Euskal Herriko Unibertsitatea/ Universidad del País Vasco

Konputazio Zientziak eta Adimen Artifiziala Saila
Departamento de Ciencias de la Computación e Inteligencia Artificial

Sobre los errores locales y globales de la integración de Ecuaciones Diferenciales Ordinarias mediante métodos de Runge-Kutta Explícitos.

Joseba Makazaga Odria

Donostia, Julio 2007

Euskal Herriko Unibertsitatea/ Universidad del País Vasco

Konputazio Zientziak eta Adimen Artifiziala Saila
Departamento de Ciencias de la Computación e Inteligencia Artificial

Sobre los errores locales y globales de la integración de Ecuaciones Diferenciales Ordinarias mediante métodos de Runge-Kutta Explícitos.

Memoria que para optar al grado de Doctor en Informática presenta

Joseba Makazaga Odria

Dirigida por
Ander Murua Uria

Donostia, Julio 2007

Agradecimientos

Laguntza handia jaso dut eta nigatik asko eman dute nire ingurukoek, horregatik, jende askori eman nahi dizkiot eskerrak, maila batean beraiei zor diedalako lan hau egin izana.

Hasteko, Anderri. Beste hainbat arrazoiren artean, zuzendari lanak bere gain hartu izanagatik. Iruditzen zait bere tesia egiteak baino lan handiagoa eman diodala, eta irakatsi didan guztiari esker egin ahal izan dut lan hau. Berari esker izan da posible.

Jarraitzeko, etxekoei, eta batez ere Mariri, eman didan laguntza guztiagatik eta etxean egin ez ditudan lan guztietaz arduratu delako.

Ez ditut ahaztu nahi sailkideak eta grafikoetako irakaskideak, bereziki, Aitor eta Ander (berriz!): hainbat erraztasun eman izan didatelako, egunerokо lana arinduz, tesian lana lasaiago egin ahal izan nezan.

Langiro atsegina izateak asko laguntzen duela argi daukat, horregatik, *Kalidadebida*ko guztiei eta kafea hartzera laguntzen didatenei emandako laguntza morala eta zientifikoa eskertu nahi diet.

Azkenik, arraroa bada ere, Osakidetzako erreumatologia taldeari ere eskerrak eman behar dizkiot, nire gaitz malapartatuarekin bizitzea posible egin dutelako.

Guztiei, eskerrik asko!

Mariri

Índice general

1. Introducción **1**

1.1. Ecuaciones Diferenciales Ordinarias 3
1.2. Resolución numérica 4
1.3. Métodos de Runge-Kutta 4
1.4. Flujo de un sistema autónomo 5
1.5. Error local y error global 6
1.6. Condiciones de orden de los Métodos de RK 7
1.7. Árboles y condiciones de orden 12
1.8. Series formales y condiciones de orden 14
1.9. Error global 19
1.9.1. Métodos de estimación del Error Global 20
1.10. Control de las longitudes de paso de la integración 23
1.11. Estabilidad lineal de los métodos de Runge-Kutta explícitos 27

2. Métodos de Tipo Runge-Kutta con estimación del Error Global **29**

2.1. Introducción 29
2.2. Error Global de los métodos de un paso 30
2.3. Una clase general de esquemas para la obtención del Error Global 32
2.4. Métodos Runge-Kutta embebidos con estimación del Error Global 35
2.5. Condiciones sobre los parámetros del método 36
2.6. Consideraciones prácticas 42
2.7. Región de estabilidad de los métodos 43
2.8. Representación binaria de los árboles 45
2.8.1. Condiciones de orden con la representación binaria de los árboles 49
2.9. Construcción de un método de 7 etapas 52

2.9.1. Aplicación de las propiedades de BSRK5 a las condiciones del método $\bar{\psi}_h(y, \bar{y})$ 53
2.10. Experimentos numéricos 59
2.11. Conclusiones de los experimentos 64
2.12. Experimentos con longitud de paso variable 64
2.13. Condiciones de los parámetros del nuevo método 67
2.14. Construcción de un método de orden 5 68
2.15. Experimentos numéricos 70
2.16. Conclusiones 74

3. Control de la Longitud de Paso Basado en Estimaciones del Error Global 77
3.1. Introducción 77
3.2. Utilización de la estimación del error global 78
3.2.1. Propagación de los errores 78
3.3. Nuevas estrategias de selección de longitud de paso 84
3.4. Implementación de la nueva estrategia 85
3.5. Experimentos numéricos con el nuevo método de ajuste de la longitud de paso 86
3.6. Conclusiones 94

4. Comparación de los métodos de Runge-Kutta 95
4.1. Introducción 95
4.2. Notación y resultados básicos 96
4.2.1. Árboles con raíz, diferenciales elementales y B-series . 96
4.2.2. Cotas de las derivadas de las funciones analíticas . . 99
4.3. Cotas para el error local de los métodos RK 101
4.3.1. Cotas de las diferenciales elementales 102
4.3.2. Cotas para la expansión de Taylor del error local . . 105
4.3.3. La dependencia de la norma elegida 114
4.4. Experimentos numéricos 119
4.4.1. El control del error local 121
4.4.2. El comportamiento de la función $D_n(\tau)$ 123
4.4.3. Construcción de métodos con $D_n(\tau)$ optimizado . . . 128
4.4.4. Comparación de diferentes métodos 129
4.4.5. Estimaciones numéricas de $L(y)$ y del error local . . . 138

5. Conclusiones y plan de trabajo futuro 147
5.1. Conclusiones 147
5.2. Mejoras en el proceso de la integración 148

5.2.1. Proceso básico de resolución de una EDO mediante métodos de RK 149
5.2.2. Control del error global 152
5.2.3. Tolerancia variable al error local 153
5.2.4. Control de la variación de la escala temporal del problema 154
5.2.5. Elección del método a utilizar en la integración . . . 157
5.3. Plan de trabajo futuro 157

Índice de figuras

2.1. Abajo a la izquierda, región de estabilidad del método para la estimación del error global basado en BSRK5. Abajo a la derecha, región de estabilidad del método BSRK5. Arriba, se puede ver que la región del método basado en BSRK5 abarca parte del eje imaginario. 60
2.2. Resultados del problema 'expsin' para $h = 2\pi/7$. En la gráfica de arriba comparamos el error global cometido (línea a tramos) con el error global estimado (línea continua), mientras que en la de abajo comparamos el error cometido por nuestro esquema (línea a tramos) con el error cometido por el esquema BSRK5 (línea continua). 62
2.3. Resultados del problema 'Arenstorf' con $h = T/14000$. Arriba, error global (línea a tramos) frente al error global estimado (línea continua). Abajo, comparación del error global del nuevo esquema (línea a tramos) frente al error global del método BSRK5 (línea continua). 63
2.4. Resultados del problema 'Arenstorf' con $h = T/3500$. La mayor tolerancia hace que el error global aumente, y en consecuencia la solución numérica se degenera. Arriba, error global (línea a tramos) frente al error global estimado (línea continua). Abajo, error global del nuevo esquema (línea a tramos) frente al error del método BSRK5 (línea continua). 65
2.5. Abajo, región de estabilidad del método para la estimación del error global basado en DOPRI5. Arriba, se puede ver que la región abarca parte del eje imaginario. 69
2.6. Problema *Arenstorf*. Arriba una tolerancia de 10^{-9} que requiere 1268 pasos. Abajo, 10^{-6} que requiere 309 pasos 72
2.7. Problema *Pleiades*. Arriba con tolerancia de 10^{-9} con 1603 pasos. Abajo tolerancia de 10^{-4} con 182 pasos. 73

2.8. Problema *expsin*. Arriba con tolerancia de 10^{-9} con 7467 pasos. Abajo tolerancia de 10^{-4} con 416 pasos. 75

3.1. A la izquierda, los valores $C(t, s)$ para una matriz 3×3 con $t = 5$. Los valores máximos de $C(t, s)$ se dan para el valor $s = \frac{t}{2}$. A la derecha, mostramos los valores $C(t, s)$ para una matriz 5×5 con $t = 10$. Las máximos valores de $C(t, s)$ vuelven a darse en la zona central del intervalo $(0, t)$. 82

3.2. Distribución del valor $\log_{10} C^*$ para 900 matrices elegidas al azar: 300 de dimensión 3x3, 300 de dimensión 4x4 y 300 de dimensión 5x5 83

3.3. Comparación de costos para el problema *Arenstorf*: nueva estrategia de ajuste de longitud de paso y la estrategia usual 87

3.4. Comparación de costos para el problema *Pleiades*: nueva estrategia de ajuste de longitud de paso y la estrategia usual . 89

3.5. Comparación de costos para el problema *Lorenz*: nueva estrategia de ajuste de longitud de paso y la estrategia usual . 90

3.6. Comparación de las longitudes de paso para el problema *Lorenz* 91

3.7. Comparación de costos para el problema de *Kepler* con excentricidad 0.5: nueva estrategia de ajuste de longitud de paso y la estrategia usual 93

4.1. Curvas de las cotas de error para el tablero de Butcher 4.39 obtenidas para el Teorema 1 con índices de truncamiento $n = 5, 6, 7$. Donde $n = 6$ (curva continua) es el índice sugerido por la observación 8. 111

4.2. Los diferentes puntos de las soluciones orbitales para los que se han calculado soluciones numéricas junto con su correspondiente error local. A la izquierda los puntos correspondientes al problema de *Kepler*, y a la derecha los del problema *Arenstorf*. 120

4.3. Los errores locales obtenidos en diferentes puntos de la solución del problema de *Kepler* con el método Dopri8 (izquierda) y el problema *Arenstorf* con el método $m4$ (derecha). Para cada punto y de la órbita se ha dado un paso con distintas longitudes de paso, y cada curva muestra cómo cambia $\log_{10} \|\delta(y, h)\|$ en función de h en cada uno de los puntos. . . 122

4.4. Los errores locales obtenidos en diferentes puntos de la solución del problema de *Kepler* (izquierda) resuelto con el método Dopri8 y el problema *Arenstorf* (derecha) resuelto con el problema *m4*. Para cada punto y de la órbita se ha dado un paso con distintos valores $\tau = h\tilde{L}(y)$ y se muestra el logaritmo decimal de la norma del error local $\delta(y, \frac{\tau}{\tilde{L}(y)})$ para cada τ. 123

4.5. La función $D_n(\tau)$ y la norma de los errores locales $\|\delta(y,h)\|$ correspondientes a cada valor $\tau = h\tilde{L}(y)$ obtenidos para los 21 puntos en la órbita de la solución del problema de *Kepler*. A la izquierda los resultados obtenidos con el método de orden 3 de la aplicación *Mathematica*, y a la derecha los correspondientes al método de orden 6. 124

4.6. La función $D_n(\tau)$ y la norma de los errores locales $\|\delta(y,h)\|$, obtenidos con el problema *Arenstorf*, en función de $\tau = h\tilde{L}(y)$. A la izquierda el método de orden 4, a la derecha el método de orden 9. 125

4.7. La función $D_n(hL(y))$ y la norma de los errores locales $\|\delta(y,h)\|$ para distintos métodos: a la izquierda, el de orden 2, el de 4 y el de 7 aplicados al problema de *Kepler*. A la derecha, el método de orden 4, el de 5 y el de 8 aplicados al problema de los tres cuerpos 126

4.8. La función $D_n(\tau)$ y los errores locales $\delta(y,h)$ correspondientes a cada valor $h = \frac{\tau}{\tilde{L}(y)}$ obtenidos tanto para los 21 puntos en la órbita de la solución del problema de los dos cuerpos (curvas oscuras ocultas bajo las curvas claras) como para los 80 puntos de la órbita de la solución del problema *Arenstorf* (curvas claras). 127

4.9. Curvas de las cotas teóricas del error local $D_n(\tau)$ del método de orden 7 (curva a trazos largos), método de orden 8 (curva a trazos cortos), Dopri 8 (curva continua) y el método de orden 9 (curva con trazos largos y cortos). 132

4.10. Comparación de las funciones de las cotas del error local $D_n(\tau)$ del método de orden 3, el de orden 4, el método *clásico de Runge-Kutta*, m4, Dopri5 y Dopri8 134

4.11. Cotas teóricas del error local de los métodos de orden 5 (curva con tramos cortos y largos) y de orden 6 (curva con tramos largos). 134

4.12. Comparación de la cota teórica del error local de Dopri8 (curva negra), m4 (curva gris) y el método de orden 5 (curva a tramos). 135
4.13. La gráfica de la cota teórica del error local, $D_n(\tau)$, y los resultados numéricos del error local $\delta(y,h)$ (para $h = \frac{\tau}{\tilde{L}(y)}$) obtenidos en la resolución del problema *Arenstorf* con los métodos Dopri8, curvas oscuras, y m4, curvas más claras. . . 136
4.14. La gráfica muestra los valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ en escala logarítmica para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2 . . . 142
4.15. La gráfica muestra los valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2 junto con los valores $\hat{L}(y)$ estimados numéricamente para $m = 2$. Ambos valores se muestran en escala logarítmica. 143
4.16. Valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ en escala logarítmica para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2 junto con los valores $\log_{10} \hat{L}(y)$ estimados numéricamente para $m = 3$. 144
4.17. Para el caso de $m = 4$ las estimaciones $\hat{L}(y)$ de $L(y)$ en los puntos mostrados en la Figura 4.2 siguen la forma de la curva mostrada por las cotas teóricas $\tilde{L}(y)$ de $L(y)$. 145
4.18. Cota teórica $\tilde{L}(y)$ de $L(y)$ junto con las estimaciones numéricas $\hat{L}(y)$ de $L(y)$ obtenidas para $m = 3$ y para $m = 4$ en cada uno de los puntos mostrados en la Figura 4.2. 146

5.1. Algoritmo general del proceso de resolución de problemas de valor inicial 151
5.2. Proceso de adecuación de la longitud de paso: la longitud de paso que en el anterior paso hubiera mantenido el error local dentro de unos márgenes muy estrechos es la que se utilizará para avanzar en la integración 156

Índice de Tablas

1.1. Condiciones para los árboles de orden menor que 5. 20
1.2. Condiciones para los árboles de orden 5. 21

2.1. Condiciones de los árboles con hojas blancas y negras de menos de 5 vértices, y el agrupamiento de árboles para que las *condiciones de independencia* tengan los términos indicados. Para que la estimación de $\psi_h(y, y+e) - \phi_h(y)$ tenga la forma que aparece en la columna de la derecha habrán de cumplirse todas las condiciones de los árboles que estén más arriba. . . 41
2.2. Descomposición estándar de los árboles de menos de 5 vértices 50
2.3. El resto de los parámetros del método basado en DOPRI5 . 70

Capítulo 1

Introducción

En este trabajo se estudia el error cometido por los métodos explícitos de Runge-Kutta en la resolución de problemas de valor inicial de sistemas de ecuaciones diferenciales ordinarias. El trabajo se basa en el análisis de las condiciones de orden de los métodos así como en el estudio de las cotas de los errores, tanto los errores locales (es decir los errores cometidos en un paso de la integración) como el error global del proceso de integración numérica.

La integración de los sistemas de Ecuaciones Diferenciales Ordinarias (EDOs) mediante métodos de un paso (en particular, los métodos de Runge-Kutta explícitos) consiste en obtener aproximaciones a la solución para un conjunto discreto de valores de la variable independiente. Basándose en la aproximación a la solución en un valor de la variable independiente se obtiene la solución numérica para el siguiente punto, y así sucesivamente.

La discretización de la variable independiente se suele controlar dinámicamente; en la medida que avanzamos en la integración del problema, y dependiendo de los errores locales estimados numéricamente para cada paso, se adecúa la longitud del paso, de forma que no se permite un error por encima de una cierta tolerancia predeterminada para el problema. El error local del método disminuye al reducir la longitud del paso, por lo que si el error estimado sobrepasa la tolerancia habrá que disminuir la longitud de paso. Evidentemente, si se dan pasos muy pequeños el coste computacional del proceso se incrementa, ya que dicho coste depende del número de pasos que se dan en la integración numérica. El error local de cada paso se suele estimar utilizando una segunda solución numérica del problema, que partiendo de la misma solución para un punto dado, mediante un segundo método de Runge-Kutta obtiene una segunda aproximación a la solución

del siguiente punto. La comparación de las dos soluciones nos puede dar una idea de cómo es el error cometido por el método de menor orden.

Aunque el error estimado de cada paso esté por debajo de la tolerancia, esta forma de controlar la integración no nos asegura que la solución final obtenida para el problema sea aceptable, puesto que no tiene en cuenta la propagación de los errores intermedios, y puede darse el caso de que dicha propagación sea tal que las soluciones numéricas proporcionadas por el método sean inaceptables. Si quisiéramos evitar este problema deberíamos controlar el error global del problema. El mayor inconveniente del control del error global es su alto coste computacional, ya que en general, equivale a la obtención de otra solución, de modo que la evolución de la diferencia entre las dos soluciones nos puede dar una idea de la propagación del error global.

El cálculo de la estimación del error local se suele realizar utilizando pares embebidos de métodos de Runge-Kutta construidos de tal forma que la segunda solución no exige casi ningún costo adicional. No podemos decir lo mismo con respecto a la obtención de la aproximación del error global. Su estimación exige nuevas evaluaciones del sistema de ecuaciones lo que sube el coste computacional.

En el segundo capítulo de este trabajo presentamos unas nuevas formas de obtener esa segunda integración del problema que nos permita la estimación del error global de modo que el coste computacional no se incremente demasiado. En el siguiente capítulo estudiamos nuevas estrategias de elección de la longitud de paso que hacen uso de la información del error global disponible, y que permiten resolver el problema de forma más eficiente.

En el cuarto capítulo presentamos un estudio de los errores locales de los métodos de Runge-Kutta y obtenemos unas cotas de los mismos. El error local de los métodos lo acotamos mediante una función que puede ser utilizada para la comparación de los distintos métodos. Por una parte, nos ofrecen criterios para la optimización de los métodos de Runge-Kutta, y por otra, esos mismos criterios nos aportan conocimientos sobre el comportamiento del error local, lo que se puede traducir en la optimización del proceso de discretización de la variable independiente, es decir nos permiten adecuar los pasos de forma que el error local que vayamos a obtener en cada paso sea más parecido a la predicción hecha.

1.1. Ecuaciones Diferenciales Ordinarias

Un sistema de ecuaciones diferenciales de primer orden es un sistema de la siguiente forma

$$\frac{dy}{dt} = f(t, y), \qquad f : \mathcal{V} \subset \mathbb{R}^{D+1} \to \mathbb{R}^D. \tag{1.1}$$

donde t es la variable independiente, y $\mathcal{V}$ es un abierto de $\mathbb{R}^{D+1}$.

Un ejemplo de ecuación diferencial que usaremos más adelante (que tomamos de [23]) es

$$y' = cos(t)y. \tag{1.2}$$

Si el valor de la función f es independiente de la variable t se dice que es un sistema *autónomo* y en ese caso tendremos un sistema de la forma

$$\frac{dy}{dt} = f(y), \tag{1.3}$$

donde $f : \mathcal{U} \subset \mathbb{R}^D \to \mathbb{R}^D$.

Un ejemplo de sistema de ecuaciones diferenciales autónomo que utilizaremos como problema test en capítulos posteriores, es el del problema restringido de los tres cuerpos, que hemos tomado de [20, pp.129–130]. Dicho problema se presenta como un sistema de segundo orden, pero introduciendo las primeras derivadas como variables del problema se convierte en el siguiente sistema de ecuaciones diferenciales de primer orden:

$$\begin{aligned} q_1' &= v_1, \\ q_2' &= v_2, \\ v_1' &= q_1 + 2v_2 - \mu' \frac{q_1 + \mu}{D_1} - \mu \frac{q_1 - \mu'}{D_2}, \\ v_2' &= q_2 - 2v_1 - \mu' \frac{q_2}{D_1} - \mu \frac{q_2}{D_2}, \end{aligned} \tag{1.4}$$

donde, $\mu = 0{,}012277471$, $\mu' = 1 - \mu$, $D_1 = ((q_1 + \mu)^2 + q_2^2)^{\frac{3}{2}}$ y $D_2 = ((q_1 - \mu')^2 + q_2^2)^{\frac{3}{2}}$.

En el caso de sistemas no autónomos, siempre podemos utilizar el equivalente autónomo

$$Y' = F(Y), \quad \text{donde} \quad Y = \begin{pmatrix} t \\ y \end{pmatrix}, \quad F(Y) = \begin{pmatrix} 1 \\ f(t, y) \end{pmatrix}. \tag{1.5}$$

Por ejemplo, el problema (1.2) puede ser expresado de forma autónoma con el siguiente sistema de ecuaciones:

$$\begin{aligned} y_1' &= 1, \\ y_2' &= cos(y_1)y_2. \end{aligned} \tag{1.6}$$

1.2. Resolución numérica

Se pretende integrar problemas de valor inicial de la forma

$$\frac{dy}{dt} = f(t, y), \quad y(t_0) = y_0 \tag{1.7}$$

sobre el intervalo $[t_0, t_f]$, es decir, buscamos la solución $y(t)$ en $t \in [t_0, t_f]$.

Cuando integramos numéricamente un sistema de ecuaciones diferenciales ordinarias mediante un método de un paso, se realiza una discretización $t_0 < t_1 < \ldots < t_N = t_f$ de la variable independiente t, con $h_n = t_n - t_{n-1}$ y obtenemos valores

$$y_n = \psi_{h_n}(t_{n-1}, y_{n-1}),$$

para $n = 1, \ldots, N$, que aproximan a los valores $y(t_n)$ $(n = 1, \ldots, N)$. Aquí, ψ_h es una familia uniparamétrica de transformaciones de la forma

$$\psi_h : \mathbb{R}^{D+1} \to \mathbb{R}^D \tag{1.8}$$

tal que para cualquier solución $y(t)$ del sistema (1.1)

$$y(t + h) \approx \psi_h(t, y(t)).$$

El ejemplo más sencillo de métodos de un paso es el método de Euler explícito, donde la transformación ψ_h viene dada por $\psi_h(t, y) = y + hf(t, y)$.

1.3. Métodos de Runge-Kutta

Para la familia de métodos de Runge-Kutta (RK) la transformación ψ_h se define como

$$\psi_h(t, y) = y + h \sum_{i=1}^{s} b_i f(t + c_i h, Y_i), \tag{1.9}$$

donde para $i = 1, \ldots, s$,

$$Y_i = y + h \sum_{j=1}^{s} a_{i,j} f(t + c_j h, Y_j). \tag{1.10}$$

Los coeficientes de los métodos de Runge-Kutta de s etapas se representan generalmente mediante el tablero de Butcher

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \ldots & a_{1s} \\ c_2 & a_{21} & a_{22} & \ldots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & & a_{ss} \\ \hline & b_1 & b_2 & \ldots & b_s \end{array} \tag{1.11}$$

El método de Runge-Kutta es explícito cuando se cumple que $a_{ij} = 0$ si $j \geq i$. Si queremos que cada etapa Y_i de (1.10) sea una aproximación de orden uno de $y(t + c_i h)$ (donde $y(t)$ es la solución del sistema (1.1) que pretendemos aproximar) se debe cumplir la condición

$$c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, 2, \ldots, s. \tag{1.12}$$

A partir de ahora, consideraremos métodos de Runge-Kutta que satisfacen la condición (1.12). Dicha condición implica además que la solución numérica que resulta de aplicar el método de Runge-Kutta al sistema autónomo equivalente (1.5) coincide con la solución numérica obtenida al aplicar dicho método al sistema no autónomo original (1.1). En lo que sigue, supondremos sin pérdida de generalidad que el sistema a resolver es autónomo.

1.4. Flujo de un sistema autónomo

Llamamos h-flujo del sistema autónomo (1.3) a la transformación paramétrica del espacio de fases

$$\phi_h : \mathbb{R}^D \to \mathbb{R}^D \qquad \text{tal que} \qquad \phi_h(y(t)) = y(t + h) \tag{1.13}$$

para cualquier solución $y(t)$ del sistema (1.3). En lo que sigue, vamos a suponer que $f(y)$ es infinitas veces diferenciable, en cuyo caso, el h-flujo (1.13) es (si existe) infinitas veces diferenciable tanto respecto a h como respecto a y.

La existencia del h-flujo de (1.3) tal como es definido en (1.13) no está garantizada ni siquiera en el caso de que $f : \mathbb{R}^D \to \mathbb{R}^D$ sea una transformación polinómica. Por ejemplo, consideremos la ecuación diferencial autónoma de dimensión $D = 1$ dada por

$$\frac{dy}{dt} = y^2. \tag{1.14}$$

En tal caso, $y(t+h) = \frac{y(t)}{1-hy(t)}$ para cualquier solución de dicha ecuación diferencial pero, obviamente, el dominio de definición de la transformación ϕ_h dada por $\phi_h(y) = \frac{y}{1-hy}$ es $\mathbb{R} - \{\frac{1}{h}\}$, y por tanto, ϕ_h no está definido en todo el espacio de fases $\mathbb{R}$, como se considera en la definición de h-flujo (1.13). En cambio, se puede definir un flujo local del sistema (1.14) como una transformación

$$\phi : \mathcal{V} = \{(h, y) \in \mathbb{R}^2 \,|\, hy < 1\} \to \mathbb{R},$$

donde para cada $(h, y) \in \mathcal{V}$, $\phi(h, y) = \phi_h(y) = \frac{y}{1-hy}$.

A menudo se tiene que la aplicación f de (1.3) no está definida para todo $\mathbb{R}^D$, como es el caso de (1.4), donde f está definida en el abierto $\mathcal{U} = \{(q_1, q_2, v_1, v_2) \in \mathbb{R}^4 \,/\, (q_1+\mu)^2 + q_2^2 \neq 0 \wedge (q_1+\mu')^2 + q_2^2 \neq 0\}$.

En general, vamos a tener que el sistema autónomo (1.3) es tal que $f : \mathcal{U} \subset \mathbb{R}^D \to \mathbb{R}^D$ donde $\mathcal{U}$ es un abierto de $\mathbb{R}^D$. En tal caso, el flujo local es una aplicación $\phi : \mathcal{V} \subset \mathbb{R}^{D+1} \to \mathbb{R}^D$ donde $\mathcal{V}$ es un abierto de $\mathbb{R}^{D+1}$ que contiene $\{0\} \times \mathcal{U}$ y es tal que para cualquier solución $y(t)$, $t \in (\alpha, \beta)$, se cumple que $\phi(h, y(t)) = y(t+h)$ siempre que $\alpha < t < t+h < \beta$.

En lo que sigue, vamos a suponer para simplificar la exposición que el sistema autónomo (1.3) es tal que $f : \mathbb{R}^D \to \mathbb{R}^D$ es infinitamente diferenciable y que el h-flujo $\phi_h : \mathbb{R}^D \to \mathbb{R}^D$ existe para cada $h \in \mathbb{R}$.

1.5. Error local y error global

Los métodos de integración de un paso, y en particular, los métodos de RK explícitos, están caracterizados por una familia de transformaciones $\psi_h : \mathbb{R}^D \to \mathbb{R}^D$ dependientes de un parámetro real h, tal que para cualquier solución $y(t)$ de (1.3), se tiene que $\psi_h(y(t)) \approx y(t+h)$ (es decir, $\psi_h \approx \phi_h$) para h suficientemente pequeño. Así, para integrar numéricamente el sistema (1.3) con valor inicial $y(t_0) = y_0$ y una discretización $t_0 < t_1 < \ldots < t_N$ de la variable independiente t, se obtiene

$$y_n = \psi_{h_n}(y_{n-1})$$

para $n = 1, 2, \ldots, N$, donde $h_n = t_n - t_{n-1}$.

Al resolver numéricamente un sistema de ecuaciones diferenciales ordinarias mediante un método de Runge-Kutta cometemos un error en cada paso, conocido como error local y que se define como

$$\delta(y,h) = \psi_h(y) - \phi_h(y). \tag{1.15}$$

Es decir, el error local es el error cometido en un único paso en el caso de que el punto de partida es el mismo para ambas transformaciones. Se dice que el método ψ_h es de orden p si $\delta(y,h) = O(h^{p+1})$ cuando $h \to 0$. No obstante, lo que finalmente interesa es la diferencia entre la solución numérica y la solución exacta del problema en cualquier punto del intervalo de integración, es decir,

$$e_n = y_n - y(t_n), \tag{1.16}$$

que se conoce como error global.

El error global satisface

$$\begin{aligned} e_n &= \psi_{h_n}(y_{n-1}) - \phi_{h_n}(y(t_{n-1})) \\ &= \delta(y_{n-1},h_n) + \phi_{h_n}(y_{n-1}) - \phi_{h_n}(y(t_{n-1})) \\ &= (\phi_{h_n}(y(t_{n-1}) + e_{n-1}) - \phi_{h_n}(y(t_{n-1}))) + \delta(y_{n-1},h_n), \end{aligned} \tag{1.17}$$

lo cual nos muestra que el error global puede ser considerado como la suma de dos errores distintos,

- Por una parte el error local debido al último paso, $\delta(y_{n-1},h_n)$.
- Y por otra, la propagación del error global del paso anterior, o dicho de otra forma, la propagación y la acumulación de los errores locales cometidos en los pasos anteriores.

1.6. Condiciones de orden de los Métodos de RK

Para obtener las condiciones de orden de un método de Runge-Kutta se debe comparar el desarrollo en serie de potencias de h de $\psi_h(y)$ con el desarrollo en serie de potencias de h del h-flujo $\phi_h(y)$.

El desarrollo en serie de potencias de h de $\phi_h(y)$ se puede obtener por medio del desarrollo en serie de Taylor de $y(t+h)$ en torno a $h=0$,

$$\phi_h(y(t)) = y(t+h) = y(t) + \sum_{j\geq 1} \frac{h^j}{j!} y^{(j)}(t), \tag{1.18}$$

En este desarrollo de Taylor nos interesa saber cómo son las diferentes derivadas $y^{(j)}(t)$. Si escribimos el sistema (1.3) teniendo en cuenta cada componente tenemos que (omitimos el argumento t en y' y en y)

$$\begin{pmatrix} (y^1)' \\ \dots \\ (y^D)' \end{pmatrix} = \begin{pmatrix} f^1(y) \\ \dots \\ f^D(y) \end{pmatrix}$$

donde $f^i : \mathbb{R}^D \to \mathbb{R}$ para $i = 1, \dots, D$.

A la hora de obtener $(y^i)''$ con $i = 1, \dots D$, hay que tener en cuenta que $(y^i)'$ depende de todas las componentes de y, por lo que, al derivar respecto de t ambos lados de la igualdad $y' = f(y)$, nos surgen las derivadas parciales

$$(y^i)'' = \sum_{j=1}^{D} \frac{\partial f^i(y)}{\partial y^j}(y^j)', \tag{1.19}$$

o mostrado en forma matricial

$$\begin{pmatrix} (y^1)'' \\ \dots \\ (y^D)'' \end{pmatrix} = \begin{pmatrix} \frac{\partial f^1(y)}{\partial y^1}) & \dots & \frac{\partial f^1(y)}{\partial y^D} \\ \vdots & & \vdots \\ \frac{\partial f^D(y)}{\partial y^1} & \dots & \frac{\partial f^D(y)}{\partial y^D} \end{pmatrix} \begin{pmatrix} f^1(y) \\ \dots \\ f^2(y) \end{pmatrix}. \tag{1.20}$$

Es decir, nos aparece la Matriz Jacobiana. Podemos simplificar la notación escribiendo

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

donde $f'(y)$ es la Matriz Jacobiana de $f(y)$.

Si tratamos de obtener las expresiones de $(y^i)'''$ para $i = 1, \dots D$, veremos que cada sumando de (1.19) vuelve a depender de todas las componentes de y, y por tanto, al derivar respecto de t, y aplicar la regla de la cadena, surgen nuevos sumatorios de derivadas parciales:

$$(y^i)''' = \sum_{j=1}^{D} \left(\left(\sum_{k=1}^{D} \frac{\partial^2 f^i(y)}{\partial y^j \partial y^k} f^j(y) f^k(y) \right) + \left(\sum_{k=1}^{D} \frac{\partial f^i(y)}{\partial y^j} \frac{\partial f^j(y)}{\partial y^k} f^k(y) \right) \right). \tag{1.21}$$

Para poder trabajar con las expresiones que nos van surgiendo podemos hacer uso de las derivadas de Frêchet (véase por ejemplo [10]), cuya definición es:

Definición 1 *Sean $y \in \mathbb{R}^D$ y $f : \mathbb{R}^D \to \mathbb{R}^D$. La Msima derivada de Frêchet de f en y, denotado por $f^{(M)}(y)$, es un operador $\underbrace{\mathbb{R}^D \times \mathbb{R}^D \times \ldots \times \mathbb{R}^D}_{M veces} \to \mathbb{R}^D$, lineal en cada operando, cuyo valor aplicado a los operandos $K_1, K_2, \ldots, K_M \in \mathbb{R}^D$ es*

$$f^{(M)}(y)(K_1, K_2, \ldots, K_M) = \sum_{i=1}^{D} \sum_{j_1=1}^{D} \sum_{j_2=1}^{D} \cdots \sum_{j_M=1}^{D} f^i_{j_1, j_2 \cdots j_M} K_1^{j_1} K_2^{j_2} \cdots K_M^{j_M} e_i \tag{1.22}$$

donde

- $K_l = (K_l^1, K_l^2, \ldots, K_l^D)^T \in \mathbb{R}^D$, $l = 1, 2, \ldots, M$,
- *para* $i, j_1, j_2, \ldots j_M \in \{1 \ldots D\}$

$$f^i_{j_1, j_2 \cdots j_M} = \frac{\partial^M}{\partial y^{j_1} \partial y^{j_2} \ldots \partial y^{j_M}} f^i(y),$$

- $e_i = (\underbrace{0, 0, \ldots 0}_{i-1}, 1, \underbrace{0, \ldots, 0}_{D-i})^T \in \mathbb{R}^D$ *(vector compuesto de ceros y un uno en la $i$ésima posición).*

Se trata en definitiva de un vector de dimensión D, ya que el primer sumatorio de (1.22) con índice i junto con los vectores e_i hacen que la expresión tenga D componentes. Por otra parte, el resto de los sumatorios hacen que en la expresión aparezcan todas las posibles derivadas parciales de orden M de las funciones $f^i(y)$ (donde las derivadas parciales se realizan respecto a las componentes del vector y).

Según la Definición 1, resulta que las derivadas $y^{(j)}$ de (1.18) se pueden expresar como combinaciones lineales de derivadas de Frêchet cuyos operandos son $f(y)$ o el resultado de derivadas de Frêchet de f. En concreto, tenemos

$$\begin{aligned}
y' &= f(y), \\
y'' &= f^{(1)}(y)(f(y)) \\
y''' &= f^{(2)}(y)(f(y), f(y)) + f^{(1)}(y)(f^{(1)}(y)(f(y))) \\
y'''' &= f^{(3)}(y)(f(y), f(y), f(y)) + f^{(1)}(y)(f^{(1)}(y)(f^{(1)}(y)(f(y)))) + \\
&\quad + f^{(1)}(y)(f^{(2)}(y)(f(y), f(y))) + 3f^{(2)}(y)(f^{(1)}(y)(f(y)), f(y)) \\
&\ldots
\end{aligned}$$

La notación que estamos utilizando se puede abreviar bastante, ya que en las derivadas de Frêchet siempre tenemos que las derivadas parciales se evalúan en y, por lo que podemos prescindir de y. De esa forma podemos simplificar las expresiones escribiendo

$$\begin{aligned} y' &= f, \\ y'' &= f^{(1)}(f), \\ y''' &= f^{(2)}(f,f) + f^{(1)}(f^{(1)}(f)), \\ y'''' &= f^{(3)}(f,f,f) + f^{(1)}(f^{(1)}(f^{(1)}(f))) + f^{(1)}(f^{(2)}(f,f)) + 3f^{(2)}(f^{(1)}(f),f), \\ &\dots \end{aligned} \tag{1.23}$$

Lo que se puede observar es que y' es la derivada de Frêchet de f de orden 0, y'' es la derivada de Frêchet de f de orden 1 operando sobre $f(y)$, y''' es combinación lineal del resultado de aplicar derivadas de Frêchet de f de orden 1 y 2, y en definitiva, se puede ver que $y^{(p)}$ es una combinación lineal de resultados de aplicar derivadas de Frêchet de f de orden hasta $p-1$ (con operandos que a su vez son resultados de aplicar derivadas de Frêchet de f de orden menor que $p-1$). A las componentes de dicha combinación lineal se les denomina *diferenciales elementales* y podemos definirlas como sigue:

Definición 2 *Una diferencial elemental de orden j ($j \geq 1$) asociada al sistema (1.3) es una transformación $g : \mathbb{R}^D \to \mathbb{R}^D$ definida como*

$$g(y) = f^{(M)}(y)(g_1(y), g_2(y), \dots, g_M(y))$$

donde $1 \leq M \leq j$, cada g_i es una diferencial elemental de orden $j_i < j$ para $i = \{1, \dots, M\}$, y $j = 1 + (j_1 + j_2 + \cdots + j_M)$. La única diferencial elemental de orden 1 es $g = f$.

Según esta definición, existe una única diferencial elemental de orden 2, $f^{(1)}(y)(f(y))$, y podemos generar las diferenciales elementales de mayor orden en base a los de menor orden; en concreto, podemos ver que hay dos diferenciales elementales de orden 3:

- $f^{(2)}(y)(f(y), f(y)) = f^{(2)}(f,f) \to j = 1 + (1+1)$
- $f^{(1)}(y)(f^{(1)}(y)(f(y))) = f^{(1)}(f^{(1)}(f)) \to j = 1 + (2)$

y cuatro de orden 4:

- $f^{(3)}(f,f,f) \to j = 1 + (1+1+1)$
- $f^{(2)}(f^{(1)}(f),f) \to j = 1 + (2+1)$
- $f^{(1)}(f^{(2)}(f,f)) \to j = 1 + (3)$
- $f^{(1)}(f^{(1)}(f^{(1)}(f))) \to j = 1 + (3)$

de la misma forma obtendríamos las 9 diferenciales elementales de orden 5, las 20 de orden 6, etc.

Se puede observar que dichas diferenciales elementales son precisamente las expresiones que nos han aparecido al calcular las derivadas $y^{(j)}(t)$, es decir, $y^{(j)}(t)$ es una combinación lineal de las diferenciales elementales de orden j.

Una vez que sabemos cómo son los $y^{(j)}(t)$ del desarrollo de Taylor de $y(t+h)$ dada en (1.18), nos interesa obtener el desarrollo en serie de potencias de h de la aplicación de un paso $\psi_h(y)$ del método de Runge-Kutta (1.9)-(1.10). Puede verse que

$$\begin{aligned}\psi_h(y) &= y + hq_{11}f + h^2q_{21}f'f + h^3(q_{31}f''(f,f) + q_{32}f'f'f) + \\ &\quad h^4(q_{41}f'''(f,f,f) + q_{42}f'f'f'f + q_{43}f'f''(f,f) + q_{44}f''(f'f,f)) + \\ &\quad O(h^5), \end{aligned} \tag{1.24}$$

donde los valores q_{ij} dependen exclusivamente de los coeficiente a_{ij}, b_i, c_i del método de Runge-Kutta. Obsérvese que de nuevo aparecen combinaciones lineales de diferenciales elementales.

Para el caso de un método de Runge-Kutta explícito de un paso se puede ver que

$$\begin{aligned} q_{11} &= \sum_{i=1}^{s} b_i, \\ q_{21} &= \sum_{i=1}^{s} b_i \sum_{j=1}^{s} a_{ij}, \\ q_{31} &= \frac{1}{2}\sum_{i=1}^{s} b_i (\sum_{j=1}^{s} a_{ij})(\sum_{k=1}^{s} a_{ik}), \\ q_{32} &= \sum_{i=1}^{s} b_i \sum_{j=1}^{s} a_{ij} \sum_{k=1}^{s} a_{jk}, \\ q_{41} &= \frac{1}{6}\sum_{i=1}^{s} b_i (\sum_{j=1}^{s} a_{ij})(\sum_{k=1}^{s} a_{ik})(\sum_{l=1}^{s} a_{il}), \end{aligned}$$

$$
\begin{aligned}
q_{42} &= \sum_{i=1}^{s} b_i \sum_{j=1}^{s} a_{ij} \sum_{k=1}^{s} a_{jk} \sum_{l=1}^{s} a_{kl}, \\
q_{43} &= \frac{1}{2} \sum_{i=1}^{s} b_i \sum_{j=1}^{s} a_{ij} (\sum_{k=1}^{s} a_{jk})(\sum_{l=1}^{s} a_{jl}), \\
q_{44} &= \sum_{i=1}^{s} b_i (\sum_{j=1}^{s} a_{ij} \sum_{k=1}^{s} a_{jk})(\sum_{l=1}^{s} a_{il}) \\
&\cdots
\end{aligned}
$$

Si igualamos (1.24) y (1.18), obtendremos las condiciones que debe cumplir un método para que su aplicación nos dé una aproximación a la solución del orden que queramos.

1.7. Árboles y condiciones de orden

En [14], Merson observa que a cada diferencial elemental que aparece en el desarrollo en serie de un paso $\psi_h(y)$ de un método de Runge-Kutta le corresponde de forma natural un árbol con raíz.

A partir de los trabajos de Butcher [4, 3], la obtención de las condiciones de orden se simplifica mucho con la utilización de árboles con raíz. Butcher muestra que, además de existir una única diferencial elemental de orden r por cada árbol con raíz con r vértices, la estructura del árbol nos permite la obtención recursiva de los valores q_{ij} que aparecen en (1.24). Los árboles con raíz se representan gráficamente como grafos en forma de árbol, en los que hay un único vértice principal, al que llamamos *raíz*, y que lo situamos debajo del resto de vértices. A la raíz unimos mediante aristas, tanto otros vértices, como otros árboles (subárboles) cuyas raíces dejan de llamarse raíces en el nuevo grafo, ya que se colocan una posición más arriba que la nueva raíz y unidas a ella mediante una arista. El árbol más simple es aquel que solo se compone de la raíz y que representamos gráficamente como $\bullet$. En un árbol también hablaremos de *hojas*, que son aquellos vértices sobre los que no hay más vértices unidos al mismo (en el árbol puede haber vértices en una posición superior a la hoja, pero nunca unidos a ella). Los vértices que no son hojas, si se quiere destacar tal condición, diremos que son *vértices internos*, y el árbol que solo tiene una hoja diremos que es un árbol *sin ramificaciones*. Abajo, a la izquierda, podemos ver un ejemplo de árbol en el que las hojas se han dibujado como círculos blancos mientras que los nodos o vértices internos son los vértices negros. A la derecha podemos ver

otro árbol, éste sin ramificaciones (y que, por tanto, sólo tiene una hoja):

Una forma elegante de obtener las condiciones de orden se basa en la utilización de **B-series** [20], que hacen uso de los árboles con raíz. Cada diferencial elemental se asocia a un árbol con raíz y para cada árbol tenemos una forma muy sencilla de obtener el coeficiente q_{ij} de (1.24). En la literatura podemos encontrar diferentes definiciones de B-series, sirvan como ejemplo los que aparecen en [20] y en [12], y aunque pertenezcan a los mismos autores, las definiciones difieren en la función de normalización utilizada para la serie. No obstante, ambas definiciones son equivalentes. La definición que vamos a utilizar es la que utilizó Murua en [16], adoptada más tarde por Butcher y Sanz-Serna en [5] y que posteriormente también ha sido utilizada por Hairer, Lubich y Wanner en [12].

Definición 3 *Sea* $\boldsymbol{a}(\emptyset), \boldsymbol{a}(\bullet), \boldsymbol{a}(\bullet\!\!-\!\!\bullet) \ldots \boldsymbol{a}(t) \ldots$ *una secuencia de coeficientes reales definidos para todo árbol* $t \in \mathcal{T}$*. Entonces se llama* B-serie *a la serie*

$$\begin{aligned} B(\boldsymbol{a})(y) &= \boldsymbol{a}(\emptyset)y + h\boldsymbol{a}(\bullet)f(y) + \frac{h^2}{2}\boldsymbol{a}(\bullet\!\!-\!\!\bullet)F(\bullet\!\!-\!\!\bullet)(y) + \ldots \\ &= \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\boldsymbol{a}(t)F(t)(y) \end{aligned} \tag{1.25}$$

Donde para cada $t \in \mathcal{T}$*,* $\rho(t)$ *indica el orden del árbol* t *(el número de vértices de* t*),* $\sigma(t)$ *indica el número de distintas simetrías que admite el árbol* t *y* $F(t)$ *es una determinada transformación del espacio de fases* $\mathbb{R}^D$*, la diferencial elemental asociada al sistema (1.3) correspondiente al árbol con raíz* t*.*

Las definiciones precisas de $F(t)$, $\rho(t)$ y de $\sigma(t)$ las daremos en la Sección 1.8.

