \alpha^2 \frac{\partial^2 u}{\partial x^2} \\ u(x, 0) &= f(x) \\ u(0, t) &= 0, \quad u(l, t) = 0\end{aligned}$$

recordamos la ecuación del método de las diferencias hacia adelante

$$\frac{u_{nm+1} - u_{nm}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0,$$

y la del método de las diferencias hacia atrás

$$\frac{u_{nm} - u_{nm-1}}{k} - \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0.$$

Escribimos la segunda para $m + 1$ en lugar de m y realizamos la media aritmética de ambas expresiones, la primera y la segunda modificada como hemos dicho; se obtiene



$$\begin{aligned}\frac{u_{nm+1} - u_{nm}}{k} - \frac{\alpha^2}{2} \left(\frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} \right. \\ \left. + \frac{u_{n+1m+1} - 2u_{nm+1} + u_{n-1m+1}}{h^2} \right) = 0.\end{aligned}$$

Es la ecuación de nuestro nuevo método. Antes de nada, vamos a usar esta expresión para calcular su orden del error de truncación. Llamamos $P(h, k)(n, m)$ a la expresión igualada a 0. Y la reescribimos substituyendo los u_{nm} por los valores $u(x_n, t_m)$ de una solución de la ecuación que estudiamos

$$u_{tt} - \alpha^2 u_{xx} = 0;$$

llamando, como otras veces $P(h, k)(x_n, t_m)$ al resultado,

$$\begin{aligned}
 P(h, k)(x_n, t_m) &= \\
 &= \frac{u(x_n, t_m + k) - u(x_n, t_m)}{k} - \\
 &\quad - \frac{\frac{\alpha^2}{2} \frac{u(x_n + h, t_m) - 2u(x_n, t_m) + u(x_n - h, t_m)}{h^2}}{2} - \\
 &\quad - \frac{\frac{\alpha^2}{2} \frac{u(x_n + h, t_m + k) - 2u(x_n, t_m + k) + u(x_n - h, t_m + k)}{h^2}}{2},
 \end{aligned}$$

con las habituales notaciones, x_n, t_m, h y k , para la red de nodos.

Desarrollamos todo en potencias de h y de k .

El primer término lo descomponemos en

$$\frac{u(x_n, t_m + k) - u(x_n, t_m)}{2k} + \frac{u(x_n, t_m + k) - u(x_n, t_m)}{2k};$$

la primera parte la desarrollamos alrededor del nodo (x_n, t_m) , punto que en principio excluimos de la notación, poniendo u, u_t , etc. para representar $u(x_n, t_m), u_t(x_n, t_m)$, etc.

$$\begin{aligned}
 u(x_n, t_m + k) &= u + k u_t + \frac{k^2}{2} u_{tt} + \mathcal{O}(k^3) \\
 u(x_n, t_m) &= u,
 \end{aligned}$$

y dicha primera parte es entonces

$$\mathbf{(1a)} = \frac{1}{2} u_t + \frac{k}{4} u_{tt} + \mathcal{O}(k^2).$$

la segunda parte la desarrollamos alrededor del nodo $(x_n, t_m + k)$, punto que también excluimos de la notación, poniendo u^* , u^*_t , etc. para representar $u(x_n, t_m + k)$, $u_t(x_n, t_m + k)$, etc.

$$u(x_n, t_m + k) = u^*$$

$$u(x_n, t_m) = u^* - k u^*_t + \frac{k^2}{2} u^*_{tt} + \mathcal{O}(k^3),$$

y dicha segunda parte es entonces

$$(1b) = \frac{1}{2} u^*_t - \frac{k}{4} u^*_{tt} + \mathcal{O}(k^2).$$

El segundo término es, como ya hemos visto en otra ocasión,

$$(2) = -\frac{\alpha^2}{2} u_{xx} + \mathcal{O}(h^2).$$

El tercero se desarrolla también alrededor del nodo $(x_n, t_m + k)$, y denotamos por u^* , u^*_x , etc. los valores $u(x_n, t_m + k)$, $u_x(x_n, t_m + k)$, etc.

$$(3) = -\frac{\alpha^2}{2} u^*_{xx} + \mathcal{O}(h^2).$$

Reuniendo todos los desarrollos,

$$P(h, k)(x_n, t_m) =$$

$$= \frac{1}{2} u_t + \frac{k}{4} u_{tt} + \mathcal{O}(k^2) + \frac{1}{2} u^*_t - \frac{k}{4} u^*_{tt} + \mathcal{O}(k^2)$$

$$- \frac{\alpha^2}{2} u_{xx} + \mathcal{O}(h^2) - \frac{\alpha^2}{2} u^*_{xx} + \mathcal{O}(h^2),$$

o bien

$$P(h, k)(x_n, t_m) = \frac{k}{4} (u_{tt} - u^*_{tt}) + \mathcal{O}(k^2) + \mathcal{O}(h^2),$$

ya que

$$\frac{1}{2} u_t - \frac{\alpha^2}{2} u_{xx} = 0 \quad \text{y} \quad \frac{1}{2} u^*_t - \frac{\alpha^2}{2} u^*_{xx} = 0.$$

Consideremos ahora el término $u_{tt} - u^*_{tt}$, o sea

$u_{tt}(x_n, t_m) - u_{tt}(x_n, t_m + k)$; desarrollando alrededor de (x_n, t_m) , tenemos

$$u_{tt}(x_n, t_m) - u_{tt}(x_n, t_m + k) = u_{tt} - u_{tt} - k u_{ttt} + \mathcal{O}(k^2).$$

Esto hace que

$$P(h, k)(x_n, t_m) = \frac{k^2}{4} u_{ttt} + \mathcal{O}(k^2) + \mathcal{O}(h^2) = \mathcal{O}(k^2) + \mathcal{O}(h^2).$$

Este es el orden que habíamos anunciado para el método de CRANK-NICOLSON.

Mejorando la ecuación del método, poniendo como es usual para las ecuaciones parabólicas

$$\lambda = \frac{\alpha^2 k}{h^2},$$

unos sencillos cálculos la dejan en

$$\begin{aligned} (1 + \lambda) u_{n m+1} - \frac{\lambda}{2} u_{n+1 m+1} - \frac{\lambda}{2} u_{n-1 m+1} &= \\ &= (1 - \lambda) u_{n m} + \frac{\lambda}{2} u_{n+1 m} + \frac{\lambda}{2} u_{n-1 m}, \end{aligned}$$

Para el usual vector

(0)

$$\mathbf{u}^{(m)} = (u_{1m}, u_{2m}, \dots, u_{N-1m}),$$

la ecuación anterior se escribe matricialmente como

$$A_1 \mathbf{u}^{(m+1)} = A_2 \mathbf{u}^{(m)},$$

donde A_1 y A_2 son las matrices

$$A_1 = \begin{bmatrix} 1 + \lambda & -\frac{\lambda}{2} & 0 & \cdots \\ -\frac{\lambda}{2} & 1 + \lambda & -\frac{\lambda}{2} & \cdots \\ 0 & -\frac{\lambda}{2} & 1 + \lambda & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad \text{y} \quad A_2 = \begin{bmatrix} 1 - \lambda & \frac{\lambda}{2} & 0 & \cdots \\ \frac{\lambda}{2} & 1 - \lambda & \frac{\lambda}{2} & \cdots \\ 0 & \frac{\lambda}{2} & 1 - \lambda & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

o sea, poniendo como antes

$$B = \begin{bmatrix} 2 & -1 & 0 & \cdots \\ -1 & 2 & -1 & \cdots \\ 0 & -1 & 2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

$$A_1 = I + \frac{\lambda}{2} B \quad \text{y} \quad A_2 = I - \frac{\lambda}{2} B$$

(I es la matriz unidad $(N - 1) \times (N - 1)$).

Como

$$\mathbf{u}^{(m+1)} = A_1^{-1} A_2 \mathbf{u}^{(m)},$$

vamos a poder estudiar la estabilidad de este método mediante el empleo del radio espectral $\rho(C)$ de la matriz

$$C = A_1^{-1} A_2 = \left(I + \frac{\lambda}{2} B\right)^{-1} \left(I - \frac{\lambda}{2} B\right).$$

Los valores propios de B eran

$$\theta_n = 4 \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N - 1.$$

le que aprovecharemos observando que son distintos y situados en el intervalo $(0, 4)$.

De las relaciones entre C , B , A_1 y A_2 , obtenemos que los valores propios de $C = A_1^{-1} A_2$ son

$$\mu_n = \frac{1 - \frac{1}{2} \lambda \theta_n}{1 + \frac{1}{2} \lambda \theta_n} = \frac{2 - \lambda \theta_n}{2 + \lambda \theta_n}, \quad n = 1, \dots, N - 1.$$

Para situar estos valores propios, estudiamos la función

$$x \longrightarrow \frac{2 - \lambda x}{2 + \lambda x}$$

para $x \in (0, 4)$; la función tiene derivada

$$x \longrightarrow \frac{-4\lambda}{(2 + \lambda x)^2} < 0$$

lo que hace que esa función sea siempre estrictamente decreciente. En 0 vale 1, y en 4 vale $\frac{2-4\lambda}{2+4\lambda}$ que será < 1 ya que la función es decreciente pero también > -1 (lo que se sigue de que $\lambda > 0$ y de que $2 > -2$ y $2 - 4\lambda > -2 - 4\lambda = -(2 + 4\lambda)$). La función toma valores en $(-1, 1)$ para $x \in (0, 4)$ y esto mismo es lo que les ocurre a los valores propios μ_n . Deducimos de ahí que

$$\rho(C) < 1$$

y que el método de CRANK-NICOLSON es estable, cualquiera que sea el valor de λ , o sea, incondicionalmente estable.



El método de CRANK-NICOLSON es enormemente ventajoso en orden y en relación a la estabilidad. Pero su puesta en práctica tiene un coste similar al de las 'diferencias hacia atrás'. En efecto, conocido el vector $\mathbf{u}^{(m)} = (u_{1m}, u_{2m}, \dots, u_{N-1m})$, el siguiente $\mathbf{u}^{(m+1)}$, dado por $A_1 \mathbf{u}^{(m+1)} = A_2 \mathbf{u}^{(m)}$, se obtiene multiplicando A_2 por $\mathbf{u}^{(m)}$, y luego resolviendo el sistema lineal

$$A_1 \mathbf{z} = A_2 \mathbf{u}^{(m)},$$

cuya solución \mathbf{z} es el vector

$$\mathbf{u}^{(m+1)}$$

que se busca y que contiene las aproximaciones al nivel de $t = t_{m+1}$.



Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-01

Se considera la matriz

$$\mathbf{B} = \begin{vmatrix} 2 & -1 & 0 & \dots \\ -1 & 2 & -1 & \dots \\ 0 & -1 & 2 & \dots \\ \cdot & \cdot & \cdot & \\ \cdot & \cdot & \cdot & \end{vmatrix}$$

de tamaño $(N - 1) \times (N - 1)$, y la matriz

$$\mathbf{A} = \mathbf{I} - \lambda \mathbf{B} = \begin{vmatrix} 1 - 2\lambda & \lambda & 0 & \dots \\ \lambda & 1 - 2\lambda & \lambda & \dots \\ 0 & \lambda & 1 - 2\lambda & \dots \\ \cdot & \cdot & \cdot & \\ \cdot & \cdot & \cdot & \end{vmatrix}$$

del mismo tamaño. Esta es la matriz del método de las 'diferencias hacia adelante' para problemas parabólicos.

a) Pruébese que los valores propios de \mathbf{B} son

$$2 - 2 \cos \frac{n \pi}{N} = 4 \left(\sin \frac{n \pi}{2N} \right)^2$$

$n = 1, \dots, N - 1$, con vectores propios

$$x^{[n]} = \left(\sin \frac{n \pi}{N}, \sin \frac{2 n \pi}{N}, \dots, \sin \frac{(N - 1) n \pi}{N} \right)$$

b) Calcúlense así los valores propios y también los vectores propios de \mathbf{A} .

```
[ > restart:
[ > interface(labeling=false):
[ > P
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-02

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

vamos a considerar el método de las 'diferencias hacia adelante'

dado por

$$u_{n,m+1} = (1 - 2\lambda) u_{n,m} + \lambda (u_{n+1,m} + u_{n-1,m})$$

con $\lambda = \frac{\alpha^2 k}{h^2}$, o poniendo

$$u^{[k]} = (u_{1k}, u_{2k}, \dots, u_{N-1k})$$

$$\begin{vmatrix} 1 - 2\lambda & \lambda & 0 & \dots \\ \lambda & 1 - 2\lambda & \lambda & \dots \\ 0 & \lambda & 1 - 2\lambda & \dots \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{vmatrix}$$

$$A = I - \lambda B = \begin{vmatrix} 1 - 2\lambda & \lambda & 0 & \dots \\ \lambda & 1 - 2\lambda & \lambda & \dots \\ 0 & \lambda & 1 - 2\lambda & \dots \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{vmatrix}$$

en forma matricial, por

$$u^{[m+1]} = A u^{[m]}$$

Se recuerda que una condición suficiente para que $\rho(A) \leq 1$ (lo que es

una condición suficiente de estabilidad) es que $\lambda \leq \frac{1}{2}$

Utilícese de nuevo (como en la parte teórica) el método matricial

para probar que, para N fijo, la condición suficiente de estabilidad es

$$\lambda \leq \frac{1}{1 + \cos\left(\frac{\pi}{N}\right)}$$

Qué significa lo anterior ?

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-03 (del EXAMEN EXTRAORDINARIO de 04JUL06)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos los métodos

'hacia adelante' y 'hacia atrás' dados por

$$\frac{u_{nm+1} - u_{nm}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

y

$$\frac{u_{nm} - u_{nm-1}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

Se escribe el segundo para $m+1$ en lugar de m , y se toma la media aritmética de ambos, como para el método de CRANK-NICOLSON, pero con coeficientes -1 y 2 , obteniéndose así un método mezcla de ambos.

Construir matricialmente dicho método y estudiar la estabilidad del método, empleando el procedimiento matricial

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-04 (del EXAMEN EXTRAORDINARIO de 09JUL03)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos los métodos

'hacia adelante' y 'hacia atrás' dados por

$$\frac{u_{nm+1} - u_{nm}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

y

$$\frac{u_{nm} - u_{nm-1}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

Se escribe el segundo para $m+1$ en lugar de m , y se toma la media aritmética de ambos, como para el método de CRANK-NICOLSON, pero con coeficientes $1/4$ y $3/4$, obteniéndose así un método mezcla de ambos.

Construir matricialmente dicho método y comprobar que es estable con independencia de las relaciones entre h y k , incrementos en x y en t .

```
> restart;
> interface(labeling=false);
Estabilidad por el método de Fourier
> u:=(n,m)->exp(I*n*b*h)*exp(m*d*k);
                                     u := (n, m) -> e(nbhI) e(mdk)
> igualdad:=u(n,m+1)-u(n,m)-(lambda/4)*(u(n+1,m)-2*u(n,m)+u(n-1,m)+3*u(n+1,m+1)-6*u(n,m+1)+3*u(n-1,m+1));
igualdad := e(nbhI) e((m+1)dk) - e(nbhI) e(mdk) - 1/4 lambda (e((n+1)bhI) e(mdk) - 2 e(nbhI) e(mdk) + e((n-1)bhI) e(mdk)
+ 3 e((n+1)bhI) e((m+1)dk) - 6 e(nbhI) e((m+1)dk) + 3 e((n-1)bhI) e((m+1)dk))
> igualdad:=simplify(igualdad/(exp(I*n*b*h)*exp(m*d*k)));
                                     igualdad := -3/2 cos(bh) lambda e(dk) - 1/2 cos(bh) lambda + e(dk) - 1 + 3/2 lambda e(dk) + 1/2 lambda
> igualdad:=subs(cos(b*h)=1-2*(sin(b*h/2))^2, igualdad);
                                     igualdad := -3/2 (1 - 2 sin(1/2 bh))^2 lambda e(dk) - 1/2 (1 - 2 sin(1/2 bh))^2 lambda + e(dk) - 1 + 3/2 lambda e(dk) + 1/2 lambda
> igualdad:=simplify(igualdad);
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-05 (del EXAMEN ORDINARIO de 24ENE11)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos los métodos

'hacia adelante' y 'hacia atrás' dados por

$$\frac{u_{n,m+1} - u_{n,m}}{k} = \alpha^2 \frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{h^2}$$

y

$$\frac{u_{n,m} - u_{n,m-1}}{k} = \alpha^2 \frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{h^2}$$

Se escribe el segundo para $m+1$ en lugar de m , y se toma la media aritmética de ambos, como para el método de CRANK-NICOLSON, pero con coeficientes -2 y 3 , obteniéndose así un método mezcla de ambos.

Construir matricialmente dicho método y estudiar la estabilidad del método, empleando el procedimiento matricial

Dígase qué condición necesaria de estabilidad se obtiene al aplicar

el procedimiento de FOURIER a dicho método

```
[ > restart:  
[ > interface(labeling=false):
```

$$igualdad := 3 \lambda e^{(dk)} - 3 \lambda e^{(dk)} \cos\left(\frac{1}{2} b h\right)^2 + \lambda - \lambda \cos\left(\frac{1}{2} b h\right)^2 + e^{(dk)} - 1$$

> `igualdad:=subs(exp(d*k)=z,igualdad);`

$$igualdad := 3 \lambda z - 3 \lambda z \cos\left(\frac{1}{2} b h\right)^2 + \lambda - \lambda \cos\left(\frac{1}{2} b h\right)^2 + z - 1$$

> `ealadk:=solve(igualdad,z);`

$$ealadk := -\frac{1 - \lambda + \lambda \cos\left(\frac{1}{2} b h\right)^2}{-3 \lambda + 3 \lambda \cos\left(\frac{1}{2} b h\right)^2 - 1}$$

> `ealadk1:=(1-lambda*sin(1/2*b*h)^2)/(1+3*lambda*sin(1/2*b*h)^2);`

$$ealadk1 := \frac{1 - \lambda \sin\left(\frac{1}{2} b h\right)^2}{1 + 3 \lambda \sin\left(\frac{1}{2} b h\right)^2}$$

> `simplify(ealadk1-ealadk);`

0

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-06 (del EXAMEN EXTRAORDINARIO de 09JUL08)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos los métodos

'hacia adelante' y 'hacia atrás' dados por

$$\frac{u_{nm+1} - u_{nm}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

y

$$\frac{u_{nm} - u_{nm-1}}{k} = \alpha^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2}$$

Se escribe el segundo para $m+1$ en lugar de m y se toma la media aritmética de ambos (con coeficientes $1/2$ y $1/2$), obteniéndose así el conocido método de CRANK-NICOLSON del que se sabe que es estable con independencia de las relaciones entre h y k , incrementos en las variables x y t .