Tanto la expansión en serie de potencias de h de la solución numérica de un método de Runge-Kutta como la expansión de la solución exacta se pueden representar como B-series. De hecho, cada Y_i en (1.10) admite una expansión en B-serie. Para la solución exacta, el coeficiente real de cada uno de los árboles es $\frac{1}{\gamma(t)}$, donde $\gamma(t)$ es la *densidad* del árbol t (que definiremos en (1.40)), mientras que para la solución numérica del método de Runge-Kutta podemos obtener esos coeficientes combinando la B-serie correspondiente a Y_i con la que corresponde a $\psi_h(y)$. En [20] podemos encontrar la forma

que toma la composición de dos B-series, que es a su vez una nueva B-serie. El resultado de la composición nos revela los coeficientes de cada árbol, que al igualarlos a $\frac{1}{\gamma(t)}$, el coeficiente de la solución exacta, nos dan las condiciones que debe cumplir el método de Runge-Kutta para que la diferencial elemental tenga el mismo coeficiente tanto en la solución exacta como en la aproximación numérica.

1.8. Series formales y condiciones de orden

En esta sección vamos a obtener el desarrollo en B-serie de la solución exacta y de la solución numérica haciendo uso de unas herramientas presentadas en [17], herramientas que utilizaremos en la sección 2.5 en el proceso de obtención de las condiciones de orden de una clase de métodos numéricos más complejos que los métodos de Runge-Kutta puros.

Ese proceso se basa en que debemos ordenar las expansiones de $\psi_{h,f}$ y de $\phi_{h,f}$ en potencias de h de forma que compartan una estructura común,

$$\psi_{h,f} = \mathrm{id} + \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\psi(t)F(t), \qquad \phi_{h,f} = \mathrm{id} + \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\phi(t)F(t) \quad (1.26)$$

donde $\mathcal{T}$ es un conjunto contable de índices a determinar (más adelante veremos que en el caso de los métodos de Runge-Kutta se puede tomar $\mathcal{T}$ como el conjunto de árboles con raíz), y para cada elemento $t \in \mathcal{T}$, el orden $\rho(t)$ es un número positivo, $\sigma(t)$ es un factor de normalización que se elegirá de forma conveniente, y donde las *diferenciales elementales* $F(t) : \mathbb{R}^D \to \mathbb{R}^D$ son transformaciones en el espacio de fases que dependen del sistema (1.1) que se pretende integrar, mientras que los *coeficientes* reales $\psi(t)$ y $\phi(t)$ no dependen del sistema.

Evidentemente, para que el método $\psi_{h,f}$ sea de orden p la condición suficiente es que para cada $t \in \mathcal{T}$ con $\rho(t) \leq p$ se cumpla $\psi(t) = \phi(t)$.

Para poder obtener esas condiciones, previamente hay que conocer la expansión (1.26), es decir, necesitamos determinar una clase apropiada de series formales

$$B(\mathbf{d}) = \mathrm{id} + \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\mathbf{d}(t)F(t) \quad (1.27)$$

donde $\mathbf{d}$ es una función $\mathcal{T} \to \mathbb{R}$, y se cumple (1.26).

En el caso de los método de Runge-Kutta conviene obtener la expansión

de $f(Y_i)$, donde suponemos que Y_i a su vez se puede representar como

$$Y_i = y + \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\psi_i(t)F(t)(y). \tag{1.28}$$

En [17] podemos encontrar la forma que toma la expansión general de una función suave k aplicada a una expansión de la forma (1.27).

Lema 1 *Sea*

$$B(d) = id + \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)}\boldsymbol{d}(t)F(t),$$

y sea k una función suave de D variables, entonces, el desarrollo de Taylor de $k \circ B(d)$ viene dada por

$$k \circ B(d) = \sum_{u\in\mathcal{F}} \frac{h^{\rho(u)}}{\sigma(u)}d'(u)X(u)k, \tag{1.29}$$

donde cada elemento $u \in \mathcal{F}$ es una tupla no ordenada de elementos de $\mathcal{T}$, que se puede interpretar formalmente como un producto conmutativo $u = t_1t_2\cdots t_m$ de elementos de $\mathcal{T}$, y para cada $u \in \mathcal{F}$, $X(u)k$ es una función suave de D variables que definimos a continuación, $d'(u) \in \mathbb{R}$, y tanto $\sigma(u)$ como $\rho(u)$ son enteros positivos. Teniendo en cuenta las repeticiones, podemos también escribir cada $u \in \mathcal{F}$ de la forma $u = t_1^{r_1}\cdots t_n^{r_n}$ con $t_i \neq t_j$ si $i \neq j$. El elemento neutro $\emptyset$ tal que $\emptyset u = u$ también pertenece a $\mathcal{F}$. Además, tenemos que:

- $X(\emptyset)k(y) = k(y)$ *y* $X(t_1\cdots t_m)k(y) = k^{(m)}(y)(F(t_1)(y),\ldots,F(t_m)(y))$,
- $d'(\emptyset) = 1$ *y* $d'(t_1\cdots t_m) = d(t_1)\cdots d(t_m)$,
- $\sigma(\emptyset) = 1$ *y* $\sigma(t_1^{r_1}\cdots t_n^{r_n}) = r_1!\cdots r_n!\sigma(t_1)^{r_1}\cdots\sigma(t_n)^{r_n}$,
- $\rho(t_1\cdots t_m) = \rho(t_1) + \cdots + \rho(t_m)$.

Si aplicamos esa expansión a $f(Y_i)$ obtenemos

$$f(Y_i) = \sum_{u\in\mathcal{F}} \frac{h^{\rho(u)}}{\sigma(u)}\psi_i'(u)X(u)f(y), \tag{1.30}$$

que sustituido a su vez en la expresión de Y_i en (1.10) nos da

$$Y_i = y + \sum_{u\in\mathcal{F}} \frac{h^{\rho(u)+1}}{\sigma(u)} \sum_{j=1}^{s} a_{ij}\psi_j'(u)X(u)f(y), \tag{1.31}$$

donde

- $X(t_1 \cdots t_m)f(y) = f^{(m)}(y)(F(t_1)(y), \ldots, F(t_m)(y))$,
- $\psi'_j(\emptyset) = 1$,
- para $m \geq 1$, y $t_1, \ldots, t_m \in \mathcal{T}$ arbitrarios $\psi'_j(u) = \psi_j(t_1) \cdots \psi_j(t_m)$.

Hasta ahora no hemos definido el conjunto $\mathcal{T}$, lo único que hemos dicho es que se trata de un conjunto contable, pero si igualamos (1.28) con (1.31) obtenemos

$$y + \sum_{t \in \mathcal{T}} \frac{h^{\rho(t)}}{\sigma(t)} \psi_i(t) F(t)(y) = y + \sum_{u \in \mathcal{F}} \frac{h^{\rho(u)+1}}{\sigma(u)} \sum_{j=1}^{s} a_{ij} \psi'_j(u) X(u) f(y) \quad (1.32)$$

donde podemos comparar los coeficientes de h^l en cada una de las expansiones. Así, si denotamos para cada $l \geq 1$, $\mathcal{T}_l = \{t \in \mathcal{T} \,/\, \rho(t) = l\}$ y $\mathcal{F}_{l-1} = \{u \in \mathcal{F} \,/\, \rho(u) = l - 1\}$ (de modo que $\mathcal{T} = \cup_{l \geq 1} \mathcal{T}_l$ y $\mathcal{F} = \cup_{l \geq 0} \mathcal{F}_l$) tendremos que

$$\sum_{t \in \mathcal{T}_l} \frac{1}{\sigma(t)} \psi_i(t) F(t)(y) = \sum_{u \in \mathcal{F}_{l-1}} \frac{1}{\sigma(u)} \sum_{j=1}^{s} a_{ij} \psi'_j(u) X(u) f(y)$$

para cada $l \geq 1$. Esto nos permite definir los conjuntos $\mathcal{T}$ y $\mathcal{F}$ de forma consistente, imponiendo que se cumplan las siguientes condiciones:

- Hay una correspondencia unívoca entre el conjunto $\mathcal{T}_l$ y $\mathcal{F}_{l-1}$. Los dos lados de la igualdad deben tener la misma estructura, y han de constar del mismo conjunto de índices. Al elemento $t \in \mathcal{T}_l$ correspondiente a un elemento $u \in \mathcal{F}_{l-1}$, lo denotaremos como $t = [u]$.
- La diferencial elemental $F(t)$ asociada a $t = [u]$ satisface

$$F(t) = X(u)f. \quad (1.33)$$

- El orden $\rho(t)$ asociado a $t = [u]$ satisface $\rho(t) = \rho(u) + 1$.
- Los pesos asociados a cada uno de ellos también deben coincidir, por lo que,

$$\psi_i(t) = \sum_{j=1}^{s} a_{ij} \psi'_j(u), \quad (1.34)$$

y por otro lado, si $t = [t_1^{r_1} \cdots t_n^{r_n}]$

$$\sigma(t) = r_1! \ldots r_n! \sigma(t_1)^{r_1} \ldots \sigma(t_n)^{r_n}. \quad (1.35)$$

Estas observaciones nos permiten definir recursivamente los conjuntos $\mathcal{T}_l$ y $\mathcal{F}_l$ (al mismo tiempo que $F(t)$ y $\sigma(t)$ para todo $t \in \mathcal{T}_l$):

- Para $l = 0$, $\mathcal{F}_0$ consta de un sólo elemento: $\mathcal{F}_0 = \{\emptyset\}$.

- Para $l = 1, 2, 3, \ldots$:
  $\mathcal{T}_l = \{[u]/u \in \mathcal{F}_{l-1}\}$,
  $\mathcal{F}_l = \{\text{m-tuplas no ordenadas } t_1 \cdots t_m, \quad t_i \in \mathcal{T}_{j_i}, \quad \sum_{i=1}^m j_i = l\}$.

El conjunto $\mathcal{T}$ admite la representación gráfica conocida como árboles con raíz, donde $t = [\emptyset]$ se representa mediante el único árbol de un sólo vértice, y el resto de elementos de $\mathcal{T}$, que son de la forma $t = [t_1 \cdots t_m]$ donde $t_1, \ldots, t_m$ son árboles con menos vértices que t, se representan recursivamente colocando una raíz unida a las raíces de las representaciones de los árboles $t_1, \ldots, t_m$. La representación típica se realiza colocando la nueva raíz debajo de los vértices que a su vez son las raíces de los árboles $t_1, \ldots, t_m$. El orden $\rho(t)$ de los elementos de $\mathcal{T}$ coincide con el número de vértices, y el único elemento de orden 1 es $[\emptyset]$ cuya representación gráfica corresponde al árbol con un único vértice $\bullet$.

Observación 1 *Las definiciones de $\sigma(t)$, $\sigma(u)$ y $F(t)$ son un tanto arbitrarias, pues basta con que*

$$\frac{F(t)}{\sigma(t)} = \frac{X(u)f}{\sigma(u)}.$$

Por ejemplo, podríamos definir $\sigma(t) = 1$ y $F(t) = \frac{1}{r_1! r_2! \cdots r_n!} X(u)f$ para cada $t = [u]$, $u = t_1^{r_1} \cdots t_n^{r_n}$ (con $t_i \neq t_j$ siempre que $i \neq j$). Adoptamos las definiciones (1.33) y (1.35) principalmente por mantener la consistencia con la notación estándar de diferencial elemental $F(t)$ asociado a un árbol con raíz t.

Podemos encontrar la recursión para $\psi_i(t)$ basándonos en (1.34), puesto que cada tupla de $\mathcal{F}$ se compone de elementos de menor orden y sabemos que $\psi'_j(u) = \psi_j(t_1)\psi_j(t_2)\ldots\psi_j(t_m)$, donde $u = t_1 t_2 \cdots t_m$. Por tanto,

$$\psi_i([\emptyset]) = \sum_{i=1}^s a_{ij}\psi'_j([\emptyset]) = \sum_{i=1}^s a_{ij},$$

$$\psi_i(t) = \sum_{i=1}^s a_{ij}\psi_j(t_1)\psi_j(t_2)\ldots\psi_j(t_m) \quad \text{con } t = [t_1 t_2 \cdots t_m]. \tag{1.36}$$

Finalmente, fijando $\sigma(\emptyset) = 1$, (1.35) determina de forma recursiva los valores de $\sigma(t)$.

Con todo esto, si tomamos $a_{s+1,i} = b_i$ $(i = 1 \dots s)$ tenemos que $\psi_{h,f} = Y_{s+1}$, y por tanto

$$\psi([\emptyset]) = \sum_{i=1}^{s} b_i,$$
$$\psi(t) = \sum_{i=1}^{s} b_i \psi_i(t_1)\psi_i(t_2)\dots\psi_i(t_m) \quad \text{con } t = [t_1, t_2, \dots t_m], \tag{1.37}$$

con lo cual tenemos la expansión (1.26) de $\psi_{h,f}$. Solo nos queda saber si $\phi_{h,f}$ admite un desarrollo de la forma

$$\phi_{h,f} = B(\phi),$$

y si lo admite, cómo es la expansión de $\phi_{h,f}$. Para ello, y basándonos en la definición del flujo, tenemos que

$$\frac{d}{dh}\phi_{h,f}(y) = f(\phi_{h,f}(y)), \tag{1.38}$$

La parte izquierda, teniendo en cuenta (1.26), puede ser expresada de la siguiente forma

$$\frac{d}{dh}\phi_{h,f}(y) = \frac{d}{dh}B(\phi)(y) = \sum_{t\in\mathcal{T}} \frac{h^{\rho(t)-1}}{\sigma(t)}\rho(t)\phi(t)F(t)(y)$$

El desarrollo de Taylor de la parte derecha de (1.38) puede ser expresada como $f(B(\phi)(y))$ y teniendo en cuenta (1.29) llegamos a

$$f(B(\phi)(y)) = \sum_{u\in\mathcal{F}} \frac{h^{\rho(u)}}{\sigma(u)}\phi'(u)(X(u)f)(y),$$

donde $\phi'(u) = \phi(t_1)\cdots\phi(t_m)$ para $u = t_1 \cdots t_m$.

Ambos lados deben coincidir, lo que nos lleva a que para $t = [u]$ se debe cumplir $\rho(t)\phi(t) = \phi'(u)$, es decir, si $t = [t_1 \cdots t_m]$

$$\phi(t) = \frac{\phi'(u)}{\rho(t)} = \frac{\phi(t_1)\cdots\phi(t_m)}{\rho(t)} \tag{1.39}$$

con $\phi([\emptyset]) = 1$.

Nótese que la definición de la serie que estamos utilizando en (1.26) difiere, en cuanto a notación, de la definición de las B-series utilizada por algunos autores en la literatura (véase por ejemplo [20]), e incluso sin utilizar el concepto o la definición de B-serie ([4, 10]), las series que se utilizaban sustituían $\frac{1}{\sigma(t)}$ por $\frac{\alpha(t)}{\rho(t)}$. No obstante, en los mencionados trabajos podemos ver que $\frac{\alpha(t)\gamma(t)}{\rho(t)!} = \frac{1}{\sigma(t)}$, donde $\gamma(t)$ se conoce como densidad del árbol. Si sustituimos en (1.25) dicha expresión vemos que lo que nosotros hemos denominado como $\phi(t)$ no es más que $\frac{1}{\gamma(t)}$ y, además, vemos que la definición recursiva (1.39) es acorde a la definición recursiva de $\gamma(t)$ utilizada en los citados trabajos:

$$\begin{aligned} \gamma([\emptyset]) &= 1, \\ \gamma([t_1, \ldots, t_m]) &= \rho([t_1, \ldots, t_m])\gamma(t_1)\cdots\gamma(t_m). \end{aligned} \tag{1.40}$$

Por otro lado, la definición de series (1.26) puede encontrarse en diversas publicaciones. Sirva como ejemplo el libro de Hairer, Lubich y Wanner [12], e incluso algunos años atrás, Butcher y Sanz-Serna en [5] y Murua en [16] utilizaron la misma notación.

Una vez definidas las expansiones de $\phi_{h,f}$ y de $\psi_{h,f}$, tenemos que las siguientes condiciones garantizan que $\psi_{h,f}$ es de orden p:

$$\psi(t) = \phi(t) \text{ para todo } t \in \mathcal{T} \text{ con } \rho(t) \leq p,$$

donde $\psi(t)$ y $\phi(t)$ vienen dados de forma recursiva en (1.37) y (1.39) respectivamente.

Podemos ver en las tablas 1.1 y 1.2 los valores que toman $\phi(t)$ y $\psi(t)$ para los árboles de grado menor que 6. En las mismas tablas se muestra la combinación de árboles de menor orden del que surge cada uno de los árboles.

1.9. Error global

La mayoría del software para la integración numérica de Ecuaciones Diferenciales trata de mantener la diferencia entre dos soluciones numéricas (tomada como estimación del error local) de cada paso por debajo de cierta tolerancia definida por el usuario. Se tiene la esperanza de que de esta forma el error global no sea demasiado grande. Si se desea información adicional sobre el error global hay que realizar un sobreesfuerzo computacional que depende del método utilizado para la estimación del error global acumulado en la integración. Podemos encontrar en la literatura diferentes formas de

Tabla 1.1: Condiciones para los árboles de orden menor que 5.

$[t_1 \cdots t_m]$	árbol	orden	$\phi(t)$	$\psi(t)$
$[\emptyset]$		1	1	$\sum_i b_i$
$[\bullet]$		2	$\frac{1}{2}$	$\sum_i b_i \sum_j a_{ij} = \sum_i b_i c_i$
$[\bullet, \bullet]$		3	$\frac{1}{3}$	$\sum_i b_i c_i^2$
$\left[\begin{smallmatrix}\bullet\\ \bullet\end{smallmatrix}\right]$		3	$\frac{\frac{1}{2}}{3} = \frac{1}{6}$	$\sum_i b_i \sum_j a_{ij} c_j$
$[\bullet, \bullet, \bullet]$		4	$\frac{1}{4}$	$\sum_i b_i c_i^3$
$\left[\bullet, \begin{smallmatrix}\bullet\\ \bullet\end{smallmatrix}\right]$		4	$\frac{1}{8}$	$\sum_i b_i c_i \sum_j a_{ij} c_j$
$\left[\begin{smallmatrix}\bullet & & \bullet\\ & \bullet & \end{smallmatrix}\right]$		4	$\frac{1}{12}$	$\sum_i b_i \sum_j a_{ij} c_j^2$
$\left[\begin{smallmatrix}\bullet\\ \bullet\\ \bullet\end{smallmatrix}\right]$		4	$\frac{1}{24}$	$\sum_i b_i \sum_j a_{ij} \sum_k a_{jk} c_k$

obtener aproximaciones del error global, sirva como referencia el trabajo de R. D. Skeel [24] que recoge y analiza trece distintas formas.

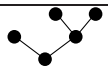
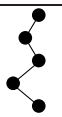
Es importante ser consciente de que en función de las propiedades del sistema de ecuaciones, del intervalo de tiempo de la integración, e incluso del método de integración que se vaya a utilizar, el error global puede ser mucho mayor que el error local, por lo que habría que prestar más atención a la evolución del error global de la integración. No obstante, el costo computacional que requieren las distintas formas de cálculo del error global hace que la mayoría de los usuarios de métodos de integración de ecuaciones diferenciales se guíen única y exclusivamente con la información sobre la estimación del error local.

1.9.1. Métodos de estimación del Error Global

La mayoría de los métodos de estimación del error global se basan en el cómputo de dos soluciones numéricas del problema, de forma que la diferencia entre las dos soluciones nos pueda dar una idea del tamaño del error.

Un método clásico para la estimación del error global es el llamado método de *Extrapolación de Richardson* y consiste en el cálculo de dos soluciones

Tabla 1.2: Condiciones para los árboles de orden 5.

$[t_1 \cdots t_m]$	árbol	orden	$\phi(t)$	$\psi(t)$
[•,•,•,•]		5	$\frac{1}{5}$	$\sum_i b_i c_i^4$
[•,•,•]		5	$\frac{1}{10}$	$\sum_i b_i c_i^2 \sum_j a_{ij} c_j$
[•,•]		5	$\frac{1}{15}$	$\sum_i b_i c_i \sum_j a_{ij} c_j^2$
[•,•]		5	$\frac{1}{20}$	$\sum_i b_i (\sum_j a_{ij} c_j)^2$
[•,•]		5	$\frac{1}{30}$	$\sum_i b_i c_i \sum_j a_{ij} \sum_k a_{jk} c_k$
[•]		5	$\frac{1}{20}$	$\sum_i b_i \sum_j a_{ij} c_j^3$
[•]		5	$\frac{1}{40}$	$\sum_i b_i \sum_j a_{ij} c_j \sum_k a_{jk} c_k$
[•]		5	$\frac{1}{60}$	$\sum_i b_i \sum_j a_{ij} \sum_k a_{jk} c_k^2$
[•]		5	$\frac{1}{120}$	$\sum_i b_i \sum_j a_{ij} \sum_k a_{jk} \sum_l a_{jl} c_l$

numéricas $\bar{y}$ y y utilizando el mismo método para las dos soluciones, pero la primera de ellas se obtiene utilizando una discretización de la variable independiente t en la que cada paso es de longitud $\bar{h}_i$ para $i = 1, 2 \ldots, N$, y la segunda solución y se obtiene haciendo que cada paso de la primera discretización se convierta en dos pasos de longitudes $\frac{\bar{h}_i}{2}$. De esta forma, y teniendo en cuenta que el error global e de un método de Runge-Kutta explícito de orden p cumple

$$e = O(H^p),$$

donde $H \geq h_i$ para $i = 1, 2, \ldots, N$, tenemos que se cumple

$$\tilde{e} = \frac{\bar{y} - y}{2^p - 1} = O(H^p)$$

con $H \geq h_i$ para $i = 1, 2, \ldots, 2N$, siendo h_i las longitudes de paso correspondientes a la solución numérica y. El valor $\tilde{e}$ nos puede dar una idea de cómo es el error global de la solución y.

Otra interesante forma de estimar el error global se presenta en [7]. En ella se propone estimar el error global mediante la resolución de una ecuación diferencial construida para tal fin, y la estimación del error global posibilita la obtención de una segunda solución numérica extrapolada. Sea el método de orden p

$$\psi_h(y) = y + h \sum_{i=1}^{s} b_i f(Y_i),$$

donde para $i = 1, \ldots, s$

$$Y_i = y + h \sum_{j=1}^{i-1} a_{i,j} f(Y_j).$$

En [7] se muestra una forma de obtener una segunda solución de orden $p+r$ de la siguiente forma:

$$\bar{\psi}_h(y, \bar{y}) = \bar{y} + h \sum_{i=s+1}^{\bar{s}} \bar{b}_i f(Y_i),$$

donde para $i = 1, \ldots, s$ el valor Y_i es el correspondiente al método $\psi_h(y)$ (de ahí que pongamos la doble dependencia de $\bar{\psi}_h$ sobre y y sobre $\bar{y}$), mientras que para $i = s + 1, \ldots, \bar{s}$ tenemos

$$Y_i = \bar{y} + h \sum_{j=1}^{i-1} a_{i,j} f(Y_j).$$

En el segundo capítulo de este trabajo mostraremos una forma más general de obtención del error global que la ofrecida en [7], y a su vez, mostraremos una forma alternativa para obtener las condiciones de los métodos ψ_h y $\bar{\psi}_h$ y la forma de trasladar dichas condiciones sobre los parámetros de los métodos.

1.10. Control de las longitudes de paso de la integración

Como ya hemos comentado, en la mayoría de los casos la integración del sistema de ecuaciones diferenciales se suele guiar con algún tipo de información sobre el error local. Evidentemente el error local no se conoce, y se realizan estimaciones de forma que el proceso no suponga un costo computacional elevado. En el caso de los métodos de Runge-Kutta, ver [20](pp 164-172), la estimación del error local se realiza utilizando métodos embebidos que no requieren el cálculo de nuevas etapas Y_i dados en (1.10), es decir, reutilizando las etapas que hacen falta para la solución numérica (1.9), se obtiene una nueva aproximación numérica $\hat{y}$ de la solución, que suele ser una aproximación de orden $\hat{p}$ inferior ($\hat{p} < p$) o superior ($\hat{p} > p$).

El tablero de Butcher que corresponde al par de métodos embebidos tiene la forma

$$\begin{array}{c|cccc} c_1 & a_{11} & & & \\ c_1 & a_{11} & a_{12} & & \\ \vdots & & & & \\ c_s & a_{s1} & a_{s2} & & a_{ss} \\ \hline & b_1 & b_2 & \dots & b_s \\ \hline & \hat{b}_1 & \hat{b}_2 & \dots & \hat{b}_s \end{array} \tag{1.41}$$

con dicho par, obtenemos dos aproximaciones $\psi_h(y)$ y $\hat{\psi}_h(y)$ del flujo $\phi_h(y)$ como

$$\psi_h(y) = y + h \sum_{i=1}^{s} b_i f(Y_i),$$
$$\hat{\psi}_h(y) = y + h \sum_{i=1}^{s} \hat{b}_i f(Y_i),$$

donde, Y_i viene dado por (1.10).

La magnitud de la resta de las dos soluciones numéricas

$$\bar{\delta}(y, h) = \psi_h(y) - \hat{\psi}_h(y) \tag{1.42}$$

nos puede dar una idea de la magnitud del error local de la solución numérica. En el caso típico en que $\hat{p} < p$ la diferencia $\bar{\delta}(y,h)$ es una estimación del error local $\hat{\delta}(y,h) = \hat{\psi}_h(y) - \phi_h(y)$ del método $\hat{\psi}_h(y)$, y no del método $\psi_h(y)$ con el que se propaga la solución. En la práctica, se trata de controlar la diferencia $\bar{\delta}(y,h)$ manteniéndola aproximadamente constante, con la esperanza de controlar indirectamente el error local del método $\psi_h(y)$. En adelante, cuando nos refiramos a la estimación del error local de un método de Runge-Kutta nos estaremos refiriendo a la diferencia $\bar{\delta}(y,h)$ dada en (1.42) entre las dos soluciones numéricas.

El procedimiento general adoptado en gran parte de las implementaciones de pares de esquemas de Runge-Kutta embebidos acepta las soluciones numéricas siempre que la estimación del error local sea, de alguna forma, menor que la tolerancia definida por quien use el sistema. La tolerancia puede ser absoluta o relativa, o una combinación de ambas. Podemos definir la tolerancia para cada componente del vector solución:

$$tol^i = Atol^i + y^i Rtol^i, \tag{1.43}$$

donde $i = 1, \ldots D$. Si $Rtol^i = 0$, estaremos trabajando con tolerancias absolutas mientras que si $Atol^i = 0$, se trabajará con tolerancias relativas.

Una vez definida la tolerancia, el proceso de integración deberá aceptar o rechazar los pasos, y para ello hay que definir algún criterio. Podemos utilizar el criterio de *error por unidad de paso,* que aceptará los pasos que tengan una estimación del error inferior a la longitud de paso multiplicado por la tolerancia, o podemos utilizar el criterio del *error por paso* que acepta los pasos que cumplan la condición

$$|y^i - \hat{y}^i| \leq tol^i, \quad i = 1, \ldots, D, \tag{1.44}$$

donde $y = (y^1, \ldots, y^D), \hat{y} = (\hat{y}^1, \ldots, \hat{y}^D)$.

Para aceptar o rechazar el paso se utiliza alguna norma dependiente de la tolerancia *tol* y se aceptará el paso si

$$\|y - \hat{y}\|_{tol} \leq 1.$$

Pueden servir como ejemplo las dos normas que mostramos a continuación:

$$\|y - \hat{y}\|_{tol} = \max_{i=1,\ldots,D} \left(\frac{|y^i - \hat{y}^i|}{Atol^i} \right)$$

$$\|y - \hat{y}\|_{tol} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(\frac{y^i - \hat{y}^i}{Atol^i} \right)^2}$$

Este tipo de normas sólo permiten tener en cuenta la tolerancia al error absoluto. Para tener en cuenta la tolerancia al error relativo se suelen sustituir las normas anteriores respectivamente por las siguientes expresiones:

$$expr_{tol} = \max_{i=1,\ldots,D} \left(\frac{|y^i - \hat{y}^i|}{Atol^i + y^i Rtol^i} \right) \tag{1.45}$$

$$expr_{tol} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left(\frac{y^i - \hat{y}^i}{Atol^i + y^i Rtol^i} \right)^2} \tag{1.46}$$

Si el valor de dichas expresiones es menor o igual que 1 se aceptará el paso, pero, por el contrario, si es mayor se rechaza la solución. Evidentemente, el rechazo de un paso significa que los cálculos realizados en la obtención de la solución numérica se han echado a perder, y se tendrá que volver a calcular una nueva solución numérica con una longitud de paso menor para que la expresión dependiente de la tolerancia dé valores aceptables. Por otro lado, tampoco interesa utilizar longitudes de paso demasiado pequeñas, ya que el coste computacional requerido se vería incrementado. La proximidad entre la expresión utilizada y el valor 1 significa que la longitud de paso se acerca al paso óptimo. Por tanto, se intenta que los pasos que se den se aproximen, sin pasarse, a la longitud máxima aceptable. Suponiendo que la estimación del error local es de orden q, se tiene que $expr_{tol} = Ch^{q+1} + O(h^{q+2})$ (donde C es el coeficiente del término dominante), de modo que para h suficientemente pequeño,

$$expr_{tol} = Ch^{q+1}.$$

De ahí, y sabiendo que para la longitud de paso actual se ha obtenido un valor que ya se conoce, es posible obtener una aproximación de la longitud de paso óptima que supuestamente hará que esa expresión se acerque al límite. Se trata de una simple regla de tres, que da como longitud de paso óptima h_{opt}:

$$h_{opt} = h \left(\frac{1}{expr_{tol}} \right)^{\frac{1}{q+1}}. \tag{1.47}$$

Tanto si se acepta el paso como si se rechaza, la nueva longitud de paso que se utilizará en la integración será la obtenida en (1.47), pero para evitar en lo posible los rechazos de los pasos de la integración conviene ser conservador a la hora de escoger la longitud de paso, de esa forma se incrementan las probabilidades de aceptarlo, evitando la perdida de tiempo computacional que provocan los rechazos. La política conservadora puede tener varias vertientes, por ejemplo, se puede limitar el valor máximo que se

permite para la longitud de un paso, o se puede también limitar el porcentaje máximo de variación de la longitud de un paso al siguiente. Hay también otras cuestiones a tener en cuenta, como la citada en [15] sobre la posibilidad de que en algún punto de la integración se anule el coeficiente dominante de la estimación del error local, lo que puede provocar una elección errónea de la longitud de paso y para evitarlo proponen la utilización de una pequeña variación en (1.47) con un pequeño incremento de coste computacional. En cualquier caso, la longitud de paso a utilizar será de la forma general

$$h_{new} = F(h_{opt}).$$

En realidad se está estimando la longitud de paso que se supone que hubiera hecho que $expr_{tol}$ fuera menor que el límite tolerable en el paso que se acaba de dar, de forma que se maximicen las posibilidades de que la solución numérica obtenida sea aceptable. Si se ha partido de y_n que es la solución numérica dada para $t = t_n$, se habrá obtenido la solución numérica $\psi_h(y_n)$ para $t = t_n + h$, y mediante el cálculo de h_{new} se calcula la longitud de paso que supuestamente se acerca a la longitud máxima aceptable. A partir de este momento de la integración hay dos opciones:

1. Si el paso dado desde t_n hasta t_n+h ha resultado ser aceptable, es decir, el valor $expr_{tol}$ obtenido es menor que la tolerancia, la solución $\psi_h(y_n)$ dada por el paso se acepta, y el proceso de integración procederá al cálculo de una nueva solución numérica para $y(t_n+h+h_{new})$ basándose en la solución numérica $\psi_h(y_n)$ aceptada. La longitud de paso h_{new} que va a utilizar es la que se supone óptima para el paso anterior pero no tiene por qué ser la longitud óptima del nuevo paso (aunque se tenga la esperanza de que sea muy parecida).

2. Por el contrario, si el paso desde $t = t_n$ hasta $t = t_n + h$ ha resultado ser inaceptable, hay que desechar los valores obtenidos $\psi_h(y_n)$ y realizar nuevos cálculos partiendo desde y_n con la longitud que se supone que será la óptima, con el objeto de obtener el resultado numérico $\psi_{h_{new}}(y_n)$ que se dará como solución numérica de $y(t_n + h_{new})$, si es que dichos cálculos son tolerables.

La diferencia que hay entre aceptar o rechazar el paso es que, tras el rechazo, volvemos a realizar los cálculos partiendo de y_n y con una longitud de paso h_{new} con el objeto de obtener la solución numérica $y_{n+1} = \psi_{h_{new}}(y_n)$ para $y(t_n + h_{new})$, mientras que si se ha aceptado el paso, realizaremos los cálculos partiendo de $y_{n+1} = \psi_h(y_n)$ y con la pretensión de obtener la solución numérica $y_{n+2} = \psi_{h_{new}}(y_{n+1})$ para $y(t_n + h + h_{new})$.

1.11. Estabilidad lineal de los métodos de Runge-Kutta explícitos

Es deseable que las propiedades cualitativas de las soluciones numéricas sean parecidas a las propiedades de la solución real del problema. En particular, si la solución exacta de un problema lineal es estable, es deseable que la solución numérica que resulta de aplicar un método dado al problema lineal sea también estable.

En el caso de los métodos de Runge-Kutta explícitos, se utiliza la ecuación test de Dahlquist:

$$y' = \lambda y, \quad y(0) = 1. \tag{1.48}$$

La solución del problema es $y(t) = e^{\lambda t}$, que es estable si y solo si $Re(\lambda) \leq 0$. La aplicación de un paso del método de Runge-Kutta con este problema equivale a

$$y_{n+1} = R(h\lambda)y_n,$$

donde $R(z)$ se conoce como *función de estabilidad lineal* del método. Obviamente, la solución numérica y_n es estable (es decir, está acotada para todo $n \geq 1$) si y solo si $|R(h\lambda)| \leq 1$ ($y_n \to 0$ cuando $n \to \infty$ si y solo si $|R(z)| < 1$). Se llama *región de estabilidad* del método al conjunto de números complejos

$$S = \{z \in \mathbb{C};\ |R(z)| \leq 1\},$$

de modo que la solución numérica es estable si y solo si $h\lambda \in S$.

Podemos calcular $R(z)$ sustituyendo $f(Y_i)$ por λY_i, en

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i f(Y_i)$$

y haciendo lo mismo, recursivamente, para

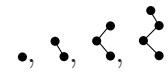
$$Y_i = y_n + \sum_{j=1}^{i-1} a_{ij} f(Y_j).$$

Si sustituimos $h\lambda$ por z llegamos a

$$y_{n+1} = R(z)y_n$$

donde $R(z)$ viene dado por

$$1+z\sum_{i}^{s} b_i + z^2 \sum_{i}^{s} b_i \sum_{j}^{i} a_{ij} + z^3 \sum_{i}^{s} b_i \sum_{j}^{i} a_{ij} \sum_{k}^{j} a_{jk} + \cdots + z^s \underbrace{\sum_{i}^{s} b_i \cdots \sum_{l}^{m} a_{lm}}_{s}.$$

Se puede observar que los coeficientes asociados a z^i corresponden a los coeficientes ψ dados en (1.37) de los árboles de orden i sin ramificaciones:  ... y, por tanto, si el método es de orden p, sabemos que los coeficientes de dichos árboles son $\frac{1}{i!}$ para $i = 1, \ldots, p$.

Capítulo 2

Métodos de Tipo Runge-Kutta con estimación del Error Global

2.1. Introducción

Generalmente la integración numérica de un problema de valor inicial de ecuaciones diferenciales ordinarias se realiza tratando de mantener la estimación del error local de cada paso por debajo de cierta tolerancia, pero no se suele tener en cuenta la evolución del error global del sistema, que es realmente de lo que depende la validez de los resultados de la integración numérica. Esto se debe al coste computacional que exige el cálculo de la estimación del error global. En este sentido, hemos trabajado en el desarrollo de metodologías para construir métodos de integración para problemas no stiff que aporten información útil sobre la propagación del error global sin que supongan un incremento substancial del coste computacional. Dichos trabajos han sido publicados y pueden ser consultados en [18] y [13]. Podemos encontrar aportaciones anteriores en el mismo sentido en el trabajo realizado por J. R. Dormand, J. P. Gilmore y P. J. Prince [7].

Nuestro trabajo se ha basado inicialmente en integraciones con longitud de paso constante. No obstante, tras los primeros resultados y con la experiencia obtenida se ha trabajado en la obtención de métodos que se adecúan a estrategias de paso variable.

2.2. Error Global de los métodos de un paso

Para simplificar, consideramos (sin pérdida de generalidad) sistemas de Ecuaciones Diferenciales Ordinarias (EDO) en la forma autónoma (1.1). El error global de los métodos de un paso (1.8) satisface la recurrencia

$$e_n = E_{h_n}(y_{n-1}, e_{n-1}),$$

donde la aplicación $E_h : R^{2D} \to R^D$ se define como $E_h(y, e) = \psi_h(y) - \phi_h(y - e)$. Evidentemente, en la medida en que desconocemos el flujo ϕ_h de (1.3) desconocemos también la aplicación E_h.

Los distintos métodos de estimación del error global propuestos en la literatura (véase por ejemplo [24]) pueden describirse como una recurrencia definida por una aplicación $\tilde{E}_h$ que de alguna forma aproxima la aplicación E_h del error global real. Por lo tanto, las estimaciones del error global en cada paso $\tilde{e}_n$ se obtienen como

$$\tilde{e}_n = \tilde{E}_{h_n}(y_{n-1}, \tilde{e}_{n-1}), \quad n = 1, \ldots, N. \tag{2.1}$$

Si las estimaciones $\tilde{e}_n$ del error global son buenas en algún sentido, se puede extrapolar y obtener una segunda solución del problema $\bar{y}_n$ como $y_n - \tilde{e}_n$ que bien podría ser una solución mejor al problema. Este nuevo punto de vista puede hacer que el proceso de la obtención de soluciones numéricas y_n junto con estimaciones del error global $\tilde{e}_n$ pueda ser considerado como el proceso de obtener dos aproximaciones y_n y $\bar{y}_n$ de la solución y posteriormente obtener la estimación del error global como la diferencia de las dos soluciones numéricas. Es decir, definimos $\bar{\psi}_h(y, \bar{y}) = \psi_h(y) - \tilde{E}_h(y, y - \bar{y})$, y obtenemos

$$\begin{aligned} y_n &= \psi_{h_n}(y_{n-1}), \\ \bar{y}_n &= \bar{\psi}_{h_n}(y_{n-1}, \bar{y}_{n-1}), \\ \tilde{e}_n &= y_n - \bar{y}_n. \end{aligned} \tag{2.2}$$

Evidentemente, cualquier proceso de la forma (2.2) puede ser interpretado como la aplicación de un método de un paso (1.8) junto con la estimación del error global de la forma (2.1), donde $\tilde{E}_h(y, e) = \psi_h(y) - \bar{\psi}_h(y, y - e)$.

Un ejemplo de la nueva interpretación de los métodos tradicionales de la literatura puede ser la siguiente: consideremos la extrapolación de Richardson basado en un método $\hat{\psi}_h$, tal y como hemos comentado en la Sección 1.9.1. La obtención de la estimación del error global consiste en el

cálculo de dos aproximaciones numéricas y_n, $\hat{y}_n$ $(n = 1, \ldots, N)$ con $\hat{y}_0 = y_0$ de la siguiente forma

$$\begin{aligned} y_n &= \hat{\psi}_{h_n/2}\left(\hat{\psi}_{h_n/2}(y_{n-1})\right), \\ \hat{y}_n &= \hat{\psi}_{h_n}\left(\hat{y}_{n-1}\right), \\ \tilde{e}_n &= \frac{1}{2^p - 1}(\hat{y}_n - y_n) \end{aligned} \tag{2.3}$$

y la solución extrapolada se obtiene como $\bar{y}_n = y_n - \tilde{e}_n$. Lo cual puede ser interpretado como un proceso de la forma (2.2) donde

$$\begin{aligned} \psi_h(y) &= \hat{\psi}_{h/2}\left(\hat{\psi}_{h/2}(y)\right), \\ \bar{\psi}_h(y, \bar{y}) &= \frac{1}{2^p - 1}\left(2^p \psi_h(y) - \hat{\psi}_h(\hat{y})\right), \quad \text{con } \hat{y} = \bar{y} - 2^p(\bar{y} - y). \end{aligned}$$

Todavía queda por aclarar qué se entiende por una aproximación buena del error global $\tilde{e}_n$. Típicamente se requiere que $\tilde{e}_n$ sea una correcta estimación del error global asintóticamente, en el sentido que

$$\tilde{e}_n = (I + O(H))e_n, \quad \text{para } nH \leq \text{Constante}, \quad H = \max_n h_n,$$

lo cual es equivalente a que la aproximación extrapolada $\bar{y}_n$ sea de orden $p + 1$, es decir,

$$\bar{y}_n - y(t_n) = O(H^{p+1}), \quad nH \leq \text{Constante}.$$

Más en general, se puede requerir que $\tilde{e}_n$ tenga para algún valor $r \geq 1$, r términos asintóticamente correctos, esto es,

$$\tilde{e}_n = (I + O(H^r))e_n,$$

ó equivalentemente, $\bar{y}_n - y(t_n) = O(H^{\bar{p}})$, con $\bar{p} = p + r$.

La extrapolación de Richardson (2.3), ofrece en general una estimación del error global asintóticamente correcto con $r = 1$, pero si el método es simétrico ([20]) nos da dos términos del error global asintóticamente correctos (es decir, $r = 2$), ya que la expansión asintótica solo tiene exponentes pares de H.

Una interesante clase de esquemas que también se ajustan en el formato (2.2) son los llamados *globally embedded RK methods* desarrollados por Dormand, Gilmore y Prince [7], donde las aplicaciones ψ_h y $\bar{\psi}_h$ se definen de la siguiente manera:

$$\psi_h(y) \;=\; y + h\sum_{i=1}^{s} b_i f(Y_i), \tag{2.4}$$

donde

$$Y_i \;=\; y + h\sum_{j=1}^{i-1} a_{ij} f(Y_j), \quad i = 1, \ldots, s, \tag{2.5}$$

y

$$\bar{\psi}_h(y, \bar{y}) \;=\; \bar{y} + h\sum_{i=s+1}^{\bar{s}} \bar{b}_i f(Y_i), \tag{2.6}$$

donde

$$Y_i \;=\; \bar{y} + h\sum_{j=1}^{i-1} a_{ij} f(Y_j), \quad i = s+1, \ldots, \bar{s}, \tag{2.7}$$

El procedimiento aplicado en [7] en la construcción de esquemas de este tipo con ordenes p y $\bar{p} = p + r$ se puede describir de la siguiente manera: Se construye un método de s etapas de orden p, con una apropiada extensión continua, y a continuación se le aplica una técnica de estimación del error global conocida como *Solving for the correction* [24], utilizando esquemas RK especialmente diseñadas para este objetivo.

Otra forma alternativa para la obtención de este tipo de esquemas con valores p y $\bar{p}$ dados, puede ser el estudio directo de las condiciones que deben satisfacer las aplicaciones ψ_h y $\bar{\psi}_h$ para que den aproximaciones de $y(t_n)$ de ordenes p y $\bar{p}$ respectivamente y trasladar dichas condiciones a los parámetros b_i, $\bar{b}_i$ y a_{ij}.

2.3. Una clase general de esquemas para la obtención del Error Global

Consideramos la siguiente generalización de los procesos de la forma (2.2): Sean $\bar{y}_0 = y_0 = y(t_0)$, y calculemos para $n = 1, 2, \ldots$,

$$\begin{aligned} y_n &= \psi_{h_n}(y_{n-1}, \bar{y}_{n-1}), \\ \bar{y}_n &= \bar{\psi}_{h_n}(y_{n-1}, \bar{y}_{n-1}), \\ \tilde{e}_n &= y_n - \bar{y}_n. \end{aligned} \tag{2.8}$$

donde, por el momento, ψ_h y $\bar{\psi}_h$ pueden considerarse como cualquier aplicación $R^{2D} \to R^D$. Se supone que los valores $y_n, \bar{y}_n$ son aproximaciones de la solución $y(t_n)$ de (1.3) con valor inicial $y(t_0) = y_0$, y que $\tilde{e}_n$ será una estimación del error global $e_n = y_n - y(t_n)$ de la aproximación y_n.