Si se eligen ahora para la media los coeficientes 2 y -1 , o bien los coeficientes -1 y 2 , describase el método que resulta y lo que el método matricial afirma de la estabilidad en el caso (uno de los dos) que el alumno elija.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-07

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

vamos a considerar el método de las 'diferencias hacia adelante'

dado por

$$u_{n,m+1} = (1 - 2\lambda) u_{n,m} + \lambda (u_{n+1,m} + u_{n-1,m})$$

con $\lambda = \frac{\alpha^2 k}{h^2}$, o poniendo

$$u^{[k]} = (u_{1k}, u_{2k}, \dots, u_{N-1k})$$

$$\begin{vmatrix} 1 - 2\lambda & \lambda & 0 & \dots \\ \lambda & 1 - 2\lambda & \lambda & \dots \\ 0 & \lambda & 1 - 2\lambda & \dots \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{vmatrix}$$

$$A = I - \lambda B = \begin{vmatrix} 1 - 2\lambda & \lambda & 0 & \dots \\ \lambda & 1 - 2\lambda & \lambda & \dots \\ 0 & \lambda & 1 - 2\lambda & \dots \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{vmatrix}$$

$$\begin{vmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{vmatrix}$$

en forma matricial, por

$$u^{[m+1]} = A u^{[m]}$$

Se recuerda que una condición suficiente para que $\rho(A) \leq 1$ (lo que es

una condición suficiente de estabilidad) es que $\lambda \leq \frac{1}{2}$

Úse el método de FOURIER para buscar una condición necesaria de estabilidad.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-08

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, se considera el retículo habitual y el método en diferencias, llamado método de DOUGLAS,

$$(1 - 6\lambda) u_{n-1m+1} + (10 + 12\lambda) u_{nm+1} + (1 - 6\lambda) u_{n+1m+1} = \\ = (1 + 6\lambda) u_{n-1m} + (10 - 12\lambda) u_{nm} + (1 + 6\lambda) u_{n+1m}$$

con $\lambda = \frac{k}{h^2}$ y las notaciones habituales.

a) Descríbase la forma matricial del método en términos de la habitual matriz

$$B = \begin{bmatrix} 2 & -1 & 0 & \dots & | \\ -1 & 2 & -1 & \dots & | \\ 0 & -1 & 2 & \dots & | \\ \cdot & \cdot & \cdot & & | \\ \cdot & \cdot & \cdot & & | \end{bmatrix}$$

cuya autovalores son

$$\rho_n = 4 \left(\sin \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N-1, \text{ para los autovectores}$$

$$x^{[n]} = \left(\sin \frac{n\pi}{N}, \sin \frac{2n\pi}{N}, \dots, \sin \frac{(N-1)n\pi}{N} \right), \quad n = 1, \dots, N-1$$

b) Calcúlense los autovalores que aparecen en el estudio, por el procedimiento matricial, de la estabilidad de este método

c) Pruébese usando dicho procedimiento matricial que el método de DOUGLAS es incondicionalmente estable

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-09 (del EXAMEN EXTRAORDINARIO de 09JUL08)

Para un problema parabólico como

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con el retículo habitual

$$(x_n, t_m), x_n = n h, n=0, \dots, N-1, t_m = m k, m=0, 1, 2, \dots$$

y poniendo el usual

$$\lambda = \frac{\alpha^2 k}{h^2}$$

se considera el método

$$u_{n m+1} - u_{n m-1} = 2 \lambda (u_{n+1 m} - 2 u_{n m} + u_{n-1 m})$$

Dígase lo que asegura el método de FOURIER para el citado método en diferencias. Calcúlese además el orden del método.

```
[ > restart;  
[ > interface(labeling=false);  
[ >
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-10

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos los métodos

'hacia adelante' y 'hacia atrás' dados por

$$\frac{u_{n,m+1} - u_{n,m}}{k} = \alpha^2 \frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{h^2}$$

y

$$\frac{u_{n,m} - u_{n,m-1}}{k} = \alpha^2 \frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{h^2}$$

Se escribe el segundo para $m+1$ en lugar de m , y se toma la media aritmética de ambos. Esto nos da el método de CRANK-NICOLSON.

Se ha comprobado mediante el método matricial

que el método es estable con independencia de las relaciones

entre h y k , incrementos en x y en t . Ahora se trata de

probar lo mismo con ayuda del método de FOURIER.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-11 (del EXAMEN EXTRAORDINARIO de 14JUL04)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, se considera el método

de RICHARDSON, que consiste en utilizar para $\frac{\delta u}{\delta t}$ la fórmula

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2)$$

y para $\frac{\delta^2 u}{\delta x^2}$ la fórmula

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$$

Constrúyase la ecuación del método, calcúlese su orden de truncación,

describese la molécula computacional que interviene en la ecuación

y utilícese el método de FOURIER para buscar, si existen, condiciones

sobre λ que sean necesarias para la estabilidad del método.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-12 (del EXAMEN ORDINARIO DE 09FEB10)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, y denotando por λ la cantidad

$$\lambda = \frac{\alpha^2 k}{h^2}, \text{ nos presentan el método en diferencias}$$

$$u_{nm+2} - 4u_{nm+1} = (4\lambda - 3)u_{nm} - 2\lambda(u_{n+1m} + u_{n-1m})$$

donde las aproximaciones $u_{nm} = u(x_n, t_m)$ tienen el sentido habitual.

Calcúlese el orden de truncación del método. Estúdiense la estabilidad del método por alguno de los procedimientos habituales.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-13 (del EXAMEN ORDINARIO DE 09FEB09)

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0, \quad x \in [0, l], \quad t \geq 0,$$

$$u(x, 0) = f(x),$$

$u(x, t) = 0$, para todo x fuera de $(0, l)$ y todo $t \geq 0$ (esto es, la solución es idénticamente nula en el exterior del dominio $x \in [0, l], t \geq 0$)

se considera el retículo habitual

$$x_n = n h, \quad h = \frac{l}{N},$$

$$t_m = m k,$$

Obtégase el método en diferencias que proporcionan las aproximaciones a las derivadas

$$f'(x) = \frac{f(x+h) - f(x)}{h} + O(h)$$

para $\frac{\delta u}{\delta t}$ y

$$f''(x) = \frac{f(x+2h) - 2f(x) + f(x-2h)}{4h^2} + O(h^2)$$

para $\frac{\delta^2 u}{\delta x^2}$. Dedúzcase su orden, la ecuación del método en términos de $\lambda = \frac{\alpha^2 k}{h^2}$, la forma matricial

del método y utilícese el método de FOURIER para analizar su estabilidad.

Nota: Obsérvese que, a la hora de aplicar el método para obtener las aproximaciones, será necesario

hacer uso de los valores auxiliares $u_{-1,m}$ y $u_{N+1,m}$, $m=0,1,2,\dots$ que se tomarán nulos a partir de

las condiciones que verifica la solución de nuestro problema.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-14

Una ecuación parabólica tiene asignado el método en diferencias

$$u_{n+1,m} - u_{n,m} + \lambda (u_{n+1,m+1} - u_{n,m+1} + u_{n,m} - u_{n-1,m}) = 0$$

para $\lambda = \frac{\alpha^2 k}{h^2}$ y las notaciones habituales.

Escríbase el método en la forma matricial y compruébese la dificultad de buscar así condiciones suficientes de estabilidad. Véase que es fácil describir la estabilidad usando el método de FOURIER.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-15

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos el método en diferencias finitas dado por

$$\frac{u_{nm} - u_{n,m-1}}{k} - \alpha^2 \frac{u_{n+1,m} - 2u_{nm} + u_{n-1,m}}{h^2} = 0$$

- Dígase qué orden posee el error de truncación de dicho método.
- Describese la molécula computacional del método.
- Úsese el método de FOURIER para estudiar la estabilidad.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-16

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \frac{\delta^2 u}{\delta x^2} = 0, \quad x \in [0,1], \quad t \geq 0,$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \quad \text{y} \quad u(1, t) = 0,$$

con condiciones de contorno homogéneas, se considera el método en diferencias dado por

$$2h(u_{n,m+1} - u_{n,m}) = u_{n+1,m} - 2u_{n,m} + u_{n-1,m} + u_{n+1,m+1} - 2u_{n,m+1} + u_{n-1,m+1}$$

para una red

$$x_n = nh, \quad h = \frac{l}{N},$$

$$t_m = mh,$$

con igual incremento en ambas variables x y t .

Estúdiense matricialmente la estabilidad de este método.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-17 (del EXAMEN ORDINARIO de 01FEB06)

Para resolver la ecuación parabólica

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

nos proporcionan el método explícito

$$u_{n m+1} = \left(1 - \frac{5\lambda}{2}\right) u_{n m} + \frac{\lambda}{12} (-u_{n+2 m} + 16 u_{n+1 m} + 16 u_{n-1 m} - u_{n-2 m})$$

donde $\lambda = \alpha^2 k / h^2$ y n, m, h y k corresponden al retículo habitual.

Dígase cuál es el error de truncación que presenta este método.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-18

Para el problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, consideramos el método en diferencias finitas dado por

$$u_{n,m+2} = 4 u_{n,m+1} + (4\lambda - 3) u_{n,m} - 2\lambda (u_{n+1,m} + u_{n-1,m}),$$

donde $\lambda = \alpha^2 k / h^2$ y n, m, h y k corresponden al retículo habitual.

a) Escribese la ecuación en la forma matricial, relacionando el resultado con la matriz tridiagonal, también habitual,

$$\mathbf{B} = \begin{bmatrix} 2 & -1 & 0 & \dots & \\ -1 & 2 & -1 & \dots & \\ 0 & -1 & 2 & \dots & \\ \cdot & \cdot & \cdot & \dots & \\ \cdot & \cdot & \cdot & \dots & \end{bmatrix}$$

de tamaño $(N - 1) \times (N - 1)$.

b) Dígase qué orden posee el error de truncación de dicho método.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-19 (del EXAMEN ORDINARIO de 01FEB06)

Para el problema dado por la ecuación parabólica

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f(x).$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con condiciones de contorno homogéneas, se tiene el siguiente método

$$\frac{u_{n,m+1} - u_{n,m-1}}{2k} - \alpha^2 \frac{u_{n+1,m} - (u_{n,m+1} + u_{n,m-1}) + u_{n-1,m}}{h^2} = 0$$

conocido como 'esquema de DU FORT - FRANKEL'.

Establézcase la ecuación sobre $\lambda = \alpha^2 k / h^2$, b y d que plantea

el método de FOURIER cuando se aplica este esquema a $e^{(inbh)} e^{(mdk)}$

Comprobar que, en cualquiera de los casos, siempre se tiene que

$$|e^{(dk)}| \leq 1$$

Qué significa esto de cara a la estabilidad del esquema.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-20 (del EXAMEN ORDINARIO de 01FEB07)

Para un problema parabólico

$$\frac{\delta u}{\delta t} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

con condiciones de contorno homogéneas, el método

$$\frac{u_{n\ m+1} - u_{n\ m}}{k} = \frac{\alpha^2}{h^2} (-u_{n+1\ m} + 2u_{n\ m} - u_{n-1\ m} + 2u_{n+1\ m+1} - 4u_{n\ m+1} + 2u_{n-1\ m+1})$$

estudiado por el procedimiento matricial habitual, es considerado como incondicionalmente estable.

Estúdiese lo que el procedimiento de FOURIER es capaz de decir sobre este método.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(12) Las Ecuaciones parabólicas

Ejercicio 12-21 (del EXAMEN ORDINARIO de 01FEB07)

Para la ecuación parabólica

$$\frac{\delta u}{\delta t} - \frac{\delta^2 u}{\delta x^2} = 0$$

nos proporcionan el método explícito

$$u_{nm+2} - 4u_{nm+1} = (4\lambda - 3)u_{nm} - 2\lambda(u_{n+1m} + u_{n-1m})$$

donde $\lambda = k / h^2$ y n, m, h y k corresponden al retículo habitual.

Dígase cuál es el error de truncación que presenta este método.