Se ha trabajado en la obtención de las condiciones que deben cumplir las aplicaciones ψ_h y $\bar{\psi}_h$ para que el proceso (2.8) obtenga una aproximación y_n de la solución $y(t_n)$ de orden p junto con estimaciones válidas $\tilde{e}_n = y_n - \bar{y}_n$ del error global e_n.

En el proceso (2.8) subyacen dos integradores de un paso que pueden definirse como $\psi_h(y) := \psi_h(y, y)$, $\bar{\psi}_h(y) := \bar{\psi}_h(y, y)$ y a los que llamaremos como *integradores de un paso subyacientes*. Sean $q, \bar{q} \geq 0$ los mayores enteros positivos tales que

$$\begin{aligned} \psi_h(y, y+e) &= \psi_h(y, y) + O(h^{q+1}||e|| + h||e||^2), && (2.9) \\ \bar{\psi}_h(y+e, y) &= \bar{\psi}_h(y, y) + O(h^{\bar{q}+1}||e|| + h||e||^2), && (2.10) \end{aligned}$$

cuando $h \to 0$ y $e \to 0$. Cuanto mayores sean los valores q y $\bar{q}$, más se asemeja la aplicación de (2.8) a la aplicación de forma independiente de los dos métodos subyacientes $y_{n+1} = \psi_h(y_n)$ y $\bar{y}_{n+1} = \bar{\psi}_h(\bar{y}_n)$. Es por ello que nos referimos a (2.9)–(2.10) como *condiciones de independencia*.

Además, nos puede interesar reducir la contribución del término $O(h||e||^2)$ en (2.9)–(2.10). En ese caso deberemos considerar nuevas condiciones de independencia en los que aparecerán nuevos términos. Por ejemplo, en algunos casos se puede reemplazar $O(h||e||^2)$ por $O(h^2||e||^2 + h||e||^3)$. Más adelante, en la sección 2.5 veremos la forma de obtener las condiciones que deben cumplir los métodos para que los métodos subyacientes sean de un orden dado y para que cumplan las condiciones de independencia que establezcamos.

Denotemos el error local de los métodos subyacientes de un paso mediante

$$\begin{aligned} \delta(y, h) &= \psi_h(y, y) - \phi_h(y), \\ \bar{\delta}(y, h) &= \bar{\psi}_h(y, y) - \phi_h(y). \end{aligned}$$

Lema 2 *Si los métodos subyacientes de un paso $\psi_h(y)$ y $\bar{\psi}_h(y)$ son respectivamente de orden p y $p+r$, y las condiciones de independencia (2.9)–(2.10) se cumplen para $q, \bar{q} \geq 0$, entonces*

$$\begin{aligned} e_n &= R_{n,n-1} e_{n-1} + \delta_n + \pi_n, \\ \bar{e}_n &= R_{n,n-1} \bar{e}_{n-1} + \bar{\delta}_n + \bar{\pi}_n, \\ \tilde{e}_n &= R_{n,n-1} \tilde{e}_{n-1} + (\delta_n - \bar{\delta}_n) + (\pi_n - \bar{\pi}_n), \end{aligned} \qquad (2.11)$$

donde $e_n = y_n - y(t_n)$, $\bar{e}_n = \bar{y}_n - y(t_n)$, $\tilde{e}_n = y_n - \bar{y}_n$, *y*

$$\begin{aligned}
\delta_n &= \delta(y_{n-1}, h_n), \quad \bar{\delta}_n = \bar{\delta}(\bar{y}_{n-1}, h_n), \\
\bar{\pi}_n &= O(h_n^{q+1}||\tilde{e}_{n-1}|| + h_n(||\tilde{e}_{n-1}||^2 + ||\bar{e}_{n-1}||^2)), \\
\pi_n &= O(h_n^{\bar{q}+1}||\tilde{e}_{n-1}|| + h_n(||\tilde{e}_{n-1}||^2 + ||e_{n-1}||^2)), \\
R_{nk} &= \frac{\partial \phi_{t_n - t_k}}{\partial y}(y(t_k)).
\end{aligned}$$

Además,

$$\begin{aligned}
e_n &= \sum_{k=1}^{n} R_{nk}(\delta_k + \pi_k), \\
\bar{e}_n &= \sum_{k=1}^{n} R_{nk}(\bar{\delta}_k + \bar{\pi}_k), \\
\tilde{e}_n &= \sum_{k=1}^{n} R_{nk}(\delta_k - \bar{\delta}_k + \pi_k - \bar{\pi}_k).
\end{aligned} \tag{2.12}$$

Observación 2 *Se deduce de (2.11) que, para que* e_n, $\bar{e}_n$, *y* $\tilde{e}_n$ *sigan patrones de propagación similares* π_n *y* $\bar{\pi}_n$ *deberían ser lo suficientemente pequeños. En este sentido se puede suponer que, para valores de* h_n *suficientemente pequeños, cuanto mayores sean* q *y* $\bar{q}$, *tendremos menores valores para* π_n *y* $\bar{\pi}_n$, *y por tanto sería preferible tener* q *y* $\bar{q}$ *lo mayores que sea posible. De todas formas, no es obvio la medida en la que en la práctica, donde los valores de* h *pueden no ser tan pequeños, es importante tener mayores o menores valores de* q *y* $\bar{q}$.

Observación 3 *Evidentemente, (2.12) no garantiza que los errores globales* e_n, $\bar{e}_n$ *y el error global estimado* $\tilde{e}_n$ *se propaguen de una forma similar, ni siquiera en el caso de que* π_n *y* $\bar{\pi}_n$ *sean insignificantes. Podemos mencionar un ejemplo en el que todo el proceso fallaría: Consideremos un sistema bidimensional, en el que las matrices Jacobianas* R_{nk} *tengan dos valores propios* λ_k^1 *y* λ_k^2 *con vectores propios* v_k^1 *y* v_k^2, *de tal forma que* $|\lambda_k^1| >> |\lambda_k^2|$. *Sea* $\delta_k = \delta_k^1 v_k^1 + \delta_k^2 v_k^2$ *y* $\bar{\delta}_k = \bar{\delta}_k^1 v_k^1 + \bar{\delta}_k^2 v_k^2$ *(para cada* k*). Si* $\delta_k^1 = \bar{\delta}_k^2$, *entonces* $\tilde{e}_n$ *puede ser mucho menor que* e_n *y* $\bar{e}_n$. *De todas formas, en general, podemos esperar que las distintas formas en las que cada* R_{nk} *afecte a* e_n, $\bar{e}_n$, *y* $\tilde{e}_n$ *(dependiendo de la dirección de* δ_k, $\bar{\delta}_k$ *y* $\delta_k - \bar{\delta}_k$*) tenderán a compensarse para distintos valores de* $k = 1, \ldots, n$.

Demostración Probaremos la primera igualdad de (2.11). La segunda se puede probar de forma análoga, mientras que la tercera se obtiene sustrayendo término a término las dos primeras igualdades de (2.11). Tenemos

que

$$\begin{aligned} e_n &= (\psi_{h_n}(y_{n-1}, \bar{y}_{n-1}) - \psi_{h_n}(y_{n-1}, y_{n-1})) \\ &\quad + (\psi_{h_n}(y_{n-1}, y_{n-1}) - \phi_{h_n}(y_{n-1})) \\ &\quad + (\phi_{h_n}(y_{n-1}) - \phi_{h_n}(y(t_{n-1}))) \,. \end{aligned}$$

De la definición de δ_n y teniendo en cuenta la condición de independencia (2.9), llegamos a

$$e_n = O(h^{q+1}||\tilde{e}_{n-1}|| + h||\tilde{e}_{n-1}||^2) + \delta_n + (\phi_{h_n}(y_{n-1}) - \phi_{h_n}(y(t_{n-1}))) \,,$$

y teniendo en cuenta el desarrollo de Taylor de $\phi_{h_n}(y(t_{n-1}) + e_{n-1})$ en torno a $e_{n-1} = 0$, podemos escribir

$$\begin{aligned} e_n &= O(h^{q+1}||\tilde{e}_{n-1}|| + h||\tilde{e}_{n-1}||^2) + \delta_n + \\ &\quad + \left(\frac{\partial \phi_{h_n}}{\partial y}(y(t_{n-1}))e_{n-1} + O(h||e_{n-1}||^2) \right) , \end{aligned}$$

lo que nos da la primera igualdad de (2.11). Las igualdades de (2.12) se obtienen de (2.11) ya que $R_{n,k} = R_{n,n-1}R_{n-1,k}$. □

Se pueden adaptar las técnicas estándares de estudio de la convergencia de los métodos de un paso para ODEs [20] con el objeto de obtener el siguiente resultado:

Teorema 1 *Bajo la hipótesis de Lema 2, si $\bar{q} \geq r$, entonces las aproximaciones numéricas obtenidas mediante el esquema (2.8) satisfacen*

$$\begin{aligned} e_n &= y_n - y(t_n) = O(H^p) \\ \bar{e}_n &= \bar{y}_n - y(t_n) = O(H^{p+r}), \end{aligned}$$

para todo n tal que $nH \leq$ Constante, $H = \max_n h_n$.

2.4. Métodos Runge-Kutta embebidos con estimación del Error Global

Ya hemos comentado que los métodos propuestos por Dormand, Gilmore y Prince (2.4)–(2.7) son de la forma (2.2). Es decir, obtienen una solución con un método Runge-Kutta puro, mientras que una segunda aproximación es obtenida combinando las etapas de la primera aproximación con otras etapas que dependen a su vez de una segunda aproximación. La segunda

solución es utilizada para la estimación del error global. Nosotros hemos propuesto una generalización de este proceso que encaja en el formato (2.8) donde las aplicaciones $\psi_h, \bar{\psi}_h : R^{2D} \to R^D$ se definen como

$$\psi_h(y, \bar{y}) = y + h \sum_{i=1}^{\bar{s}} b_i f(Y_i), \tag{2.13}$$

$$\bar{\psi}_h(y, \bar{y}) = \bar{y} + h \sum_{i=1}^{\bar{s}} \bar{b}_i f(Y_i), \tag{2.14}$$

donde para $i = 1, \ldots, \bar{s}$,

$$Y_i = \mu_i y + \bar{\mu}_i \bar{y} + h \sum_{j=1}^{i-1} a_{ij} f(Y_j), \tag{2.15}$$

donde $\bar{\mu}_i = 1 - \mu_i$ para $i = 1, \ldots, \bar{s}$.

Se pueden obtener esquemas que encajan en la forma (2.2) con el simple requerimiento en (2.13)–(2.15) de que para algún $s < \bar{s}$,

$$\begin{aligned} \mu_i &= 1, \quad i = 1, \ldots, s \\ b_i &= 0, \quad i = s+1, \ldots, \bar{s} \end{aligned}$$

Así mismo, si además de las anteriores restricciones imponemos las siguientes

$$\begin{aligned} \bar{b}_i &= 0, \quad i = 1, \ldots, s, \\ \mu_i &= 0, \quad i = s+1, \ldots, \bar{s}, \end{aligned}$$

obtendremos la familia de los métodos globalmente embebidos ó *globally embedded RK methods* definidos en (2.4)–(2.7).

En el esquema definido por (2.13)–(2.15) subyacen dos métodos Runge-Kutta de un paso, es decir, tanto $\psi_h(y, y)$ como $\bar{\psi}_h(y, y)$ son dos métodos Runge-Kutta que encajan en la definición estándar (1.9)–(1.10).

2.5. Condiciones sobre los parámetros del método

Para poder aplicar el Lema 2 y el Teorema 1, nos hace falta saber el orden de los métodos Runge-Kutta subyacientes, al igual que para la obtención de q y $\bar{q}$ para los que se cumplen las condiciones de independencia (2.9)–(2.10). En la introducción se muestra una forma de obtener sistemáticamente las

condiciones que han de cumplir los coeficientes de un método RK explícito para que el método sea de un orden dado.

Necesitamos conocer la expansión de $\psi_h(y, y+e)$ y de $\bar{\psi}_h(y, y+e)$ en potencias de h para poder igualarla a la expansión del flujo $\phi_h(y)$ y así obtener las condiciones que deben cumplir los parámetros del esquema.

Para conocer el desarrollo en serie de potencias de h de $\psi_h(y, y+e)$ y de $\bar{\psi}_h(y, y+e)$ vamos a seguir el procedimiento seguido en la sección 1.8, es decir, suponiendo que la serie correspondiente a cada Y_i tiene una forma general dada, deduciremos los valores de cada elemento de la serie.

Supongamos que Y_i se puede representar como

$$Y_i = y + \sum_{t \in \mathcal{T}_*} \frac{h^{\rho(t)}}{\sigma(t)} \psi_i(t) F(t)(y, e), \tag{2.16}$$

donde $\mathcal{T}_*$ es un conjunto contable de índices (veremos que se puede tomar como el conjunto de árboles con hojas blancas y negras y nodos internos negros), y para cada $t \in \mathcal{T}_*$, el orden $\rho(t)$ es un número positivo, $\sigma(t)$ es un factor que se elegirá de forma conveniente, la diferencial elemental $F(t)$ es una aplicación $F(t) : \mathbb{R}^{2D} \to \mathbb{R}^D$ que depende del sistema (1.1) que se pretende integrar, y $\psi_i(t)$ es un coeficiente real que no depende del sistema.

Lo primero que nos hace falta conocer es la expansión de $f(Y_i)$ y podemos obtenerlo aplicando el Lema 1:

$$f(Y_i) = \sum_{u \in \mathcal{F}_*} \frac{h^{\rho(u)}}{\sigma(u)} \hat{\psi}_i(u) X(u) f(y, e), \tag{2.17}$$

donde cada elemento $u \in \mathcal{F}_*$ es una tupla no ordenada de elementos de $\mathcal{T}_*$, que se puede interpretar formalmente como un producto conmutativo $u = t_1 t_2 \cdots t_m$ de elementos de $\mathcal{T}_*$, y el elemento neutro, que denotaremos como $\emptyset$, es la tupla compuesta por 0 elementos de $\mathcal{T}_*$ y también pertenece a $\mathcal{F}_*$. Para cada $u \in \mathcal{F}_*$, $\rho(u) \in \mathbb{Z}^+$, $\hat{\psi}_i(u) \in \mathbb{R}$, así como la aplicación $X(u)f(y) : \mathbb{R}^{2D} \to \mathbb{R}^D$, están determinados según el Lema 1 a partir de los valores de $\rho(t)$, $\psi_i(t)$ y de la definición de $F(t) : \mathbb{R}^{2D} \to \mathbb{R}^D$ para los índices de $t \in \mathcal{T}_*$.

Si sustituimos la expresión correspondiente a $f(Y_i)$ dada por (2.17) en (2.15), llegamos a

$$Y_i = y + \bar{\mu}_i e + \sum_{u \in \mathcal{F}_*} \frac{h^{\rho(u)+1}}{\sigma(u)} \Big(\sum_{j=1}^{\bar{s}} a_{ij} \hat{\psi}_j(u)\Big) X(u)[f](y, e), \tag{2.18}$$

que debe ser equiparable a (2.16).

Sean

$$\mathcal{T}_{*l} = \{t \in \mathcal{T}_* \quad / \quad \rho(t) = l\}$$

y

$$\mathcal{F}_{*l} = \{t_1 \cdots t_m \quad / \quad t_i \in \mathcal{T}_{*j_i}, \quad \sum_{i=1}^{m} j_i = l\}.$$

Teniendo en cuenta las relaciones entre el conjunto $\mathcal{T}_*$ y el conjunto $\mathcal{F}_*$, de la equiparación de (2.18) con (2.16) fijándonos en los coeficientes de h^l, podemos definir recursivamente tanto el conjunto $\mathcal{T}_*$ como $\mathcal{F}_*$ de forma coherente.

1. Para el caso h^0, en (2.18) tenemos el término $\bar{\mu}_i e$, por lo que elegimos un elemento en $\mathcal{T}_{*0}$, que denotamos como $\circ$, para el que
   - $\rho(\circ) = 0$,
   - $\sigma(\circ) = 1$,
   - $F(\circ)(y, e) = e$,
   - $\psi_i(\circ) = \bar{\mu}_i$
2. Para los casos de h^l con $l = 1, 2, 3, \ldots$, tenemos que los elementos $u \in \mathcal{F}_{*l-1}$ son las m-tuplas no ordenadas $t_1 \cdots t_m$, con $t_i \in \mathcal{T}_{*j_i}$, y $\sum_{i=1}^{m} j_i = l - 1$, y para cada uno de estos elementos $u \in \mathcal{F}_*$, elegimos un único elemento en $\mathcal{T}_{*l}$, que denotamos como $t = [u]$.

Hemos comentado en el Lema 1 que la tupla con 0 elementos de $\mathcal{T}_*$ (el elemento neutro si consideramos las tuplas como productos conmutativos de elementos de $\mathcal{T}_*$) forma parte de $\mathcal{F}_*$, y lo denotamos como $\emptyset$. Para el árbol $t = [\emptyset]$ correspondiente a dicho elemento neutro, y representado a partir de ahora mediante $\bullet$, definimos

- $\rho(\bullet) = \rho(\emptyset) + 1 = 1$,
- $\sigma(\bullet) = \sigma(\emptyset) = 1$,
- $F(\bullet)(y, e) = X(\emptyset)[f](y, e) = f(y)$,
- $\psi_i(\bullet) = \sum_{j=1}^{\bar{s}} a_{ij} \hat{\psi}_j(\emptyset) = \sum_{j=1}^{\bar{s}} a_{ij}$.

Para el resto de elementos de $\mathcal{F}_*$, sean $t_1, \ldots, t_m \in \mathcal{T}_*$ tales que $u = t_1 \cdots t_m$, entonces, para el árbol $t = [u] \in \mathcal{T}_*$ elegimos

- $\rho([t_1 \cdots t_m]) = \rho(u) + 1 = \rho(t_1) + \ldots + \rho(t_m) + 1$,

- $F([t_1 \cdots t_m])(y,e) = X(t_1 \cdots t_m])[f](y,e) =$
  $= f^{(m)}(F(t_1)(y,e), \ldots, F(t_m)(y,e))$,

- la recursión que nos posibilitará la obtención de las condiciones sobre los parámetros del método,

$$\psi_i([t_1 \cdots t_m]) = \sum_{j=1}^{\bar{s}} a_{ij}\hat{\psi}_j(u) = \sum_{j=1}^{\bar{s}} a_{ij}\psi_j(t_1)\ldots\psi_j(t_m), \qquad (2.19)$$

- Además, si $u = t_1^{r_1} \cdots t_n^{r_n}$ donde $t_1, \ldots t_n$ son distintos dos a dos, entonces

$$\sigma([t_1^{r_1} \cdots t_n^{r_n}]) = r_1! \cdots r_n! \sigma(t_1)^{r_1} \cdots \sigma(t_n)^{r_n}.$$

El conjunto $\mathcal{T}_*$ se puede representar como el conjunto de los arboles con hojas blancas y negras y nodos internos negros. Es decir, podemos representar gráficamente estos elementos, por una parte, como el árbol con un único nodo blanco, y por otra, como los árboles con raíz y vértices negros, pero a los que hemos añadido un conjunto de hojas blancas, siendo el número de hojas blancas mayor o igual que 0. Las hojas blancas corresponden a los elementos $\circ \in \mathcal{T}_{*0}$ que siempre se sitúan como hojas, no como nodos internos al árbol, ya que la raíz de un árbol es siempre un vértice negro excepto en el caso del árbol $\circ$ que tiene un único vértice y es blanco. El conjunto de árboles con raíz con vértices negros $\mathcal{T}$, considerado en el caso de los métodos de Runge-Kutta estándar, es obviamente un subconjunto de $\mathcal{T}_*$. Además, si $t \in \mathcal{T}_*$ solo tiene vértices negros, las definiciones de $\rho(t)$, $\sigma(t)$ y $F(t)$ coinciden con las dadas en la Sección 1.8 para las B-series. Otra interesante observación es que las hojas blancas no tienen ningún efecto en $\rho(t)$, por lo que la potencia de h a la que afecta el árbol solo depende de los nodos negros. Más concretamente, $\rho(t)$ es el número de vértices negros. En consecuencia, el número de árboles que hay para una potencia h^p con $p > 0$ es ilimitado ya que podemos añadir tantas hojas blancas como queramos sin alterar el valor $\rho(t)$.

Ahora que conocemos la expansión de $f(Y_i)$, lo podemos sustituir en (2.13) y haciendo uso de (2.17) obtendremos

$$\begin{aligned} \psi(y, y+e) &= y + \sum_{u \in \mathcal{F}_*} \frac{h^{\rho(u)+1}}{\sigma(u)} \left(\sum_{i=1}^{\bar{s}} b_i \hat{\psi}_i(u) \right) X(u)[f](y,e) \\ &= y + \sum_{t \in \mathcal{T}_*} \frac{h^{\rho(t)}}{\sigma(t)} \psi(t) F(t)(y,e), \end{aligned} \qquad (2.20)$$

donde para los diferentes elementos $t \in \mathcal{T}_*$ tenemos que

- para $t = \circ$, se cumplen $\psi(\circ) = 0$, $\sigma(\circ) = 1$, $\rho(\circ) = 0$ y $F(\circ)(y, e) = e$,
- para $t = [\emptyset] = \bullet$, se cumplen $\psi(\bullet) = \sum_{i=1}^{s} b_i$, $\sigma(\bullet) = 1$, $\rho(\bullet) = 1$ y $F(\bullet)(y, e) = f(y)$,
- y para el resto de elementos $t = [t_1 \cdots t_m]$, tenemos que

$$\psi([t_1 \cdots t_m]) = \sum_{i=1}^{\bar{s}} b_i \psi_i(t_1) \cdots \psi_i(t_m), \tag{2.21}$$

con $\psi_i(t)$ dado en (2.19),

$$F([t_1 \cdots t_m])(y, e) = f^{(m)}(F(t_1)(y, e), \ldots, F(t_m)(y, e)),$$

y si expresamos el árbol t de la forma $t = [t_1^{r_1} \cdots t_n^{r_n}]$ donde $t_1, \ldots t_n$ son distintos dos a dos, entonces

$$\sigma([t_1^{r_1} \cdots t_n^{r_n}]) = r_1! \cdots r_n! \sigma(t_1)^{r_1} \cdots \sigma(t_n)^{r_n}.$$

Para la expansión del flujo, sabemos de la Sección 1.8 que puede ser expandido como una B-serie

$$\phi_{h,f} = B(\phi), \tag{2.22}$$

donde $\phi(t)$ viene dada por la recursión que se muestra en (1.39), y donde los elementos que aparecen en la serie corresponden a los árboles con raíz con vértices negros $\mathcal{T} \subset \mathcal{T}_*$. Definiendo $\phi(t) = 0$ si $t \in \mathcal{T}_* - \mathcal{T}$, entonces

$$\phi_{h,f} = y + \sum_{t \in \mathcal{T}_*} \frac{h^{\rho(t)}}{\sigma(t)} \phi(t) F(t)(y, e), \tag{2.23}$$

Por tanto, para que el método cumpla las condiciones de orden que nos interese, solo nos resta hacer que los coeficientes de cada árbol en (2.20) sean iguales a los coeficientes de los mismos árboles en (2.23). Los árboles que aparecen en el desarrollo en potencias de h en (2.20) son los árboles con raíz negra y hojas blancas y negras. Estos árboles junto con las condiciones que surgen de cada uno de ellos se muestran en la Tabla 2.1.

Para el caso del segundo método $\bar{\psi}_h$ definido en (2.14) podemos obtener la expansión de igual manera que para ψ_h. La expansión de $f(Y_i)$ dada en (2.17) la podemos sustituir en (2.14) con lo que obtenemos

Tabla 2.1: Condiciones de los árboles con hojas blancas y negras de menos de 5 vértices, y el agrupamiento de árboles para que las *condiciones de independencia* tengan los términos indicados. Para que la estimación de $\psi_h(y, y+e) - \phi_h(y)$ tenga la forma que aparece en la columna de la derecha habrán de cumplirse todas las condiciones de los árboles que estén más arriba.

árbol	Condición: $\psi(t) = \phi(t)$	Estimación $\psi_h(y, y+e) - \phi_h(y)$
h		
	$\sum_i b_i = 1$	
h^2		
	$\sum_i b_i c_i = \frac{1}{2}$	
$h\|\|e\|\|$		
	$\sum_i b_i \bar{\mu}_i = 0$	$O(h^3 + h^2\|\|e\|\| + h\|\|e\|\|^2)$
h^3		
	$\sum_i b_i c_i^2 = \frac{1}{3}$	
	$\sum_i b_i \sum_j a_{ij} c_j = \frac{1}{6}$	$O(h^4 + h^2\|\|e\|\| + h\|\|e\|\|^2)$
$h^2\|\|e\|\|$		
	$\sum_i b_i c_i \bar{\mu}_i = 0$	
	$\sum_i b_i \sum_j a_{ij} \bar{\mu}_j = 0$	$O(h^4 + h^3\|\|e\|\| + h\|\|e\|\|^2)$
$h\|\|e\|\|^2$		
	$\sum_i b_i \bar{\mu}_i^2 = 0$	$O(h^4 + h^3\|\|e\|\| + h^2\|\|e\|\|^2 + h\|\|e\|\|^3)$
h^4		
	$\sum_i b_i c_i^3 = \frac{1}{4}$	
	$\sum_i b_i c_i \sum_j a_{ij} c_j = \frac{1}{8}$	
	$\sum_i b_i \sum_j a_{ij} c_j^2 = \frac{1}{12}$	
	$\sum_i b_i \sum_j a_{ij} \sum_k a_{jk} c_k = \frac{1}{24}$	$O(h^5 + h^3\|\|e\|\| + h^2\|\|e\|\|^2 + h\|\|e\|\|^3)$
$h^3\|\|e\|\|$		
	$\sum_i b_i c_i^2 \bar{\mu}_i = 0$	
	$\sum_i b_i c_i \sum_j a_{ij} \bar{\mu}_j = 0$	
	$\sum_i b_i \bar{\mu}_i \sum_j a_{ij} c_j = 0$	
	$\sum_i b_i \sum_j a_{ij} c_j \bar{\mu}_j = 0$	
	$\sum_i b_i \sum_j a_{ij} \sum_k a_{jk} \bar{\mu}_k = 0$	$O(h^5 + h^4\|\|e\|\| + h^2\|\|e\|\|^2 + h\|\|e\|\|^3)$

$$\begin{aligned}\bar{\psi}(\bar{y}+e,\bar{y}) &= \bar{y}+\sum_{u\in\mathcal{F}_*}\frac{h^{\rho(u)+1}}{\sigma(u)}\left(\sum_{i=1}^{\bar{s}}\bar{b}_i\hat{\psi}_i(u)\right)X(u)[f](y,e)\\ &= \bar{y}+\sum_{t\in\mathcal{T}_*}\frac{h^{\rho(t)}}{\sigma(t)}\bar{\psi}(t)F(t)(y,e),\end{aligned}\tag{2.24}$$

donde $\rho(t)$ y $\sigma(t)$ se definen igual que en (2.20) y $\bar{\psi}(t)$ está definido para cada $t\in\mathcal{T}_*$ como sigue:

$$\begin{aligned}\bar{\psi}(\circ) &= 0,\\ \bar{\psi}(\bullet) &= \sum_{i=1}^{\bar{s}}\bar{b}_i,\\ \bar{\psi}(t) &= \sum_{i=1}^{\bar{s}}\bar{b}_i\bar{\psi}_i(t_1)\cdots\bar{\psi}_i(t_m)\ \text{ si } t=[t_1\cdots t_m]\end{aligned}$$

con

$$\begin{aligned}\bar{\psi}_i(\circ) &= \mu_i,\\ \bar{\psi}_i(\bullet) &= \sum_{j=1}^{\bar{s}}a_{ij},\\ \bar{\psi}_i([t_1\cdots t_m]) &= \sum_{j=1}^{\bar{s}}a_{ij}\bar{\psi}_j(t_1)\ldots\bar{\psi}_j(t_m).\end{aligned}$$

Obtendremos las condiciones correspondientes comparando (2.23) con el desarrollo en serie de potencias de h de $\bar{\psi}_h(\bar{y}+e,\bar{y})$ dado en (2.24), pero no mostramos las condiciones de orden del método subyaciente ni las condiciones de independencia (2.10) que han de cumplir los parámetros del método $\bar{\psi}_h(\bar{y}+e,\bar{y})$, porque son muy similares a las condiciones mostradas en la Tabla 2.1 para el método $\psi_h(y,y+e)$. Por simetría, basta con sustituir en la Tabla 2.1 b_i y $\bar{\mu}_i$ por $\bar{b}_i$, y μ_i respectivamente para obtener las condiciones del método $\bar{\psi}_h(\bar{y}+e,\bar{y})$.

2.6. Consideraciones prácticas

Supongamos que el Teorema 1 se cumple con $r\geq 1$ para el esquema (2.13)–(2.15). Entonces, tiene sentido proporcionar la aproximación de mayor orden $\bar{y}_n$, en lugar de y_n, como la solución numérica. En ese caso, $\tilde{e}_n$

ya no es una estimación del error global asintóticamente correcta, sino una *estimación incierta* [24]. Así, esperamos que $\tilde{e}_n = y_n - \bar{y}_n$ sea mayor que $\bar{e}_n = \bar{y}_n - y(t_n)$ para secuencias de pasos h_n suficientemente pequeños, y a su vez, esperamos que el error exacto $\bar{e}_n$ y el estimado $\tilde{e}_n$ se propaguen de una forma similar. En este sentido, no hace falta que $\bar{p} = p + r$ sea mucho mayor que p, ya que en ese caso, $\tilde{e}_n$ sería una estimación excesivamente conservadora de $\bar{e}_n$ para secuencias de h_n suficientemente pequeñas.

Si comparamos la aplicación del esquema (2.8) definido por (2.13)–(2.15) con la aplicación del método Runge-Kutta subyaciente de orden $\bar{p}$, $\hat{y}_n = \bar{\psi}_{h_n}(\hat{y}_{n-1}, \hat{y}_{n-1})$, podemos observar por un lado que en cada paso requieren prácticamente el mismo esfuerzo computacional, y por otro lado, que cuanto mayor sea $\bar{q}$, más similares son $\bar{y}_n$ y $\hat{y}_n$ (para pequeños valores de h_n y $\tilde{e}_n$). Por tanto, podemos esperar para valores razonables de $\bar{q}$ en la condición de independencia (2.10), que las dos aproximaciones $\bar{y}_n$ y $\hat{y}_n$ muestren una precisión parecida. En concreto, bajo las hipótesis del Teorema 1 se puede demostrar que,

$$\bar{y}_n - y(t_n) = (I + O(H^{\bar{q}-r}))(\hat{y}_n - y(t_n)), \tag{2.25}$$

para $nH \leq$ Constante y $H = \max_n h_n$.

Estas consideraciones hacen que esperemos que el cálculo de $\bar{y}_n$ junto con la *estimación incierta* $\tilde{e}_n$ del error global ofrezca unas soluciones numéricas comparables a las que se obtendrían con el método Runge-Kutta subyaciente $\hat{y}_n$, por lo que obtendríamos información sobre el error global sin pérdida sustancial de eficiencia del proceso numérico.

2.7. Región de estabilidad de los métodos

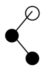
A la hora de calcular un paso del método numérico con el problema de testeo (1.48) de Dahlquish, con el objeto de analizar la región de estabilidad (lineal) para los métodos Runge-Kutta embebidos con estimación del error global, hay que sustituir $f(Y_i)$ por λY_i en (2.13) y en (2.14) y hacer recursivamente lo mismo para cada Y_i dado por (2.15). Con estas sustituciones obtenemos la siguiente expresión

$$\begin{pmatrix} y_{n+1} \\ \bar{y}_{n+1} \end{pmatrix} = \begin{pmatrix} R_{11}(z) & R_{12}(z) \\ R_{21}(z) & R_{22}(z) \end{pmatrix} \begin{pmatrix} y_n \\ \bar{y}_n \end{pmatrix}$$

donde $z = h\lambda$ y

$$R_{11}(z) = \quad 1 + \quad z \sum_i b_i \mu_i + z^2 \sum_i b_i \sum_j a_{ij} \mu_j + \cdots +$$

$$
\begin{aligned}
& +z^{\bar{s}}\underbrace{\sum_i b_i \sum_j a_{ij}\cdots\sum_l^m a_{lm}\mu_m}_{\bar{s}}, \\
R_{12}(z) = \quad & z\sum_i b_i(1-\mu_i) + z^2\sum_i b_i\sum_j a_{ij}(1-\mu_i) + \cdots + \\
& +z^{\bar{s}}\underbrace{\sum_i b_i \sum_j a_{ij}\cdots\sum_l^m a_{lm}(1-\mu_m)}_{\bar{s}}, \\
R_{21}(z) = \quad & z\sum_i \bar{b}_i\mu_i + z^2\sum_i \bar{b}_i\sum_j a_{ij}\mu_j + \cdots + \\
& +z^{\bar{s}}\underbrace{\sum_i \bar{b}_i \sum_j a_{ij}\cdots\sum_l^m a_{lm}\mu_m}_{\bar{s}}, \\
R_{22}(z) = \quad 1+ \quad & z\sum_i \bar{b}_i(1-\mu_i) + z^2\sum_i \bar{b}_i\sum_j a_{ij}(1-\mu_i) + \cdots + \\
& +z^{\bar{s}}\underbrace{\sum_i \bar{b}_i \sum_j a_{ij}\cdots\sum_l^m a_{lm}(1-\mu_m)}_{\bar{s}}.
\end{aligned}
$$

Los coeficientes asociados a z^i son los valores $\psi(t)$ definidos en la sección 2.5 (cuyo valor obtenemos mediante la recursión (2.21) y que mostramos en la Tabla 2.1) correspondientes a los árboles t sin ramificaciones con una hoja blanca:  ,  ...

Por otra parte, si sumamos $R_{11}(z)+R_{12}(z)$ (o en su caso $R_{21}(z)+R_{22}(z)$) obtenemos la función de estabilidad del método Runge-Kutta subyaciente $\psi_h(y)$ (respectivamente $\bar{\psi}_h(y)$), definida en la Sección 1.11, en la que los coeficientes de z^i son los valores que toma la función $\psi(t_i)$, definida en (1.26) y cuyo valor se obtiene mediante la recursión (1.34), donde t_i es el árbol con i nodos que no tiene ramificaciones ni nodos blancos.

La región de estabilidad lineal es aquella para la que los autovalores de la matriz 2×2

$$
M(z) = \begin{pmatrix} R_{11}(z) & R_{12}(z) \\ R_{21}(z) & R_{22}(z) \end{pmatrix} \tag{2.26}
$$

son en módulo menores que o iguales a 1.

2.8. Representación binaria de los árboles

De la equiparación de (2.20) con (2.22) se obtienen las condiciones necesarias para definir el conjunto $\mathcal{T}_*$, y a su vez, hemos obtenido las recursiones que nos posibilitan la obtención de las condiciones sobre los parámetros del método que corresponden a cada uno de los elementos de $\mathcal{T}_*$. No obstante, necesitamos algún método para construir y representar el conjunto de los arboles $\mathcal{T}_*$.

En [17] Murua muestra una forma de representar y construir el conjunto de árboles $\mathcal{T}_*$. En ella se define la *descomposición estándar* de los árboles, más concretamente de una generalización de árboles llamados *N-Trees*, que son árboles que admiten N tipos de nodos en su estructura. En nuestro caso, admitimos dos tipos de nodos en un árbol $t \in \mathcal{T}_*$, los nodos blancos y negros, pero en la raíz solo admitimos los nodos negros, excepto el árbol con un único nodo blanco $\circ$. Además de la descomposición estándar establece una relación de orden entre los N-árboles, y proporciona un algoritmo para la construcción de la tabla correspondiente a la descomposición estándar para los árboles desde el orden 1 hasta cualquier orden $p_{\max}$ dado. En nuestro caso, para un árbol t hay que distinguir entre el orden del árbol $\rho(t)$ y el número de vértices del árbol $|t|$. El número de vértices de un árbol viene dado por:

- $|t| = 1$ si $t = \bullet$ o $t = \circ$,
- $|t| = 1 + |t_1| + \ldots + |t_m|$ si $t = [t_1 \cdots t_m]$.

Como hemos dicho, las hojas blancas no influyen en el orden del árbol, pero en [17] Murua supone que ambos coinciden. No obstante el algoritmo de construcción de árboles nos sirve igualmente, ya que aunque se base en el número de vértices podemos obtener todos los árboles en función de ese número en vez de obtenerlos en función del orden.

Para poder definir la descomposición estándar de los árboles necesitamos definir una operación binaria entre dos árboles, a menudo conocida como el producto de Butcher en la literatura del análisis numérico de ecuaciones diferenciales ordinarias:

Definición 4 *Dados $t_1, t_2 \in \mathcal{T}_*$, el producto de Butcher $t_1 \cdot t_2 \in \mathcal{T}_*$ se define como:*

- $t_1 \cdot t_2 = [t_2]$ *si* $t_1 = [\emptyset]$,
- $t_1 \cdot t_2 = [t_1' \cdots t_m' t_2]$ *si* $t_1 = [t_1' \cdots t_m']$.

Obsérvese que los elementos $t = [t_1 \cdots t_m]$ de $\mathcal{T}_*$ los definimos en función de sus componentes $t_1, \ldots, t_m$, y no importa cómo estén ordenadas las componentes, es decir, son tuplas no ordenadas, por lo que el producto de Butcher cumple la propiedad $(t_1 \cdot t_2) \cdot t_3 = (t_1 \cdot t_3) \cdot t_2$.

El producto de Butcher nos posibilita la descomposición de un árbol $t \neq [\emptyset]$ en dos árboles (t_a, t_b) de forma que $t = t_a \cdot t_b$. Sin embargo, dicha descomposición no es única. Para establecer una descomposición única, nos basaremos en una relación de orden total del conjunto $\mathcal{T}_*$.

Sea $\mathcal{T}_*^i = \{t \in \mathcal{T}_* \quad / \quad |t| = i\}$, suponiendo que tenemos ordenado el conjunto $\mathcal{T}_*^{r-} = \mathcal{T}_*^1 \cup \mathcal{T}_*^2 \cup \ldots \cup \mathcal{T}_*^{r-1}$ (con $r > 1$) de tal forma que $t_a < t_b$ si $|t_a| < |t_b|$,

- llamaremos *descomposición estándar* de $t = [t_1 t_2 \cdots t_m] \in \mathcal{T}_*^r$ con $t_1 \leq t_2 \leq \ldots \leq t_m$ al par $\mathrm{dec}(t) = (\mathrm{dec}_1(t), \mathrm{dec}_2(t))$ donde $\mathrm{dec}_1(t) = [t_1 t_2 \cdots t_{m-1}]$ y $\mathrm{dec}_2(t) = t_m$.

- y a su vez, dados $t_1, t_2 \in \mathcal{T}_*^r$, si su descomposición estándar es $t_1 = (t_{1a}, t_{1b})$ y $t_2 = (t_{2a}, t_{2b})$, diremos que $t_1 < t_2$ si se cumple una de estas dos condiciones:

$$t_{1a} < t_{2a}, \tag{2.27}$$

$$t_{1a} = t_{2a} \text{ y } t_{1b} < t_{2b}. \tag{2.28}$$

- Finalmente, los dos elementos de $\mathcal{T}_*^1 = \{\bullet, \circ\}$ no se pueden descomponer, y establecemos el orden $\bullet < \circ$, con lo que completamos la definición de la relación de orden.

La definición de *descomposición estándar* junto con la relación de orden ofrecen una forma simple de representar el conjunto de árboles $\mathcal{T}_*$. Identificamos cada árbol de $\mathcal{T}_*$ con un número positivo en función de la relación de orden establecida; empezando desde 1 para $\bullet$ y 2 para $\circ$, asociamos a cada árbol el número que le corresponde a su posición según el orden establecido por la relación de orden definida junto con la descomposición estándar.

A la hora de construir los árboles de cierto número de vértices p hay que tener en cuenta que la descomposición estándar del árbol t obliga a que el par $(\mathrm{dec}_1(t), \mathrm{dec}_2(t))$ esté compuesto por dos árboles de número de vértices menor que p, pero cuya suma sea igual a p. Además, al ser $\mathrm{dec}_2(t)$ el máximo posible, obliga a que los posibles números que puedan combinarse con un $\mathrm{dec}_1(t)$ dado sean mayores o iguales a $\mathrm{dec}_2(\mathrm{dec}_1(t))$. Estas dos condiciones nos van a guiar a la hora de construir los árboles, empezando desde el árbol $\bullet = 1$ y $\circ = 2$ que no pueden descomponerse.

Para obtener todos los árboles con un número de vértices p dado, el algoritmo a seguir es bastante simple: supongamos que tenemos construidos de forma ordenada según la relación de orden dada todos los árboles con menos vértices que p, y supongamos a su vez que tenemos el número del primer árbol de cada número de vértices $q < p$, que denotaremos como first(q). Por tanto, el siguiente árbol que debemos generar será first(p), y todos esos árboles que ya hemos generado están numerados desde 1 hasta first$(p) - 1$. Por ejemplo, para el caso de que queramos crear los árboles de 5 vértices, partimos de que tenemos los árboles de menos de 5 vértices, numerados desde 1 hasta 22 (ver Tabla 2.2) y tenemos que first$(1) = 1$, first$(2) = 3$, first$(3) = 5$, y first$(4) = 10$ y el siguiente a generar será first$(5) = 23$.

Para generar los árboles de p vértices en orden ascendente, debemos tener en cuenta primeramente, la condición (2.27) de la relación de orden dada, por lo que los primeros árboles a generar serán los que tengan $\mathrm{dec}_1(t)$ mínimo, es decir debemos empezar desde 1 e ir subiendo hasta $p - 1$. En nuestro ejemplo, empezaremos a combinar los árboles de un vértice con los de 4 vértices, luego los de 2 con los de 3, los de 3 con los de 2 y terminaremos combinando los de 4 vértices con los de 1 único vértice.

Pero entre los árboles de p vértices con $\mathrm{dec}_1(t)$ dado, el orden lo establece la condición (2.28), por lo que para generar estos árboles habrá que empezar desde el mínimo $\mathrm{dec}_2(t)$ posible. Como hemos dicho, el mínimo $\mathrm{dec}_2(t)$ posible debe ser mayor o igual a $\mathrm{dec}_2(\mathrm{dec}_1(t))$, de ahí que para construirlos debamos empezar desde el mayor entre $\mathrm{dec}_2(\mathrm{dec}_1(t))$ y first$(p - q)$. Siguiendo con el ejemplo, supongamos que estamos tratando de generar los árboles t que se obtienen de combinar el árbol  de 4 vértices, cuya descomposición estándar es (3,3) y su identificador es el número 15, con los árboles de un vértice. En ese caso, $\mathrm{dec}_2(t)$ debería ser como mínimo 3, ya que $\mathrm{dec}_2(15) = 3$ y first$(5 - 4) = 1$, pero no hay árboles de un vértice mayores o iguales a 3, por lo que no habrá árboles t de 5 vértices cuya descomposición estándar tenga $\mathrm{dec}_1(t) = 15$.

Por otra parte, no admitimos los árboles que tengan vértices internos blancos. Para saber si un árbol t es aceptable o no podemos utilizar un test que se asegure de que tanto $\mathrm{dec}_1(t)$ como $\mathrm{dec}_2(t)$ sean aceptables: para ello diremos que un árbol es de tipo 1 si su raíz es un nodo negro, y el árbol es de tipo 2 si tiene raíz blanca. Si queremos aceptar árboles sin vértices internos blancos, basta con que $\mathrm{dec}_1(t)$ sea siempre del tipo 1, lo que hará que el árbol resultante tenga raíz negra (la raíz del resultado de la multiplicación de Butcher de dos árboles el la raíz del primer árbol) mientras que para $\mathrm{dec}_2(t)$ tenemos dos posibilidades, la primera es que sea un vértice blanco

sin nada más $\mathrm{dec}_2(t) = 2$ (estaríamos añadiendo una hoja blanca unida a la raíz al árbol $\mathrm{dec}_1(t)$), y la segunda es que sea un árbol de tipo 1 (y su descomposición estándar volverá a tener como primer elemento un árbol de tipo 1).