```
[ > restart:  
[ > interface(labeling=false):
```

13. Las Ecuaciones hiperbólicas

- 1 El problema hiperbólico y el método explícito en diferencias
- 2 Condición necesaria de estabilidad; el procedimiento de FOURIER

El problema hiperbólico

Abordaremos un tipo de problemas que suelen ser dependientes del tiempo, y que supondremos ya en la forma

$$\frac{\partial^2 u(x, t)}{\partial t^2} - \alpha^2 \frac{\partial^2 u(x, t)}{\partial x^2} = 0, \quad x \in [0, l], t \in [0, +\infty),$$

ecuación en un dominio D que es el rectángulo 'semiinfinito'

$$D = [0, l] \times [0, +\infty).$$

El problema se completa con una doble condición inicial; la ya usada

$$u(x, 0) = f_1(x), \quad x \in [0, l],$$

pero también una condición sobre la derivada respecto de t

$$\frac{\partial u(x, 0)}{\partial t} = f_2(x), \quad x \in [0, l],$$



y las condiciones de contorno (sobre el contorno físico tanto en $x = 0$ como en $x = l$)

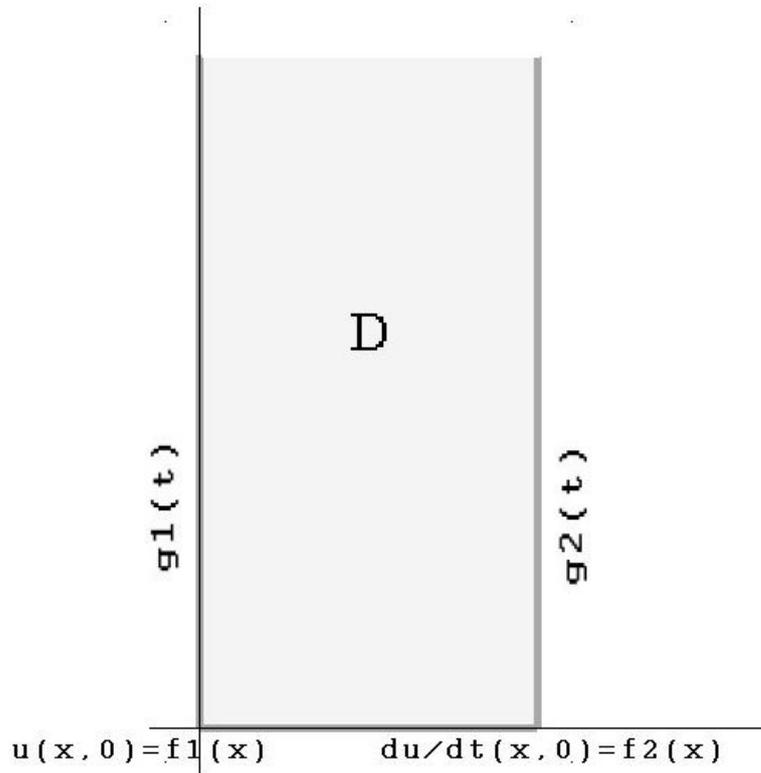
$$u(0, t) = g_1(t), \quad u(l, t) = g_2(t), \quad t \in [0, +\infty).$$

Se llamaba 'ecuación de ondas' a esta ecuación.

La segunda condición inicial es aquí imprescindible para que la solución sea única (lo que se pretende en cada caso). Como veremos juega también un papel decisivo para los valores de arranque en el método en diferencias que vamos a estudiar.



Recordemos con una figura el diseño del dominio:



El método explícito en diferencias

Consideramos en D el retículo definido como explicamos. Para la x ,

$$x_0, x_1, \dots, x_N, \quad x_n = n h, \quad h = \frac{l}{N},$$

Para t no es posible, sin embargo, hacer una división para calcular la longitud del subintervalo. Lo que se hace es tomar una longitud $k > 0$ arbitraria y fabricar con ella los t_m

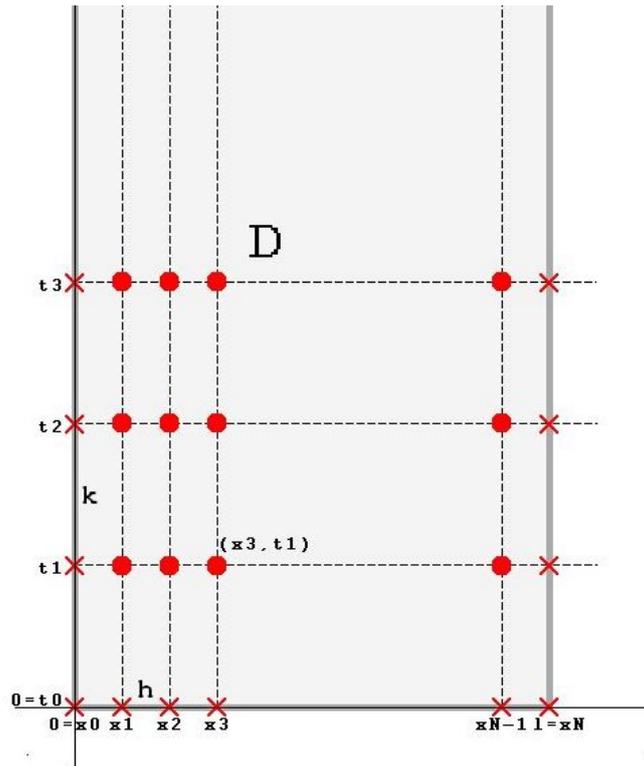
$$t_0, t_1, t_2, \dots, \quad t_m = m k, \quad k > 0.$$

Ahora hay un retículo bidimensional formado en principio por una cantidad infinita de nodos (x_n, t_m) en los que intentaremos aproximar la solución del problema por valores calculados

$$u_{nm} \sim u(x_n, t_m).$$

Comenzaremos por construir un método, llamado a veces método **explícito en diferencias**.

Observemos como antes el dominio D con sus nodos (x_n, t_m) , $n = 0, 1, \dots, N-1, N$, $m = 0, 1, 2, \dots$



Marcados con un círculo rojo están los llamados 'nodos interiores' del rectángulo. Son aquellos con

$$n = 1, \dots, N-1 \text{ y } m = 1, 2, \dots$$

Los restantes, marcados con un aspa, son los 'nodos del contorno'. Son aquellos con

$$m = 0 \text{ y con } n = 0 \text{ o } n = N.$$

Ciertos nodos interiores van a dar origen a las ecuaciones en diferencias que van a formar el método numérico. Los nodos exteriores sirven, como antes, para proporcionar procedimientos de arranque del método.

El nodo (x_n, t_m) permite establecer una ecuación escribiendo la expresión

$$\frac{\partial^2 u(x_n, t_m)}{\partial t^2} - \alpha^2 \frac{\partial^2 u(x_n, t_m)}{\partial x^2} = 0$$

de manera que sólo intervengan valores de la solución u y no de sus derivadas.



En ambas segundas derivadas parciales empleamos ahora la misma fórmula de aproximación

$$\begin{aligned} f''(x) &= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} - \frac{h^2}{12} f^{(iv)}(\xi) \\ &= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2), \end{aligned}$$

En el primer caso (respecto de t) la empleamos para cada $f(t) = u(x_n, t)$, para la que

$$\frac{\partial^2 u(x_n, t_m)}{\partial t^2} = f''(t_m),$$

para $x = t_m$ y h el k del retículo, recordando que $t_{m+1} = t_m + k$ y $t_{m-1} = t_m - k$. Así obtenemos

$$\frac{\partial^2 u(x_n, t_m)}{\partial t^2} = \frac{u(x_n, t_{m+1}) - 2u(x_n, t_m) + u(x_n, t_{m-1}))}{k^2} - \frac{k^2}{12} \frac{\partial^4 u(x_n, \eta_m)}{\partial t^4}.$$



En el segundo (respecto de x) la empleamos para cada $f(x) = u(x, t_m)$, para la que

$$\frac{\partial^2 u(x_n, t_m)}{\partial x^2} = f''(x_n),$$

para $x = x_n$ y h el h del retículo, recordando que $x_{n+1} = x_n + h$ y $x_{n-1} = x_n - h$. Así obtenemos

$$\frac{\partial^2 u(x_n, t_m)}{\partial x^2} = \frac{u(x_{n+1}, t_m) - 2u(x_n, t_m) + u(x_{n-1}, t_m))}{h^2} - \frac{h^2}{12} \frac{\partial^4 u(\xi_n, t_m)}{\partial x^4}.$$

Reuniendo ambos términos, acabamos dando como ecuación del método explícito en diferencias con

$$\frac{u_{n\ m+1} - 2u_{nm} + u_{n\ m-1}}{k^2} - \alpha^2 \frac{u_{n+1\ m} - 2u_{nm} + u_{n-1\ m}}{h^2} = 0,$$

con un error de truncación

$$\mathcal{O}(h^2) + \mathcal{O}(k^2).$$



El proceso de mejora de la ecuación nos lleva a poner

$$\lambda = \frac{\alpha k}{h};$$

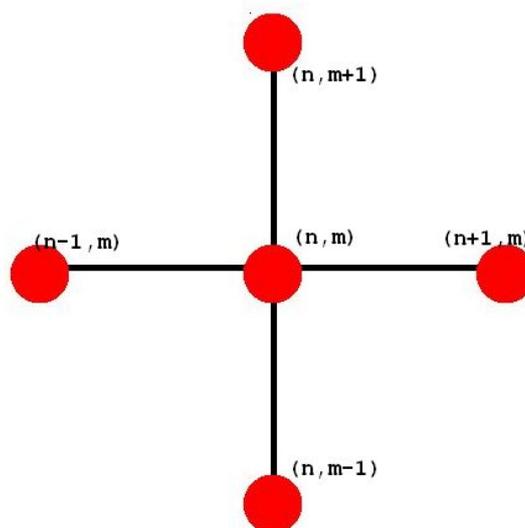
(lo habitual en el caso hiperbólico). Reemplazando este λ en la ecuación que teníamos, resulta

$$u_{nm+1} = 2(1 - \lambda^2)u_{nm} + \lambda^2(u_{n+1m} + u_{n-1m}) - u_{nm-1}$$

para la ecuación creada alrededor de (x_n, t_m) y con validez para $m = 1, 2, \dots$

También está preparada para usarse como fórmula explícita, pero con alguna dificultad que pronto resolveremos.

La **molécula computacional** de este método es



método explícito en diferencias

siendo el punto del centro el que ha servido para crear la ecuación del método.

En lo que respecta a la aportación de las condiciones iniciales y las condiciones de contorno a la asignación de valores a las aproximaciones que no deben ser incógnitas, tendremos ahora

$$\begin{aligned}u_{n0} &= f_{1n}, & n &= 0, \dots, N, \\u_{0m} &= g_{1m}, & m &= 0, 1, 2, \dots, \\u_{Nm} &= g_{2m}, & m &= 0, 1, 2, \dots,\end{aligned}$$

donde hemos puesto $f_{1n} = f_1(x_n)$, $g_{1m} = g_1(t_m)$ y $g_{2m} = g_2(t_m)$. Esto sirve en principio para equilibrar el número de ecuaciones e incógnitas en el cálculo de un cierto número de iteraciones para un cierto número de 'pisos' de aproximaciones. Pero deja en el aire algo esencial: cómo usar la fórmula del método para calcular sucesivamente valores aproximados partiendo de los valores precedentes?

Por ejemplo, para calcular de forma explícita u_{n2} se puede usar $u_{n0} = f_{1n}$, pero se necesitaría también u_{n-11} , u_{n1} y u_{n+11} , que no están calculados aún.



Es posible emplear la notación matricial para expresar este método. Se usan los vectores

$$\mathbf{u}^{(m)} = (u_{1m}, u_{2m}, \dots, u_{N-1m}),$$

donde $\mathbf{u}^{(m)}$ es de nuevo el vector que agrupa las aproximaciones para el valor de t_m , pero únicamente en los nodos interiores.

Además, nos hace falta situarnos en el caso en que las condiciones de contorno son 'homogéneas', esto es, idénticamente nulas. En esa situación, la fórmula de la ecuación del método explícito en diferencias se puede expresar como

$$\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)} - \mathbf{u}^{(m-1)}, \quad m = 1, 2, \dots$$

ahora para la matriz



$$A = \begin{bmatrix} 2(1 - \lambda^2) & \lambda^2 & 0 & \cdots \\ \lambda^2 & 2(1 - \lambda^2) & \lambda^2 & \cdots \\ 0 & \lambda^2 & 2(1 - \lambda^2) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

y donde el vector $\mathbf{u}^{(m)}$ se multiplica como vector columna. También es posible relacionar la matriz A con la misma B del capítulo precedente

$$B = \begin{bmatrix} 2 & -1 & 0 & \cdots \\ -1 & 2 & -1 & \cdots \\ 0 & -1 & 2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

que no contiene el parámetro λ , y de la que conocemos los valores propios. Ahora

$$A = 2I - \lambda^2 B,$$

recordando que A , B y la unidad I son de tamaño $(N - 1) \times (N - 1)$.



Volvemos al arranque como método explícito que ya hemos comentado. En la notación vectorial, el arranque requiere el conocimiento de $\mathbf{u}^{(0)}$ y de $\mathbf{u}^{(1)}$. El primero ya es conocido pues vale

$$\mathbf{u}^{(0)} = (f_{11}, f_{12}, \dots, f_{1N-1}),$$

con valores en principio exactos excepto los pormenores de la representación en ordenador. Para $\mathbf{u}^{(1)} = (u_{11}, u_{21}, \dots, u_{N-11})$ con

$$u_{n1} \sim u(x_n, t_1) = u(x_n, k),$$

vamos a usar justamente la condición

$$\frac{\partial u(x, 0)}{\partial t} = f_2(x), \quad x \in [0, l],$$

que aún no habíamos hecho intervenir.



Ponemos la derivada $\frac{\partial u(x_n, t)}{\partial t}$ como derivada primera ordinaria

$$f'(t) = \frac{\partial u(x_n, t)}{\partial t}$$

para las funciones

$$f : t \longrightarrow u(x_n, t).$$

Como estamos interesados en $\frac{\partial u(x_n, t_1)}{\partial t} = \frac{\partial u(x_n, k)}{\partial t}$, empleamos la fórmula