Tenemos por un lado el test que indica si un árbol es válido o no:

```
int test(int u1, int u2)
{
if (tipo[u1] == 1) && ((tipo[u2] == 1)||(u2 == 2))
      return(1);
  else return(0);
}
```

Y por otro lado el código que genera las tablas con la información de cada árbol:

- numero de vértices que tiene,
- orden al que corresponde,
- su descomposición estándar (dec1 y dec2)
- tipo (siempre 1 a excepción del árbol 2)

Además, genera una tabla, la tabla *first*, donde para cada índice i guarda el primer árbol con i vértices.

```
first[1] = 1;
numvert[1] = 1;
numvert[2] = 1;
tipo[1]= 1;
tipo[2]= 2;
dec_1[1]=1; dec_2[1]=0;
dec_1[2]=2; dec_2[2]=0;
orden[1] = 1;
orden[2] = 0;
siguiente = 3;
for (p = 2; p < p_max; p++) {// árboles con p vértices
   first[p]= siguiente;
   for (q = 1; q < p; q++) {
      //trataremos de combinar todos los árboles con i vértices
      for (u1 = first[q]; u1 <  first[q+1]; q++) {
```

```
        u2min= max(first[p-q],dec_2[u1]);
        for (u2 = u2min; u2 < first[p-q+1]; u2++) {
           if (test(u1,u2)) {
              numvert[siguiente] = p; // = q + p-q;
              tipo[siguiente] = tipo[u1];  // = 1;
              dec_1[siguiente]= u1;
              dec_2[siguiente]= u2;
              orden[siguiente] = orden[u1]+orden[u2];
              siguiente ++;
              } //if
           } //  for (u2...
        } // for (u1...
     } // for (q...
  } // for (p...
```

Con este algoritmo podemos obtener los árboles junto con su descomposición y la información sobre el número de vértices, $\rho(t)$ y el tipo del árbol. En la Tabla 2.2 vemos los resultados que obtiene el código para los árboles con menos de 5 vértices. Podemos ver a su vez, cómo entre los árboles de 4 vértices aparecen tanto árboles de orden cuatro como de tres, dos e incluso orden 1, y si siguiéramos adelante con el proceso de obtención de árboles veríamos que entre los árboles de n vértices siempre hay árboles de orden 1 hasta n; el número de hojas blancas reduce el orden del árbol respecto al número de vértices del árbol, y siempre podemos crear el árbol con raíz negra y el resto de vértices pueden ser hojas blancas que salen de la raíz. Todo ello nos indica que para este tipo de métodos el desarrollo en serie (2.20) tiene un número infinito de elementos para cada potencia de $h > 0$.

2.8.1. Condiciones de orden con la representación binaria de los árboles

En la sección 2.5 hemos dado a conocer las expansiones en potencias de h de $\psi_h(y, y+e)$ y de $\bar{\psi}_h(y+e, y)$ mediante (2.20) y (2.24) respectivamente, y en dichas expansiones la recursión para obtener los coeficientes $\psi(t)$ de cada árbol vienen dadas por (2.21). Así mismo, se han dado las recursiones para obtener $\sigma(t)$, $\rho(t)$ y $F(t)(y, e)$. Todas las recursiones mencionadas se basan en la forma de representar los árboles mediante la tupla de subárboles que quedan al quitar la raíz del árbol. Pero si representamos los árboles mediante su descomposición estándar, conviene que podamos obtener todos los valores que aparecen en (2.20) basándonos en esa descomposición.

Tabla 2.2: Descomposición estándar de los árboles de menos de 5 vértices

árbol	descomposición	tipo	N° vértices	$\rho(t)$	gráfica
1	(1, 0) $\bullet \cdot \emptyset$	1	1	1	
2	(2, 0) $\circ \cdot \emptyset$	2	1	0	
3	(1, 1) $\bullet \cdot \bullet$	1	2	2	
4	(1, 2) $\bullet \cdot \circ$	1	2	1	
5	(1, 3) $\bullet \cdot$	1	3	3	
6	(1, 4) $\bullet \cdot$	1	3	2	
7	(3, 1) $\cdot \bullet$	1	3	3	
8	(3, 2) $\cdot \circ$	1	3	2	
9	(4, 2) $\cdot \circ$	1	3	1	
10	(1, 5) $\bullet \cdot$	1	4	4	
11	(1, 6) $\bullet \cdot$	1	4	3	
12	(1, 7) $\bullet \cdot$	1	4	4	
13	(1, 8) $\bullet \cdot$	1	4	3	
14	(1, 9) $\bullet \cdot$	1	4	2	
15	(3, 3) $\cdot \cdot$	1	4	4	
16	(3, 4) $\cdot \cdot$	1	4	3	
17	(4, 3) $\cdot \cdot$	1	4	3	
18	(4, 4) $\cdot \cdot$	1	4	2	
19	(7, 1) $\cdot \bullet$	1	4	4	
20	(7, 2) $\cdot \circ$	1	4	3	
21	(8, 2) $\cdot \circ$	1	4	2	
22	(9, 2) $\cdot \circ$	1	4	1	

El orden y el número de vértices de cada elemento se pueden obtener de las recursiones

$$\begin{aligned}
&\rho(\bullet) = 1, \rho(\circ) = 0,\\
&\rho(t) = \rho(\mathrm{dec}_1(t)) + \rho(\mathrm{dec}_2(t))\\
&|\bullet| = 1, |\circ| = 1,\\
&|u| = |\mathrm{dec}_1(t)| + |\mathrm{dec}_2(t)|
\end{aligned}$$

En cuanto a los valores de $\phi(t)$ de (2.22), y de $\psi(t)$ de (2.20) tenemos por un lado los casos triviales:

$$\begin{aligned}
\psi(\bullet) &= \sum_{i=1}^{s} b_i,\\
\psi(\circ) &= 0\\
\phi(\bullet) &= 1,\\
\phi(\circ) &= 0.
\end{aligned}$$

mientras que para los casos generales hemos de mirar el origen de la recursiones dadas en (2.22) y en (2.20) para $t = [t_1 \cdots t_m]$. Por la aplicación del Lema 1, tenemos que a cada árbol $t \in \mathcal{T}_*$ le corresponde una tupla $u = t_1 t_2 \cdots t_m \in \mathcal{F}_*$. Supongamos sin perder generalidad que $t_1 \leq t_2 \leq \ldots \leq t_m$ por lo que la descomposición estándar del árbol t es $t = ([t_1 \cdots t_{m-1}], t_m)$. Si continuáramos descomponiendo recursivamente la parte izquierda de la descomposición llegaríamos a

$$t = \underbrace{(\cdots(}_{m} 1, t_1), t_2), \cdots, t_m).$$

Teniendo en cuenta esta descomposición de t, junto con la recursión dada para $\psi(t)$ en (2.21)

$$\psi(t) = \sum_{i=1}^{s} b_i \psi_i(t_1) \cdots \psi_i(t_m),$$

la recursión correspondiente a $\psi_i(t)$ dada en (2.21)

$$\psi_i(t) = \sum_{j=1}^{s} a_{ij} \psi_j(t_1) \cdots \psi_j(t_m),$$

y la recursión de $\phi(t)$ dada en (1.39)

$$\phi(t) = \frac{1}{\rho(t)} \phi'(t) = \frac{1}{\rho(t)} \phi(t_1) \cdots \phi(t_m),$$

podemos establecer las siguientes recursiones para $\psi(t)$ y para $\phi(t)$:

$$\begin{aligned}
&\hat{\psi}_i'(\bullet) = 1,\\
&\hat{\psi}_i(\bullet) = \sum_{j=1}^{s} a_{ij},\\
&\hat{\psi}_i(\circ) = \mu_i,\\
&\hat{\psi}_i'(t) = \hat{\psi}_i'(\mathrm{dec}_1(t))\hat{\psi}_i(\mathrm{dec}_2(t)),\ 1 \le i \le s,\\
&\hat{\psi}_i(t) = \sum_{j=1}^{s} a_{ij}\hat{\psi}_j'(t),\ 1 \le i \le s,\\
&\psi(t) = \sum_{i=1}^{s} b_i\hat{\psi}_i'(t),\\
&\phi'(\bullet) = 1,\\
&\phi'(t) = \phi'(\mathrm{dec}_1(t))\phi(\mathrm{dec}_2(t)),\\
&\phi(t) = \frac{\phi'(t)}{\rho(t)}.
\end{aligned}$$

2.9. Construcción de un método de 7 etapas

Para ver si los esquemas generales globalmente embebidos (2.13)–(2.15) pueden realmente ser tan eficientes como los métodos de Runge-Kutta estándares, hemos construido un método de orden 5 (más precisamente $\bar{s} = 8$, $p = 5$, $\bar{p} = 4$, $q = 2$, $\bar{q} = 1$) basándonos en un método de RK muy eficiente, a saber, el método de RK explícito de orden 5 de Bogacki y Shampine [2] implementado en el código RKSUITE [8]. Nos referiremos a éste método como BSRK5.

Hemos determinado los coeficientes b_i $(1 \le i \le 7)$, y a_{ij} $(1 \le i, j \le 7)$ de tal forma que el método subyaciente de orden 5 sea el esquema BSRK5. Al igual que es típico en la construcción de esquemas de Runge-Kutta localmente embebidos, hemos hecho que y_{n-1} y y_n sean respectivamente la primera y última etapa del esquema (2.13)–(2.15) (FSAL: First Same As Last). Ello se consigue haciendo que $a_{8i} = b_i$ $(1 \le i \le 7)$, $\mu_1 = 1$ y $\mu_8 = 1$. Por tanto, aunque el método resultante sea formalmente de ocho etapas, requiere únicamente siete evaluaciones de $f(y)$ en cada paso.

El resto de parámetros, es decir, $\bar{b}_i$ $(1 \le i \le 8)$ y μ_i $(2 \le i \le 7)$, se han elegido de tal forma que el método de RK subyaciente $\bar{\psi}_h(y) = \bar{\psi}_h(y, y)$

sea de orden 4 y las condiciones de independencia (2.9)–(2.10) se satisfagan para $\bar{q} = 1$ y $q = 2$.

Para que $\bar{\psi}_h(y, y+e) = \bar{\psi}_h(y, y) + O(h^2||e|| + h||e||^2)$, se debe cumplir que en la expansión (2.24) el coeficiente del árbol  se anule, ya que es el único árbol con término proporcional a $h\|e\|$.

Por otro lado, para que $\psi_h(y+e, y) = \psi_h(y, y) + O(h^3||e|| + h||e||^2)$ en la expansión (2.20), se deben anular los coeficientes de , y . Los dos últimos arboles son los que aparecen multiplicando a $h^2\|e\|$. Por tanto, las condiciones de independencia conllevan que se tengan que cumplir las siguientes cuatro condiciones:

$$\sum_i \bar{b}_i \mu_i = 0 \tag{2.29}$$

$$\sum_i b_i (1 - \mu_i) = 0, \tag{2.30}$$

$$\sum_i b_i c_i (1 - \mu_i) = 0, \tag{2.31}$$

$$\sum_i b_i \sum_j a_{ij} (1 - \mu_j) = 0. \tag{2.32}$$

No obstante, los coeficientes del método BSRK5 cumplen ciertas propiedades, que como veremos seguidamente, hacen que las condiciones (2.29)-(2.32) no sean independientes entre sí. En la construcción de métodos explícitos de Runge-Kutta es usual que se impongan ciertas condiciones que simplifican las condiciones de orden de los métodos. Seguidamente veremos que las condiciones simplificadoras que verifica el esquema BSRK5 simplifican a su vez las condiciones requeridas para nuestro método embebido.

2.9.1. Aplicación de las propiedades de BSRK5 a las condiciones del método $\bar{\psi}_h(y, \bar{y})$

A la hora de construir métodos de Runge-Kutta explícitos se deben establecer los valores que toman las variables que aparecen en el tablero de Butcher del método a construir, y dependiendo del orden del método, los valores del tablero deberán cumplir más o menos condiciones. Estas condiciones son las que surgen al igualar elementos de los desarrollos en serie de potencias de h de la solución numérica ψ_h y los elementos del flujo exacto ϕ_h, dadas en (1.26).

En la Sección 1.8 hemos explicado la forma de obtener la condición correspondiente a cada uno de los árboles. No obstante, se pueden estable-

cer condiciones generales, no asociadas a un único árbol, de forma que las condiciones derivadas individualmente de cada árbol se simplifiquen entre sí, haciendo que algunas de ellas sean redundantes. Estas simplificaciones son utilizadas muy ampliamente en la construcción de familias de métodos de Runge-Kutta.

Una de las ventajas que ofrecen las simplificaciones es que las expresiones que surgen en los sistemas de ecuaciones resultantes son mucho más manejables, y por otro lado, nos posibilitan la obtención de los valores del tablero de Butcher de forma escalonada.

Las propiedades que vamos a utilizar en la simplificación son las siguientes:

$$\sum_{i=1}^{s} b_i a_{ij} = b_j(1-c_j), \quad j = 1, \ldots, s \tag{2.33}$$

$$\sum_{j=1}^{s} a_{ij} c_j - \frac{c_i^2}{2} = 0, \quad i = 3, \ldots, s \tag{2.34}$$

$$\sum_{j=1}^{s} a_{ij} c_j - \frac{c_i^3}{3} = 0, \quad i = 3, \ldots, s \tag{2.35}$$

En particular, el esquema de Runge-Kutta explícito BSRK5 de Bogacki y Shampine [2] en el que basamos la construcción de nuestro esquema globalmente embebido cumple las condiciones (2.33)-(2.35).

La propiedad dada en (2.33) permite escribir la condición (2.32) como

$$\sum_{j=1}^{s} b_j(1-c_j)(1-\mu_j) = \sum_i b_i(1-\mu_i) - \sum_i b_i c_i(1-\mu_i) = 0$$

Esto implica que si se cumplen (2.30) y (2.31) también se cumple (2.32).

Para que el método subyaciente $\bar{\psi}_h(y, y)$ sea de orden 4, deben cumplirse las condiciones que surgen de los árboles con raíz y vértices negros con cuatro o menos vértices, es decir, han de cumplirse las 8 condiciones que obtenemos al igualar $\bar{\psi}(t)$ a $\phi(t)$ para los árboles de la Tabla 1.1, donde $\bar{\psi}(t)$ y $\phi(t)$ representan los coeficientes correspondientes al árbol t en las expansiones en potencias de h de $\bar{\psi}_h(y)$ y de $\phi_h(y)$ respectivamente, y que hemos dado en (1.37) y en (1.39). Sin embargo, podemos reducir el número de condiciones gracias a las propiedades (2.33)–(2.35) que cumple el esquema BSRK5. Supongamos que las condiciones asociadas a  ,  ,  y  se cumplen, es decir,

$$\bar{\psi}(\bullet) = 1,\ \bar{\psi}(\bullet\!\!-\!\!\bullet) = \frac{1}{2},\ \bar{\psi}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) = \frac{1}{3} \ \text{ y } \ \bar{\psi}(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) = \frac{1}{4}.$$

Vamos a demostrar que las condiciones $\bar{\psi}(t) = \phi(t)$ asociadas a los árboles $t =$  se reducen a $\bar{b}_2 = 0$ y $\sum_i \bar{b}_i a_{i2} = 0$.

1. Teniendo en cuenta que la condición asociada a  se cumple, podemos reescribir
$$\bar{\psi}(\text{}) = \phi(\text{}) = \frac{1}{6}$$
como
$$\left(\bar{\psi}(\text{}) - \frac{1}{6}\right) - \frac{1}{2}\left(\bar{\psi}(\text{  }) - \frac{1}{3}\right) = 0,$$
que equivale a
$$\sum_i \bar{b}_i (\sum_j a_{ij} c_j - \frac{c_i^2}{2}) = 0. \tag{2.36}$$
Sabemos que el esquema BSRK5 cumple la propiedad dada en (2.34) para todo $i > 2$, por lo que todos los sumandos de la expresión (2.36) se anulan, excepto la correspondiente a $i = 2$, es decir, (2.36) se reduce a
$$-\bar{b}_2 \frac{c_2^2}{2} = 0.$$
Por tanto, basta que $\bar{b}_2 = 0$ para que se cumpla la condición asociada al árbol  .

2. Podemos reescribir igualmente la condición asociada a  como
$$\left(\bar{\psi}(\text{  }) - \frac{1}{8}\right) - \frac{1}{2}\left(\bar{\psi}(\text{  }) - \frac{1}{4}\right) = 0,$$
o de forma equivalente ,
$$\sum_i \bar{b}_i c_i (\sum_j a_{ij} c_j - \frac{c_i^2}{2}) = 0,$$
que vuelve a reducirse a la condición $\bar{b}_2 = 0$.

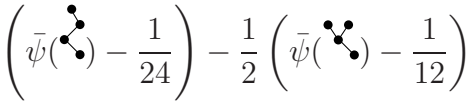
3. El caso de  lo podemos tratar como
$$\left(\bar{\psi}(\text{  }) - \frac{1}{12}\right) - \frac{1}{3}\left(\bar{\psi}(\text{  }) - \frac{1}{4}\right) = 0,$$

que se puede reescribir como

$$\sum_i \bar{b}_i (\sum_j a_{ij} c_j^2 - \frac{c_i^3}{3}) = 0. \tag{2.37}$$

En este caso hay que tener en cuenta que el esquema BSRK5 cumple (2.35) para $i > 2$, por lo que dicha condición se reduce a $\bar{b}_2 = 0$.

4. Para  , consideramos

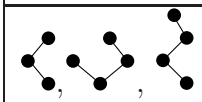


es decir,

$$\sum_i \bar{b}_i \sum_j a_{ij} (\sum_k a_{jk} c_k - \frac{c_j^2}{2}) = 0,$$

y, teniendo en cuenta (2.34), basta con que se cumpla $\sum_i \bar{b}_i a_{i2} = 0$.

Por tanto, teniendo en cuenta que el método subyaciente $\psi_h(y, y)$ es el método BSRK5, las condiciones para que el método subyaciente $\bar{\psi}_h(y, y)$, sea de orden 4 son:

	$\sum_i \bar{b}_i = 1$
	$\sum_i \bar{b}_i c_i = \frac{1}{2}$
	$\sum_i \bar{b}_i c_i^2 = \frac{1}{3}$
	$\sum_i \bar{b}_i c_i^3 = \frac{1}{4}$
	$\bar{b}_2 = 0$
	$\sum_i \bar{b}_i a_{i2} = 0$

(2.38)

Los parámetros que hay que establecer son $\bar{b}_i$ $(1 \leq i \leq 8)$ y μ_i $(2 \leq i \leq 7)$, en total son 14 parámetros, pero las 6 condiciones de la tabla de arriba junto con las tres condiciones de independencia (2.29), (2.30) y (2.31), nos dejan cinco parámetros libres, y para elegirlos hemos tratado de minimizar una función objetivo, en la que hemos incluido los siguientes elementos:

- Por un lado hemos tratado de minimizar la dependencia entre los dos métodos, para lo que hemos introducido en la función objetivo

el cuadrado de los coeficientes de algunos árboles que inciden en las condiciones de independencia. En concreto, para el método $\bar{\psi}(y, y+h)$ hemos introducido los cuadrados de los coeficientes de los árboles  y  , mientras que para $\psi(y+e, y)$ los cuadrados de los coeficientes de los árboles  y  . Dichos coeficientes toman los siguientes valores:

$$\sum_i \bar{b}_i c_i \mu_i,$$
$$\sum_i \bar{b}_i \sum_j a_{i,j} \mu_j,$$
$$\sum_i b_i c_i^2 (1-\mu_i),$$
$$\sum_i b_i c_i \sum_j a_{i,j}(1-\mu_j)$$

- Por otra parte, hemos tratado de minimizar el efecto de las ecuaciones de los árboles de orden 5 del método Runge-Kutta subyaciente $\bar{\psi}_h(y, y)$ que buscamos. Es decir, hemos incluido en la función objetivo la suma de los cuadrados de los nueve coeficientes de error $\bar{\psi}(t) - \phi(t)$ correspondientes a los nueve árboles t de orden 5.

Por último, hemos tratado de que la región de estabilidad lineal del método incluya un intervalo del eje imaginario centrado en el origen. Para ello, para cada mínimo local obtenido para nuestra función objetivo, hemos desechado los que no cumplen esta condición. Es decir, sean $\lambda_1(z)$ y $\lambda_2(z)$ los valores propios de la matriz de estabilidad $M(z)$ dada en (2.26), queremos que $|\lambda_1(iy)| \leq 1$ y $|\lambda_2(iy)| \leq 1$ para todo $y \in \mathbb{R}$ suficientemente pequeño. Por las condiciones de independencia (2.29)–(2.32), tenemos que

$$R_{12}(z) = O(z^3) \text{ y } R_{21}(z) = O(z^2) \text{ cuando } z \to 0. \tag{2.39}$$

Ello implica que

$$\lambda_1(z) = R_{11}(z) + O(z^5),\ \lambda_2(z) = R_{22}(z) + O(z^5) \text{ cuando } z \to 0.$$

Por otro lado, puesto que en nuestro caso el método de Runge-Kutta subyaciente $\psi_h(y, y)$ (BSRK5) es de orden 5, y el método de Runge-Kutta subyaciente $\bar{\psi}_(y, y)$ es de orden 4, tenemos que

$$R_{11}(z) + R_{12}(z) = e^z + O(z^6), \quad R_{21}(z) + R_{22}(z) = e^z + O(z^5),$$

lo cual implica, teniendo en cuenta (2.39), que

$$R_{11}(z) = 1 + z + \frac{z^2}{2} + \frac{R_{11}^{(3)}(0)}{6}z^3 + \frac{R_{11}^{(4)}(0)}{24}z^4 + O(z^5),$$
$$R_{22}(z) = 1 + z + \frac{R_{22}^{(2)}(0)}{2}z^2 + O(z^3).$$

Por tanto, tenemos para $z = iy$, $y \in \mathbb{R}$, que

$$\lambda_1(iy) = \left(1 - \frac{y^2}{2} + \frac{R_{11}^{(4)}(0)}{24}y^4\right) + i\left(y - \frac{R_{11}^{(3)}(0)}{6}y^3\right) + O(y^5),$$
$$\lambda_2(iy) = \left(1 - \frac{R_{22}^{''}(0)}{2}y^2\right) + iy + O(y^3),$$

cuando $y \to 0$, de modo que

$$|\lambda_1(iy)|^2 = 1 + \left(\frac{1}{4} + \frac{R_{11}^{(4)}(0)}{12} - \frac{R_{11}^{(3)}(0)}{3}\right)y^4 + O(y^5),$$
$$|\lambda_2(iy)|^2 = 1 + \left(1 - R_{22}^{''}(0)\right)y^2 + O(y^3).$$

Tenemos pues, finalmente, que si se verifican las dos desigualdades

$$\begin{aligned} &3 + R_{11}^{(4)}(0) < 4R_{11}^{(3)}(0), \\ &R_{22}^{''}(0) > 1, \end{aligned} \tag{2.40}$$

entonces la región de estabilidad del método incluye un intervalo del eje imaginario centrado en el origen.

Tras una intensiva búsqueda realizada con una rutina numérica de búsqueda de mínimos locales, y tras haber eliminado los métodos que no cumplen las condiciones (2.40), hemos elegido los cinco parámetros libres como:

$$\mu_2 = -\frac{539}{261}, \quad \mu_3 = -\frac{969}{500}, \quad \mu_4 = \frac{621}{101},$$
$$\mu_5 = \frac{2780}{401}, \quad \bar{b}_8 = \frac{-26}{225}.$$

Con lo que el resto de parámetros queda determinada por las tres ecuaciones (2.29)–(2.31) de independencia y por las condiciones dadas en (2.38) que son las necesarias para que el método subyaciente $\bar{\psi}_h(y, y)$ sea de orden 4. Los

valores que toman son:

$$\mu_6 = \frac{4768373937707}{2789060864000}, \quad \mu_7 = -\frac{1492653337169}{294320767000}$$
$$\bar{b}_1 = \frac{272606507613}{3565852942400}, \quad \bar{b}_2 = 0, \quad \bar{b}_3 = \frac{6645196186371}{25350985762375},$$
$$\bar{b}_4 = \frac{84608482815521}{331114916080000}, \quad \bar{b}_5 = -\frac{11356676118237}{89146323560000},$$
$$\bar{b}_6 = \frac{129399657242}{278582261125}, \quad \bar{b}_7 = \frac{1300793}{7056000}.$$

La región de estabilidad que le corresponde al método puede verse en el gráfico inferior izquierdo de la Figura 2.1, y en el gráfico de arriba puede observarse cómo el eje imaginario entra en la región de estabilidad hasta una altura aproximada de $y = 1{,}2$. La región de estabilidad, en comparación con la del método BSRK5, mostrado en el gráfico inferior derecho de la misma figura, no es muy amplia, y en general, los métodos obtenidos en la búsqueda tienen el mismo problema. Hemos buscado métodos con región de estabilidad más amplia, introduciendo en la función objetivo el módulo de valores de la función de estabilidad en ciertos puntos, pero los métodos obtenidos muestran peores resultados en cuanto al efecto en los coeficientes de error de los árboles de orden 5, por lo que hemos optado por conformarnos con una región de estabilidad no muy amplia.

2.10. Experimentos numéricos

Al principio trabajamos con longitud de paso constante (es decir $h_n = h$ para todo n) para poder ver si las expectativas que teníamos se cumplían y para poder calibrar la utilización de estas técnicas a la hora de estimar el error global. En [18] se presentaron los primeros resultados de los experimentos numéricos realizados con longitud de paso constante.

A continuación podemos ver los resultados obtenidos para dos problemas de valor inicial:

1. El primer problema se extrajo de [23]. Se trata de un sistema de dimensión $D = 1$ y el problema de valor inicial se define como

$$y' = \cos(t)y, \quad y(0) = 1, \quad t \in [0, 94{,}6].$$

La solución es $y(t) = e^{\sin(t)}$, una función periódica con periodo 2π. Nos referiremos a este problema como *expsin*.

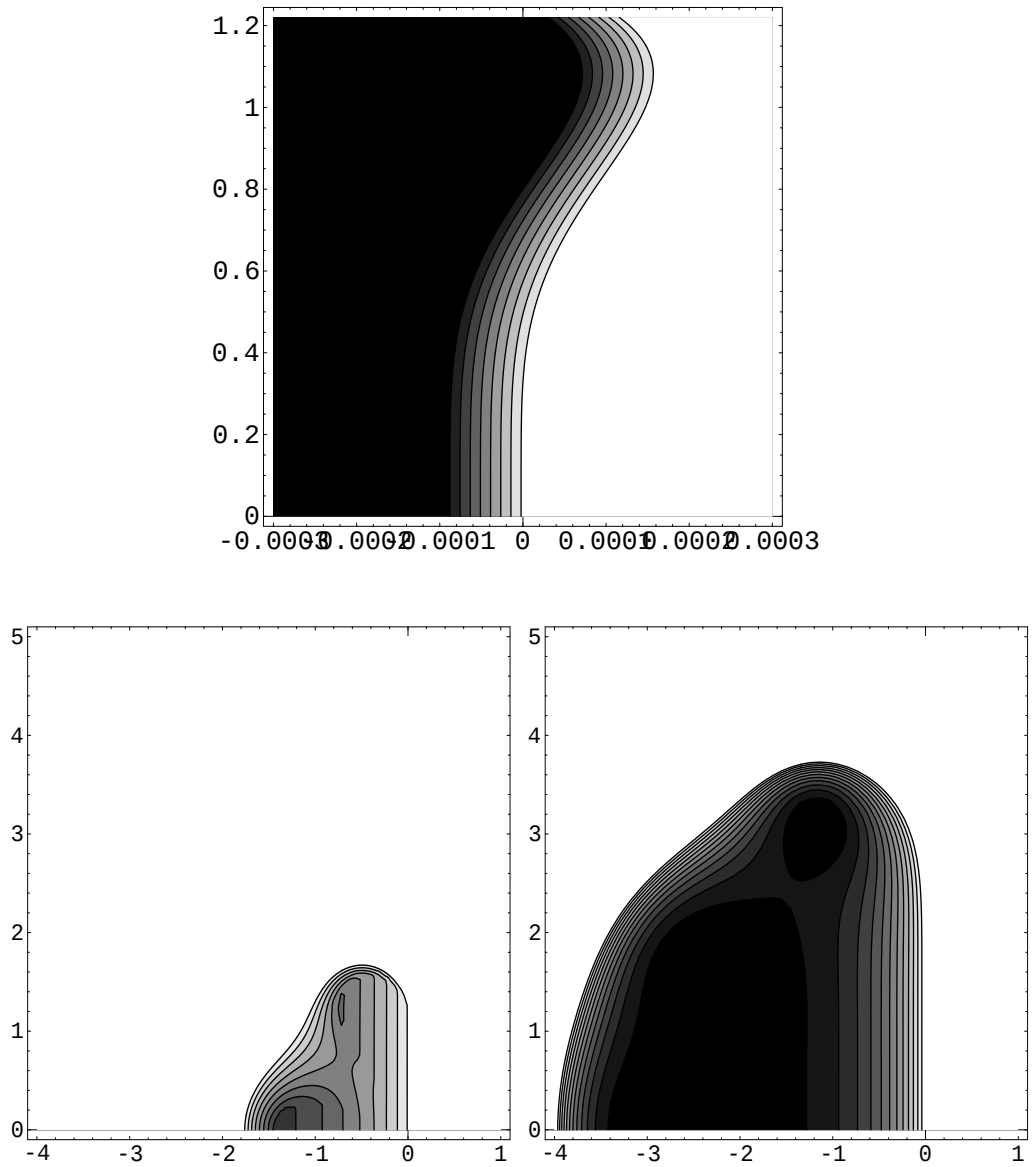


Figura 2.1: Abajo a la izquierda, región de estabilidad del método para la estimación del error global basado en BSRK5. Abajo a la derecha, región de estabilidad del método BSRK5. Arriba, se puede ver que la región del método basado en BSRK5 abarca parte del eje imaginario.

2. El segundo ejemplo, al que llamaremos *Arenstorf*, se trata de un problema de valor inicial de dimensión 4 extraido de [20, pp.129–130] que mostramos en (1.4). Corresponde a una solución periódica del problema restringido de los tres cuerpos.

El objeto de estos experimentos es comprobar en qué medida el método de orden 5(4) creado siguiendo el esquema (2.13–2.15) puede proporcionar estimaciones útiles del error global, y si es tan preciso como el método Runge-Kutta subyaciente de orden 5, es decir, el método BSRK5.

Para cada uno de los problemas y dada una longitud de paso h, mostramos dos gráficas, ambas en escala logarítmica, tanto el eje del tiempo como el eje de la norma $\| \ \|_\infty$ del error: La primera gráfica compara el error global exacto de nuestra solución numérica $\bar{y}_n$ de orden 5 (línea a tramos) con el error global estimado por el proceso numérico (línea continua). En la segunda gráfica se comparan el error global exacto de nuestra solución numérica $\bar{y}_n$ (línea a tramos) con el error global exacto de la solución del método estándar Runge-Kutta subyaciente $\hat{y}_n$ (en este caso el método BSRK5, que aparece con línea continua).

Hemos integrado el ejemplo 'expsin' en el intervalo $[0, 30\pi]$ con longitud de paso constante $h = 2\pi/7$. Las dos gráficas correspondientes a esta integración pueden verse en la Figura 2.2. Los resultados obtenidos son prometedores ya que el error global estimado por el método es mayor que el error global exacto (gráfica de la arriba), tal y como esperábamos para valores de h suficientemente pequeños, y además, nuestro método muestra un comportamiento parecido al BSRK5, es decir, se puede considerar como una pequeña perturbación de la solución BSRK5 (gráfica de abajo). Por todo ello, en este caso, nuestro esquema RK general globalmente embebido ha sido tan eficiente como el método Runge-Kutta subyaciente, y al mismo tiempo, proporciona información útil sobre el error global cometido.

Otro problema con el que hemos realizado experimentos es el problema *Arenstorf*. Hemos integrado el problema sobre un periodo $[0, T]$ ($T =$ 17,0652165...), primero con una longitud de paso $h = T/14000$ (Figura 2.3). Los resultados siguen siendo tan positivos como en el caso de *expsin*.

En un nuevo experimento se ha utilizado una longitud de paso mayor que el anterior, $h = T/3500$ (Figura 2.4), con el objeto de analizar lo que sucede cuando el proceso de integración ofrece resultados completamente erróneos. La primera gráfica, Figura 2.3, muestra que la estimación del error global refleja de forma satisfactoria la propagación del error global verdadero, y que, al igual que con el anterior problema, la solución ofrecida por el esquema es una pequeña perturbación de la solución ofrecida por el método Runge-Kutta subyaciente.

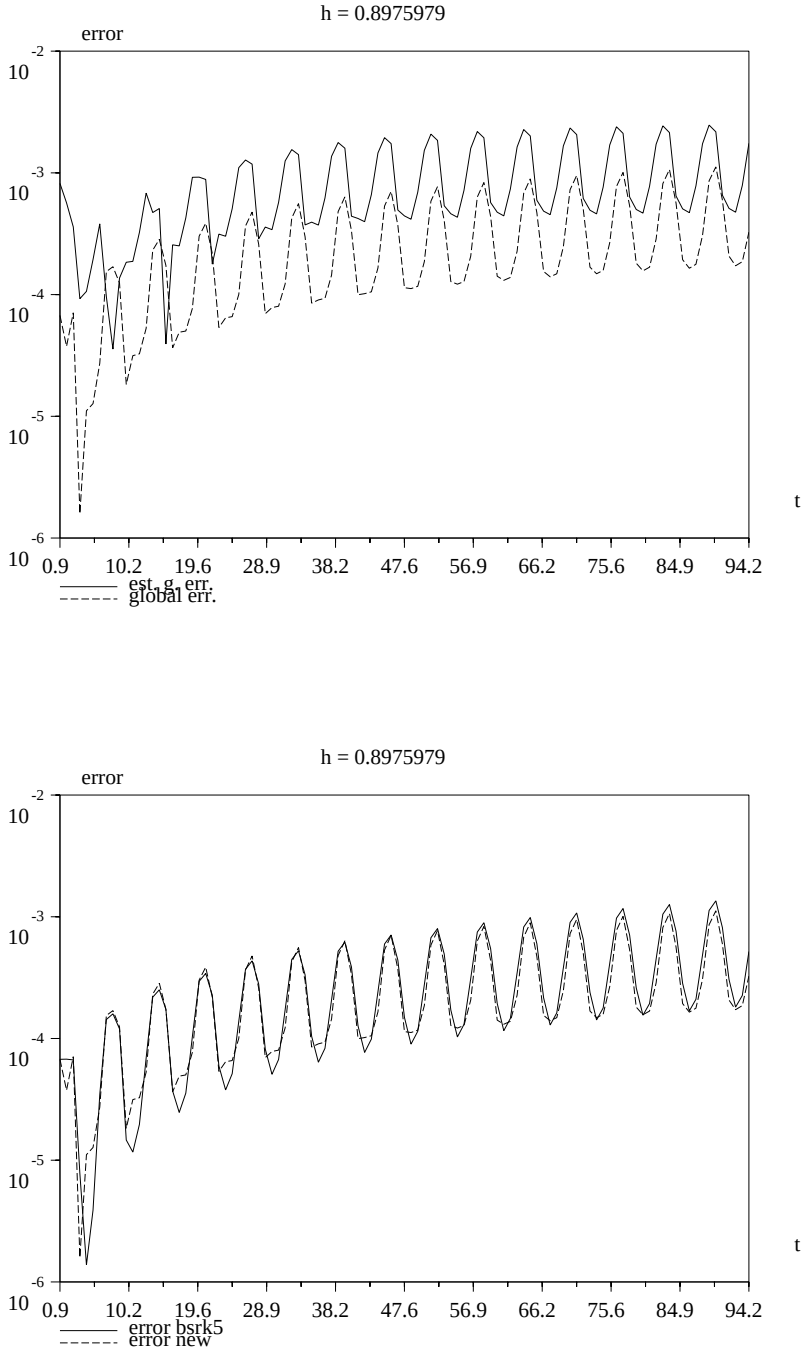


Figura 2.2: Resultados del problema 'expsin' para $h = 2\pi/7$. En la gráfica de arriba comparamos el error global cometido (línea a tramos) con el error global estimado (línea continua), mientras que en la de abajo comparamos el error cometido por nuestro esquema (línea a tramos) con el error cometido por el esquema BSRK5 (línea continua).

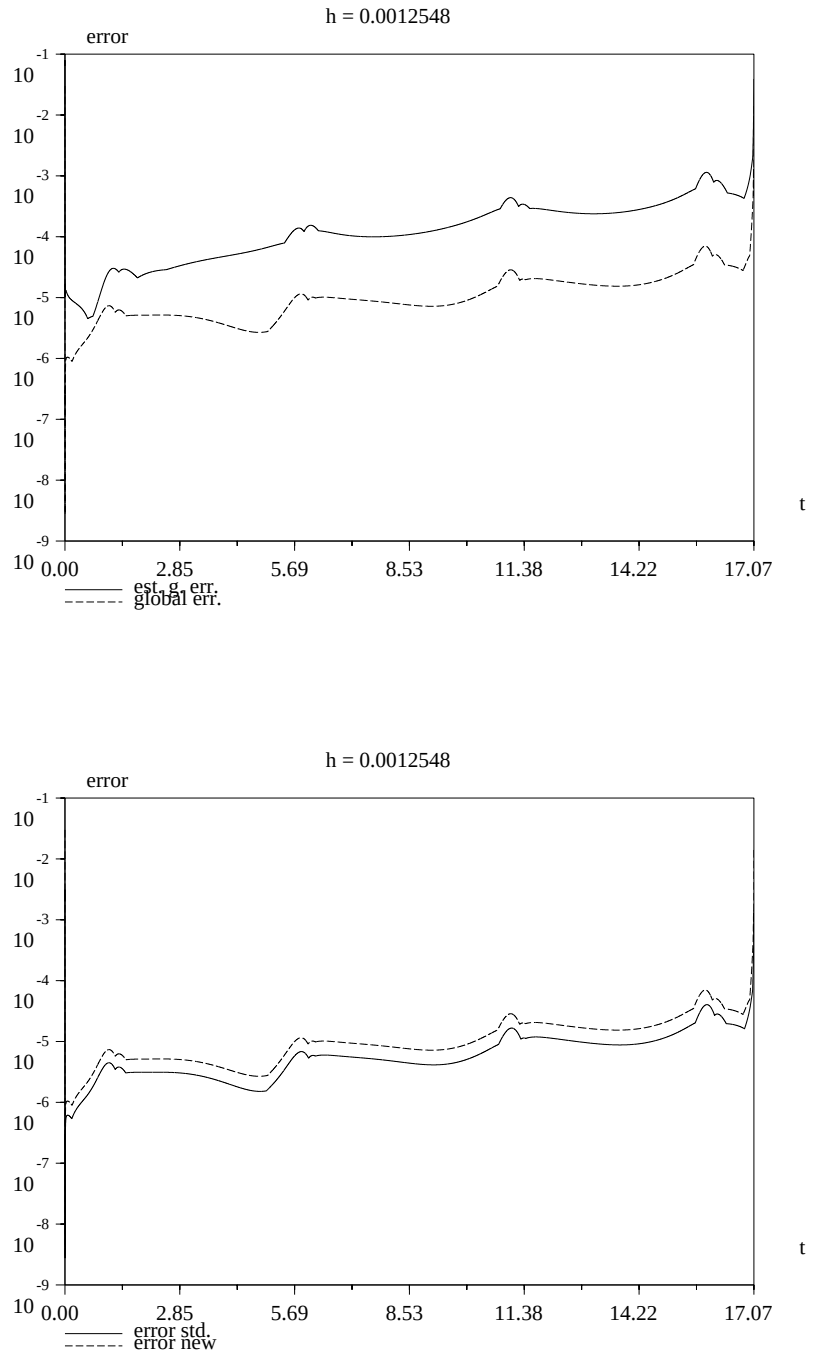


Figura 2.3: Resultados del problema 'Arenstorf' con $h = T/14000$. Arriba, error global (línea a tramos) frente al error global estimado (línea continua). Abajo, comparación del error global del nuevo esquema (línea a tramos) frente al error global del método BSRK5 (línea continua).

No obstante, la segunda integración, Figura 2.4, para el que hemos utilizado una longitud de paso superior, ofrece unos resultados que muestran que la aproximación numérica correspondiente al método de orden 5, $\bar{y}_n$ (curva discontinua de la gráfica de abajo), ofrece valores completamente erróneos antes del final del intervalo de integración, mientras que la solución ofrecida por el método BSRK5, $\hat{y}_n$ (curva continua), es considerablemente mejor. Esto, aparentemente se debe al hecho de que la aproximación de orden 4 y_n se degrada antes que la solución de orden 5 BSRK5 $\hat{y}_n$, de forma que $\tilde{e}_n = y_n - \bar{y}_n$ deja de ser un valor pequeño y despreciable, y por tanto el término $O(h||\tilde{e}_{n-1}||^2)$ en $\bar{\pi}_n$ (Lema 2) domina la propagación del error $\bar{e}_n$.

2.11. Conclusiones de los experimentos

Los experimentos resultan muy esperanzadores, y de hecho continuamos realizando más experimentos, sobre todo en la línea de poder corregir y mejorar los resultados cuando el error empieza a dejar de ser insignificante. La degradación de la solución de orden 4 obliga a reconsiderar la condiciones de independencia y a hacer que la repercusión de la estimación del error global $\tilde{e}_n$ en $\bar{e}_n$ se minimice. Se debe avanzar, por tanto, en dos direcciones: por un lado, hay que extender la utilización de estas técnicas a la integración de problemas con longitud de paso variable, y por otro lado, debemos buscar pares de métodos en los que las condiciones de independencia aseguren que la repercusión de los errores de un método no tengan tanta incidencia en el otro método del par, y así evitar que las soluciones numéricas obtenidas mediante estas técnicas, cuando el error global empieza a ser considerable, se alejen tanto de las soluciones ofrecidas por los métodos subyacientes.

La búsqueda de pares de métodos, en los que las condiciones de independencia aseguren que la degradación de los resultados obtenidos no difieran tanto de los resultados obtenidos por los métodos subyacientes no ha sido muy fructífera en la clase general de esquemas Runge-Kutta embebidos con estimación del error global (2.13)– (2.15).

2.12. Experimentos con longitud de paso variable

La extensión de la aplicación de las nuevas técnicas a una implementación con longitud de paso variable ha sido realizada de forma satisfactoria y los resultados obtenidos han sido publicados en [13].