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(\xi) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h),$$

para $x = t_0 = 0$ y el incremento h igual al k del retículo. Eso nos da

$$\frac{\partial u(x_n, 0)}{\partial t} = \frac{u(x_n, k) - u(x_n, 0)}{k} - \frac{k}{2} \frac{\partial^2 u(x_n, \eta)}{\partial t^2},$$



y, teniendo en cuenta que

$$\frac{\partial u(x_n, 0)}{\partial t} = f_2(x_n),$$

tenemos

$$f_2(x_n) = \frac{u(x_n, k) - u(x_n, 0)}{k} + \mathcal{O}(k).$$

Despejamos $u(x_n, k)$, que es lo que nos interesa,

$$u(x_n, k) = u(x_n, 0) + k f_2(x_n) + \mathcal{O}(k^2),$$

y esta expresión es la que, con error $\mathcal{O}(k^2)$, permite poner, para $u_{n1} \sim u(x_n, k)$,

$$u_{n1} = u_{n0} + k f_{2n},$$

término general del vector $\mathbf{u}^{(1)}$. O sea, se toma

$$\mathbf{u}^{(1)} = (f_{11} + k f_{21}, f_{12} + k f_{22}, \dots, f_{1N-1} + k f_{2N-1}).$$

Ahora, el conocimiento de $\mathbf{u}^{(0)}$ y de $\mathbf{u}^{(1)}$ permite arrancar el método calculando $\mathbf{u}^{(2)}$ y sucesivamente los demás, mediante

$$\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)} - \mathbf{u}^{(m-1)}, \quad m = 1, 2, \dots$$



Lo anterior no es la única forma de conseguir una aproximación para $\mathbf{u}^{(1)}$. Hay otras posibilidades, incluso con menor error de truncación, lo que puede resultar atractivo ya que estamos calculando valores de arranque, que conviene que sean lo más precisos posible. Por ejemplo, desarrollando $u(x_n, k) = u(x_n, t_1)$ alrededor de $u(x_n, 0) = u(x_n, t_0)$, obtenemos

$$u(x_n, k) = u(x_n, 0) + k \frac{\partial u(x_n, 0)}{\partial t} + \frac{k^2}{2} \frac{\partial^2 u(x_n, 0)}{\partial t^2} + \mathcal{O}(k^3),$$

y, como la solución $u(x, t)$ del problema verifica

$$\frac{\partial^2 u(x_n, 0)}{\partial t^2} = \alpha^2 \frac{\partial^2 u(x_n, 0)}{\partial x^2},$$

basta recordar que $u(x, 0) = f_1(x)$ para obtener

$$\frac{\partial^2 u(x_n, 0)}{\partial t^2} = \alpha^2 f_1''(x_n).$$

Esto deja

$$u_{n1} = f_{1n} + k f_{2n} + \frac{k^2}{2} \alpha^2 f_1''(x_n),$$

donde $f_1''(x_n)$ se calculará a partir de la expresión de $f_1(x)$. Y con error de truncación $\mathcal{O}(k^3)$.

Claro, que es posible que $f_1(x)$ no venga dada por una fórmula, sino por una tabla en los valores x_n relativos al retículo. En ese caso es posible poner

$$f_1''(x_n) = \frac{f_1(x_{n+1}) - 2f_1(x_n) + f_1(x_{n-1}))}{h^2} + \mathcal{O}(h^2)$$

usando la fórmula de aproximación de la derivada segunda, y obtener la aproximación

$$\begin{aligned} u_{n1} &= f_{1n} + k f_{2n} - \lambda^2 f_{1n} + \frac{\lambda^2}{2} f_{1n+1} + \frac{\lambda^2}{2} f_{1n-1} = \\ &= (1 - \lambda^2) f_{1n} + \frac{\lambda^2}{2} f_{1n+1} + \frac{\lambda^2}{2} f_{1n-1} + k f_{2n} \end{aligned}$$

para los valores de las u_{n1} que componen el vector $\mathbf{u}^{(1)}$.

Ejemplo 1

Traemos también aquí un ejemplo que se encuentra en el programa realizado en 'Maple' **Cap13_ejemplo1.mws** para el problema hiperbólico

$$\frac{\partial^2 u}{\partial t^2} - 4 \frac{\partial^2 u}{\partial x^2} = 0,$$

en el dominio $x \in [0, 1]$, $t \in [0, +\infty)$, con condiciones homogéneas de contorno y condiciones iniciales $u(x, 0) = \text{sen}(\pi x)$ y $\partial u(x, 0)/\partial t = 0$ ambas para $x \in [0, 1]$.

Este problema tiene como única solución la función

$$u(x, y) = \text{sen}(\pi x) \cos(2 \pi t).$$

Haremos los cálculos con 6 cifras decimales significativas con redondeo.

Notemos que $\alpha = 2$.

Para nuestros cálculos construimos un retículo dividiendo $[0, 1]$ para x en $N = 10$ subintervalos, tomando $h = 1/N = 0,1$ y luego $x_n = 0 + n h$, lo que hace



$x_0 = 0.$, $x_1 = 0,1$, $x_2 = 0,2$, $x_3 = 0,3$, $x_4 = 0,4$, $x_5 = 0,5$,
 $x_6 = 0,6$, $x_7 = 0,7$, $x_8 = 0,8$, $x_9 = 0,9$, $x_{10} = 1$.

Vamos a realizar varias integraciones, para diferentes valores de k , llegando todas ellas al valor $t = 1$.

Comenzamos tomando para t el valor $k = 0,05$, lo que hace $t_0 = 0.$, $t_1 = 0,05$, $t_2 = 0,1$, ... y $t_{20} = 1$. Con este valor,

$$\lambda = \frac{\alpha k}{h} = 1.;$$

y obtenemos la matriz

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots \\ 1 & 0 & 1 & \cdots \\ 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Integramos el problema hasta $t_{20} = 1$.

Para el contorno de x tenemos que $u_{0m} = 0$. y $u_{10m} = 0$. para todo m (condiciones homogéneas).



Los valores de arranque son para $\mathbf{u}^{(0)}$ los $u_{n0} = f_{1n} = \text{sen}(\pi 0,1 n)$, o sea

$u_{10} = ,309017$, $u_{20} = ,587785$, $u_{30} = ,809017$, $u_{40} = ,951057$, $u_{50} = 1.$, $u_{60} = ,951058$, $u_{70} = ,809020$, $u_{80} = ,587789$, $u_{90} = ,309020$ y como $\partial u(x, 0)/\partial t = f_2(x) \equiv 0$, obtenemos en este sencillo caso que $f_{1n} + k f_{2n} = f_{1n}$ y que $\mathbf{u}^{(1)} = \mathbf{u}^{(0)}$.

No hay mas que repetir 20 veces la iteración

$$\mathbf{u}^{(m+1)} = A\mathbf{u}^{(m)} - \mathbf{u}^{(m-1)},$$

a partir de $\mathbf{u}^{(0)}$ y $\mathbf{u}^{(1)}$ para obtener $\mathbf{u}^{(20)}$, que da los valores aproximados de la solución en $t_{20} = 1.$, y que vale

$$\mathbf{u}^{(20)} = (0,309024, 0,587769, 0,809011, 0,951047, 1,00002, \\ , 0,951030, 0,809028, 0,587785, 0,309023,$$

además, claro, de los valores nulos en los extremos. Así pues



$$\begin{array}{ll} u(0, 1.) \sim u_{020} = 0. & u(0,1, 1.) \sim u_{120} = 0,309024 \\ u(0,2, 1.) \sim u_{220} = 0,587769 & u(0,3, 1.) \sim u_{320} = 0,809011 \\ u(0,4, 1.) \sim u_{420} = 0,951047 & u(0,5, 1.) \sim u_{520} = 1,00002 \\ u(0,6, 1.) \sim u_{620} = 0,951030 & u(0,7, 1.) \sim u_{720} = 0,809028 \\ u(0,8, 1.) \sim u_{820} = 0,587785 & u(0,9, 1.) \sim u_{920} = 0,309023 \\ u(1, 1.) \sim u_{1020} = 0. , & \end{array}$$

Vamos a comparar estos valores aproximados (obtenidos con 6 cifras decimales significativas con redondeo) con los verdaderos valores de la solución. Se obtiene el mayor error para

$$u_{620} = 0,951030 \text{ para } u(0,6, 1.) = 0,951058 \text{ con error} = 0,000028.$$

Como el error de truncación es $\mathcal{O}(h^2) + \mathcal{O}(k^2)$ y, en este caso

$$h^2 + k^2 = 0,1^2 + 0,05^2 = 0,0125,$$

es posible que este error sea un poco bajo para lo que podíamos esperar.



Y eso es lo que en verdad sucede, debido a una combinación del valor de $\lambda = 1$ y a la forma de la ecuación. Vamos a verlo.

Como $\lambda = \alpha k/h = 1$, se tiene que $\alpha k = h$ y que $\alpha^2 k^2 = h^2$. Por otra parte, el error de truncación $\mathcal{O}(h^2) + \mathcal{O}(k^2)$ es, más exactamente,

$$-\alpha^2 \frac{h^2}{12} \frac{\partial^4 u(\xi_n, t_m)}{\partial x^4} + \frac{k^2}{12} \frac{\partial^4 u(x_n, \eta_m)}{\partial t^4}.$$

Pero esta expresión es también de hecho un par de series infinitas de la forma

$$\begin{aligned} & -2\alpha^2 \left(\frac{h^2}{4!} \frac{\partial^4 u(x_n, t_m)}{\partial x^4} + \frac{h^4}{6!} \frac{\partial^6 u(x_n, t_m)}{\partial x^6} + \dots \right) + \\ & + 2 \left(\frac{k^2}{4!} \frac{\partial^4 u(x_n, t_m)}{\partial t^4} + \frac{k^4}{6!} \frac{\partial^6 u(x_n, t_m)}{\partial t^6} + \dots \right) \end{aligned}$$

que organizamos, aislando los términos más importantes, como

$$\begin{aligned} & -\frac{2}{4!} \left(\alpha^2 h^2 \frac{\partial^4 u(x_n, t_m)}{\partial x^4} - k^2 \frac{\partial^4 u(x_n, t_m)}{\partial t^4} \right) - \\ & -\frac{2}{6!} \left(\alpha^2 h^4 \frac{\partial^6 u(x_n, t_m)}{\partial x^6} - k^4 \frac{\partial^6 u(x_n, t_m)}{\partial t^6} \right) - \dots \end{aligned}$$

Ahora bien, sabemos que la solución cumple

$$\frac{\partial^2 u}{\partial t^2} - \alpha^2 \frac{\partial^2 u}{\partial x^2} = 0$$

y, por lo tanto,

$$\begin{aligned} \frac{\partial^4 u}{\partial t^4} &= \frac{\partial^2}{\partial t^2} \left(\frac{\partial^2 u}{\partial t^2} \right) = \frac{\partial^2}{\partial t^2} \left(\alpha^2 \frac{\partial^2 u}{\partial x^2} \right) = \frac{\partial^2}{\partial x^2} \left(\alpha^2 \frac{\partial^2 u}{\partial t^2} \right) = \\ &= \frac{\partial^2}{\partial x^2} \left(\alpha^4 \frac{\partial^2 u}{\partial x^2} \right) = \alpha^4 \frac{\partial^4 u}{\partial x^4} \end{aligned}$$

luego el primero de los términos en que estaba organizado el error

vale

$$\alpha^2 h^2 \frac{\partial^4 u}{\partial x^4} - k^2 \frac{\partial^4 u}{\partial t^4} = \alpha^2 h^2 \frac{\partial^4 u}{\partial x^4} - \alpha^4 k^2 \frac{\partial^4 u}{\partial x^4} = \alpha^2 (h^2 - \alpha^2 k^2) \frac{\partial^4 u}{\partial x^4} = 0$$