Para poder aplicar de forma eficiente el nuevo esquema de funcionamiento, hace falta que la longitud de paso sea variable, y las estrategias usuales

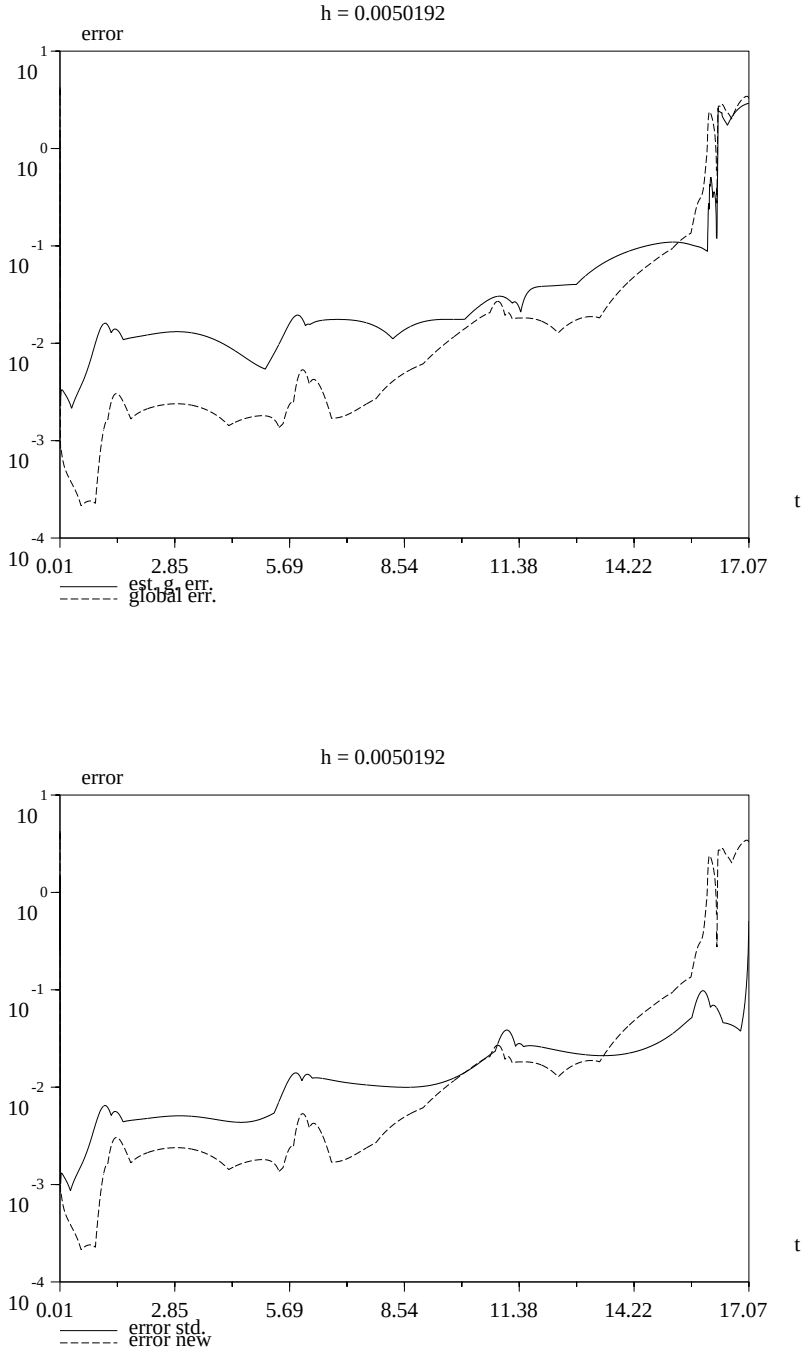


Figura 2.4: Resultados del problema 'Arenstorf' con $h = T/3500$. La mayor tolerancia hace que el error global aumente, y en consecuencia la solución numérica se degenera. Arriba, error global (línea a tramos) frente al error global estimado (línea continua). Abajo, error global del nuevo esquema (línea a tramos) frente al error del método BSRK5 (línea continua).

de adecuación de la longitud de paso requieren una estimación del error local dada como la diferencia entre dos soluciones numéricas (ver [23], pp. 334–340, y [20], pp. 167). Evidentemente, esa estimación no debe ser computacionalmente costosa, por lo que se utilizan los valores intermedios $f(Y_i)$ ya disponibles. En el caso de los nuevos esquemas planteados en (2.13)–(2.15), los valores intermedios dependen tanto de y_{n-1} como de $\overline{y}_{n-1}$, por lo que se ven afectados por el tamaño de la estimación del error global $\tilde{e}_{n-1}$. Este efecto puede provocar ciertas dificultades, parecidas a las que hemos comentado en la sección de los experimentos numéricos, que pueden ser evitadas (a expensas de perder algunos parámetros del método) de la siguiente forma. Si hacemos que se cumplan las siguientes condiciones

$$\mu_i = 1, \qquad i = 1, \ldots, s, \tag{2.41}$$
$$b_i = 0, \qquad i = s+1, \ldots, \overline{s}, \tag{2.42}$$

la transformación ψ_h deja de depender del valor de $\overline{y}$. Realmente ψ_h pasa a ser una transformación Runge-Kutta explícita, perteneciente a la familia definida en (1.9)–(1.10), por lo que la estimación del error local puede realizarse de la forma usual.

El cumplimiento de las condiciones (2.41)–(2.42) supone que nuestro esquema sea de la forma (2.2), por lo que el proceso de obtención del error global se puede interpretar como (2.1), con $\tilde{E}_h(y,h) := \psi_h(y) - \overline{\psi}_h(y, y-\tilde{e})$.

Si además de las condiciones (2.41)–(2.42) se cumplen las siguientes condiciones

$$\overline{b}_i = 0, \qquad i = 1, \ldots, s,$$
$$\mu_i = 0, \qquad i = s+1, \ldots, \overline{s},$$

se obtendría la familia de pares de métodos Runge-Kutta globalmente embebidos de Dormand, Gilmore y Prince [7].

Al restringir la clase general de esquemas Runge-Kutta embebidos con estimación del error global (2.13)–(2.15) mediante la imposición de las condiciones (2.41)–(2.42), hemos obtenido una subclase cuyas características nos permiten el cálculo de la estimación del error local utilizando las vías usuales ya que la solución numérica se obtiene con un método de Runge-Kutta estándar. Además, ello nos evita los problemas de la temprana degradación de la solución numérica debido al crecimiento de la estimación del error global. Por todo ello, en lo que sigue nos hemos centrado en el estudio del comportamiento de esa subclase de esquemas.

2.13. Condiciones de los parámetros del nuevo método

En la sección 2.5 hemos comentado cómo podemos determinar los valores apropiados para los parámetros b_i, $\overline{b}_i$, a_{ij} y μ_i a la hora de obtener métodos del tipo (2.13)– (2.15). Estos parámetros deben ser elegidos, una vez establecidos los valores de s y $\overline{s}$, en función de ciertos criterios.

1. El error local $\delta(y,h) = \psi_h(y) - \phi_h(y)$ del esquema Runge-Kutta ha de ser tan pequeño como sea posible. Normalmente se requiere que $\delta(y,h) = O(h^{p+1})$ para algún $p \geq 1$ (es decir, que el método sea de orden p) con constantes razonablemente pequeños en $O(h^{p+1})$.

2. La diferencia $\overline{\psi}_h(y+e,y) - \phi_h(y)$ también ha de ser tan pequeña como sea posible. Cuanto menor sea esta diferencia más similar será la transformación $\tilde{E}_h(y,e)$ a la transformación del error global $E_h(y,e)$. Nuestro objetivo es la obtención de estimaciones de $\overline{\psi}_h(y+e,y)-\phi_h(y)$ de la forma

$$\overline{\psi}_h(y+e,y) - \phi_h(y) = O(h^{q_0} + h^{q_1}||e|| + h^{q_2}||e||^2 + \cdots) \qquad (2.43)$$

   con q_k enteros positivos suficientemente altos, y constantes de error pequeños.

El primer criterio es típico para los métodos Runge-Kutta explícitos, y se sabe cómo obtener las condiciones de los coeficientes del método para lograr un orden prescrito utilizando árboles con raíz [4], [20]. Cada árbol genera una condición en términos de los parámetros b_i, a_{ij} del método, y el método será de orden p si se satisfacen todas las condiciones de los árboles con número de vértices negros menor o igual que p.

Para el segundo criterio volvemos a aplicar el procedimiento explicado en la Sección 2.5, basado en la utilización de árboles con raíz con vértices de dos colores. En la tabla 2.1 podemos encontrar las condiciones que genera cada árbol y los valores que van tomando los exponentes q_k ($k = 0, 1, 2, \ldots$). El valor q_k depende de las condiciones generadas por los árboles con k vértices blancos y nos indica el número de vértices negros que tiene el primer árbol cuya condición no se cumple. Es decir si $q_1 = 5$ significa que la condición asociada a cualquier árbol con un único nodo blanco con menos de 5 vértices negros se cumple. Los parámetros μ_i aparecen en las condiciones correspondientes a los árboles con nodos blancos, por lo que, q_0 depende única y exclusivamente de los parámetros a_{ij} y $\overline{b}_i$. Además, $p = q_0 - 1$, nos indica el orden del método Runge-Kutta subyaciente $\overline{\psi}_h(y,y)$.

2.14. Construcción de un método de orden 5

Hemos construido un esquema del tipo (2.13)– (2.15) que satisface (2.41)–(2.42) basado en el conocido método Runge-Kutta explícito de orden 5 de Dormand y Prince [6], al que llamaremos *DOPRI5* . Más en concreto, hemos elegido los parámetros s, b_i, a_{ij} $(1 \leq i, j \leq s)$ de forma que $\psi_h(y)$ coincida exactamente con un paso del método DOPRI5. Por tanto, tenemos que $s = 6$, y para nuestro esquema hemos elegido $\overline{s} = 10$. Tal y como suele ser habitual en la construcción de pares de métodos embebidos, hemos hecho que y_{n-1} y y_n sean la primera y última etapa del esquema (2.13)– (2.15), para lo cual han de cumplirse $a_{7i} = b_i$ $(1 \leq i \leq s = 6)$, $\mu_1 = 1$, y $\mu_7 = 1$.

El resto de los parámetros a_{ij} $(8 \leq i \leq 10,\ j < i)$, μ_8, μ_9, μ_{10} y b_i $(1 \leq i \leq 10)$ han sido determinados de forma que se cumpla la estimación

$$\overline{\psi}(y, \overline{y}) - \phi(\overline{y}) = O(h^7 + h^4 \|e\| + h^2 \|e\|^2 + h \|e\|^3), \tag{2.44}$$

es decir, , $q_0 = 7$, $q_1 = 4$, $q_2 = 2$, $q_3 = 1$ en (2.43).

Esta condición reduce el número de parámetros libres a 10, y hemos tratado de elegirlos de tal forma que se minimicen las constantes de error de los términos correspondientes a h^7, a $h^4 \|e\|$ y a $h^2 \|e\|^2$. Es decir, hemos tratado de minimizar numéricamente en el sentido de la suma de los cuadrados de los errores, las condiciones correspondientes a los árboles que faltan por cumplirse para que la estimación (2.44) sea de la forma $O(h^8 + h^5 \|e\| + h^3 \|e\|^2 + h \|e\|^3)$.

Tras realizar una intensiva búsqueda numérica y tras una comparación de los resultados de la búsqueda, especialmente en cuanto a la región de estabilidad lineal, hemos elegido un conjunto particular de valores para los parámetros libres que determinan el resto de parámetros a_{ij} $(8 \leq i \leq 10,\ j < i)$, μ_8, μ_9, μ_{10} y $\overline{b}_i$ $(1 \leq i \leq 10)$ del método con el que hemos realizado los experimentos. Estos parámetros pueden verse en la tabla 2.3, cuyos valores aparecen racionalizados con una precisión de 10^{-20}.

En la Figura 2.5 mostramos la región de estabilidad correspondiente al método dado en la Tabla 2.3. El eje vertical corresponde al eje imaginario y el horizontal representa el eje real. En la gráfica de abajo podemos ver la región de estabilidad y en la de arriba se puede ver que parte del eje imaginario entra en la región de estabilidad.

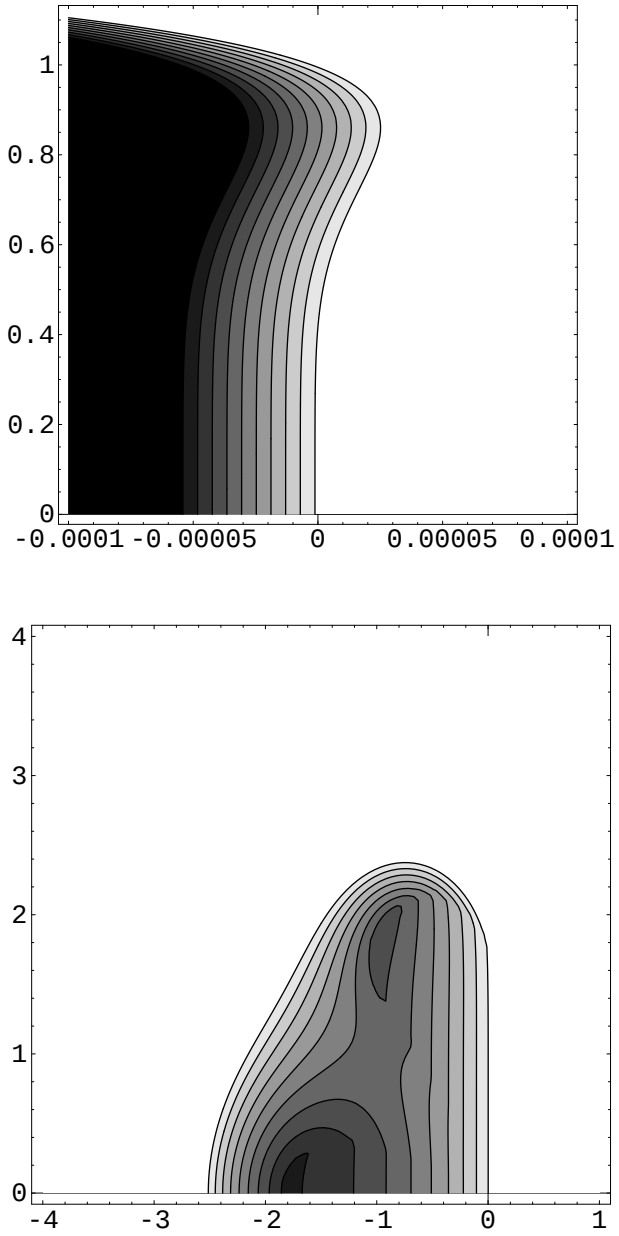


Figura 2.5: Abajo, región de estabilidad del método para la estimación del error global basado en DOPRI5. Arriba, se puede ver que la región abarca parte del eje imaginario.

Tabla 2.3: El resto de los parámetros del método basado en DOPRI5

i	$1-\mu_i$	c_i	$a_{8,i}$	$a_{9,i}$	$a_{10,i}$	$\overline{b}_i$
1	0	0	$\frac{26251126}{75292183}$	$-\frac{126276029}{115017392}$	$\frac{89178409}{82486612}$	$\frac{56696811}{789712427}$
2	0	$\frac{1}{5}$	$-\frac{30511879}{68834945}$	$\frac{153409379}{49308629}$	$-\frac{275044175}{99029299}$	0
3	0	$\frac{3}{10}$	$\frac{11490887}{155205387}$	$-\frac{107711621}{48274693}$	$\frac{115406143}{68971088}$	$-\frac{47431484}{279691831}$
4	0	$\frac{4}{5}$	$\frac{700737845}{174891007}$	$-\frac{675136779}{64711289}$	$\frac{140298385}{24130572}$	$\frac{72791025}{357831874}$
5	0	$\frac{8}{9}$	$-\frac{5336}{941}$	$\frac{559269939}{36928210}$	$-\frac{344040692}{42025591}$	$\frac{17490085}{349505178}$
6	0	1	$\frac{5735}{1214}$	$-\frac{669687859}{52442748}$	$\frac{121333564}{17575013}$	$-\frac{66245097}{563676842}$
7	0	1	$-\frac{2507}{898}$	$\frac{193952703}{25738526}$	$-\frac{190380249}{47005513}$	$-\frac{24}{611}$
8	$\frac{140719960}{143529893}$	$\frac{204}{823}$	0	$\frac{169021117}{130072535}$	$-\frac{12078143}{165601005}$	$\frac{40757463}{82884629}$
9	$\frac{941}{896}$	$\frac{579}{1036}$	0	0	$\frac{56747365}{92317949}$	$\frac{33159666}{111811519}$
10	$\frac{92493035}{95359057}$	1	0	0	0	$\frac{42422453}{199331202}$

2.15. Experimentos numéricos

El comportamiento del esquema a la hora de estimar el error global ha sido analizado mediante experimentos numéricos. El objetivo de de los experimentos es comprobar si las estimaciones del error global que nos ofrece el método implementado con longitud de paso variable son útiles y válidos para varios problemas. En este apartado consideramos tres problemas de valor inicial:

1. El primero de los problemas ha sido extraido de [20]. Se trata de un problema de valor inicial de dimensión 4, al que llamamos *Arenstorf*. Los valores iniciales corresponden a una solución periódica del problema restringido de los tres cuerpos. Anteriormente hemos utilizado este mismo problema para mostrar los problemas de independencia que sufrían los métodos estudiados con longitud de paso fijo.

2. El segundo, al que llamamos *Pleiades*, es uno de los que aparecen en el conjunto de problemas IVPTest [11] para analizar los métodos de resolución de problemas de valor inicial. Se trata de un problema de dimensión 28.

3. El tercero de los problemas ha sido extraido de [23]. Se trata del problema unidimensional mostrado en (1.2) y su equivalente autónomo en (1.6). Su solución es $y(t) = e^{sin(t)}$, una función periódica cuyo periodo es 2π. Este problema también lo hemos utilizado en el estudio del comportamiento de los métodos con paso fijo, donde nos hemos referido a él como el problema *expsin*.

Para cada uno de los problemas mostramos dos gráficas, una de ellas ha sido obtenida utilizando una tolerancia muy pequeña, mientras que la otra corresponde a los resultados obtenidos con una tolerancia mayor.

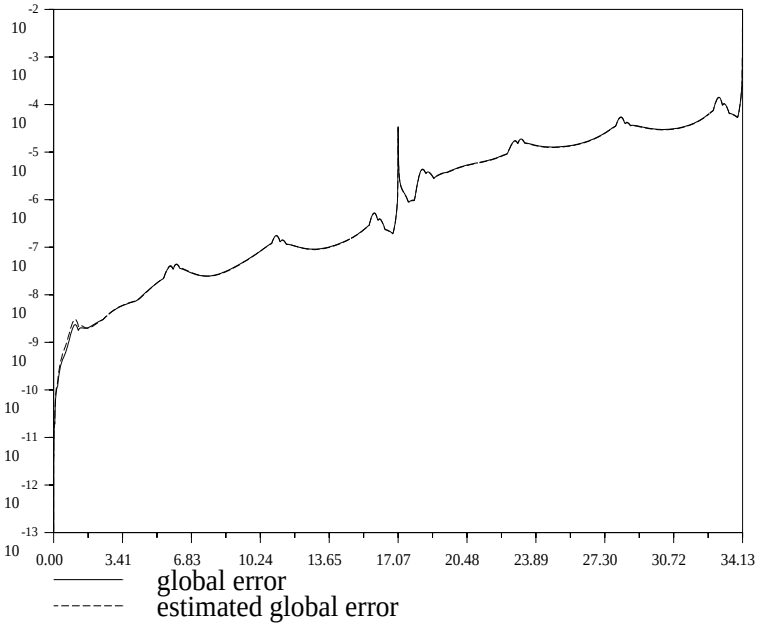
En la figura 2.6 vemos los resultados obtenidos con el problema *Arenstorf* en la integración sobre dos periodos. El eje horizontal corresponde al tiempo mientras que el vertical muestra los valores de la norma infinita del error global en escala logarítmica. En las gráficas se pueden observar dos curvas, una corresponde a la estimación del error global obtenida por nuestro esquema mientras que la otra muestra el error global exacto. La comparación de ambas curvas nos lleva a la conclusión de que tanto el error global estimado como el exacto tienen un comportamiento muy parecido. El cambio de la tolerancia tampoco afecta tanto a los resultados, en la gráfica de arriba se ha utilizado una tolerancia de 10^{-9} mientras que en la de abajo podemos observar los resultados obtenidos con una tolerancia de 10^{-6}. Evidentemente, al relajar la tolerancia el número de pasos requeridos para la integración baja, en este caso de 1268 a 309, y el error global obtenido se ve incrementado, pero la estimación del error global sigue reflejando de forma satisfactoria ese incremento y comportamiento.

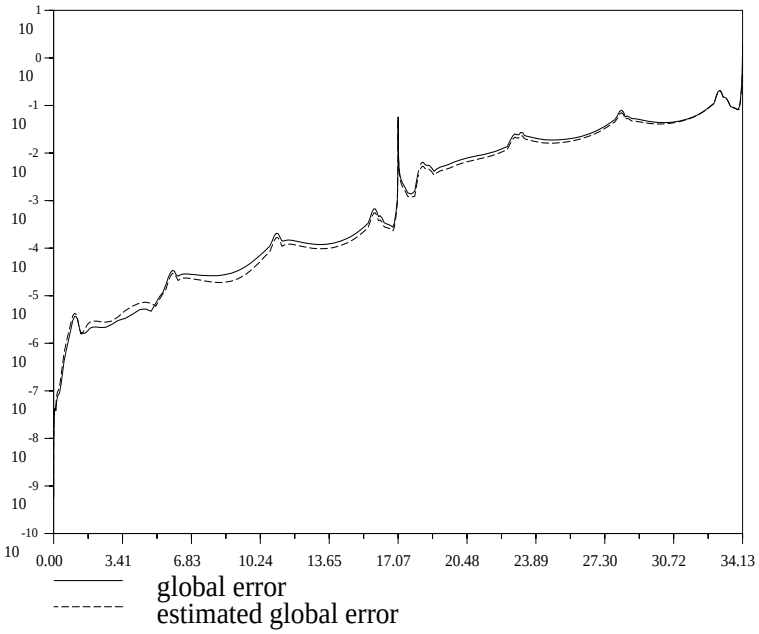
Podemos observar los resultados obtenidos con el problema *Pleiades* en la figura 2.7. Volvemos a mostrar dos integraciones, las dos con diferentes tolerancias, una con tolerancia de 10^{-9} que requiere 1603 pasos y la otra con tolerancia de 10^{-4} con 182 pasos. Nuevamente vemos que las estimaciones del error global, tanto con una tolerancia como con la otra vuelven a comportarse como pequeñas perturbaciones del error global exacto, es decir, podemos repetir lo dicho para el problema *Arenstorf* para el caso del problema *Pleiades*.

Otra cuestión a destacar es que el esquema cumple las condiciones (2.41)– (2.42) por lo que la solución numérica ofrecida por el método coincide con la solución numérica del método Runge-Kutta explícito *DOPRI5*. Evitando de esta forma los problemas de degeneración debido a la influencia del error global que padecíamos con los pares que no cumplen esas dos condiciones.

Otro ejemplo de los resultados obtenidos, pero en este caso con más diferencias que en los anteriores experimentos, son los correspondientes al

Figura 2.6: Problema *Arenstorf*. Arriba una tolerancia de 10^{-9} que requiere 1268 pasos. Abajo, 10^{-6} que requiere 309 pasos





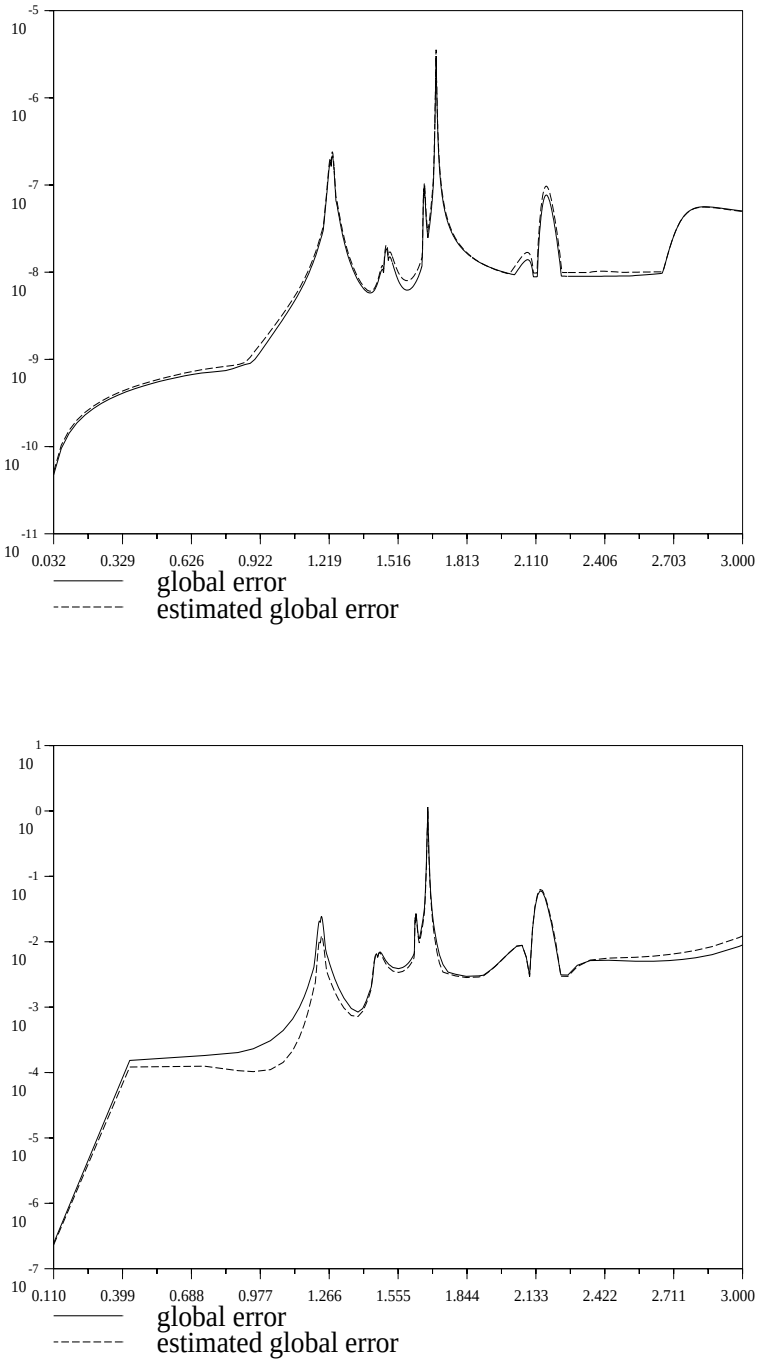


Figura 2.7: Problema *Pleiades*. Arriba con tolerancia de 10^{-9} con 1603 pasos. Abajo tolerancia de 10^{-4} con 182 pasos.

tercero de los problemas. En la figura 2.8 mostramos dos gráficas, cada una obtenida con diferentes tolerancias, y en ellas se aprecia una mayor diferencia entre la estimación del error global y los valores reales del error global. De todas formas, hay que decir que el comportamiento de los dos errores siguen siendo parecidos, y que, por tanto, siguen siendo unas estimaciones válidas, que dan información útil sobre el comportamiento del error global.

2.16. Conclusiones

Todos los experimentos realizados indican que la información obtenida en torno al error global es una estimación aceptable, por lo que, surgen nuevas expectativas con respecto a su utilización. En concreto, una primera utilización puede darse en la toma de decisiones sobre la validez de las soluciones ofrecidas por el sistema, es decir, podremos utilizar nuevas tolerancias globales que indiquen hasta qué punto son o dejan de ser aceptables los resultados, lo cual puede ayudarnos a la hora de realizar integraciones costosas, ya que podremos detener los procesos, en los que, aún siendo aceptables todos y cada uno de los pasos según el sistema tradicional de control de las longitudes de paso basado en tolerancias locales, puede llegarse a soluciones cuya validez se aleje de lo que es aceptable globalmente.

A la hora de realizar los experimentos numéricos, muchas veces nos hemos encontrado con problemas cuyo error crece exponencialmente, y en estos casos, los pequeños errores aceptados en cada paso, errores por debajo de la tolerancia, se ven amplificados según avanza la integración. Todo esto nos lleva a que al final de la integración las soluciones numéricas obtenidas no sean aceptables, es decir, obtenemos soluciones para los que el error global es demasiado grande. En estos casos la cuestión que se puede plantear es doble, por una parte, se debe detectar la situación lo antes posible, y por otra, se debería utilizar la información sobre el error global para tratar de corregir la situación, o incluso para abaratar el costo computacional de la integración.

En este último sentido, parece posible el abaratamiento del coste computacional ya que el mantenimiento de una tolerancia local sin tener en cuenta la evolución del error global puede no ser una buena política, especialmente cuando la propagación de los errores de los pasos previos superan los efectos del error local en el error global. Es decir, parece no tener demasiado sentido el tratar de mantener el error local de un paso por debajo de una tolerancia dada, cuando se sabe que la propagación de los errores locales de pasos anteriores provocan un error muy por encima de dicha tolerancia.

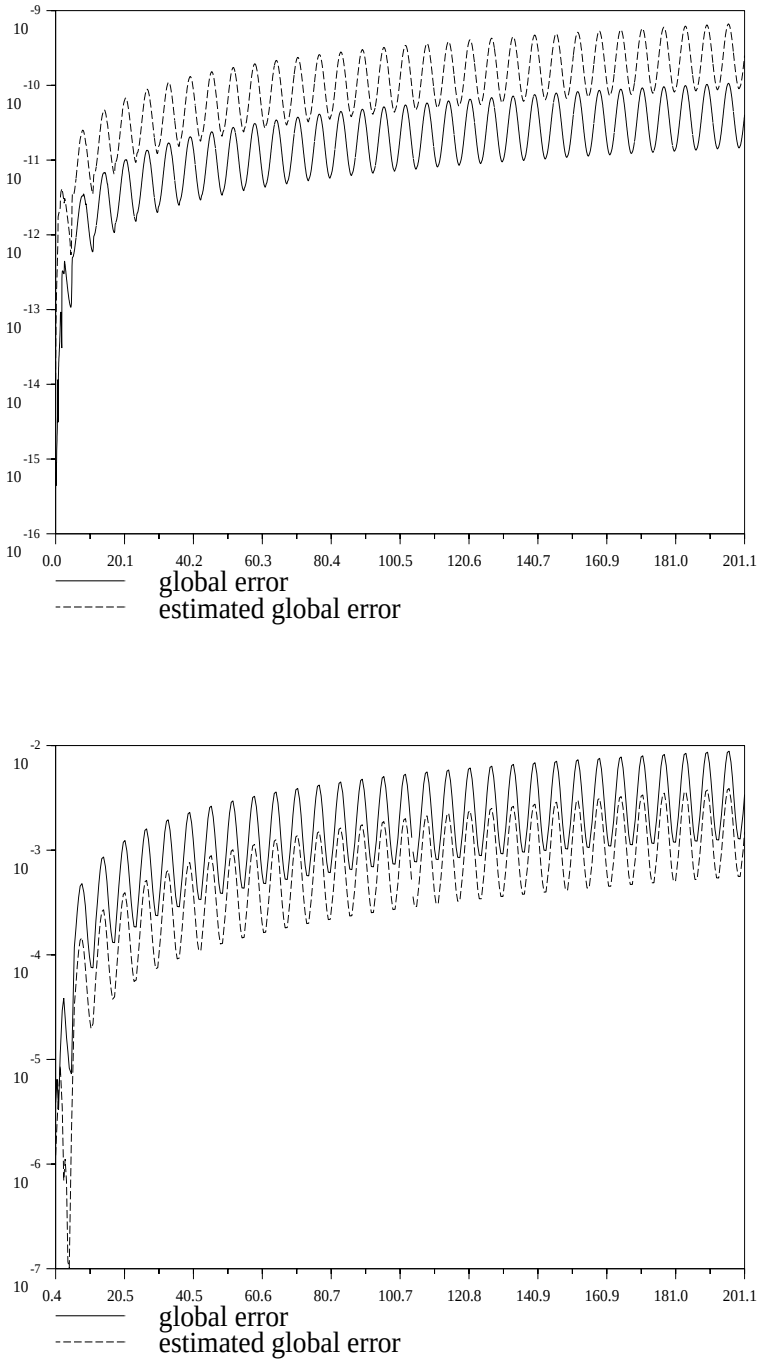


Figura 2.8: Problema *expsin*. Arriba con tolerancia de 10^{-9} con 7467 pasos. Abajo tolerancia de 10^{-4} con 416 pasos.

En el siguiente capítulo exploraremos las posibilidades de incrementar la eficiencia de las estrategias de ajuste de la longitud de paso utilizando la estimación del error global.

Capítulo 3

Control de la Longitud de Paso Basado en Estimaciones del Error Global

3.1. Introducción

El control del proceso de integración de las Ecuaciones Diferenciales Ordinarias (1.1) mediante métodos Runge-Kutta explícitos se basa en el control de la discretización $t_0 < t_1 < \cdots < t_N = t_f$ de la variable independiente t, siendo $h_n = t_n - t_{n-1}$ la longitud del n-ésimo paso. Dicho control se realiza manteniendo la diferencia entre dos soluciones numéricas, que se toma como estimación del error local (1.15) de cada paso, por debajo de cierta tolerancia, adecuando la longitud de paso a la longitud óptima, en el sentido de que con la longitud utilizada la estimación del error local se aproxime a la tolerancia sin superarlo. No obstante, el error global (1.16), que es lo que realmente es importante cuando tenemos que valorar los resultados obtenidos en una integración numérica, puede crecer substancialmente con respecto al error local según avanza la integración. Cuando integramos un problema cuyo error global crece exponencialmente, la propagación de los pequeños errores aceptados en cada paso tienen un efecto en el error global que crece con el avance de la variable independiente, por lo que podemos pensar que puede no tener sentido el algoritmo de control de la integración comentado anteriormente. Recientes trabajos confirman que la elección de la longitud de paso puede ser mejorada, por ejemplo, Jitse Niessen en [19] busca la longitud de paso óptima para la integración de diferentes problemas y cita entre las conclusiones que los errores locales iniciales tienen mucha

mayor importancia que las finales, por lo que las longitudes de paso iniciales deberían ser menores que las finales.

Los esquemas planteados en el capitulo sobre métodos de Runge-Kutta con estimación del error global (2.13)-(2.14) nos ofrecen estimaciones del error global, y hemos explorado la posibilidad de utilizarlas para incrementar la eficiencia de las estrategias de ajuste de la longitud de paso. Parece posible aumentar la eficiencia del integrador con el hecho de aumentar la tolerancia del error local cuando la propagación de los errores previos se ve aumentada.

Proponemos, por una parte, utilizar la información del error global para determinar los efectos de los errores previos en cada paso y, por otra, retocar la política de ajuste de la longitud de paso haciendo uso de dicha información.

3.2. Utilización de la estimación del error global

Se pretende buscar una condición para el error local (1.15) que garantice que su efecto en el error global (1.16) sea comparable al efecto de la propagación de los errores locales previos.

En (1.17) descomponíamos el error global como la suma del error local y la propagación de los errores previos. Asumiendo que el error global es suficientemente pequeño podemos decir que el error en el n-ésimo paso satisface

$$e_n \simeq R_{n,n-1}e_{n-1} + \delta_n \tag{3.1}$$

donde $\delta_n = \delta(y_{n-1}, h_n) = \psi_{h_n}(y_{n-1}) - \phi_{h_n}(y_{n-1})$ y $R_{n,j}$ es la matriz Jacobiana del $(t_n - t_j)$-flujo $\phi_{t_n-t_j}(y)$,

$$R_{n,j} = \frac{\partial \phi_{t_n-t_j}}{\partial y}\left(y(t_j)\right).$$

Para simplificar la exposición asumiremos a partir de ahora que (3.1) se cumple exactamente. Debido a las propiedades del flujo se cumple que $R_{n,j} = R_{n,k}R_{k,j}$ para $n \geq k \geq j$, lo que implica que

$$e_n = R_{n,j}e_j + \sum_{k=j+1}^{n} R_{n,k}\delta_k, \quad 1 \leq j < n. \tag{3.2}$$

3.2.1. Propagación de los errores

Para encontrar las condiciones que buscamos para el error local, hay que ver cómo y en qué condiciones se propagan los errores de los pasos

anteriores.

Lema 3 *Asumiendo que (3.2) se cumple, si*

$$\|\delta_k\| \leq \frac{h_k \|R_{k,j}\| \varepsilon_j}{C}, \tag{3.3}$$

$$\varepsilon_j = \frac{\|e_j\|}{t_j - t_0}, \tag{3.4}$$

$$C = \max_{j \leq k \leq n} \left(\frac{\|R_{n,k}\| \|R_{k,j}\|}{\|R_{n,j}\|} \right), \tag{3.5}$$

entonces, se cumple

$$\varepsilon_n \leq \|R_{n,j}\| \varepsilon_j$$

para $1 \leq j < n$.

Demostración De la hipótesis del lema y de (3.2) llegamos a la desigualdad

$$\|e_n\| \leq \|R_{n,j}\| (t_j - t_0)\varepsilon_j + \sum_{k=j+1}^{n} \|R_{n,k}\| \frac{\|R_{k,j}\| h_k \varepsilon_j}{C},$$

que podemos reescribir como

$$\|e_n\| \leq \|R_{n,j}\| \varepsilon_j \left((t_j - t_0) + \sum_{k=j+1}^{n} h_k \frac{1}{C} \frac{\|R_{n,k}\| \|R_{k,j}\|}{\|R_{n,j}\|} \right),$$

y por la definición de C llegamos a

$$\|e_n\| \leq \|R_{n,j}\| \varepsilon_j \left((t_j - t_0) + \sum_{k=j+1}^{n} h_k \right) = \|R_{n,j}\| \varepsilon_j (t_n - t_0),$$

que nos lleva al resultado requerido. □

Lema 4 *Bajo los supuestos del Lema 3, si*

$$\varepsilon_{n-1} \leq \|R_{n-1,j}\| \varepsilon_j,$$
$$\|\delta_n\| \leq \frac{1}{C^2} \|R_{n,n-1}\| h_n \varepsilon_{n-1},$$

entonces se cumple que

$$\varepsilon_n \leq \|R_{n,j}\| \varepsilon_j. \tag{3.6}$$

Demostración Utilizando las dos desigualdades de las hipótesis del lema, llegamos a

$$\|\delta_n\| \leq \frac{1}{C^2} \|R_{n,n-1}\| \, h_n \, \|R_{n-1,j}\| \, \varepsilon_j = \frac{1}{C} \frac{\|R_{n,n-1}\| \, \|R_{n-1,j}\|}{C} h_n \varepsilon_j,$$

y teniendo en cuenta que, por definición de C, se cumple

$$\frac{\|R_{n,n-1}\| \, \|R_{n-1,j}\|}{C} \leq \|R_{n,j}\|,$$

llegamos a la desigualdad (3.3) que es la condición del Lema 3, cuya aplicación nos da la desigualdad que buscábamos. □

La forma típica de ajustar la longitud de paso (véase [23, pp. 335]) depende de la tolerancia τ establecida por el usuario y de una norma $\|\cdot\|$ con el que se mide el error local. En el caso del criterio de *error por unidad de paso* se asume que la longitud de paso óptima en el n-ésimo paso es el mayor h_n para el que la norma del error local satisface

$$\|\delta_n\| \leq h_n \, \tau. \tag{3.7}$$

Si tenemos en cuenta (3.2), con $j = 0$ y suponiendo que $e_0 = 0$, la condición (3.7) nos limita el error en el paso n dándonos la siguiente cota:

$$||e_n|| \leq \tau \sum_{k=1}^{n} ||R_{n,k}|| h_k. \tag{3.8}$$

Hay que decir que la condición (3.8) no garantiza que el error global en un cierto momento t sea menor que la tolerancia establecida, pero, en general, se puede tener la esperanza de que el error global vaya en proporción a la tolerancia τ (la cota (3.8) puede dar una idea de esa proporcionalidad). En la práctica, la norma del error local es desconocida, por lo que en su lugar se utilizan estimaciones de las mismas.

Análisis del parámetro C para problemas lineales

Obviamente el parámetro C depende del problema que se quiere resolver. Para el caso de los problemas lineales, la solución de $y' = Ay$ es

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

en cuyo caso tenemos que $\|R_{nj}\| = \left\|e^{(t_n - t_j)A}\right\|$. Teniendo en cuenta que

$$\frac{\|R_{n,k}\| \, \|R_{k,j}\|}{\|R_{n,j}\|} = C(t_n - t_j, t_n - t_k), \tag{3.9}$$

donde

$$C(t,s) = \frac{\left\|e^{sA}\right\| \left\|e^{(t-s)A}\right\|}{\left\|e^{tA}\right\|},$$

según la definición de C dada en (3.5) tenemos que

$$C = \max_{j\le k\le n} C(t_n - t_j, t_n - t_k) \le \sup_{\substack{t,s\in\mathcal{R}\\ 0\le s\le t}} C(t,s) =: C^*,$$

lo cual implica que siempre se cumple que $C \ge 1$. Además, con la norma euclídea, si la matriz A es simétrica, se cumple que $C = 1$: En ese caso, si A es de dimensión $n \times n$, podemos escribirla como $A = P^T D P$ donde P es una matriz ortonormal y D es una matriz diagonal, y la diagonal de D está compuesta por los valores propios $\lambda_1, \lambda_2, \ldots, \lambda_n$ de A . En ese caso tenemos que $e^{tA} = P^T e^{tD} P$. Puesto que la norma euclídea de una matriz B es

$$\|B\|_2 = \sqrt{\rho(B^T B)}$$

donde ρ indica el radio espectral, tenemos que

$$\left\|e^{tA}\right\| = \sqrt{\rho(P^T e^{tD} P P^T e^{tD} P)},$$

y teniendo en cuenta que, al ser P una matriz ortonormal, PP^T es la matriz identidad, llegamos a

$$\left\|e^{tA}\right\| = \sqrt{\rho(P^T \left(e^{tD}\right)^2 P)} = \sqrt{\rho\left(e^{2tD}\right)} = e^{t|\lambda_{\text{máx}}|}$$

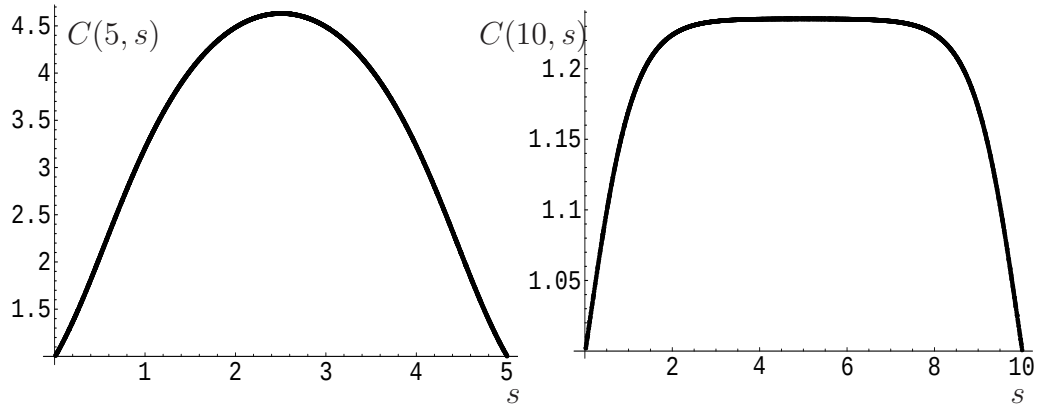
donde $|\lambda_{\text{máx}}| = \text{máx}(|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|)$. Si sustituimos estos valores en $C(t,s)$ llegamos a

$$C(t,s) = \frac{e^{s|\lambda_{\text{máx}}|} e^{(t-s)|\lambda_{\text{máx}}|}}{e^{t|\lambda_{\text{máx}}|}} = \frac{e^{t|\lambda_{\text{máx}}|}}{e^{t|\lambda_{\text{máx}}|}} = 1.$$

No obstante, para sistemas lineales con matriz A no simétrica C^* puede ser considerablemente mayor.

Hemos realizado una serie de experimentos numéricos para obtener C^* para distintas matrices A. Las matrices se han elegido utilizando la función *Random[]* de la aplicación *Mathematica*, lo que nos da valores reales elegidos de forma pseudoaleatoria distribuidos uniformemente entre 0 y 1. En cuanto a las dimensiones de las matrices hemos utilizado matrices de 3×3, 4×4 y 5×5. Para cada una de las matrices tomamos distintos valores de t y de s $(0 \le s \le t)$ y obtuvimos el valor que toma $C(t,s)$.

Figura 3.1: A la izquierda, los valores $C(t,s)$ para una matriz 3×3 con $t=5$. Los valores máximos de $C(t,s)$ se dan para el valor $s=\frac{t}{2}$. A la derecha, mostramos los valores $C(t,s)$ para una matriz 5×5 con $t=10$. Las máximos valores de $C(t,s)$ vuelven a darse en la zona central del intervalo $(0,t)$.



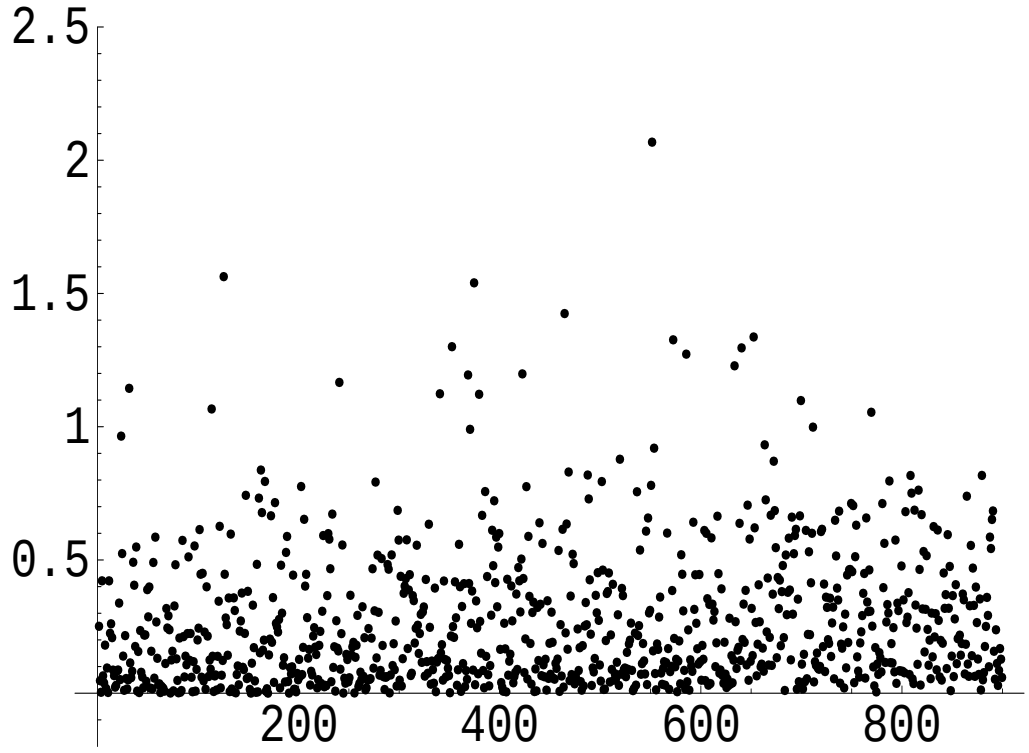
En la Figura 3.1 podemos observar el comportamiento de $C(t,s)$ para dos matrices diferentes, con un valor de t prefijado y haciendo variar s en el intervalo $(0,t)$. La gráfica de la izquierda corresponde a una matriz 3×3 con $t=5$, la de la derecha a una matriz 5×5 con $t=10$. El eje horizontal corresponde al valor s y el eje vertical a $C(t,s)$.

En las gráficas de la Figura 3.1 se puede ver el comportamiento de $C(t,s)$ para t prefijado: en los extremos del intervalo el valor se aproxima a 1 y en la medida que nos alejamos de los extremos del intervalo el valor de $C(t,s)$ crece, tomando los valores máximos en la zona central del intervalo. Este comportamiento es general, es decir, aunque mostremos las gráficas de dos matrices el resto de matrices proporcionan resultados con el mismo comportamiento. Lo que nos interesa es conocer el máximo de dichos valores, es decir C^*, ya que C, dado en (3.5), es menor o igual que dicho valor. Hemos observado experimentalmente que con un valor de t suficientemente grande $(t>10)$, los valores obtenidos para $C(t,s)$ con $s\approx t/2$, se acercan mucho a C^*.

En la Figura 3.2 mostramos los valores de C^* de 900 matrices elegidas al azar (de forma pseudoaleatoria con valores uniformemente distribuidos entre 0 y 1), de ellas, las primeras 300 corresponden a matrices de dimensión 3x3, las 300 siguientes a matrices de dimensión 4x4 y las 300 siguientes a matrices de dimensión 5x5. La altura de cada punto indica el logaritmo decimal del valor de C^* correspondiente a cada matriz.

Los resultados nos indican que para matrices de mayor dimensión se obtiene, por regla general, valores de C^* mayores. En la Figura 3.2 la con-

Figura 3.2: Distribución del valor $\log_{10} C^*$ para 900 matrices elegidas al azar: 300 de dimensión 3x3, 300 de dimensión 4x4 y 300 de dimensión 5x5



centración de puntos para las matrices de 3x3 es mayor para valores de C^* entre 1 y 1.5 (logaritmo decimal entre 0 y 0.176). Esta afirmación se puede ver mejor si mostramos los valores de C^* en una tabla, en ella hemos puesto el número de matrices que tienen C^* en rangos de valores que van de menor a mayor, y separando las matrices de distintas dimensiones:

intervalo de $\log_{10} C^*$	3×3	4×4	5×5
(0.000, 0.176)	174	148	118
(0.176, 0.477)	83	104	117
(0.477, 1.000)	38	37	59
(1.000, 2.500)	4	10	5

3.3. Nuevas estrategias de selección de longitud de paso

En [13] propusimos una nueva estrategia para la selección de la longitud de paso para los métodos Runge-Kutta explícitos. La nueva estrategia se basa en el Lema 4, y consiste en primeramente elegir una longitud de paso inicial h_1 en la forma estándar, exigiendo que cumpla (3.7), y a partir del segundo paso, $n = 2$, determinar la longitud de paso h_n de forma que cumpla

$$\|\delta_n\| \leq \frac{h_n \varepsilon_{n-1}}{C^2}, \tag{3.10}$$

donde ε y C son los definidos en (3.5–3.3).

Como $||R_{n,n-1}|| \geq 1$, se puede aplicar el Lema 4, que nos da $\varepsilon_n \leq \|R_{n,1}\| \, \varepsilon_1$. Y, además, como h_1 se ha elegido de forma que cumpla $||\varepsilon_1|| \leq \tau$, obtenemos la cota

$$\|e_n\| \leq \tau \, \|R_{n,1}\| \, (t_n - t_0). \tag{3.11}$$

Podemos decir que si comparamos esta cota con la cota (3.8), que corresponde al error global obtenido utilizando la estrategia estándar de ajuste de la longitud de paso, es una cota razonable. Nótese que el ratio entre las dos cotas crece, como mucho linealmente con el tiempo, y la nueva estrategia nos permitirá la utilización de pasos mas largos en la medida que avance la integración (suponiendo que la secuencia $\varepsilon_n = e_n/(t_n - t_0)$ es creciente).

La condición (3.10) se puede tomar como la condición (3.7) con una tolerancia variable $\tau = \frac{\varepsilon_{n-1}}{C^2}$. En la práctica, proponemos modificar la estrategia típica sustituyendo la tolerancia τ por el valor resultante en

$$\tau' = \text{máx}(\tau, K\varepsilon_{n-1}) \tag{3.12}$$

donde K es un parámetro prefijado. Idealmente,

$$K = \frac{1}{C^2}. \tag{3.13}$$

En la práctica, no vamos a conocer el valor exacto de C definido en (3.5). Sabemos que $C \geq 1$ por lo que K debe ser un valor que cumpla $0 \leq K \leq 1$.

Si elegimos un valor de $K = 0$, la nueva estrategia se convierte en la estrategia estándar, ya que máx$(\tau, K\varepsilon_{n-1}) = \tau$, y en ese caso estaríamos siguiendo el procedimiento normal de ajuste de longitud de paso. Por otro lado, si hacemos que K tenga un valor demasiado grande, de forma que $K > \frac{1}{C^2}$ y la condición (3.10) no se cumpla, entonces no se puede garantizar la cota (3.11), lo que puede provocar un crecimiento demasiado veloz del error global $||e_n||$.

Desafortunadamente el valor óptimo de $K \in (0, 1]$ depende del problema que se quiera resolver, y sería deseable obtener un método automático de ajuste del parámetro K, para poder incluir la implementación de la nueva estrategia de una forma robusta. Ese método automático debería ajustar el valor de K según se avance en la integración del problema. Para los problemas cuyo error global crece muy suavemente debería elegir valores cercanos a 0, mientras que para aquellos problemas cuyo error global crece exponencialmente debería ajustar K a valores cercanos a 1. La búsqueda de dicho tratamiento automático puede suponer nuevas vías de investigación en la línea de la optimización de las estrategias de ajuste de la longitud de paso.

3.4. Implementación de la nueva estrategia

Al igual que se hace con las estrategias de ajuste de longitud de paso usuales (véase [20, pp.167-168], [23, pp.334-347]), nuestra estrategia también debe ser cauta a la hora de modificar la tolerancia, y debe evitar en lo posible incrementos demasiado grandes. La implementación de la nueva estrategia se ha realizado teniendo en cuenta mecanismos de seguridad y ralentización parecidos a los mecanismos mencionados en la sección 1.10.

Cada vez que se decida incrementar la tolerancia τ' dada en (3.12), limitamos el cambio a realizar poniendo un margen de seguridad, que por ejemplo puede ser que en un único cambio, como mucho, se permita doblar la tolerancia. Otro posible mecanismo de control es que la tolerancia variable τ', independientemente de los cambios que haya podido sufrir durante el proceso de integración, no pueda superar una determinada cota prefijada,

por ejemplo, no permitiendo que en ningún momento la tolerancia variable τ' sea mayor que 100 veces la tolerancia inicial.

Por otro lado, para poder decidir que se va a realizar un cambio de la tolerancia es muy importante contar con la mejor información posible, por lo que solo se permite el cambio de la tolerancia cada cierto número de pasos. En este sentido, hemos implementado un mecanismo que cada cierto tiempo, obtiene una estimación más exacta del error local. Para ello nos basamos en la aplicación de dos pasos consecutivos con la misma longitud de paso. Con las etapas de los dos pasos se puede obtener un método Runge-Kutta más preciso, ya que disponemos del doble de etapas intermedias, y aunque la longitud de paso sea mas larga, se puede obtener una aproximación mejor del error local con lo que podemos establecer un factor de corrección de la estimación del error local. El mecanismo de paso doble hay que controlarlo, ya que, la frecuencia de realizar los pasos dobles puede provocar que la integración no siga la secuencia óptima de longitudes de paso, y de esa forma incrementaríamos el costo computacional. No obstante si la frecuencia es muy baja es posible que no aprovechemos al máximo la potencialidad de mejora de la nueva estrategia de ajuste de longitud de paso, por lo que también estaríamos perdiendo eficiencia.

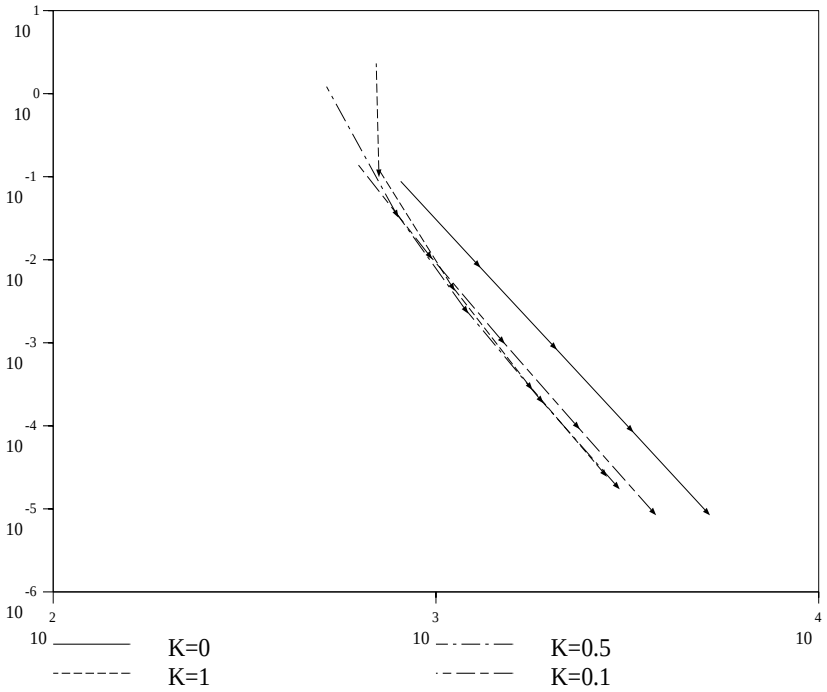
3.5. Experimentos numéricos con el nuevo método de ajuste de la longitud de paso

Hemos incluido la implementación de la nueva estrategia basada en (3.10) y en (3.12) en nuestro código de resolución de ODEs y hemos llevado a cabo varios experimentos numéricos con diferentes problemas de valor inicial. Podemos ver los resultados obtenidos con cuatro problemas distintos:

1. El problema definido en [20, pp.129–130], el cual hemos mostrado en (1.4) y comentado en la Sección 2.15 (al que hemos llamado *Arenstorf*). Se trata de un problema cuyo error crece muy rápidamente.

2. Al igual que el anterior, hemos utilizado este segundo problema también en la Sección 2.15. Se trata del problema que hemos tomado del conjunto de problemas IVPTest [11] para analizar métodos de resolución de problemas de valor inicial, en dicho conjunto al problema elegido lo llaman el problema de *Pleiades* y es un problema de dimensión 28.

Figura 3.3: Comparación de costos para el problema *Arenstorf*: nueva estrategia de ajuste de longitud de paso y la estrategia usual



3. El problema de valor inicial que hemos tomado de [20, pp. 120-121] conocido como problema de *Lorenz*.

4. El problema de *Kepler* de los dos cuerpos [23, pp. 91] con excentricidad = 0,5.

Con el objetivo de analizar la eficiencia del nuevo sistema de ajuste de la longitud de paso mostramos en las gráficas el número de pasos necesarios contra el error global al final de la integración. Los dos ejes se muestran en escala logarítmica, y en cada gráfica mostramos diferentes líneas de eficiencia: para obtener cada una de las lineas se ha integrado el problema varias veces, cada vez con una tolerancia distinta, por lo que obtenemos la curva que nos relaciona el número de pasos con el error global.

En cada gráfica tenemos varias curvas de eficiencia, en todas ellas una corresponde a la estrategia típica de ajuste de longitud de paso (linea continua), mientras que el resto de líneas corresponden a la nueva estrategia de ajuste de longitud de paso planteada en la Sección 3.3 para diferentes

valores de K. Los valores que hemos utilizado para K son $K = 0{,}1$, $K = 0{,}5$ y $K = 1$ (la estrategia estándar es la que corresponde a $K = 0$).

Podemos ver los resultados obtenidos con el problema *Arenstorf* en la Figura 3.3. La linea continua corresponde a la estrategia usual de ajuste de la longitud de paso, y se puede ver que independientemente de la tolerancia, es decir, independientemente del número de pasos utilizados para la integración, la estrategia usual es claramente menos eficiente que la nueva estrategia planteada por nosotros.

De las tres lineas de eficiencia obtenidas con la nueva estrategia, y que mostramos para este problema, la mejor corresponde a $K = 0{,}5$, aunque cuando la tolerancia es muy exigente la línea correspondiente a $K = 1$ obtiene prácticamente los mismos resultados. No obstante, para tolerancias relajadas se ve que los resultados obtenidos con $K = 1$ son malos, es decir hay un crecimiento grande del error global. Esto se debe, al parecer, a que en estos casos $K = 1 > \frac{1}{C^2}$, y la condición (3.10) deja de cumplirse, lo que hace que la cota del error (3.8) tampoco se cumpla. Para este problema, la mejora de la eficiencia en la integración cuando $K = 0{,}5$ es alrededor del 33 % para todas las tolerancias, o dicho de otra forma, la estrategia usual es un 50 % mas costosa que la nueva estrategia.

La Figura 3.4 muestra la diferencia entre la estrategia usual de ajuste de la longitud de paso y la estrategia derivada de la Sección 3.3 para el problema *Pleiades*. En este caso los mejores resultados se han logrado con $K = 1$, mientras que los peores vuelven a ser los obtenidos con la estrategia usual de ajuste de la longitud de paso. La ganancia de la eficiencia para este problema está entre el 20 % y el 45 % respecto a la estrategia usual. Las tolerancias relajadas son las que muestran las menores ganancias, mientras que en la medida en que la tolerancia se vuelve más exigente la ganancia en el costo computacional se incrementa hasta llegar a un incremento que parece estabilizarse.

Los resultados para el problema *Lorenz* se muestran en la Figura 3.5. Volvemos a obtener unos resultados que muestran que la nueva estrategia mejora la eficiencia de la integración sin perder la precisión de los resultados. En este problema los resultados son muy buenos para todos los valores de K, y en concreto para $K = 1$ y para $K = 0{,}5$ la ganancia en la eficiencia es de al rededor del 45 % respecto a la estrategia usual, es decir, la estrategia usual tiene un costo de casi el doble que la nueva estrategia propuesta.

Podemos comparar en la Figura 3.6 las longitudes de paso utilizadas en la integración del problema *Lorenz* utilizando las dos estrategias: la estrategia usual, que solo atiende al error local y la nueva estrategia con $K = 0{,}5$ que tiene en cuenta la propagación del error. La línea continua muestra las

Figura 3.4: Comparación de costos para el problema *Pleiades*: nueva estrategia de ajuste de longitud de paso y la estrategia usual

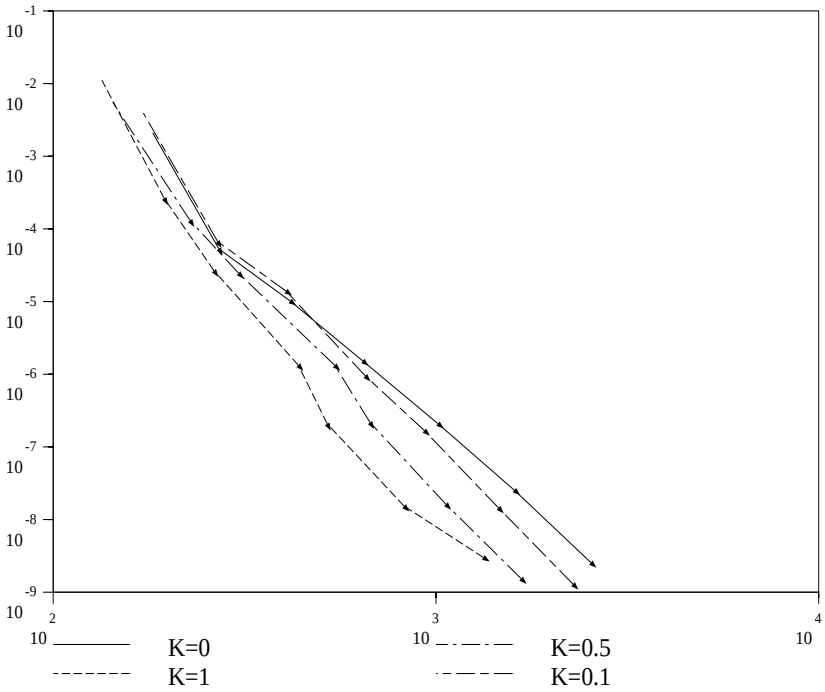


Figura 3.5: Comparación de costos para el problema *Lorenz*: nueva estrategia de ajuste de longitud de paso y la estrategia usual

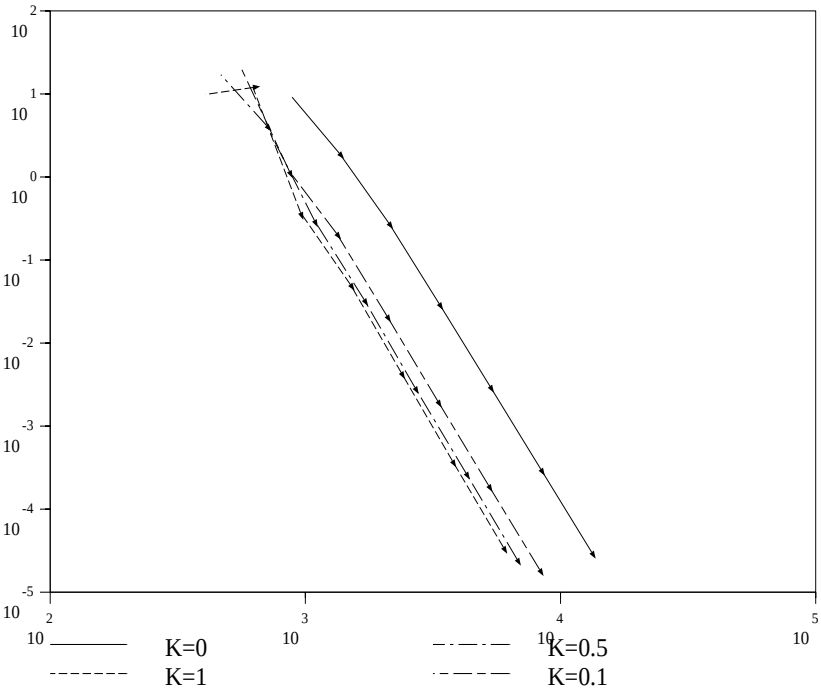
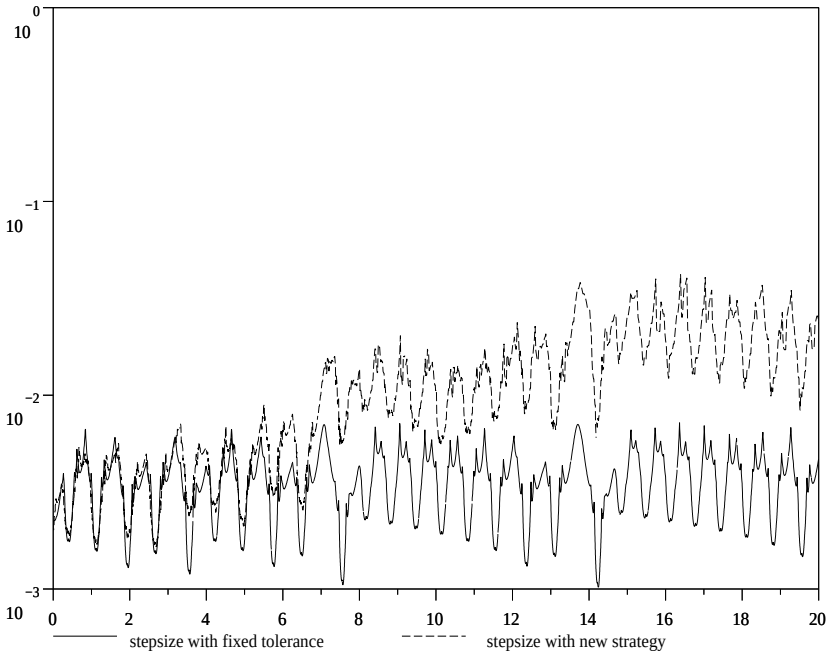


Figura 3.6: Comparación de las longitudes de paso para el problema *Lorenz*



longitudes de paso que determina el sistema usual, mientras que la discontinua muestra las longitudes de paso que acepta la nueva estrategia. Se ve claramente que las longitudes de paso aceptadas van creciendo según avanza la integración, esto se debe a que la propagación de los errores se incrementa según avanza la integración y por ello las tolerancias locales pueden ser más relajadas, ya que el incremento del error global se debe sobre todo a la propagación de los errores de los pasos anteriores.

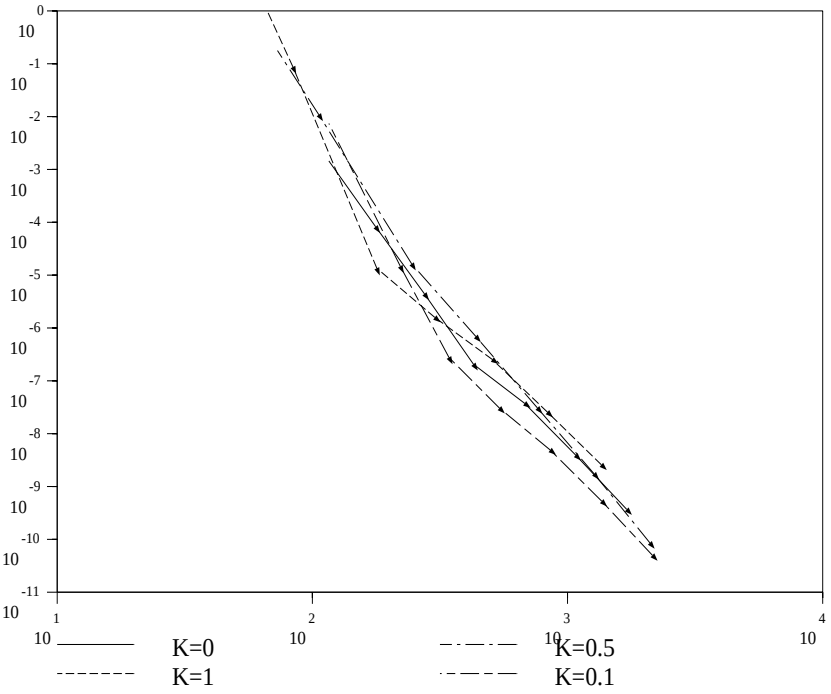
Hasta ahora hemos visto el comportamiento de la nueva estrategia con problemas cuyos errores crecen muy rápidamente, y podemos decir que los resultados son francamente buenos. Por otra parte, también nos hemos planteado la cuestión de cual es el comportamiento de la nueva estrategia cuando el problema no ofrece un crecimiento del error global tan acelerado. En la Figura 3.7 podemos ver el comportamiento del nuevo ajuste de la longitud de paso con el problema de *Kepler* con excentricidad 0.5, es decir, un problema relativamente fácil en el que el error global no crece tan rápidamente, ya que crece de forma cuadrática. En este caso la nueva estrategia actúa como si se tratara de un cambio de tolerancia: no ofrece grandes diferencias respecto a la estrategia usual. Todas la líneas de eficiencia se sobreponen, hay algunas lineas que para alguna tolerancia son un poco mejores que otras. Quizás la línea correspondiente a $K = 0{,}1$ muestra una insignificante ganancia respecto a la estrategia usual cuando las tolerancias son exigentes, pero en general, todas muestran comportamientos parecidos, por lo que se puede decir que en este caso no hay ni ganancias ni pérdidas de eficiencia.

Otra posible cuestión que se puede plantear es si la estimación del error global puede verse afectada por la elección de otra secuencia de longitudes de paso. Hemos realizado muchos experimentos numéricos para analizar y aclarar esta cuestión y hemos visto que la calidad de la estimación del error global no se ve afectada por la nueva selección de la longitud de paso. Los resultados y las gráficas obtenidas son muy parecidas a las mostradas en la Sección 2.15 del capítulo sobre el error global. Hay que tener en cuenta que la motivación teórica dada en la Sección 2.3 no depende de la secuencia $\{h_n\}$ utilizada en la integración numérica del problema. Lo único requerido es que cada h_n sea lo suficientemente pequeño, de forma que la estimación (2.43)

$$\psi_h(y+e,y) - \phi_h(y) = O(h^{q_0} + h^{q_1}||e|| + h^{q_2}||e||^2 + \cdots)$$

tenga sentido. Por tanto, mientras se cumpla el criterio (2.43), es de esperar que las estimaciones del error global sean parecidas.

Figura 3.7: Comparación de costos para el problema de *Kepler* con excentricidad 0.5: nueva estrategia de ajuste de longitud de paso y la estrategia usual



3.6. Conclusiones

Podemos concluir diciendo que la nueva estrategia basada en el control del error local (3.10) y en la tolerancia variable (3.12) ofrece la posibilidad de un ahorro computacional considerable para los problemas que presentan una propagación de los errores que hace que el error global crezca rápidamente, mientras que para los problemas que no presentan esa característica sigue siendo una estrategia tan válida como la usual. Su implementación permite al usuario la elección de la estrategia pero le obliga a proporcionar el valor del parámetro K (3.13) que depende del problema a resolver. Sería interesante disponer de algún método heurístico que eligiera para cada problema un valor de K cercano al óptimo.

Capítulo 4

Comparación de los métodos de Runge-Kutta

4.1. Introducción

En este capítulo presentamos un procedimiento general para obtener estimaciones rigurosas del error local de los métodos de Runge-Kutta aplicados a Ecuaciones Diferenciales Ordinarias. Bajo la suposición de que f es analítica real en un conjunto abierto del espacio de fases daremos un procedimiento para obtener, para cada método particular de orden p, una estimación del error local $\delta(y,h)$ de la forma $||\delta(y,h)|| \leq h\, C(y)\, D(hL(y))$, donde $C(y)$ y $L(y)$ son constantes que solo dependen de f y de y, y la función $D(\tau)$ depende sólo del tablero de Butcher del esquema de Runge-Kutta. La función del error local $D(\tau)$ está definida para $0 \leq \tau \leq \kappa$ donde $\kappa > 0$ es una constante que depende del tablero de Butcher. De esta manera el error local se puede estimar para todo $h \leq \frac{1}{\kappa L(y)}$.

La correspondiente función $D(\tau)$ puede ser utilizada para comparar la precisión teórica de los diferentes métodos de Runge-Kutta. Es decir, proponemos la utilización de $D(\tau)$ como indicador del comportamiento del error local cometido por el esquema de Runge-Kutta aplicado a Ecuaciones Diferenciales Ordinarias generales que cumplen nuestras suposiciones. Mostraremos experimentos numéricos que apoyan esta propuesta.

Podremos evaluar la eficiencia teórica de diferentes esquemas de Runge-Kutta con el mismo número de etapas efectivas, incluso métodos de diferentes órdenes, comparando los diagramas de sus correspondientes funciones $D(\tau)$. Usamos este enfoque para elegir entre diferentes métodos de Runge-Kutta los esquemas más eficientes para cada rango de tolerancias.

4.2. Notación y resultados básicos

Dado un sistema de ecuaciones diferenciales ordinarias autónomo (1.3) consideremos la aplicación de un paso $\psi_h : \mathbb{R}^d \to \mathbb{R}^d$ a dicho sistema tal y como hemos definido en (1.9)–(1.10).

Las etapas internas Y_i del método definidas en (1.10) pueden ser consideradas en sí mismas como métodos de Runge-Kutta. Así, cualquier esquema de Runge-Kutta $\psi_h(y)$ se puede escribir de forma unívoca de la siguiente manera

$$\psi_h(y) = y + h \sum_{i=1}^{s} b_i f(\psi_h^i(y)), \tag{4.1}$$

donde $b_i \neq 0$, $i = 1, \ldots, s$, y $\psi_h^i(y)$ es un esquema de Runge-Kutta.

4.2.1. Árboles con raíz, diferenciales elementales y B-series

Sea $\mathcal{T}_n$ el conjunto de árboles con raíz de n vértices, y $\mathcal{T} = \cup_{n\geq 1}\mathcal{T}_n$ el conjunto de todos los árboles con raíz. Como decíamos en la sección 1.8, el árbol con un único vértice lo denotamos como $\bullet$, y el árbol $t = [t_1 \cdots t_k] \in \mathcal{T}$ lo representamos uniendo las raíces de $t_1, \cdots, t_k$ a un nuevo vértice que es la raíz de t. La raíz la colocamos debajo de las raíces de los árboles que representan a $t_1, \cdots, t_k$. Es posible que en los árboles $t_1 \cdots t_k$ haya árboles repetidos, por lo que a veces nos interesará denotar al árbol t como $[t_1^{j_1} \cdots t_m^{j_m}]$ donde j_i indica el número de veces que aparece el árbol t_i, $i = 1, \ldots, m$.

Recordemos que el número de vértices de $t \in \mathcal{T}$ lo denotamos como $|t|$, es decir, $|\bullet| = 1$ y $|[t_1 \cdots t_k]| = 1 + |t_1| + \cdots + |t_k|$ para $t_1, \cdots, t_k \in \mathcal{T}$.

Para cada $t \in \mathcal{T}$ denotamos como $\omega(t)$ el número racional positivo definido de forma recursiva como

$$\begin{aligned} \omega(\bullet) &= 1, \\ \omega([t_1^{j_1} \cdots t_m^{j_m}]) &= k^k \prod_{i=1}^{m} \left(\frac{\omega(t_i)}{j_i} \right)^{j_i}, \end{aligned} \tag{4.2}$$

donde $k = \sum_{i=1}^{m} j_i$, $t = [t_1^{j_1} \cdots t_m^{j_m}]$, y los árboles $t_1, \cdots, t_m$ son distintos dos a dos.

Como ya hemos visto en la Sección 1.7, tanto el h-flujo ϕ_h, definido en (1.13), como la aplicación de un paso ψ_h de un método de Runge-Kutta (4.1), pueden ser desarrollados formalmente como B-series $\phi_h(y) = B(\phi, y)$,

$\psi_h(y) = B(\psi, y)$, con $\phi(\emptyset) = \psi(\emptyset) = 1$, donde la definición de B-serie viene dada por (1.25). Además, Y_i $(1 \leq i \leq s)$ admite un desarrollo en B-serie de la forma $B(\psi_i, y)$, con $\psi_i(\emptyset) = 1$.

Los coeficientes $\phi(t)$ para cada $t \in \mathcal{T}$ de la B-serie correspondiente al h-flujo vienen dados por $\frac{1}{\gamma(t)}$, con $\gamma(t)$ dado en (1.40), o de forma equivalente por (1.39) que volvemos a mostrar aquí:

$$\begin{aligned} \phi(\bullet) &= 1, \\ \text{y para} \quad t &= [t_1 \cdots t_k] \\ \phi(t) &= \frac{\phi(t_1) \cdots \phi(t_m)}{\rho(t)}, \end{aligned} \tag{4.3}$$

donde $\rho(t)$ indica el orden del árbol t y equivale al número de vértices del árbol.

Los coeficientes $\psi(t)$ y $\psi_i(t)$ para cada $t \in \mathcal{T}$ son polinomios en los parámetros b_i, a_{ij}, y vienen dados por las recursiones (1.37) y (1.36) respectivamente. Para facilitar la lectura mostramos las dos recursiones de nuevo:

$$\begin{aligned} \psi(\bullet) &= \sum_{i=1}^{s} b_i, \\ \psi_i(\bullet) &= c_i = \sum_{j=1}^{s} a_{ij}, \\ \text{y para} \quad t &= [t_1 \cdots t_k] \\ \psi(t) &= \sum_{i=1}^{s} b_i \psi_i(t_1) \cdots \psi_i(t_k), \end{aligned} \tag{4.4}$$

$$\psi_i(t) = \sum_{j=1}^{s} a_{ij} \psi_j(t_1) \cdots \psi_j(t_k). \tag{4.5}$$

Según la definición del error local de un método numérico dado en (1.15) y teniendo en cuenta las expansiones en forma de B-series de $\phi_h(y)$ y de $\psi_h(y)$ tenemos:

Proposición 1 *El error local (1.15) puede ser desarrollado como una B-serie $B(\delta, y)$ con*

$$\begin{aligned} &\delta(\emptyset) = 0, \\ &\delta(t) = \psi(t) - \phi(t) \ \textit{para} \ t \in \mathcal{T}. \end{aligned} \tag{4.6}$$

Recordemos que en la Sección 1.8 al definir las B-series mediante (1.25) asociamos a cada árbol con raíz $t \in \mathcal{T}$ una *diferencial elemental* $F(t) : \mathbb{R}^d \to \mathbb{R}^d$, y teniendo en cuenta (1.33) junto con el Lema 1 tenemos que

$$\begin{aligned} F(\bullet) &= f \\ F([t_1 \cdots t_k])(y) &= f^{(k)}(y)(F(t_1)(y), \ldots, F(t_k)(y)). \end{aligned} \tag{4.7}$$

Claramente, las diferenciales elementales no dependen de los parámetros b_i, a_{ij} del esquema de Runge-Kutta. Además, normalizamos cada término de la B-serie con cierto número $\sigma(t)$ (la *simetría* del árbol con raíz t) definido en (1.35).

Podemos escribir el desarrollo en serie de potencias de h del error local como

$$\delta(y, h) = \sum_{n \geq p+1} h^n \delta_n(y) \tag{4.8}$$

donde p es el orden de consistencia del método. Sabemos por la Proposición 1 que (4.8) se puede expresar como una B-serie, es decir, cada $\delta_n(y)$ es una combinación lineal de diferenciales elementales, una diferencial elemental por cada árbol con raíz de n vértices, y por tanto, combinando (4.8) con la B-serie dada en la Proposición 1, podemos escribir el desarrollo en serie de potencias de h de $\delta(y, h)$ como

$$\delta(y, h) = \sum_{n \geq p+1} h^n \sum_{t \in \mathcal{T}_n} \frac{\delta(t)}{\sigma(t)} F(t)(y), \tag{4.9}$$

donde $\mathcal{T}_n$ es el conjunto de árboles de n vértices y, para cada árbol con raíz t, el coeficiente $\delta(t)$ es un polinomio de los parámetros $b_i, a_{i,j}$ del método de Runge-Kutta dado por (1.9)-(1.10).

Podemos obtener estimaciones rigurosas de $\delta(y, h)$ acotando las diferenciales elementales $F(t)(y)$. Las suposiciones básicas sobre el sistema (1.1) y algunas de las técnicas utilizadas en este capítulo están inspiradas en la teoría del análisis regresivo del error de los integradores numéricos para ecuaciones diferenciales ordinarias [1], [9], [21].

En [9], Hairer y Lubich obtienen cotas para las diferenciales elementales $F(t)$ (también para los coeficientes de $\delta(t)$) que pueden ser utilizadas para obtener cotas rigurosas para el error local de los métodos de Runge-Kutta. Asumen que f es una función analítica real en y, y que $||f(z)||_1 \leq M$ para todo $z \in \mathbb{C}^d$ tal que $||z - y||_\infty \leq R$ (donde $|| \cdot ||_1$ es la norma-1 y $|| \cdot ||_\infty$ es la norma del máximo), y obtienen cotas para cada $||F(t)(y)||_1$ que pueden

ser expresadas de la forma

$$||F(t)(y)||_1 \leq \nu(t)\, M \left(\frac{M}{R}\right)^{n-1}, \tag{4.10}$$

donde n es el número de vértices del árbol con raíz t, y $\nu(t)$ es un número asociado a t. Es de señalar que estas cotas dependen en general de la dimensión d del sistema: considérense las diferenciales elementales del sistema $\dot{y}_1 = g(y_1)$, $\dot{y}_2 = g(y_2), \ldots, \dot{y}_d = g(y_d)$ con valor inicial $(y_1, y_2, \ldots, y_d) = (y_0, y_0, \ldots, y_0)$, donde $g : \mathbb{R} \to \mathbb{R}$. La diferencial elemental correspondiente a cada $t \in \mathcal{T}$ de este sistema evaluadas en $(y_0, y_0, \ldots, y_0) \in \mathbb{R}^d$ coinciden con d copias de la diferencial elemental del sistema simple $\dot{y} = g(y)$, con valor inicial $y = y_0$. Supongamos que $||g(z)||_1 \leq M$ para todo $z \in \mathbb{C}^d$ tal que $||z - y||_\infty \leq R$, por lo que en el sistema simple obtendríamos las cotas de las diferenciales elementales dependientes de M y de R, dadas en (4.10). Pero en el caso a considerar, tenemos que $||g(z)||_1 \leq d \cdot M$ para todo $z \in \mathbb{C}^d$ tal que $||z - y||_\infty \leq R$, por lo que tendríamos las siguientes cotas para la diferencial elemental $F(t)$ del sistema de dimensión d

$$||F(t)(y)||_1 \leq \nu(t)\, d \cdot M \left(\frac{d \cdot M}{R}\right)^{n-1} \leq \nu(t)\, M \left(\frac{M}{R}\right)^{n-1} d^n.$$

Como alternativa, consideramos suposiciones que solo implican una única (y arbitraria) norma (en lugar de las dos mencionadas, la norma-1 y la norma máxima) que nos proporciona cotas de $||F(t)(y)||$ que son independientes de la dimensión d. Nuestra suposición 1 está estrechamente relacionada con las consideradas (para la norma máxima) por Reich [21] (ver también [12]).

4.2.2. Cotas de las derivadas de las funciones analíticas

Si una aplicación $r : \{s \in \mathbb{C} : \ |s| \leq \rho\} \to \mathbb{C}^l$ es analítica, entonces las estimaciones de Cauchy dan las cotas

$$||r^{(j)}(0)||_l \leq \frac{j!}{\rho^j} \sup_{|s| \leq \rho} ||r(s)||_l, \tag{4.11}$$

donde $||\cdot||_l$ es una norma arbitraria en $\mathbb{C}^l$.

Para cada $s \in \mathbb{C}$ tal que $|s| < \rho$, podemos acotar cada coeficiente de la serie de Taylor mediante las estimaciones de Cauchy, y el resto del desarrollo de Taylor truncado puede ser representado como

$$r(s) - \sum_{j=0}^{n-1} \frac{s^j}{j!} r^{(j)}(0) = \frac{s^n}{2\pi i} \oint_{|w|=\rho} \frac{r(w)}{w^n (w - s)}\, dw,$$

lo que nos lleva a la cota

$$\left|\left|r(s)-\sum_{j=0}^{n-1}\frac{s^j}{j!}r^{(j)}(0)\right|\right|_l \leq \xi(|s|/\rho)\left(\frac{|s|}{\rho}\right)^n \sup_{|s|\leq\rho}||r(s)||_l, \tag{4.12}$$

donde para cada $\tau \geq 0$,

$$\xi(\tau)=\int_0^1 \frac{1}{\sqrt{1+\tau^2-2\,\tau\,\cos(2\,\pi\theta)}}\,d\theta < \frac{1}{1-\tau}. \tag{4.13}$$

Las estimaciones de Cauchy se pueden generalizar a las aplicaciones analíticas $r:\mathcal{V}\subset\mathbb{C}^m\to\mathbb{C}^l$ de la siguiente manera. Dados $\rho_1,\cdots,\rho_m>0$,

$$\left|\left|r^{(j_1,\cdots,j_m)}(0)\right|\right|_l \leq \frac{j_1!}{\rho_1^{j_1}}\cdots\frac{j_m!}{\rho_m^{j_m}}\sup_{|s_i|\leq\rho_i}||r(s)||_l, \tag{4.14}$$

donde $s=(s_1,\cdots,s_m)$, y $r^{(j_1,\cdots,j_m)}(s)=\frac{\partial^{j_1}}{\partial s_1^{j_1}}\cdots\frac{\partial^{j_m}}{\partial s_m^{j_m}}r(s_1,\ldots,s_m)$. Supongamos ahora que tenemos una cota de $||r(s)||_l$ para $|s|_1\leq 1$ ($|s|_1=|s_1|+\cdots+|s_m|$). En este caso, (4.14) aplicado a $\rho_1,\ldots,\rho_m>0$ arbitrarios tales que $\rho_1+\cdots+\rho_m=1$ nos da la cota

$$\left|\left|r^{(j_1,\cdots,j_m)}(0)\right|\right|_l \leq \frac{j_1!}{\rho_1^{j_1}}\cdots\frac{j_m!}{\rho_m^{j_m}}\sup_{|s|_1\leq 1}||r(s)||_l.$$

Para obtener la mejor opción de los valores de ρ_i, debemos hallar el mínimo de $\frac{j_1!}{\rho_1^{j_1}}\cdots\frac{j_m!}{\rho_m^{j_m}}$ sujeto a la restricción $\rho_1+\cdots+\rho_m=1$. Podemos hallar la solución de dicho problema de minimización con restricciones utilizando el método de los multiplicadores de Lagrange, lo que nos lleva a

$$\rho_k=\frac{j_k}{j_1+j_2+\ldots+j_m},\ k=1,\ldots,m.$$

Por tanto, (4.14) implica el siguiente resultado:

Lema 5 *Sea $r:\mathcal{V}\in\mathbb{C}^m\to\mathbb{C}^d$ analítica, entonces se cumple que*

$$\left|\left|r^{(j_1,\cdots,j_m)}(0)\right|\right|_l \leq (j_1+\cdots+j_m)^{j_1+\cdots+j_m}\frac{j_1!}{j_1^{j_1}}\cdots\frac{j_m!}{j_m^{j_m}}\sup_{|s|_1\leq 1}||r(s)||_l. \tag{4.15}$$

Ahora nos interesa obtener cotas para las derivadas de frêchet $g^{(k)}(y)$ (de orden k) de una aplicación analítica $g:\mathcal{U}\subset\mathbb{C}^d\to\mathbb{C}^l$ para $y\in\mathcal{U}$. Más exactamente, queremos obtener cotas del resultado $g^{(k)}(y)(v_1,\ldots,v_k)$,

es decir, cotas de la aplicación k-lineal $g^{(k)}(y)$ actuando sobre los vectores $v_1, \ldots, v_k \in \mathbb{C}^d$. Si se da el caso de que algunos de los vectores de $(v_1, \ldots, v_k)$ se repiten se pueden obtener mejores cotas. En este contexto puede ser más conveniente el uso de la notación $(v_1^{j_1}, \ldots, v_m^{j_m}) := (v_1, \ldots, v_k)$, donde j_i indica el número de copias de v_i para $i = 1, \ldots, m$ $(j_1 + \cdots + j_m = k)$. Con esta notación, se cumple que

$$g^{(k)}(y)(v_1^{j_1}, \ldots, v_m^{j_m}) = r^{(j_1, \cdots, j_m)}(0), \tag{4.16}$$

donde $r(s_1, \ldots, s_m) = g(y + \sum_{i=1}^m s_i v_i)$.