porque, como habíamos dicho, $\lambda = \alpha k/h = 1$ asegura que $h^2 - \alpha^2 k^2 = 0$.

Esto justificaría ya que el error sea más pequeño de lo previsto, pero es que, además, se puede probar de igual manera que todos los otros términos del error son también 0. O sea, es nulo el error de truncación en este caso, y el error que aparece se debe sólo a los errores de redondeo al trabajar con 6 cifras decimales.

Cualquier otro valor de λ , incluso más pequeño, proporciona error de truncación acorde con lo que se espera. Por ejemplo, tomamos ahora $k = 0,01$, más pequeño, lo que da

$$\lambda = \frac{\alpha k}{h} = 0,2;$$

la matriz A que se obtiene es

$$A = \begin{bmatrix} 1,92 & 0,04 & 0 & \dots \\ 0,04 & 1,92 & 0,04 & \dots \\ 0 & 0,04 & 1,92 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$

Integramos el problema hasta $t_{100} = 1$. repitiendo 100 veces la iteración

$$\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)} - \mathbf{u}^{(m-1)},$$

a partir de los mismos $\mathbf{u}^{(0)}$ y $\mathbf{u}^{(1)}$ que antes. La aproximación de la solución en $t_{100} = 1$. vale

$$\mathbf{u}^{(100)} = (0,309115, 0,588073, 0,809184, 0,951293, 1,00015, \\ , 0,951123, 0,809198, 0,587854, 0,309154,$$

además, claro, de los valores nulos en los extremos.

El mayor error se da para

$u_{2100} = 0,588073$ para $u(0,2, 1.) = 0,587785$ con error = 0,000288. y es más coherente con su orden $\mathcal{O}(h^2) + \mathcal{O}(k^2)$ y con el valor

$$h^2 + k^2 = 0,1^2 + 0,01^2 = 0,0101,$$

Termina el ejemplo una elección mayor de $k = 0,1$, lo que hace

$$\lambda = \frac{\alpha k}{h} = 2.;$$

La matriz A es

$$A = \begin{bmatrix} -6 & 4 & 0 & \dots \\ 4 & -6 & 4 & \dots \\ 0 & 4 & -6 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

y las iteraciones necesarias para llegar a $t_{10} = 1$. son solamente 10

La aproximación de la solución en $t_{10} = 1$. vale

$$\mathbf{u}^{(100)} = (-4004,44, 10368,4, -18951,5, 26741,2, -29921,0, \\ , 26779,6, -19010,1, 10422,5, -4035,78 ,$$

además, claro, de los valores nulos en los extremos.

El error ahora es desproporcionado, llegando a

$$u_{5,10} = -29921,0 \text{ para } u(0,5, 1.) = 1. \text{ con error} = 29922,0.$$

y proviene solamente de la inestabilidad del método en este caso, ya que

$$h^2 + k^2 = 0,1^2 + 0,1^2 = 0,02.$$



No hemos buscado aún condiciones sobre λ que garanticen la estabilidad del método explícito en diferencias. No va a ser sencillo y lo primero que hay que observar es que carecemos de la posibilidad de uso del procedimiento matricial. El hecho de que la iteración matricial sea ahora

$$\mathbf{u}^{(m+1)} = A \mathbf{u}^{(m)} - \mathbf{u}^{(m-1)},$$

hace que $\rho(A) < 1$ no sea ahora suficiente para garantizar la estabilidad.

En su lugar obtendremos que $\lambda \leq 1$ es, para el método explícito en diferencias, una condición necesaria para la estabilidad, lo que nos asegura inestabilidad para los valores $\lambda > 1$.

Justamente, acabamos de notar experimentalmente que $\lambda = 2$ proporciona un resultado inestable. Junto a ello hemos visto que tanto $\lambda = 1$ como $\lambda = 0,2$ correspondían a respuestas estables del método.



El procedimiento de FOURIER

Para el problema hiperbólico en $D = [0, l] \times [0, +\infty)$

$$\begin{aligned}\frac{\partial^2 u}{\partial t^2} - \alpha^2 \frac{\partial^2 u}{\partial x^2} &= 0 \\ u(x, 0) &= f_1(x) \\ \frac{\partial u(x, 0)}{\partial t} &= f_2(x) \\ u(0, t) &= 0, \quad u(l, t) = 0\end{aligned}$$

y el método explícito en diferencias

$$u_{nm+1} = 2(1 - \lambda^2)u_{nm} + \lambda^2(u_{n+1m} + u_{n-1m}) - u_{nm-1}$$

con $\lambda = \alpha k/h$, vamos a usar el procedimiento de FOURIER, que proporciona condiciones necesarias sobre λ para la estabilidad del método.

h y k , n y m están asociados, como siempre al retículo habitual.



Que

$$v_{nm} = e^{i nbh} e^{mdk}$$

verifique la ecuación del método explícito en diferencias significa que

$$\begin{aligned}e^{i nbh} e^{(m+1)dk} &= 2(1 - \lambda^2)e^{i nbh} e^{mdk} + \\ &+ \lambda^2(e^{i(n+1)bh} e^{mdk} + e^{i(n-1)bh} e^{mdk}) - e^{i nbh} e^{(m-1)dk}.\end{aligned}$$

La división por los factores no nulos $e^{i nbh}$ y e^{mdk} proporciona

$$e^{dk} = 2(1 - \lambda^2) + \lambda^2(e^{i bh} + e^{-i bh}) - e^{-dk}.$$

Usando

$$e^{i bh} + e^{-i bh} = 2 \cos bh,$$

queda

$$e^{dk} = 2(1 - \lambda^2) + 2\lambda^2 \cos bh - e^{-dk}.$$

Usando además

$$\cos 2a = 1 - 2 \operatorname{sen}^2 a,$$



para $a = bh/2$,

$$e^{dk} = 2 - 2\lambda^2 + 2\lambda^2 - 4\lambda^2 \operatorname{sen}^2 \frac{bh}{2} - e^{-dk},$$

o sea,

$$e^{dk} - 2 \left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right) + e^{-dk} = 0,$$

expresión que escribimos como trinomio de segundo grado en e^{dk}

$$\left(e^{dk} \right)^2 - 2 \left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right) e^{dk} + 1 = 0.$$

Como ocurre en todo trinomio de ese tipo con término independiente 1, el producto de las dos soluciones de e^{dk} es 1. Supongamos ahora que el método explícito en diferencias es estable; entonces, para ambas soluciones,

$$|e^{dk}| \leq 1$$

por lo que (al ser 1 su producto) ambas son de módulo exactamente 1.



Si el discriminante del trinomio

$$\Delta = 4 \left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right)^2 - 4$$

fuese > 0 ambas soluciones de módulo 1 serían reales y distintas (luego ± 1), lo que contradice el hecho de que su producto es 1. Esto obliga a que $\Delta \leq 0$, o sea

$$4 \left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right)^2 - 4 \leq 0,$$

lo que va implicando

$$\left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right)^2 - 1 \leq 0,$$

$$\left(1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right)^2 \leq 1,$$



$$\left| 1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \right| \leq 1,$$

$$-1 \leq 1 - 2\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \leq 1,$$

$$-2 \leq -2\lambda^2 \operatorname{sen}^2 \frac{bh}{2},$$

$$\lambda^2 \operatorname{sen}^2 \frac{bh}{2} \leq 1$$

para b arbitrario, o sea,

$$\lambda^2 \leq 1, \text{ y}$$

$$\lambda \leq 1,$$

que constituye así una condición necesaria para la estabilidad del método explícito en diferencias, por lo que este método será inestable siempre que $\lambda > 1$, como ya habíamos adelantado.

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-01

Se considera la 'ecuación de convección-difusión'

$$\frac{\delta u}{\delta t} + \alpha^2 \frac{\delta u}{\delta x} = \beta^2 \frac{\delta^2 u}{\delta x^2}$$

y el método en diferencias finitas para esta ecuación

$$\frac{u_{nm+1} - u_{nm}}{k} + \alpha^2 \frac{u_{n+1m} - u_{n-1m}}{2h} - \beta^2 \frac{u_{n+1m} - 2u_{nm} + u_{n-1m}}{h^2} = 0$$

Dígase cuál es el orden del error de truncación. Descríbase la molécula computacional.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-02

Para el problema hiperbólico

$$\frac{\delta^2 u}{\delta t^2} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f_1(x), \quad \frac{\delta u}{\delta t}(x, 0) = f_2(x),$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con el retículo habitual

$$h = \frac{l}{N}, \quad x_n = n h, \quad x_0 < x_1 < \dots < x_N,$$

$$k > 0, \quad t_m = m k, \quad t_0 < t_1 < t_2 < \dots,$$

se hacen las aproximaciones

$$\text{para } \frac{\delta^2 u}{\delta t^2}, \quad f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$$

$$\text{para } \frac{\delta^2 u}{\delta x^2}, \quad f''(x) = \frac{-f(x+2h) + 16f(x+h) - 30f(x) + 16f(x-h) - f(x-2h)}{12h^2} + O(h^4)$$

Dedúzcase el método en diferencias que resulta, su orden, y describese la molécula computacional.

Usando el método de FOURIER, búsqese la condición sobre $\lambda = \frac{\alpha k}{h}$ para

que el método sea estable.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-03 (del EXAMEN ORDINARIO de 09FEB04)

Para la ecuación

$$\frac{\delta u}{\delta t} + \frac{\delta u}{\delta x} - \frac{\delta^2 u}{\delta x^2} = f(x,t)$$

se considera un retículo con $h = k$ y el método en diferencias finitas

$$u_{n m + 1} = 2(1 - \lambda) u_{n m} - (1 - \lambda) u_{n + 1 m} + \lambda u_{n - 1 m} + h f_{n m}$$

donde se ha puesto $\lambda = 1/h$. Dígase cuál es el orden del error de truncación.

Describese la molécula computacional.