Lema 6 *Sea $g : \mathcal{U} \subset \mathbb{C}^d \to \mathbb{C}^l$ analítica, $y \in \mathcal{U}$, $v_1, \cdots, v_m \in \mathbb{R}^D$, y sean $||\cdot||_d$, $||\cdot||_l$ normas en $\mathbb{C}^d$ y $\mathbb{C}^l$ respectivamente. Sean $j_1, \cdots, j_m$ enteros positivos tales que $k = j_1 + \cdots + j_m \geq 1$, entonces se cumple que*

$$||g^{(k)}(y)(v_1^{j_1}, \ldots, v_m^{j_m})||_l \leq k^k \frac{j_1!}{j_1^{j_1}} \cdots \frac{j_m!}{j_1^{j_m}} ||v_1||_d^{j_1} \cdots ||v_m||_d^{j_m} \sup_{||z-y||_d \leq 1} ||g(z)||_l. \tag{4.17}$$

Demostración Debido a la k-linealidad de $g^{(k)}(y)$, podemos asumir sin perdida de generalidad que $||v_i||_d = 1$, $1 \leq i \leq m$. Ahora, (4.16) junto con (4.15) nos lleva a (4.17). □

Observación 4 *La ecuación (4.16) también se cumple para $r(s) = g(y + \sum s_i v_i) - g(y)$, y por tanto el Lema 6 también se cumple con* $\sup ||g(z)||_l$ *reemplazado por* $\sup ||g(z) - g(y)||_l$.

4.3. Cotas para el error local de los métodos RK

Nuestra intención es la obtención de un procedimiento que nos dé cotas significativas del error local (1.15) de un paso de un método de Runge-Kutta (1.9) aplicado a un $y \in \mathcal{U}$ dado para un sistema dado (1.3). Utilizaremos las cotas proporcionadas por las estimaciones de Cauchy, por lo que tenemos que asumir las condiciones en las que se pueden aplicar dichas estimaciones. Haremos la siguiente suposición

Suposición 1 *Dado $y \in \mathcal{U} \subset \mathbb{R}^d$ y una norma $||\cdot||$ en $\mathbb{C}^d$, la transformación $f : \mathcal{U} \subset \mathbb{R}^d \to \mathbb{R}^d$ puede ser extendida analíticamente a*

$$\{z \in \mathbb{C}^d : ||z - y|| \leq 1\}.$$

En lo que sigue, denotamos

$$L = \sup_{||z-y|| \leq 1} ||f(z)||. \tag{4.18}$$

Primero vamos a acotar, bajo la Suposición 1, las diferenciales elementales $F(u)(y)$, de forma que los primeros términos de la expansión (4.9) puedan ser acotadas para cada esquema particular de Runge-Kutta. Seguidamente obtendremos cotas para el resto del desarrollo de Taylor truncado del error local $\delta(y,h)$.

4.3.1. Cotas de las diferenciales elementales

Lema 7 *Bajo la suposición 1, se cumple para cada árbol con raíz $t \in \mathcal{T}$ que*

$$\frac{1}{\sigma(t)}||F(t)(y)|| \leq \omega(t)\, L^{|t|}, \tag{4.19}$$

donde $\sigma(t)$ y $\omega(t)$ están definidas recursivamente en (1.35) y (4.2) respectivamente.

Demostración Mediante la inducción sobre el número de vértices $|t|$ de los árboles. Es trivial para $|t| = 1$ (es decir $t = \bullet$), ya que $F(\bullet)(y) = f(y)$, $\sigma(\bullet) = 1$ y $\omega(\bullet) = 1$. Esto nos lleva a $||f(y)|| \leq L$ que se cumple por la Suposición 1. Para un árbol dado $t \in \mathcal{T}$ con $|t| > 1$, sean $t_1, \ldots, t_m$ m árboles con raíz distintos de forma que $t = [t_1^{r_1} \cdots t_m^{r_m}]$. Por definición de $\sigma(t)$, dada en (1.35), y de $F(t)(y)$, dada en (4.7), tenemos que

$$\frac{1}{\sigma(t)}||F(t)(y)|| = \frac{1}{r_1!\cdots r_m!\sigma(t_1)^{r_1}\cdots\sigma(t_m)^{r_m}}||f^{(k)}(y)(F(t_1)(y)^{r_1}\cdots F(t_m)(y)^{r_m})||,$$

entonces la aplicación del Lema 6 con $v_i = F(t_i)(y)$, $(1 \leq i \leq m)$, $g = f$, y la norma $||\cdot||$, nos lleva a

$$\frac{1}{r_1!\cdots r_m!\sigma(t_1)^{r_1}\cdots\sigma(t_m)^{r_m}}||f^{(k)}(y)(F(t_1)(y)^{r_1}\cdots F(t_m)(y)^{r_m})|| \leq \frac{1}{\sigma(t_1)^{r_1}\cdots\sigma(t_m)^{r_m}} L k^k \frac{1}{r_1^{r_1}\cdots r_m^{r_m}} ||F(t_1)(y)||^{r_1}\cdots||F(t_m)(y)||^{r_m}$$

donde podemos aplicar la hipótesis de inducción a los subárboles t_i, y llegamos a

$$\frac{1}{\sigma(t)}||F(t)(y)|| \leq L k^k \prod_{i=1}^{m}\left(\frac{1}{r_i}\omega(t_i)L(y)^{|t_i|}\right)^{r_i} = L^{1+|t_1|r_1+\cdots|t_m|r_m} k^k \prod_{i=1}^{m}\left(\frac{\omega(t_i)}{r_i}\right)^{r_i} \tag{4.20}$$

que nos lleva a la desigualdad requerida, teniendo en cuenta el número de vértices del árbol t, $|t| = 1 + |t_1|r_1 + \cdots + |t_m|r_m$, y la definición de $\omega(t)$, dada en (4.2). □

Ejemplo 1 *Consideremos el problema de* Kepler *de los dos cuerpos,* $\ddot{q} = -q/(q_1^2 + q_2^2)^{3/2}$*, donde* $q = (q_1, q_2)$*. Podemos escribir el problema como un sistema de ecuaciones diferenciales ordinarias de primer orden (1.3), con* $y = (q, p)$ *y* $f(y) = (p, -q/(q_1^2 + q_2^2)^{3/2})$*. Consideramos la norma en* $\mathbb{C}^4$ *dada por* $||(Q, P)|| = \max(||Q||, ||P||)$*, donde* $||Q|| = \sqrt{|Q_1|^2 + |Q_2|^2}$ *para cada* $Q \in \mathbb{C}^2$*. Para cada* $Q, P \in \mathbb{C}^2$*, denotamos* $\langle P, Q\rangle = P_1\overline{Q}_1 + P_2\overline{Q}_2$*. Queremos ver si la Suposición 1 se cumple con esta norma para* $(q, p) \in \mathbb{R}^4$ *dado. La función* $f(Q, P)$ *es analítica en* $Q, P \in \mathbb{C}^4$ *si* $Q_1^2 + Q_2^2 \neq 0$*, y* $||f(Q, P)|| = \max(||P||, K(Q))$*, donde*

$$K(Q) = \frac{(|Q_1|^2 + |Q_2|^2)^{1/2}}{|Q_1^2 + Q_2^2|^{3/2}} = \frac{||Q||}{|\langle Q, \overline{Q}\rangle|^{3/2}}. \tag{4.21}$$

Buscamos primero una cota inferior de $|\langle Q, \overline{Q}\rangle|$ *suponiendo que* $||(Q, P) - (q, p)|| \leq 1$*, y por tanto* $||\Delta Q|| \leq 1$*, donde* $\Delta Q = Q - q$*.*

Tenemos que

$$\langle Q, \overline{Q}\rangle = \langle q, \overline{q}\rangle + \langle \Delta Q, \overline{\Delta Q}\rangle + \langle \Delta Q, \overline{q}\rangle + \langle \overline{\Delta Q}, q\rangle,$$

y por tanto,

$$||q||^2 = |\langle Q, \overline{Q}\rangle - \langle \Delta Q, \overline{\Delta Q}\rangle - \langle \Delta Q, \overline{q}\rangle - \langle \overline{\Delta Q}, q\rangle|,$$

y aplicando la desigualdad triangular y la de Schwartz llegamos a

$$||q||^2 \leq |\langle Q, \overline{Q}\rangle| + ||\Delta Q||^2 + 2||\Delta Q||||q||,$$

y finalmente

$$|\langle Q, \overline{Q}\rangle| \geq ||q||^2 - ||\Delta Q||^2 - 2||\Delta Q||\,||q||.$$

Así pues, $|\langle Q, \overline{Q}\rangle| \geq 0$ *siempre que* $||q||^2 \geq ||\Delta Q||^2 + 2||\Delta Q||\,||q||$*, es decir, si* $||q|| \geq (1 + \sqrt{2})||\Delta Q||$*, en cuyo caso,*

$$K(Q) \leq \frac{||q|| + ||\Delta Q||}{(||q||^2 - 2||q||\,||\Delta Q|| - ||\Delta Q||^2)^{3/2}}. \tag{4.22}$$

Entonces la Suposición 1 se cumple en $y = (q, p)$ *con* $L \leq \max(||p|| + 1, (||q|| + 1)/(||q||^2 - 2||q|| - 1)^{3/2})$ *siempre que* $||q|| > 1 + \sqrt{2}$*. En ese caso, el Lema 7 nos permite acotar cada diferencial elemental* $F(u)(q, p)$*. No obstante, tales cotas se pueden ajustar considerablemente, en la manera que veremos en la Subsección 4.3.3.*

Observación 5 *Se puede demostrar en la misma línea de la demostración del Lema 7 que, bajo la Suposición 1, se cumple para cada árbol con raíz $t \in \mathcal{T}$ que*

$$\frac{1}{\sigma(t)}||F(t)(y)|| \leq \omega(t)\,||f(y)||^{\langle t\rangle}\,L^{|t|-\langle t\rangle}, \tag{4.23}$$

donde $\langle t\rangle$ indica el número de hojas *de $t \in \mathcal{T}$, que se define recursivamente como $\langle \bullet\rangle = 1$ y $\langle [t_1 \cdots t_k]\rangle = \langle t_1\rangle + \cdots + \langle t_k\rangle$. Además, la Observación 4 muestra que L (4.18) puede ser sustituido por*

$$L^* = \sup_{||z-y||\leq 1} ||f(z) - f(y)||. \tag{4.24}$$

En general, (4.23) (con o sin L sustituido por L^*) nos dará cotas más ajustadas que (4.19) para cada diferencial elemental, pero la correspondiente cota de la B-serie tiene una estructura más compleja. Sin embargo, (4.23) muestra que $L^{|t|}$ en (4.19) puede ser sustituido por $\min(L^*, ||f(y)||)^{|t|-1}||f(y)||$, *o también por*

$$L^{|t|-1}||f(y)||. \tag{4.25}$$

Corolario 1 *Dada una B-serie (1.25), el término correspondiente a la n-sima potencia de h puede ser acotado bajo la Suposición 1 como*

$$\left|\left|\sum_{t\in\mathcal{T}_n} \frac{c(t)}{\sigma(t)}F(t)(y)\right|\right| \leq L^n \sum_{t\in\mathcal{T}_n} \omega(t)|c(t)|. \tag{4.26}$$

Observación 6 *Se pueden obtener mejores cotas que (4.26) para combinaciones lineales particulares de diferenciales elementales. Por ejemplo, se puede probar que*

$$\left|\left|\sum_{t\in\mathcal{T}_n} \frac{1}{\sigma(t)\gamma(t)}F(t)(y)\right|\right| \leq L^n.$$

(Esto se puede demostrar considerando el Lema 9 de más adelante, aplicando las estimaciones de Cauchy y teniendo en cuenta que la expansión en B-series de la solución exacta es $B(1/\gamma, y)$). Cotas de combinaciones lineales de diferenciales elementales de esta forma nos pueden llevar a mejores cotas que (4.26) para B-series particulares (se puede ver, por ejemplo, que esto sucede con las B-series correspondientes a métodos de Runge-Kutta implícitos con simplificaciones en las condiciones de orden del tipo $\sum_j a_{ij}c_j^{n-1} = \frac{c_i^n}{n}$).

Ejemplo 2 *Consideremos el modelo de Lotka-Volterra*

$$\dot{u} = u(v-2),$$
$$\dot{v} = v(1-u).$$

Queremos obtener cotas para $F(u)(2,3)$*, con la norma del máximo, aplicando el Lema 7. Por tanto, tenemos que calcular* L *de la Suposición 1 para* $y = (2,3)$ *y* $f(u,v) = (u(v-2), v(1-u))$*. Si* $u, v \in \mathbb{C}$ *son tales que* $||(u,v) - (2,3)|| \leq 1$*, tenemos que*

$$|u-2| \leq 1 \text{ y } |v-3| \leq 1,$$

y en particular, $|u| \leq 3$*,* $|u-1| \leq 2$*,* $|v| \leq 4$*,* $|v-2| \leq 2$*, de modo que el supremo* L *de* $||f(u,v)|| = \max(|u(v-2)|, |v(1-u)|)$ *para* $u, v \in \mathbb{C}$ *tales que* $||(u,v) - (2,3)|| \leq 1$ *satisface la desigualdad* $L \leq \max(6,8) = 8$*. Además,* $||f(3,4)|| = 8$*, de modo que para la norma del máximo* $L = 8$*, lo que nos permite acotar cada diferencial elemental aplicando el Lema 7.*

4.3.2. Cotas para la expansión de Taylor del error local

Consideremos el desarrollo en serie de potencias de h de un paso $\psi_h(y)$ de un esquema de Runge-Kutta (1.9)

$$\psi_h(y) = y + \sum_{n \geq 1} h^n g_n(y), \tag{4.27}$$

donde para cada $n \geq 1$, $g_n(y) = \sum_{t \in \mathcal{T}_n} \frac{\psi(t)}{\sigma(t)} F(t)(y)$, con $\psi(t)$ dado en (4.4) y siendo $\mathcal{T}_n$ el conjunto de árboles con n vértices.

El Corolario 1 junto con el desarrollo en B-serie (4.9) da las siguientes cotas para los términos del error local y para los términos $g_n(y)$ de (4.27).

Lema 8 *Bajo la Suposición 1, el* n*-ésimo término* $\delta_n(y)$ *del error local (1.15), (4.8) del esquema de Runge-Kutta (1.9) y el* n*-ésimo término* $g_n(y)$ *de (4.27) pueden ser acotados como*

$$||\delta_n(y)|| \leq L^n \, d'_n, \quad d'_n = \sum_{u \in \mathcal{T}_n} \omega(u) \left| \psi(u) - \frac{1}{\gamma(u)} \right|, \tag{4.28}$$

$$||g_n(y)|| \leq L^n \, \hat{d}'_n, \quad \hat{d}'_n = \sum_{u \in \mathcal{T}_n} \omega(u) |\psi(u)|. \tag{4.29}$$

Donde $\psi(u)$ *está dado en 4.4,* $\omega(u)$ *en 4.2 y* $\gamma(u)$ *es la densidad del árbol* u *dada por (1.40).*

Acotando las diferenciales elementales, y usando el Lema 8 podemos acotar cada término de la expansión de Taylor del error local (1.15). Pero para acotar el error local tenemos que truncar la expansión de Taylor y necesitamos alguna cota para el resto. Considerando (4.8) como una función de la variable compleja h, las estimaciones de Cauchy nos pueden ayudar a acotar tanto los términos como el resto de la expansión de Taylor truncada.

Bajo la Suposición 1 podemos obtener esas cotas obteniendo una segunda cota para los términos de la expansión de Taylor. La comparación de las dos cotas nos va a servir de guía a la hora de la toma de decisión de cómo truncar la expansión de Taylor del error local.

Sea ϕ_h la extensión compleja ($h \in \mathbb{C}$) del h-flujo de (1.1). Lo siguiente es un resultado estándar

Lema 9 *Bajo la Suposición 1, $\phi_h(y)$ es analítico en $\{h \in \mathbb{C} : \ |h| \leq \frac{1}{L}\}$, y*

$$||\phi_h(y) - y|| \leq |h|L, \quad para \quad |h| \leq \frac{1}{L}. \tag{4.30}$$

Nos interesa un resultado similar para $\psi_h(y)$.

Definición 1 *Dado un esquema de Runge-Kutta explícito $\psi_h(y)$, consideremos los números reales positivos $K > 0$ tales que para cada ODE (1.1) y cada norma en $\mathbb{R}^d$ para los que cumpla la Suposición 1, $\psi_h(y)$ es analítico en $\{h \in \mathbb{C} : \ |h| \leq \frac{1}{KL}\}$ y $||\psi_h(y) - y|| \leq |h|KL$ para $|h| \leq \frac{1}{KL}$. Denotamos como κ el ínfimo de tales K.*

Veremos que κ está bien definida:

Proposición 2 *Para cada esquema de Runge-Kutta explícito de la forma (4.1), existe $K \geq 0$ tal que la Suposición 1 garantiza que $\psi_h(y)$ es analítica en $\{h \in \mathbb{C} : \ |h| \leq \frac{1}{KL}\}$, y*

$$||\psi_h(y) - y|| \leq |h|KL, \quad para \quad |h| \leq \frac{1}{KL}. \tag{4.31}$$

Demostración Cualquier esquema de Runge-Kutta explícito de s etapas puede escribirse como (4.1) donde cada $\psi_h^i(y)$ es un esquema de Runge-Kutta explícito de s_i etapas con $s_i \leq i < s$. El resultado se cumple de forma trivial cuando $s = 0$ (es decir, $\psi_h(y) = y$). En otro caso, podemos asumir por inducción sobre s que el resultado se cumple para los esquemas de Runge-Kutta ψ_h^i, $i = 1, \ldots, s$. Para cada i, sea κ_i dado por la Definición 1 para el esquema de Runge-Kutta $\psi_h^i(y)$ en (4.1). Por la suposición 1, $f(\psi_h^i(y))$ es analítico (como función de la variable compleja h) y $||f(\psi_h^i(y))|| \leq L$ para

$|h| \leq \frac{1}{\kappa_i L}$. Por lo tanto, $\psi_h(y)$ es analítico y $||\psi_h(y) - y|| \leq |h|L \sum_{i=1}^{s} |b_i|$ para $|h| \leq \frac{1}{\hat{\kappa} L}$, donde $\hat{\kappa} = \text{máx}_{1 \leq i \leq s} \kappa_i$. Finalmente, el resultado requerido se cumple para $K = \text{máx}(\hat{\kappa}, \sum_{i=1}^{s} |b_i|)$. □

Observación 7 *Para esquemas de Runge-Kutta explícitos de la forma (4.1) con $b_i > 0$, $i = 1, \ldots, s$, se cumple que*

$$\kappa = \text{máx}(\kappa_1, \ldots, \kappa_s, \sum_{i=1}^{s} b_i), \tag{4.32}$$

donde κ_i viene dado por la Definición 1 para el esquema de Runge-Kutta $\psi_h^i(y)$ en (4.1). Para verlo, consideremos el problema de valor inicial $y' = 1$ con $y(0) = 0$. Tomemos la norma dada por $||y|| = |y|$. La Suposición 1 se cumple en $y = 0$ para dicha norma, y $||f(z)|| = 1 = L$ para cualquier z tal que $||z - y|| \leq 1$. Por la Definición 1, $|h \sum_{i=1}^{s} b_i| = ||\psi_h(y) - y|| \leq |h|\kappa$ siempre que $|h| \leq \frac{1}{\kappa}$. Por lo tanto $\sum_{i=1}^{s} b_i \leq \kappa$ y de la demostración de la Proposición 2 obtenemos el valor de κ dado en (4.32).

Lema 10 *Bajo la Suposición 1, el n-ésimo término de $\delta_n(y)$ del error local (1.15), (4.8) y el n-ésimo término $g_n(y)$ de la expansión de Taylor (4.27) del esquema de Runge-Kutta (1.9) pueden ser acotados como*

$$||\delta_n(y)|| \leq L^n \, d_n'', \quad d_n'' = 1 + \sum_{i=1}^{s} |b_i| \kappa_i^{n-1}, \tag{4.33}$$

$$||g_n(y)|| \leq L^n \hat{d}_n'', \quad \hat{d}_n'' = \sum_{i=1}^{s} |b_i| \kappa_i^{n-1} \tag{4.34}$$

donde cada κ_i viene dado por la Definición 1 para el esquema de Runge-Kutta $\psi_h^i(y)$ en (4.1). Y para el resto, tenemos

$$||\delta(y,h) - \sum_{j=p+1}^{n-1} \delta_j(y) h^j|| \leq (hL)^n \left(\xi(hL) + \sum_{i=1}^{s} |b_i| \kappa_i^{n-1} \xi(\kappa_i hL) \right),$$

$$||\psi_h(y) - \sum_{j=1}^{n-1} g_j(y) h^j|| \leq (hL)^n \sum_{i=1}^{s} |b_i| \kappa_i^{n-1} \xi(\kappa_i hL).$$

Demostración Acotamos el n-ésimo término de la expansión de Taylor del error local

$$\delta(y,h) = \left(y + h \sum_{i=1}^{s} b_i f(Y_i) \right) - \phi_h(y)$$

acotando independientemente el n-ésimo término de la expansión de $\phi_h(y) - y$ y $hb_i f(Y_i)$, $1 \leq i \leq s$. Según el Lema 9, $\phi_h(y) - y$ es una función analítica de h acotada (en la norma $||\cdot||$) por $|h|L$ para $|h| \leq 1/L$. De la demostración de la Proposición 2, tenemos que $hb_i f(\psi_h^i(y))$ es analítico y está acotado por $|h||b_i|L$ para $|h| \leq 1/(\kappa_i L)$. Por lo tanto, si aplicamos (4.11) y (4.12) por un lado para $r(h) = \phi_h(y) - y$ y por el otro para $r(h) = hb_i f(\psi_h^i(y))$ obtenemos, para la primera

$$||\phi_h(y) - y|| \leq \left(\sum_{j=1}^{n-1} |h|^j L^j + \xi(hL)|h|^n L^n \right) \sup_{|h| \leq \frac{1}{L}} ||\phi_h(y) - y||$$

lo que nos lleva a

$$||\phi_h(y) - y|| \leq \sum_{j=1}^{n-1} h^{j+1} L^{j+1} + \xi(hL)(hL)^{n+1}.$$

Mientras que para la segunda función tenemos que

$$hb_i f(\psi_h^i(y)) \leq \sum_{j=1}^{n-1} h^j (\kappa_i L)^j h|b_i|L + \xi(h\kappa_i L)(h\kappa_i L)^n h|b_i|L$$

Teniendo en cuenta todas las cotas, obtenemos finalmente,

$$\begin{aligned} \delta(y,h) \quad \leq \quad & \sum_{j=1}^{n-1} h^{j+1} L^{j+1} \left(1 + \sum_{i=1}^{s} |b_i| \kappa_i^j \right) + \\ & + h^{n+1} L^{n+1} \left(\xi(hL) + \sum_{i=1}^{s} \xi(h\kappa_i L) \kappa_i^n |b_i| \right) \end{aligned}$$

donde tenemos las estimaciones requeridas. □

El Lema 8 y el Lema 10 implican el siguiente resultado.

Teorema 1 *Bajo la Suposición 1, el error local (1.15), (4.8) de un esquema de Runge-Kutta (1.9) de orden p y $\psi_h(y) - y$ pueden ser acotados, para cada $n \geq p+1$, como $||\delta(y,h)|| \leq D_n(hL)$, y $||\psi_h(y) - y|| \leq \hat{D}_n(hL)$, donde*

$$D_n(\tau) = \sum_{j=p+1}^{n-1} d_j \tau^j + r_n(\tau)\tau^n, \tag{4.35}$$

$$\hat{D}_n(\tau) = \sum_{j=1}^{n-1} \hat{d}_j \tau^j + \hat{r}_n(\tau)\tau^n, \tag{4.36}$$

y $d_n = \min(d'_n, d''_n)$ *(*d'_n *y* d''_n *dados en (4.28) y (4.33) respectivamente),* $\hat{d}_n = \min(\hat{d}'_n, \hat{d}''_n)$ *(*$\hat{d}'_n$ *y* $\hat{d}''_n$ *dados en (4.29) y (4.34) respectivamente), y*

$$r_n(\tau) \;=\; \xi(\tau) + \sum_{i=1}^{s} |b_i| \kappa_i^{n-1} \xi(\kappa_i \tau), \tag{4.37}$$

$$\hat{r}_n(\tau) \;=\; \sum_{i=1}^{s} |b_i| \kappa_i^{n-1} \xi(\kappa_i \tau), \tag{4.38}$$

donde cada κ_i *está determinada como en el Lema 10.*

Si $b_i < 0$ para algún $i \in 1, 2, \ldots, s$, las cotas que se derivan del Teorema 1 para $||\psi_h(y) - y||$ pueden ser utilizadas para acotar κ, siempre que los correspondientes valores κ_i de los esquemas de Runge-Kutta internos $\psi_h(y)$ sean conocidos o se disponga de cotas superiores de los parámetros κ_i. Para $||\psi_h(y) - y||$, una cota alternativa a la dada en el Teorema 1 puede ser obtenida basándonos en la cota del error local $||\psi_h(y) - y|| \leq ||\phi_h(y) - y|| + ||\delta(y, h)|| \leq hL + D_n(hL)$ para $|hL| \leq 1$.

Observación 8 *En los cálculos realizados con diferentes esquemas hemos observado que, si para un esquema de Runge-Kutta de orden p dado,* $n \geq p + 1$ *es tal que* $d'_n \geq d''_n$*, entonces* $d'_j \geq d''_j$ *para todos los* $j \geq n$*. Esto sugiere que una opción adecuada para el índice de truncamiento n en el Teorema 1 resulta ser el primer* $n \geq p + 1$ *tal que* $d'_n \geq d''_n$*. Mediante el siguiente ejemplo mostramos cómo elegimos el índice de truncamiento, y a su vez, lo comparamos con las cotas que se obtienen con otros índices de truncamiento.*

Ejemplo 3 *Para el esquema de Runge-Kutta de segundo orden y de dos etapas dado por el tablero de Butcher*

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

tenemos que $\kappa_1 = 0$ *(como para cualquier método de Runge-Kutta explícito) y de la Observación 7 obtenemos que* $\kappa_2 = 1$*, de forma que el Lema 10*

se cumple con $d''_n = 3/2$ *para todo* n. *El Lema 8 nos da* $d'_3 = 1/3$, $d'_4 = 7/8$, $d'_5 = 8/5$, *y* $d'_n \approx 2d'_{n-1}$ *para mayores índices* n. *Siguiendo el criterio descrito en la Observación 8 elegimos* $n = 5$ *en el Teorema 1, llegando a*

$$D_5(\tau) = \frac{1}{3}\tau^3 + \frac{7}{8}\tau^4 + \frac{3}{2}\xi(\tau)\tau^5.$$

Para el método de Runge-Kutta de orden 3 *y de* 3 *etapas dado en el tablero,*

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ \frac{3}{4} & 0 & \frac{3}{4} & \\ \hline & \frac{2}{9} & \frac{1}{3} & \frac{4}{9} \end{array} \tag{4.39}$$

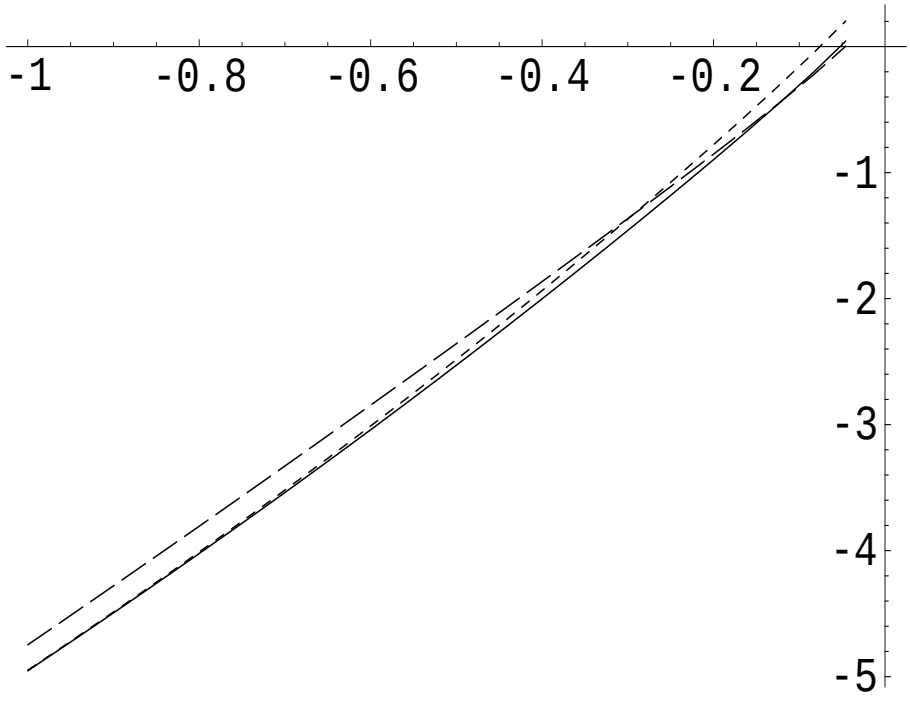
tenemos que la Proposición 2 se cumple con $\kappa_i = c_i$, $i = 1, 2, 3$, *y teniendo en cuenta los valores que obtenemos según el Lema 8 (*$d'_4 = \frac{1}{16}$, $d'_5 = \frac{241}{640}$, $d'_6 = \frac{25141}{20736}$...*) y los obtenidos por el Lema 10 (*$d''_4 = \frac{59}{48}$, $d''_5 = \frac{223}{192}$, $d''_6 = \frac{857}{768} < d'_6$...*), debido al criterio descrito en la Observación 8 tomamos* $n = 6$ *en el Teorema 1, y llegamos a*

$$D_6(\tau) = \frac{1}{16}\tau^4 + \frac{241}{640}\tau^5 + \left(\frac{1}{96}\xi(\frac{\tau}{2}) + \frac{27}{256}\xi(\frac{3\tau}{4}) + \xi(\tau)\right)\tau^6.$$

En la Figura 4.1 mostramos las curvas correspondientes a $D_6(\tau)$, $D_5(\tau)$ *y* $D_7(\tau)$ *del método de RK dado por el tablero de Butcher (4.39) en escala logarítmica (eje vertical), frente a los valores de* τ, *también en escala logarítmica (eje horizontal). La curva correspondiente a* $D_6(\tau)$ *es la curva continua, mientras que la de tramos cortos corresponde a* $D_7(\tau)$ *y la de tramos largos a* $D_5(\tau)$. *Se puede observar que para valores pequeños de* τ *la función* $D_7(\tau)$ *toma valores muy cercanos a* $D_6(\tau)$ *y para valores grandes de* τ *tanto* $D_5(\tau)$ *como* $D_6(\tau)$ *toman valores similares. Teniendo en cuenta todos los valores que puede tomar* τ, *siempre es la curva correspondiente a* $D_6(\tau)$ *la que nos da las menores cotas.*

Observación 9 *Siguiendo las pautas de la demostración de la Proposición 2 y la Observación 7, para esquemas de Runge-Kutta explícitos de la forma (4.1), siempre tenemos que* $\kappa \leq \max(\kappa_1, \ldots, \kappa_s, \sum_{i=1}^s |b_i|)$. *No obstante, en el caso de que algunos valores* b_i *sean negativos, si se da el caso de que* $\sum_{i=1}^s |b_i| > \max(\kappa_1, \ldots, \kappa_s)$, *normalmente suele ser posible mejorar (decrementar) la cota de* κ.

Figura 4.1: Curvas de las cotas de error para el tablero de Butcher 4.39 obtenidas para el Teorema 1 con índices de truncamiento $n = 5, 6, 7$. Donde $n = 6$ (curva continua) es el índice sugerido por la observación 8.



Ejemplo 4 *Consideremos el método de Runge-Kutta dado en el siguiente tablero de Butcher.*

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} \tag{4.40}$$

Tenemos que $\kappa_1 = 0$ *y, como los coeficientes son positivos para* $\psi_h^2(y)$, $\kappa_2 = 1/2$. *De la demostración de la Proposición 2 llegamos a que* $\kappa_3 \leq 3$, *pero es posible que esta cota para* κ_3 *se pueda mejorar, ya que* $c_3 = 1 < 3 = \sum |a_{3j}|$. *Consideremos el método de Runge-Kutta* $\psi_h^3(y)$ *para el cual queremos encontrar un* $K > 0$ *tal que bajo la Suposición 1, si* $h \leq \frac{1}{KL}$, *se cumple que* $||\psi_h^3(y) - y|| \leq |h|KL$.

Sea $K > 0$ *tal que* $\hat{D}_n(\frac{1}{K}) = 1$, *en ese caso, según el Teorema 1 tenemos que* $||\psi_h^3(y) - y|| \leq \hat{D}_n(\frac{hKL}{K})$, *y si* $h \leq \frac{1}{KL}$ *tenemos que* $hKL \leq 1$. *Sea* $s = hKL$. *Según la definición de* $\hat{D}_n(\tau)$ *dada en (4.36), y teniendo en cuenta que para* $s \leq 1$ *se cumple* $\hat{r}_n(\frac{s}{K}) \leq \hat{r}_n(\frac{1}{K})$ *llegamos a*

$$\hat{D}_n\left(\frac{s}{K}\right) \leq s\left(\sum_{j=1}^{n-1} d_j \left(\frac{1}{K}\right)^j + \left(\frac{1}{K}\right)^n \hat{r}_n\left(\frac{1}{K}\right)\right),$$

es decir,

$$||\psi_h^3(y) - y|| \leq s\hat{D}_n\left(\frac{1}{K}\right) \leq |h|KL.$$

Por lo que nos interesa obtener el valor de K *que cumpla*

$$||\psi_h^3(y) - y|| \leq \hat{D}_n(\frac{1}{K}) = 1, \tag{4.41}$$

para cualquier $n > p$. *Por otra parte, teniendo en cuenta que* $\psi_h(y) - y = \phi_h(y) - y + \delta_h(y)$ *y que* $\phi_h(y) - y \leq hL$ *para* $h \leq \frac{1}{L}$, *tenemos una segunda cota del error local y de ahí podemos obtener un valor de* K *que cumpla*

$$||\psi_h^3(y) - y|| \leq |hL| + D_n(\frac{1}{K}) = 1 \tag{4.42}$$

En el caso concreto de nuestro ejemplo, para acotar el error local de ψ_h^3 *mediante (4.42) necesitamos* $D_n(\frac{1}{K})$ *del Teorema 1, y para ello nos hacen falta los valores* d'_n *del Lema 8 y* d''_n *del Lema 10 correspondientes al método de Runge-Kutta* ψ_h^3. *Según (4.33), tenemos* $d''_k = 1 + 2^{2-k}$ *para cada* k, *y mediante (4.28) obtenemos* $d'_1 = 0$, $d'_2 = 1/2$, $d'_3 = 1/3$, $d'_{3,4} = 5/8 < d''_4 =$

$5/4$, $d'_5 = 11/8 > d''_5 = 9/8$ *(y* $d'_k \approx 2d'_{k-1}$ *para valores superiores de* k*). Por lo tanto, siguiendo lo indicado en la Observación 8, elegimos* $n = 5$*, y obtenemos*

$$||\psi_h^3(y) - y|| \leq hL + D_5(hL)$$

para $|hL| \leq 1$*, donde*

$$D_5(\tau) = \frac{1}{2}\tau^2 + 1/3\tau^3 + \frac{5}{8}\tau^4 + \left(\xi(\tau) + \frac{1}{8}\xi(\tau/2)\right)\tau^5.$$

Por otro lado, para utilizar la cota (4.41) necesitamos $\hat{D}_n(\tau)$ *y para obtenerlo hemos de obtener los valores* $\hat{d}'_n$ *del Lema 8 y* $\hat{d}''_n$ *del Lema 10 correspondientes al método de Runge-Kutta* ψ_h^3*: según (4.34) tenemos que* $\hat{d}''_k = 2^{2-k}$ *para cada* k*, y de (4.29) obtenemos* $\hat{d}'_1 = 1 < \hat{d}''_1$ *pero* $\hat{d}'_2 = 1 = \hat{d}''_2$ *(y* $\hat{d}'_k = \hat{d}''_k$ *para* $k \geq 2$*). Si tomamos*

$$||\psi_h^3(y) - y|| \leq \hat{D}_2(hL),$$

donde

$$\hat{D}_2(\tau) = \tau + \xi(\tau/2)\tau^2$$

la solución K *de la ecuación* $\hat{D}_2(\frac{1}{K}) = 1$ *es* $K \approx 1{,}6291$*. No obstante, y puesto que* $\hat{d}'_3 = \hat{d}''_3$*, podemos obtener la solución* K *para* $\hat{D}_3(\frac{1}{K}) = 1$ *con* $\hat{D}_3(\tau) = \tau + \tau^2 + \frac{1}{2}\tau^3\xi(\tau/2)$ *y llegamos a que* $K \approx 1{,}74225$*. Evidentemente la primera solución es mejor que esta última, y por lo tanto* $\kappa \leq K \approx 1{,}6291 \leq \frac{44}{27}$*.*

De igual forma la solución de $\frac{1}{K} + D_5(\frac{1}{K}) = 1$ *es* $K \approx 1{,}69202$ *lo que no mejora nuestra anterior cota para* κ_3 *(*$1{,}6291 \leq \frac{44}{27}$*).*

Ahora, una vez acotado el valor de κ_3 *con* $\frac{44}{27}$*, siguiendo lo establecido en la Observación 7, hemos de tomar como valor de* $\kappa_4 = \kappa_3$*, ya que* $\max(\kappa_1, \ldots, \kappa_s) > \sum_{i=1}^{s} |b_i|$*. Y con estos valores, la correspondiente función* $D_n(\tau)$ *(con valor óptimo* $n = 6$*) dado por el Teorema 1 para el esquema de Runge-Kutta original* $\psi_h(y)$ *dado en (4.40), esta acotada por*

$$D_6(\tau) \leq \frac{5\,\tau^4}{24} + \frac{109\,\tau^5}{120} + \tau^6\left(\frac{\xi(\frac{\tau}{2})}{48} + \xi(\tau) + \frac{82458112\,\xi(\frac{44\tau}{27})}{43046721}\right),$$

ya que $d'_4 = 5/24 < d''_4 = 1{,}80394$*,* $d'_5 = 109/120 < d''_5 = 2{,}21561$ *y* $d'_6 = 7655/2592 > d''_6 = 2{,}93332$*.*

4.3.3. La dependencia de la norma elegida

Obviamente, el valor de L en la Suposición 1, y consecuentemente, todas las cotas obtenidas hasta el momento en esta sección dependen de la norma que hayamos elegido.

Ejemplo 5 *Consideremos de nuevo el problema de los dos cuerpos. Pero ahora consideremos una familia bi-paramétrica de normas: dados* $\rho_1, \rho_2 > 0$, *sea la norma* $||(Q,P)||_{\rho_1,\rho_2} = \text{máx}(||Q||/\rho_1, ||P||/\rho_2)$, *con* $Q, P \in \mathbb{C}^2$. *Entonces tenemos que* $||f(Q,P)||_{\rho_1,\rho_2} \leq \text{máx}(||P||/\rho_1, ||K(Q)||/\rho_2)$, *donde* $K(Q)$ *está dado en (4.21). De la desigualdad (4.22), llegamos a la conclusión de que la Suposición 1 se cumple para esta norma para los valores de* $(q,p) \in \mathbb{R}^4$ *tales que* $||q|| > (1+\sqrt{2})\rho_1$. *Por tanto, para* $(q,p) \in \mathbb{R}^4$ *arbitrario con* $\|q\| \neq 0$, *la Suposición 1 se cumple para la norma dada si* $\rho_1 < \frac{\|q\|}{1+\sqrt{2}}$.

Así mismo, llegamos a que

$$L \leq \text{máx}\left(\frac{||p|| + \rho_2}{\rho_1}, \frac{||q|| + \rho_1}{\rho_2(||q||^2 - 2\rho_1||q|| - \rho_1^2)^{3/2}}\right).$$

Si $y = (q,p)$ *es tal que la función Hamiltoniana* $H(q,p) = 1/2||p||^2 - 1/||q||$ *es negativa (trayectorias elípticas del movimiento de los dos cuerpos), entonces* $||p|| < \sqrt{2}||q||^{-1/2}$, *y por lo tanto,*

$$L \leq \tilde{L}(y) := \text{máx}\left(\frac{\sqrt{2}||q||^{-1/2} + \rho_2}{\rho_1}, \frac{||q|| + \rho_1}{\rho_2(||q||^2 - 2\rho_1||q|| - \rho_1^2)^{3/2}}\right).$$

Esto junto con el Lema 7 nos da diferentes cotas (dependiendo de ρ_1 *y* ρ_2*) para las diferenciales elementales* $F(u)(q,p)$. *Con el objeto de optimizar tales cotas, determinamos para cada* $y = (q,p)$, *los valores de los parámetros* $\rho_1 \in (0, \frac{1}{1+\sqrt{2}}||q||)$ *y* $\rho_2 > 0$ *que minimicen* $\tilde{L}(y)$. *Obsérvese que* $\tilde{L}(y)$ *es el mayor de dos valores; el primero creciente con respecto al valor de* ρ_2 *y el segundo decreciente con respecto a* ρ_2, *por lo que el valor mínimo de* $\tilde{L}(y)$ *se da con el valor de* ρ_2 *que hace que*

$$\frac{\sqrt{2}||q||^{-1/2} + \rho_2}{\rho_1} = \frac{||q|| + \rho_1}{\rho_2(||q||^2 - 2\rho_1||q|| - \rho_1^2)^{3/2}}.$$

cuyo valor mínimo se da para $\rho_1 \approx 0{,}27396\,||q||$ *y* $\rho_2 \approx 0{,}709832\,||q||^{-1/2}$, *y con dichos valores obtenemos* $\tilde{L}(y) \approx 7{,}75297\,||q||^{-3/2}$.

Supongamos que queremos acotar las diferenciales elementales $F(u)(y)$ en una norma dada $||\cdot||$ en $\mathbb{C}^d$. Esta norma, en lo que sigue, será tratada como una norma fija.

Como hemos visto, el Lema 7 con esta norma puede no dar cotas ajustadas para las diferenciales elementales. Incluso puede pasar que la Suposición 1 no se cumpla para esta norma.

Definición 2 *Dados $f : \mathcal{U} \subset \mathbb{R}^d \to \mathbb{R}^d$ y una norma $||\cdot||$ en $\mathbb{C}^d$, supongamos que para $y \in \mathcal{U}$ y $S \in GL(d)$, donde $GL(d)$ es el conjunto de matrices cuadradas invertibles de dimensión d, la Suposición 1 se cumple para la norma $||\cdot||_S$ en $\mathbb{C}^d$ definida como*

$$||z||_S = ||Sz||. \tag{4.43}$$

En ese caso denotamos

$$L_S(y) = \sup_{||z-y||_S \leq 1} ||f(z)||_S. \tag{4.44}$$

Si para un $S \in GL(d)$ no se cumple la Suposición 1 para la norma $||\cdot||_S$, entonces denotamos $L_S(y) = \infty$.

Finalmente, denotamos

$$L(y) = \inf_{S \in GL(d)} L_S(y), \tag{4.45}$$

y $C_S = ||S||^{-1}$. Por el Teorema 1 tenemos que el error local (1.15), (4.8) de un esquema de Runge-Kutta de orden p (1.9) puede ser acotado para la norma $||\cdot||_S$ y para cada $n \geq p+1$ como

$$||\delta(y,h)||_S \leq D_n(hL_S(y)),$$

donde $D_n(\tau)$ viene dado por (4.35). Si volvemos a la norma original $||\cdot||$ tenemos, por tanto, para cada $n \geq p+1$,

$$||\delta(y,h)|| = ||S^{-1}\delta(y,h)||_S \leq ||S^{-1}||\ ||\delta(y,h)||_S \leq C_S D_n(hL_S(y)). \tag{4.46}$$

Ejemplo 6 *Volvamos al problema de los dos cuerpos. Sean q, p tales que se encuentren en la solución del problema de valor inicial con*

$$\begin{aligned} q(0) &= (1-e, 0), \\ p(0) &= \left(0, \sqrt{\frac{1+e}{1-e}}\right). \end{aligned}$$

En ese caso, $1-e \leq ||q|| \leq 1+e$. *Sea* $||\cdot||$ *la norma de* $\mathbb{C}^4$ *considerada en el Ejemplo 1. En el Ejemplo 5 hemos visto que* $L_S(q,p) \leq 7{,}75297||q||^{-\frac{3}{2}}$ *para*

$$S = \begin{pmatrix} \frac{1}{\rho_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\rho_1} & 0 & 0 \\ 0 & 0 & \frac{1}{\rho_2} & 0 \\ 0 & 0 & 0 & \frac{1}{\rho_2} \end{pmatrix},$$

con $\rho_1 = 0{,}27396||q||$, $\rho_2 = 0{,}709832||q||^{\frac{-1}{2}}$ *(por tanto* $L_S(q,p) \leq 7{,}75297||q||^{-\frac{3}{2}}$*)* *y* $C_S = ||S^{-1}|| = \max(\rho_1, \rho_2)$, *es decir,*

$$C_S = \max\left(0{,}27396\,||q||, 0{,}709832\,||q||^{-1/2}\right) \leq \frac{0{,}709832}{\sqrt{1-e}}.$$

Ejemplo 7 *Consideremos el modelo de Lotka-Volterra*

$$\begin{aligned} \dot{u} &= u(v-2), \\ \dot{v} &= v(1-u). \end{aligned}$$

Sea $||\cdot||$ *la norma máxima. Queremos obtener cotas de de las diferenciales elementales* $||F(t)(2,3)||$.