```
[ > restart:  
[ > interface(labeling=false):
```

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-04 (del EXAMEN ORDINARIO de 09FEB04)

Para el problema hiperbólico

$$\frac{\delta^2 u}{\delta t^2} - \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f_1(x), \quad \frac{\delta u}{\delta t}(x, 0) = f_2(x),$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

se considera un retículo

$$h = \frac{l}{N}, \quad x_n = n h, \quad x_0 < x_1 < \dots < x_N,$$

$$k = h, \quad t_m = m k = m h, \quad t_0 < t_1 < t_2 < \dots,$$

donde se utiliza el mismo incremento para x y para t . Para crear un método en diferencias se aproximan ambas derivadas segundas mediante la fórmula

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$$

Dígase qué ecuación en diferencias resulta, describábase la molécula computacional y escribábase el método en forma matricial. Finalmente, discútase la estabilidad del método empleando alguno de los procedimientos habituales.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-05 (del EXAMEN ORDINARIO de 04FEB08)

Para el problema hiperbólico

$$\frac{\delta^2 u}{\delta t^2} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f_1(x), \quad \frac{\delta u}{\delta t}(x, 0) = f_2(x),$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

con el retículo habitual

$$h = \frac{l}{N}, \quad x_n = n h, \quad x_0 < x_1 < \dots < x_N,$$

$$k > 0, \quad t_m = m k, \quad t_0 < t_1 < t_2 < \dots,$$

se hacen las aproximaciones

$$\text{para } \frac{\delta^2 u}{\delta t^2}, \quad f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2)$$

$$\text{para } \frac{\delta^2 u}{\delta x^2}, \quad f''(x) = \frac{-f(x+2h) + 6f(x+h) - 10f(x) + 6f(x-h) - f(x-2h)}{2h^2} + O(h^2)$$

Dedúzcase el método en diferencias que resulta, su orden, y describese la molécula computacional.

Además, dígase lo que afirma el método de FOURIER sobre $\lambda = \frac{\alpha k}{h}$ cuando

se analiza la estabilidad del método.

```
[ > restart:
[ > interface(labeling=false):
```

Ejercicios de

(13) Las Ecuaciones hiperbólicas

Ejercicio 13-06 (del EXAMEN EXTRAORDINARIO de 01JUL05)

Para el problema hiperbólico

$$\frac{\delta^2 u}{\delta t^2} - \alpha^2 \frac{\delta^2 u}{\delta x^2} = 0$$

$$u(x, 0) = f_1(x), \quad \frac{\delta u}{\delta t}(x, 0) = f_2(x),$$

$$u(0, t) = 0 \text{ y } u(1, t) = 0,$$

se considera el retículo usual $[0, 1] \times [0, +\infty)$ y el método en diferencias finitas

$$u_{n_{m+1}} - 2u_{n_m} + u_{n_{m-1}} - \frac{\lambda^2}{3} (u_{n_{+2m}} - u_{n_{+1m}} - u_{n_{-1m}} + u_{n_{-2m}}) = 0$$

donde se ha puesto $\lambda = \alpha k / h$. Dígase cuál es el orden de truncación del método.

Además, dígase qué ocurre cuando se estudia la estabilidad del método mediante el procedimiento de FOURIER.

```
[ > restart:
[ > interface(labeling=false):
```

14. El Método de Líneas (MOL)

1 Semidiscretización de la ecuación del calor

El Método de Líneas (MOL)

Se considera el siguiente problema, ligado a una ecuación parabólica

$$\begin{aligned}\frac{\partial u(x, t)}{\partial t} &= \frac{\partial^2 u(x, t)}{\partial x^2}, & x \in [0, 1], t \geq 0, \\ u(x, 0) &= \text{sen } \pi x, \\ u(0, t) &= u(1, t) = 0\end{aligned}$$

que representa, por ejemplo, la distribución de temperaturas en una barra de longitud 1 con distribución inicial $\text{sen } \pi x$, $x \in [0, 1]$ y temperatura 0 en los extremos, a lo largo de todo el proceso. Es fácil comprobar que su única solución viene dada por

$$u(x, t) = e^{-\pi^2 t} \text{sen } \pi x$$

(o sea, la temperatura tiende a 0 con rapidez).

Definimos en $x \in [0, 1]$ los nodos

$$x_n = n \cdot h, \quad \text{con } h = \frac{1}{N}, \quad n = 0, 1, \dots, N-1, N$$

o sea, nodos

$$0 = x_0 < x_1 < x_2 < \dots < x_{N-1} < x_N = 1$$

separados por distancias h .

Para cada x_n nos vamos a referir a la semirecta 'vertical' (x_n, t) , $t \geq 0$ y a considerar en ella una función de t , descripción de la solución de la ecuación original en derivadas parciales, pero limitada a la semirecta, en definitiva

$$u_n(t) = u(x_n, t), \quad t \in [0, +\infty].$$

Nótese que, para los extremos de la barra, las $u_0(t)$ y $u_N(t)$ valen obligatoriamente

$$\begin{aligned}u_0(t) &= u(x_0, t) = u(0, t) = 0 \\ u_N(t) &= u(x_N, t) = u(1, t) = 0\end{aligned}$$

pero no son conocidas las que corresponden a los otros índices $n = 1, \dots, N - 1$.

Como se observa, hemos hecho una discretización en una de las variables, la x , pero no en la otra, t , que sigue siendo continua. Esto justifica el nombre dado a este procedimiento, que acaba obteniendo las soluciones (generalmente numéricas) a lo largo de las líneas $x = x_n, t \in [0, +\infty]$.

Vamos a ver cómo somos capaces de decir qué es lo que verifican las funciones

$$u_n(t) = u(x_n, t), \quad t \in [0, +\infty], \quad n = 0, 1, \dots, N - 1, N.$$

Fijado un valor de $t \in [0, +\infty]$, la función (una diferente para cada valor de t)

$$f : x \longrightarrow u(x, t)$$

es tal que

$$\frac{\partial^2 u(x, t)}{\partial x^2} = f''(x),$$



y, en particular,

$$\frac{\partial^2 u(x_n, t)}{\partial x^2} = f''(x_n).$$

Aproximando esta derivada por la fórmula usual, acabamos obteniendo

$$\frac{\partial^2 u(x_n, t)}{\partial x^2} = \frac{u(x_{n-1}, t) - 2u(x_n, t) + u(x_{n+1}, t))}{h^2} + \mathcal{O}(h^2),$$

Ahora bien, $u(x, t)$ verifica

$$\frac{\partial^2 u(x_n, t)}{\partial x^2} = \frac{\partial u(x_n, t)}{\partial t} = u'_n(t), \quad n = 1, \dots, N - 1,$$

y, en definitiva

$$u'_n(t) = \frac{u_{n-1}(t) - 2u_n(t) + u_{n+1}(t))}{h^2} + \mathcal{O}(h^2).$$

Es decir, las funciones $u_n(t)$ consideradas antes son solución del sistema lineal de ecuaciones diferenciales ordinarias



$$u'_n = \frac{1}{h^2} (u_{n-1} - 2u_n + u_{n+1}), \quad n = 1, \dots, N-1,$$

con un error de truncación de $\mathcal{O}(h^2)$.

Agrupando las soluciones escalares en el vector de funciones

$$\mathbf{u}(t) = (u_1(t), u_2(t), \dots, u_{N-1}(t)),$$

y teniendo en cuenta que las funciones $u_0(t)$ y $u_N(t)$ son (condiciones de contorno) idénticamente nulas, escribimos matricialmente este sistema como

$$\mathbf{u}' = \frac{1}{h^2} A \mathbf{u},$$

donde A es la matriz

$$A = \begin{bmatrix} -2 & 1 & 0 & \cdots \\ 1 & -2 & 1 & \cdots \\ 0 & 1 & -2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}.$$



Por otra parte, la condición inicial (para u)

$$u(x, 0) = \text{sen } \pi x,$$

da origen ahora a las condiciones iniciales para este sistema

$$u_n(0) = u(x_n, 0) = \text{sen } \pi x_n = \text{sen}(\pi n h),$$

que garantizan solución única del sistema.

Resumiendo, el **Método de Líneas (MOL)** permite resolver numéricamente la ecuación parabólica en derivadas parciales como un sistema lineal y de coeficientes constantes de ecuaciones diferenciales ordinarias. Naturalmente, aquí entran en juego los métodos de resolución numérica de este tipo de sistemas.

Es conveniente aclarar el carácter más o menos 'stiff' que pueda tener este sistema lineal, y es lo que examinamos a continuación. Este hecho nos moverá a usar unos u otros métodos, dependiendo del carácter citado.



Que el sistema anterior sea 'stiff' depende de los autovalores de la matriz $\frac{1}{h^2} A$. A estas alturas ya nos son conocidos y valen

$$\mu_n = -\frac{4}{h^2} \left(\operatorname{sen} \frac{n\pi}{2N} \right)^2, \quad n = 1, \dots, N-1.$$

(A es ahora la matriz $-B$, para la matriz B ya tratada en el capítulo destinado al caso parabólico). Son reales y negativos, y más negativos cuando se hace decrecer el tamaño de h , que es el objetivo si se desea minimizar el error de truncación del sistema aproximado, cuya truncación es del orden $\mathcal{O}(h^2)$.

O sea, nos encontramos no sólo ante un problema 'stiff', sino que nos interesa de alguna manera acentuar dicho carácter.

Puesto que se trata de un ejemplo sencillo, sabemos que la (única) solución del sistema lineal es

$$u_n(t) = e^{-4N^2 \left(\operatorname{sen} \frac{\pi}{2N} \right)^2 t} \operatorname{sen} \frac{n\pi}{N}, \quad n = 1, \dots, N-1,$$

hecho que vamos a comprobar. Por un lado,

$$u'_n(t) = -4N^2 \left(\operatorname{sen} \frac{\pi}{2N} \right)^2 e^{-4N^2 \left(\operatorname{sen} \frac{\pi}{2N} \right)^2 t} \operatorname{sen} \frac{n\pi}{N}.$$

Por el otro, recordando que $h = 1/N$,

$$\begin{aligned} & \frac{1}{h^2} (u_{n-1}(t) - 2u_n(t) + u_{n+1}(t)) \\ &= N^2 e^{-4N^2 \left(\operatorname{sen} \frac{\pi}{2N} \right)^2 t} \left(\operatorname{sen} \frac{(n-1)\pi}{N} - 2 \operatorname{sen} \frac{n\pi}{N} + \operatorname{sen} \frac{(n+1)\pi}{N} \right). \end{aligned}$$

La parte $N^2 e^{-4N^2 \left(\operatorname{sen} \frac{\pi}{2N} \right)^2 t}$ es común a ambas expresiones; para el resto, aplicando

$$\operatorname{sen}(a+b) + \operatorname{sen}(a-b) = 2 \operatorname{sen} a \cos b$$

para $a = n\pi/N$ y $b = \pi/N$, se tiene

$$\begin{aligned}
 & \operatorname{sen} \frac{(n-1)\pi}{N} - 2 \operatorname{sen} \frac{n\pi}{N} + \operatorname{sen} \frac{(n+1)\pi}{N} \\
 &= -2 \operatorname{sen} \frac{n\pi}{N} + 2 \operatorname{sen} \frac{n\pi}{N} \cos \frac{\pi}{N} \\
 &= -2 \operatorname{sen} \frac{n\pi}{N} \left(1 - \cos \frac{\pi}{N}\right),
 \end{aligned}$$

y aplicando ahora

$$\cos 2a = 1 - 2 \operatorname{sen}^2 a, \quad a = \frac{\pi}{2N},$$

la igualdad continua como

$$\begin{aligned}
 &= -2 \operatorname{sen} \frac{n\pi}{N} \cdot 2 \left(\operatorname{sen} \frac{\pi}{2N}\right)^2 \\
 &= -4 \left(\operatorname{sen} \frac{\pi}{2N}\right)^2 \operatorname{sen} \frac{n\pi}{N},
 \end{aligned}$$

lo que prueba que, efectivamente

$$u'_n(t) = \frac{1}{h^2} (u_{n-1}(t) - 2u_n(t) + u_{n+1}(t)).$$



Además, estas soluciones $u_n(t)$ del sistema verifican también la condición inicial

$$u_n(0) = \operatorname{sen}(\pi n h) = \operatorname{sen} \frac{n\pi}{N}.$$

Desde el punto de vista numérico, ayudándose de métodos adecuados para sistemas 'stiff', es posible, haciendo que $h \rightarrow 0$, conseguir una aproximación suficiente de las soluciones sobre nodos situados sobre las semirrectas $x = x_n$.

Desde el punto de vista matemático, las soluciones $u_n(t)$ deberían también aproximarse a las trazas sobre $x_n = nh = n/N$ de la solución conocida $u(x, t)$ del problema parabólico original.

Y así es cuando $h \rightarrow 0$ o $N \rightarrow \infty$ ya que

$$\lim_{N \rightarrow \infty} \frac{\operatorname{sen} \frac{\pi}{2N}}{\frac{\pi}{2N}} = 1,$$

y

$$\lim_{N \rightarrow \infty} u_n(t) = e^{-\pi^2 t} \operatorname{sen} n\pi h = e^{-\pi^2 t} \operatorname{sen} \pi x_n = u(x_n, t).$$



El Método de Líneas (MOL) que hemos descrito en un ejemplo lleva una Ecuación en derivadas parciales a un sistema de Ecuaciones diferenciales ordinarias cuando se buscan soluciones numéricas para valores nodales de una de las variables, $0 = x_0 < x_1 < x_2 < \dots < x_{N-1} < x_N = 1$ en el ejemplo. Naturalmente que, a continuación, la búsqueda de la solución numérica de este sistema lleva a la discretización en la otra variable, y las soluciones se calculan para nodos $0 = t_0 < t_1 < t_2 < \dots$. Finalmente, el proceso total es un método en diferencias para la Ecuación en derivadas parciales, pero ejecutado en dos tiempos independientes. La discretización en t en el ejemplo, es en principio independiente de la inicial en x , e incluso puede evitarse esta discretización cuando el sistema resultante presenta una solución analítica razonablemente sencilla.

En este sentido, el MOL tiene un interés que puede ir más allá de lo numérico, ligando una Ecuación en derivadas parciales con un problema, ya en una sola variable, de Ecuaciones diferenciales ordinarias. Y es este campo, numéricamente o no, hay gran cantidad de resultados utilizables.

Finalmente no hay que olvidar el carácter 'stiff' del sistema que resulta en la generalidad de los casos, que lleva a que los avances hacia valores más y más cercanos de los nodos x_n sean prudentes.

Ejercicios de

(14) El método de líneas (M.O.L.)