Por ejemplo, para la matriz

$$S = \begin{pmatrix} 1 & 0{,}0711014 \\ 0{,}588736 & 1 \end{pmatrix},$$

obtenemos $L_S(u,v) = 4{,}3803$ *y* $C_S = ||S^{-1}|| = 1{,}65815$.

Ejemplo 8 *En este caso vamos a considerar el problema restringido de los tres cuerpos dado en (1.4) y la norma* $||\cdot||$ *de* $\mathbb{C}^4$ *considerada en el Ejemplo 1. Buscaremos una cota* $\tilde{L}_S(q,v)$ *de* $L_S(q,v)$ *para*

$$S = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_1 & 0 & 0 \\ 0 & -\lambda_3 & \lambda_2 & 0 \\ \lambda_3 & 0 & 0 & \lambda_2 \end{pmatrix}$$

con $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$ *arbitrarios.*

El problema (1.4) se puede escribir de la siguiente forma:

$$\begin{aligned} q' &= v, \\ v' &= q + 2v^T - \mu' \frac{q^+}{(r^+)^3} - \mu \frac{q^-}{(r^-)^3}, \end{aligned}$$

donde μ *y* μ' *son parámetros del problema,* $q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$, $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, $q^+ = \begin{pmatrix} q_1 + \mu \\ q_2 \end{pmatrix}$, $q^- = \begin{pmatrix} q_1 - \mu' \\ q_2 \end{pmatrix}$, $v^T = \begin{pmatrix} v_2 \\ -v_1 \end{pmatrix}$, $r^+ = \sqrt{(q_1^+)^2 + (q_2^+)^2}$ *y* $r^- = \sqrt{(q_1^-)^2 + (q_2^-)^2}$.

Para obtener una cota de $L_S(q, v)$, *hay que buscar una cota de* $\|f(Q, V)\|_S$, *donde* $Q = q + \Delta Q$ *y* $V = v + \Delta V$, *para los valores* (Q, V) *que cumplen* $\|(\Delta Q, \Delta V)\|_S \leq 1$. *Se tiene que*

$$\begin{aligned} \|(\Delta Q, \Delta V)\|_S &= \|(\lambda_1 \Delta Q, \lambda_2 \Delta V + \lambda_3 \Delta Q^T)\| \\ &\leq \max\left(|\lambda_1| \|\Delta Q\|, |\lambda_2| \|\Delta V\| + |\lambda_3| \|\Delta Q\|\right), \end{aligned}$$

de donde llegamos a que si $|\lambda_1| > |\lambda_3|$ *y*

$$\|\Delta Q\| \leq \frac{1}{|\lambda_1|}, \tag{4.47}$$

$$\|\Delta V\| \leq \frac{1}{|\lambda_2|}\left(1 - \frac{|\lambda_3|}{|\lambda_1|}\right) \tag{4.48}$$

entonces $\|(\Delta Q, \Delta V)\|_S \leq 1$.

Por otra parte, se tiene que

$$\|f(Q, V)\|_S = \|Sf(Q, V)\|,$$

donde

$$S \cdot f(Q, V) = \left(\lambda_1 V, \lambda_2(Q + 2V^T) + \lambda_3 V^T + \lambda_2\left(-\mu' \frac{Q^+}{(R^+)^3} - \mu \frac{Q^-}{(R^-)^3}\right)\right)$$

y $R^+ = \sqrt{(Q_1^+)^2 + (Q_2^+)^2}$, $R^- = \sqrt{(Q_1^-)^2 + (Q_2^-)^2}$, $Q^+ = q^+ + \Delta Q$ *y* $Q^- = q^- + \Delta Q$. *Siguiendo la notación utilizada en el Ejemplo 1, se llega a*

$$\begin{aligned} \|f(Q, V)\|_S \leq \max(& |\lambda_1| \|V\|, \\ & \|\lambda_2(Q + 2V^T) + \lambda_3 V^T\| + |\lambda_2| \left(\mu' K(Q^+) + \mu K(Q^-)\right)) \end{aligned}$$

donde $K(Q)$ *viene dado en (4.21). Buscamos, por tanto, el máximo de dos valores sujetos a las restricciones (4.47)-(4.48). El primero,* $|\lambda_1| \|V\|$, *se puede acotar mediante*

$$|\lambda_1| \|V\| \leq |\lambda_1| \left(\|v\| + \|\Delta V\|\right) \leq |\lambda_1| \left(\|v\| + \frac{1}{|\lambda_2|}\left(1 - \frac{|\lambda_3|}{|\lambda_1|}\right)\right),$$

mientras que para el segundo término, por una parte, se tiene que

$$\|\lambda_2(Q+2V^T)+\lambda_3 V^T\| \leq C,$$

con

$$C = \|\lambda_2 q + (\lambda_3+2\lambda_2)v^T\| + \frac{|\lambda_2|}{|\lambda_1|} + \frac{|\lambda_3+2\lambda_2|}{|\lambda_2|}\left(1-\frac{|\lambda_3|}{|\lambda_1|}\right), \tag{4.49}$$

y, por otra parte, utilizando la cota (4.22) de $K(Q)$ dada en el Ejemplo 1, se tiene la cota

$$|\lambda_2|\left(\mu' K(Q^+)+\mu K(Q^-)\right) \leq |\lambda_2|\left(\mu' C^+ + \mu C^-\right),$$

donde

$$C^+ = \frac{\|q^+\| + \frac{1}{|\lambda_1|}}{\left(\|q^+\|^2 - \frac{2\|q^+\|}{|\lambda_1|} - \frac{1}{|\lambda_1|^2}\right)^{\frac{3}{2}}}, \tag{4.50}$$

y

$$C^- = \frac{\|q^-\| + \frac{1}{|\lambda_1|}}{\left(\|q^-\|^2 - \frac{2\|q^-\|}{|\lambda_1|} - \frac{1}{|\lambda_1|^2}\right)^{\frac{3}{2}}}. \tag{4.51}$$

Por todo ello, se llega a

$$\begin{aligned} \|f(Q,V)\|_S &\leq \max\left(|\lambda_1|\left(\|v\| + \frac{|\lambda_1|-|\lambda_3|}{|\lambda_2||\lambda_1|}\right), C + |\lambda_2|(\mu' C^+ + \mu C^-)\right) \\ &=: \tilde{L}_S(q,v) \end{aligned} \tag{4.52}$$

donde C, C^+ y C^- son los dados en (4.49), (4.50) y (4.51) respectivamente.

Finalmente, definimos

$$\tilde{L}(q,v) = \min_{\lambda_1,\lambda_2,\lambda_3} \tilde{L}_S(q,v)$$

restringido a la condición $|\lambda_1| > |\lambda_3|$. La obtención de $\tilde{L}(q,v)$ es un problema de minimización con respecto a los parámetros λ_1, λ_2 y λ_3, y en los experimentos numéricos que mostramos más adelante, para el problema Arenstorf *(1.4) se ha utilizado la aplicación* Mathematica *para buscar $\tilde{L}(q,v)$ en cada (q,v) de la trayectoria de la solución del problema, y hemos observado que los valores dados por*

$$G(q,v) = 4\sqrt{\frac{\mu'}{(r^+)^3} + \frac{\mu}{(r^-)^3}} \tag{4.53}$$

se aproximan mucho a las cotas $\tilde{L}_S(q,v)$ de $L_S(q,v)$ obtenidas por Mathematica.

A partir de ahora, suponemos para simplificar la presentación, que para cada $y \in \mathcal{U}$, $\exists S \in GL(d)$ tal que $L_S(y) = L(y)$. En tal caso, denotamos $C(y) = C_S$ y de (4.46) tenemos que

$$||\delta(y,h)|| \leq C(y) D_n(hL(y)).$$

Observación 10 *Si se diera el caso de que no exista $S \in GL(d)$ tal que $L_S(y) = L(y)$, podríamos definir para cada $C > 0$*

$$L_C(y) = \inf_{||S^{-1}|| \leq C} L_S(y)$$

de forma que

$$\lim_{C \to \infty} L_C(y) = L(y).$$

En tal caso, se tiene para cada $C > 0$ que para cada $n \geq p+1$,

$$||\delta(y,h)|| \leq C\, D_n(hL_C(y)).$$

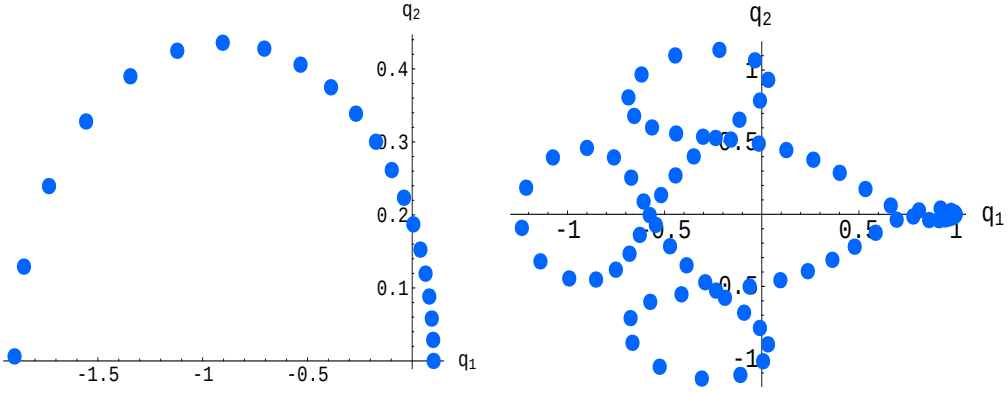
4.4. Experimentos numéricos

En esta sección presentamos una serie de experimentos numéricos con diferentes métodos aplicados a diferentes problemas.

En los experimentos numéricos se han tenido en cuenta los nueve métodos considerados por el software Mathematica, una aplicación extensamente usada que ofrece distintos métodos de Runge-Kutta explícitos para la resolución numérica de ecuaciones diferenciales ordinarias. Además de esos métodos también hemos tenido en cuenta los métodos de orden 5 [6] y de orden 8 propuestos por Dormand y Prince, cuya construcción se comenta en [20] (donde los dan a conocer como Dopri5 y Dopri8). Otro método interesante para la comparación es *el método clásico de Runge-Kutta* de orden 4, que denotaremos como *RK4* y que mostramos en el siguiente tablero de Butcher:

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array} \tag{4.54}$$

Figura 4.2: Los diferentes puntos de las soluciones orbitales para los que se han calculado soluciones numéricas junto con su correspondiente error local. A la izquierda los puntos correspondientes al problema de *Kepler*, y a la derecha los del problema *Arenstorf*.



Este método tiene la particularidad de que todos los valores del tablero de Butcher son positivos.

Por otro lado, hemos construido un nuevo método de Runge-Kutta teniendo en cuenta la Proposición 2, y en especial, la Observación 7, tratando de optimizar la cota del error local (1.15), (4.8) dada en el Teorema 1. Este último método lo denotamos como *m4*.

Para cada uno de los métodos que se han tomado en consideración hemos obtenido la función $D_n(\tau)$ derivada del Teorema 1 y dada en (4.35), con el índice de truncamiento apropiado según la Observación 8, y la hemos comparado con los errores locales obtenidos en los experimentos numéricos. Los problemas que hemos elegido son el problema de *Kepler* de los dos cuerpos, el mismo que hemos mostrado y utilizado en el ejemplo 1, y el problema de valor inicial de dimensión 4 extraido de [20, pp.129–130] que mostramos en (1.4). Este problema corresponde a una solución periódica del problema restringido de los tres cuerpos y en la sección 2.10 lo hemos denominado *Arenstorf*. Ambos problemas tienen una solución periódica, y para cada una de dichas soluciones hemos tomado diferentes puntos de las soluciones para el cálculo de los errores locales. Estos puntos, proyectados en el plano (q_1, q_2), se muestran en la Figura 4.2 (la gráfica de la izquierda corresponde al problema de *Kepler* y la de la derecha al problema restringido de los tres cuerpos).

Hemos utilizado estos puntos para calcular el error local con diferentes longitudes de paso para compararlos con las cotas teóricas del error.

En la segunda parte de los experimentos numéricos comparamos los

métodos mencionados mediante la comparación de las funciones $D_n(\tau)$ dadas en (4.35) correspondientes a cada uno de ellos.

4.4.1. El control del error local

Hemos acotado el error local (1.15) de los métodos de Runge-Kutta mediante la función $D_n(hL(y))$, es decir, la cota depende tanto de la longitud del paso h como de $L(y)$. Observamos que en la resolución numérica de cualquier problema, en caso de que utilicemos la misma longitud de paso, los errores locales que obtenemos en cada punto difieren mucho entre si.

Hemos realizado experimentos numéricos para ver si los errores locales que se obtienen para valores prefijados de $hL(y)$ se asemejan y los resultados han sido altamente satisfactorios. Los experimentos han confirmado lo que la teoría daba a entrever: si la longitud de paso que se utiliza en cada punto y del espacio de fases es aquel que hace que se mantenga constante el valor $hL(y)$, los errores locales que se obtienen son de magnitud parecida, a diferencia de lo que ocurre si lo que se mantiene constante es la longitud de paso.

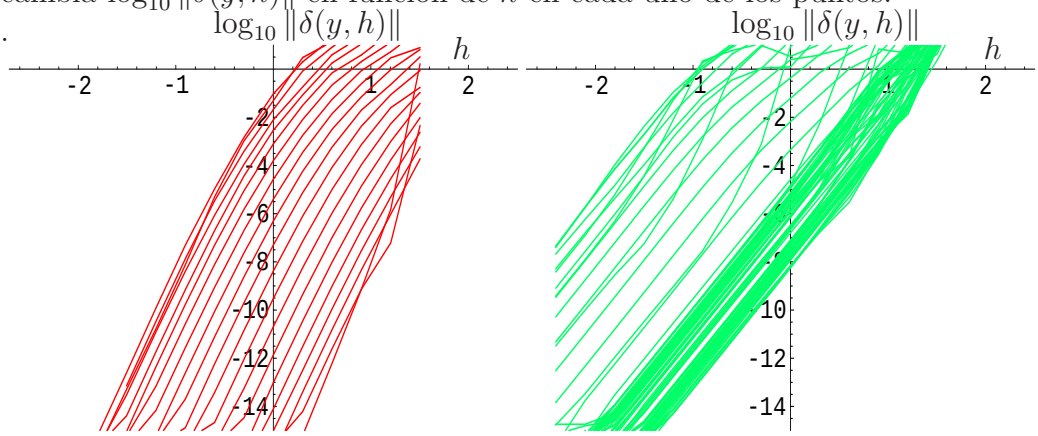
Para mostrar dicho comportamiento, podemos observar la Figura 4.3 donde aparecen dos gráficas, una de ellas corresponde al problema de *Kepler*, y la otra al problema *Arenstorf*. En ellas aparecen varias curvas, una por cada punto mostrado en la Figura 4.2, y cada curva muestra los errores locales $\delta(y, h)$ obtenidos frente a la longitud de paso h. Hay que destacar que las curvas de los errores locales son muy distantes entre si, aunque sean paralelas, por lo que se puede deducir que el error local en cada punto no solo depende de la longitud de paso, sino también de la *dificultad o variabilidad* del problema, por lo que podemos decir que hay una variación de la escala de tiempo.

En la gráfica de la izquierda podemos ver los errores locales obtenidos con el método *m4*, que construimos en la Sección 4.4.3, en la resolución del problema de *Kepler*. La *dificultad* del problema va variando por lo que los errores locales también varían.

La figura de la derecha muestra más curvas agrupadas, pero sigue habiendo puntos con magnitudes de errores locales muy distintos entre sí. La dificultad del problema es muy grande al comienzo y al final de la órbita, mientras que en el resto del problema no hay tanta variación, de ahí que en la gráfica correspondiente a este problema veamos más curvas agrupadas. Las curvas de la norma del error local $\|\delta(y, h)\|$ corresponden al método Dopri8.

Puesto que la cota teórica $D_n(hL(y))$ del error local $\|\delta(y, h)\|$ depende de

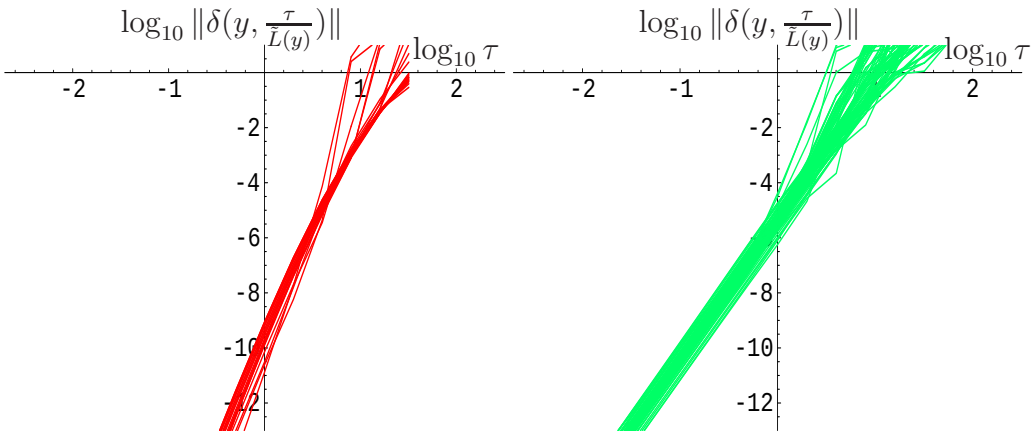
Figura 4.3: Los errores locales obtenidos en diferentes puntos de la solución del problema de *Kepler* con el método Dopri8 (izquierda) y el problema *Arenstorf* con el método $m4$ (derecha). Para cada punto y de la órbita se ha dado un paso con distintas longitudes de paso, y cada curva muestra cómo cambia $\log_{10} \|\delta(y,h)\|$ en función de h en cada uno de los puntos.



$hL(y)$, nos preguntamos si las longitudes de paso que vayamos a utilizar son aquellas que hacen que $hL(y)$ tenga valores prefijados, los errores locales que obtengamos en los distintos puntos y sean de magnitud muy parecida. En la Figura 4.4 consideramos los mismos problemas resueltos con los mismos métodos que los mostrados en la Figura 4.3, pero en vez de mostrar las gráficas de los errores locales $\|\delta(y,h)\|$ en función de h, consideramos, para distintos puntos y, el error local $\|\delta(y, \frac{\tau}{L(y)})\|$ en función de $\tau = hL(y)$. En la Figura 4.4 observamos que las distintas curvas obtenidas (en doble escala logarítmica) para $\|\delta(y, \frac{\tau}{L(y)})\|$ para distintos puntos y de las órbitas de cada uno de los problemas se encuentran concentrados en una estrecha banda, lo que muestra que de alguna forma, $L(y)$ nos da la *variación de la escala de tiempo* del problema, por lo que si adecuamos la longitud de paso h a lo largo de la integración de tal forma que $hL(y)$ toma un valor prefijado τ, obtenemos errores locales de parecida magnitud.

En la práctica, no disponemos de una expresión explícita de $L(y)$ en ninguno de los dos problemas considerados, de modo que en su lugar consideramos una cota superior $\tilde{L}(y)$ de $L(y)$. La cota $\tilde{L}(y)$ se ha obtenido siguiendo el procedimiento mostrado en el ejemplo 5 para el problema de *Kepler* de los dos cuerpos. Y para el problema *Arenstorf* llegamos a la cota mostrada en el Ejemplo 8.

Figura 4.4: Los errores locales obtenidos en diferentes puntos de la solución del problema de *Kepler* (izquierda) resuelto con el método Dopri8 y el problema *Arenstorf* (derecha) resuelto con el problema *m4*. Para cada punto y de la órbita se ha dado un paso con distintos valores $\tau = h\tilde{L}(y)$ y se muestra el logaritmo decimal de la norma del error local $\delta(y, \frac{\tau}{\tilde{L}(y)})$ para cada τ.



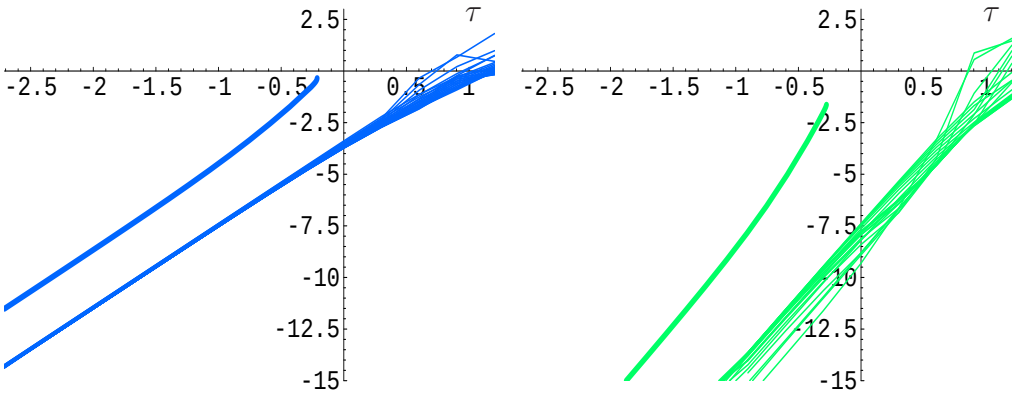
4.4.2. *El comportamiento de la función* $D_n(\tau)$

Queremos mostrar que la cota teórica del error local, dada por la función $D_n(\tau)$, muestra un comportamiento parecido a los errores locales obtenidos en las resoluciones numéricas de los problemas, es decir, que la relación entre la norma del error local $\delta(y, h)$ y la cota del error local dada por $D_n(hL(y))$ varía relativamente poco con respecto de h y de y.

Para cada uno de los puntos preestablecidos en la solución de los problemas (que se muestran en la Figura 4.2), se ha obtenido, siguiendo el procedimiento mostrado en el ejemplo 5, una cota $\tilde{L}(y)$ de $L(y)$, y hemos obtenido diferentes soluciones numéricas junto con su correspondiente error local (1.15) para diferentes valores de $h\tilde{L}(y)$: $h\tilde{L}(y) = \frac{1}{2^8}, \frac{1}{2^7}, \ldots, 8, 16$.

En la Figura 4.5 mostramos los resultados obtenidos con el problema de los dos cuerpos resuelto mediante dos métodos distintos. Cada gráfica contiene varias curvas, una gruesa y varias finas. Para todas ellas el eje horizontal corresponde a los valores $\tau = h\tilde{L}(y)$, pero el significado del eje vertical varía: en el caso de la curva gruesa el eje vertical corresponde a los valores de la función $D_n(\tau)$, mientras que para las curvas finas el eje vertical representa la norma del error local $\|\delta(y, \frac{\tau}{\tilde{L}(y)})\|$, y cada curva corresponde a los errores cometidos en uno de los puntos mostrados en la Figura 4.2. Para cada uno de estos puntos y se han obtenido, utilizando un método

Figura 4.5: La función $D_n(\tau)$ y la norma de los errores locales $\|\delta(y,h)\|$ correspondientes a cada valor $\tau = h\tilde{L}(y)$ obtenidos para los 21 puntos en la órbita de la solución del problema de *Kepler*. A la izquierda los resultados obtenidos con el método de orden 3 de la aplicación *Mathematica*, y a la derecha los correspondientes al método de orden 6.



dado, distintas soluciones numéricas $\psi_h(y)$ y su correspondiente error local $\delta(y,h)$, una por cada valor de $\tau = h\tilde{L}(y)$ preestablecido. Es precisamente ese conjunto de errores locales lo que nos muestra cada curva fina, es decir, la norma del error local $\delta(y, \frac{\tau}{\tilde{L}(y)})$ dado en (1.15) en función de $\tau = h\tilde{L}(y)$ para cada punto y. Y todo ello en escala logarítmica, tanto un eje como el otro. Obviamente, el error obtenido crece al incrementar el valor $\tau = h\tilde{L}(y)$. El gráfico de la izquierda ha sido obtenido con el método de orden 3 utilizado en el *Mathematica*, mientras que el de la derecha corresponde al método de orden 6 utilizado en la misma aplicación.

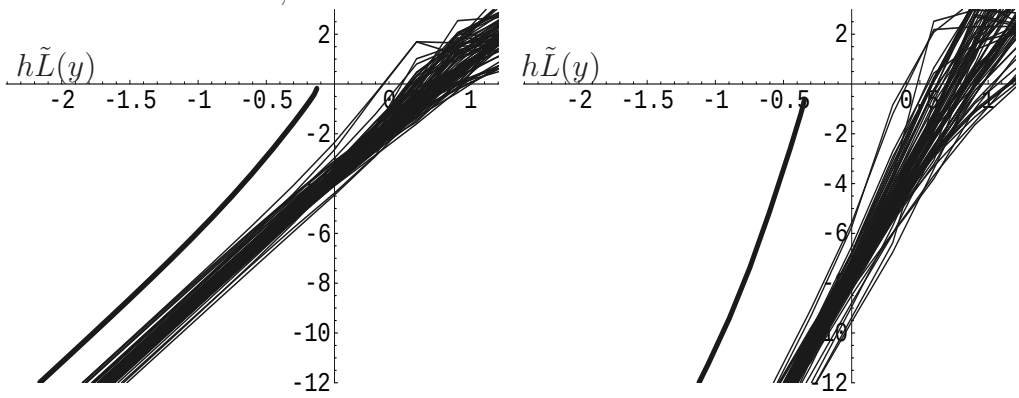
Se puede observar que las 21 curvas finas están unas muy cerca de otras, o que incluso se sobreponen, es decir, la relación entre $D_n(h\tilde{L}(y))$ y $\delta(y,h)$ es más o menos constante, independientemente del punto y donde hayamos obtenido los resultados. Por ello podemos decir que dicha relación depende poco del punto y y de la longitud de paso, es decir:

$$\frac{D_n(h\tilde{L}(y))}{\delta(y,h)} \approx \text{Cte.} \tag{4.55}$$

Por otra parte, se puede ver que la curva gruesa, la que corresponde a la cota teórica del error local, está, como era de esperar, por encima de los valores obtenidos numéricamente para los errores locales.

En la Figura 4.6 mostramos los resultados obtenidos con otro problema, en este caso el problema *Arenstorf*, y otros dos métodos: el método de

Figura 4.6: La función $D_n(\tau)$ y la norma de los errores locales $\|\delta(y,h)\|$, obtenidos con el problema *Arenstorf*, en función de $\tau = h\tilde{L}(y)$. A la izquierda el método de orden 4, a la derecha el método de orden 9.

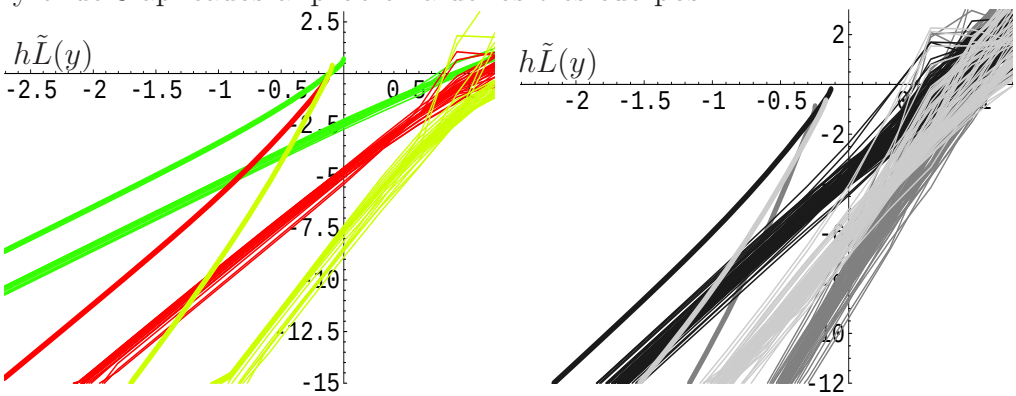


Runge-Kutta de orden 4 y el de orden 9 que utiliza la aplicación *Mathematica*. Para este otro problema se han obtenido curvas de la norma del error local $\delta(y,h)$ en bastantes más puntos que en el problema de *Kepler*, es decir, en cada gráfica hay tres veces más curvas finas que en la Figura 4.5. No obstante, aunque los métodos y el problema sean distintos, volvemos a apreciar que la relación mostrada en (4.55) se vuelve a mantener más o menos constante, tanto para el método de orden 4 como para el de orden 9.

Otra cuestión que nos surge y que debe ser aclarada es si la relación dada en (4.55) varía al cambiar de método. Para ello, en la Figura 4.7 mostramos los resultados superpuestos que hemos obtenido con tres métodos distintos. En la gráfica de la izquierda vemos los resultados con los métodos de orden 2, orden 4 y orden 7 aplicados todos ellos al problema de los dos cuerpos. En ella se puede ver que si para algún valor $h\tilde{L}(y)$ un método obtiene un error local $\delta(y,h)$ menor que otro método, sus respectivas funciones $D_n(h\tilde{L}(y))$ muestran el mismo proceder, es decir, si para valores altos de $h\tilde{L}(y)$ un método A resuelve el problema con un error local $\delta(y,h)$ menor que el método B entonces para valores altos de $h\tilde{L}(y)$ el método A tiene un valor de $D_n(h\tilde{L}(y))$ menor que el método B.

Mostramos también, en la gráfica de la derecha de la Figura 4.7, los resultados obtenidos con el problema *Arenstorf* mediante los métodos de Runge-Kutta de orden 4, de orden 5 y de orden 8 utilizados por la aplicación *Mathematica*. En esta nueva gráfica también se puede decir que si un método obtiene mejores resultados numéricos que otros métodos para algunos valores de $h\tilde{L}(y)$ la cota teórica del error local $D_n(h\tilde{L}(y))$ correspondiente a dicho método también muestra valores inferiores a los valores

Figura 4.7: La función $D_n(hL(y))$ y la norma de los errores locales $\|\delta(y,h)\|$ para distintos métodos: a la izquierda, el de orden 2, el de 4 y el de 7 aplicados al problema de *Kepler*. A la derecha, el método de orden 4, el de 5 y el de 8 aplicados al problema de los tres cuerpos

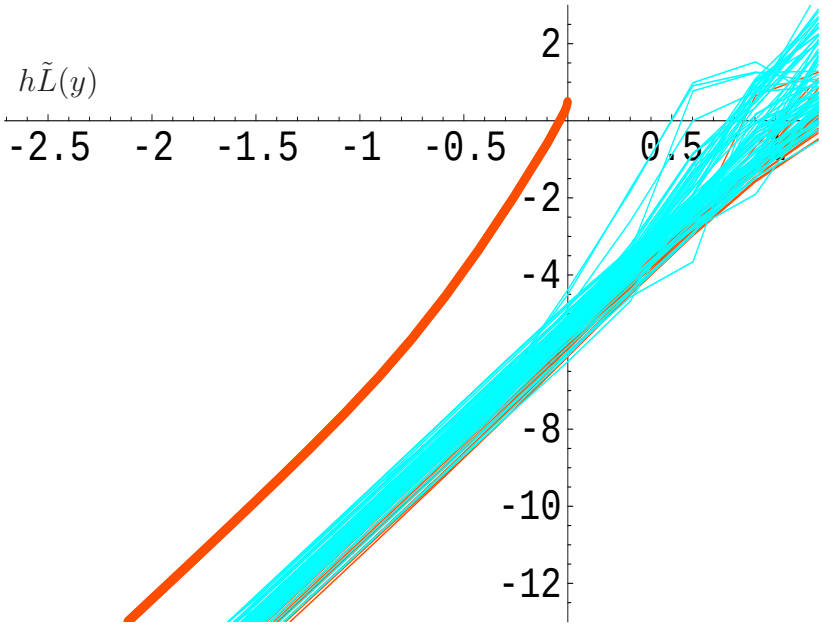


de las cotas correspondientes para dichos métodos.

Tras los resultados comentados hasta ahora se puede decir que la relación (4.55) varía poco independientemente del punto donde obtengamos los errores locales, es decir, para un método dado, incluso en la integración de distintos sistemas de ecuaciones diferenciales, los errores locales que cabe esperar deberían mantener esa relación. Los resultados numéricos obtenidos para un método de Runge-Kutta con distintos problemas parecen indicar que dicha relación se mantiene: en la Figura 4.8 mostramos los errores locales obtenidos con el método de Runge-Kutta *m4*, cuya construcción comentamos en la Sección 4.4.3. Hemos resuelto numéricamente el problema de los dos cuerpos y el problema *Arenstorf* utilizando este método, y como siempre, hemos establecido los distintos puntos en las órbitas de la solución de los dos problemas, y para cada punto hemos obtenido una solución numérica junto con el error local para distintas longitudes de paso (para valores preestablecidos de $\tau = h\tilde{L}(y)$). Para cada punto mostramos la curva que se obtiene uniendo los distintos valores del error local y podemos observar que todas las curvas correspondientes a los errores locales obtenidos en las distintas integraciones aparecen sobrepuestas unas encima de otras, y no pueden diferenciarse entre si. Si dibujamos primero las curvas obtenidas con el problema de los dos cuerpos y luego los obtenidos con el problema *Arenstorf* las del segundo problema ocultan las curvas correspondientes al primer problema.

En las figuras que hemos comentado podemos observar los resultados obtenidos con todos los métodos utilizados en la aplicación *Mathematica*,

Figura 4.8: La función $D_n(\tau)$ y los errores locales $\delta(y,h)$ cometidos con el método *m4*, que corresponden a cada valor $h = \frac{\tau}{\tilde{L}(y)}$ obtenidos tanto para los 21 puntos en la órbita de la solución del problema de los dos cuerpos (curvas oscuras ocultas bajo las curvas claras) como para los 80 puntos de la órbita de la solución del problema *Arenstorf* (curvas claras).



todos ellos, desde el método de orden 2 hasta el método de orden 9, aparecen en alguna de las gráficas. Y de todas estas gráficas podemos concluir que la relación (4.55) depende poco del punto donde obtengamos el error local, y a su vez depende poco del método que utilicemos en dicho cálculo. Por lo que podemos decir que la función $D_n(\tau)$ correspondiente a un método puede ser utilizada para la comparación del error local $\delta(y,h)$ que se puede esperar con la utilización de dicho método.

4.4.3. Construcción de métodos con $D_n(\tau)$ optimizado

Lo que queremos hacer es comparar los diferentes métodos comparando las funciones $D_n(\tau)$ correspondientes a dichos métodos. La función $D_n(\tau)$ del método depende de los valores κ resultantes del Lema 10, por lo que son de gran importancia a la hora de la comparación. Si tenemos en cuenta la Observación 7, podemos obtener un método para el que $\kappa = 1$. Una condición suficiente para ello es que los valores del tablero de Butcher del método sean positivos, y que $0 \leq c_i = \sum_j a_{ij} \leq 1$ para cada $i = 1, \ldots, s$.

Basándonos en una familia de métodos de orden 4 con 6 etapas hemos realizado una búsqueda de métodos con valores a_{ij} positivos y que a su vez optimizan las condiciones de los árboles de orden 5, 6 y 7 en el sentido de mínimos cuadrados, junto con el objetivo de que posea un área de estabilidad relativamente amplia, y hemos llegado a obtener el siguiente método al que denotamos *m4*:

$$
\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{3}{10} & \frac{3}{10} & & & & & \\
\frac{3}{10} & 0 & \frac{3}{10} & & & & \\
\frac{3}{5} & 0 & 0 & \frac{3}{5} & & & \\
\frac{11}{14} & \frac{1018823}{7137144} & \frac{39}{289} & \frac{108904}{892143} & \frac{2754557}{7137144} & & \\
\frac{9}{10} & \frac{506699231}{4850396100} & \frac{296}{829} & \frac{7782703}{88189020} & \frac{4}{197} & \frac{266709499}{808399350} & \\
\hline
 & b_1 & b_2 & b_3 & b_4 & b_5 & b_6
\end{array}
\tag{4.56}
$$

donde

$$
\begin{aligned}
b_1 &= \frac{24635841840343}{250724932259886} \\
b_2 &= \frac{25247197004665}{125362466129943}
\end{aligned}
$$

$$
\begin{aligned}
b_3 &= \frac{25247197004665}{125362466129943} \\
b_4 &= \frac{629053720835935}{3259424119378518} \\
b_5 &= \frac{905130449608}{8622815130631} \\
b_6 &= \frac{933206410860}{4643054301109}
\end{aligned}
$$

4.4.4. Comparación de diferentes métodos

Nuestra intención es la obtención del mejor método para diferentes rangos de valores de $hL(y)$, pero para poder comparar diferentes esquemas de Runge-Kutta deberíamos tener en cuenta el número de etapas internas que usa en cada paso, es decir, el costo computacional requerido en cada paso. Dicho costo computacional depende fundamentalmente del número de evaluaciones de la función $f(y)$ que requiere el método de Runge-Kutta (1.9-1.10), es decir, una evaluación por etapa. Así, por ejemplo, podemos comparar el método de orden 8 utilizado en *Mathematica* con el método Dopri8 ya que ambos tienen el mismo número de etapas internas en cada paso. Pero en el caso de que esto no sea así, la comparación debe tener en cuenta esa diferencia. Si queremos comparar el método de orden 4 de *Mathematica*, que tiene cuatro etapas, con el método de orden 8, que tiene 12 etapas, deberíamos dar un paso de longitud h con el método de orden 8 y tres pasos de longitud $\frac{h}{3}$ con el método de orden 4. De esta forma el coste computacional en ambos casos es parecido así como el tamaño de paso. Esto significa que debemos comparar el método de orden 8 con el método resultante de combinar tres subpasos del método de orden 4.

Los métodos que hemos considerado para la comparación tienen un número de etapas internas muy diferentes entre si, por lo que no siempre hemos trabajado con la composición de subpasos de forma que tengamos métodos con un coste computacional idéntico. Por ejemplo, si quisiéramos comparar el método de orden 7 que tiene 10 etapas, los dos métodos de orden 8 compuestos por 12 etapas y el de orden 9 que tiene 15 etapas, tendríamos que considerar los métodos obtenidos mediante 6 subpasos del método de orden 7 (60 etapas internas), 5 subpasos de los métodos de orden 8 (60 etapas internas) y 4 subpasos del método de orden 9 (60 etapas internas). No obstante, hemos comparado éstos cuatro métodos sin tener en cuenta la diferencia de costo computacional que tienen, porque aún así, ha sido posible obtener conclusiones claras en cuanto a su eficiencia teórica relativa.

Para la comparación de los diferentes métodos mencionados, y con la idea de realizar una comparación en igualdad de condiciones, ó, por lo menos, en condiciones similares, en cuanto a costo computacional, hemos obtenido las siguientes composiciones de los métodos:

método	etapas	subpasos de la composición	etapas resultantes
orden 3	3	4	12
orden 4	4	3	12
RK4	4	3	12
m4	6	2	12
Dopri5	6	2	12
orden 5	7	1	7
orden 6	8	1	8
orden 7	10	1	10
orden 8	12	1	12
Dopri8	12	1	12
orden 9	15	1	15

Para cada uno de los métodos compuestos hemos obtenido su correspondiente función $D_n(\tau)$ para compararlas entre sí. La composición de n subpasos de idéntica longitud de un método cualquiera equivale a un método de Runge-Kutta cuyo tablero de Butcher tiene la siguiente forma:

$$\begin{array}{c|cccc} \frac{C}{n} & \frac{A}{n} & & & \\ \frac{1}{n}+\frac{C}{n} & \frac{B}{n} & \frac{A}{n} & & \\ \dots & \dots & \dots & \dots & \\ \frac{n-1}{n}+\frac{C}{n} & \frac{B}{n} & \frac{B}{n} & \dots & \frac{A}{n} \\ \hline & \frac{b_i}{n} & \frac{b_i}{n} & \dots & \frac{b_i}{n} \end{array}$$

Donde $\frac{A}{n}$ es la matriz del tablero de Butcher compuesto por los valores $\frac{a_{ij}}{n}$ con $i, j = 1 \dots s$ del método en cuestión, donde $\frac{B}{n}$ es la matriz cuadrada de $s \times s$ en la que cada fila tiene los valores $\frac{b_i}{n}$ $(i = 1 \dots s)$ correspondientes al método, y donde C es el vector compuesto por los valores c_i del tablero de Butcher del método.

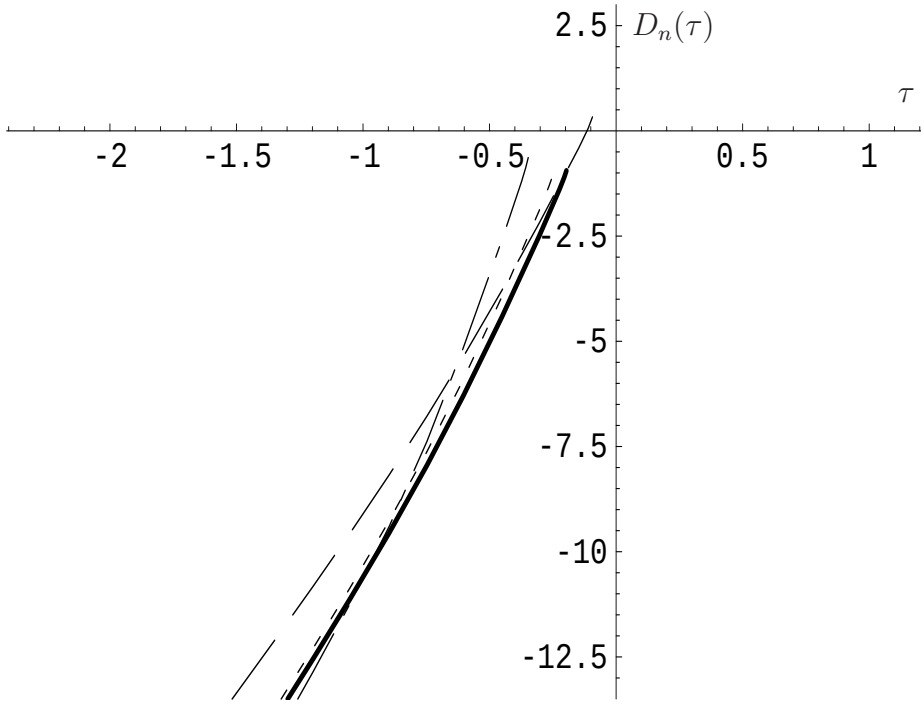
Para estos métodos compuestos obtenemos su correspondiente función $D_n(\tau)$ que acota la norma del error local $\|\delta(y,h)\|$ para $h = \frac{\tau}{L(y)}$ basándonos en el Teorema 1, al igual que hacíamos en el Ejemplo 4.

Mostramos las gráficas de las funciones $D_n(\tau)$ de los métodos de orden alto, es decir, de orden 7, 8 y 9, en la Figura 4.9. Ya hemos comentado antes que estos métodos los hemos comparado sin componerlos, con la diferencia de coste computacional que tienen entre ellos. Cuando obtuvimos la función $D_n(\tau)$ del método de orden 9 observamos que el parámetro κ que nos da el Lema 10 para este método tenia un valor muy alto, y la gráfica de la función $D_n(\tau)$ nos confirmó los malos resultados que esperábamos para este método. En la figura se pueden ver cuatro curvas, la que corresponde al método de orden 9 es la curva compuesta por trazos largos y cortos. La curva continua corresponde al método Dopri8, la curva con trazos largos corresponde al método de orden 7, y finalmente, la curva con trazos cortos, la que va paralela a la continua, es la del método de orden 8 utilizada por *Mathematica*.

En la gráfica de la Figura 4.9 podemos ver que la curva correspondiente al método de orden 8 utilizado por *Mathematica* está en todo momento por encima de la curva continua, que corresponde a Dopri8. Los dos métodos tienen el mismo número de etapas, por lo que concluimos que Dopri8 obtiene mejores cotas del error local para todos los valores de $\tau = hL(y)$ que el método de orden 8 de *Mathematica*. En cuanto al método de orden 7, para valores pequeños de τ, claramente Dopri8 obtiene mejores resultados, mientras que para valores grandes de τ los resultados para ambos métodos empiezan a igualarse. Teniendo en cuenta que el costo computacional del método de orden 7 es menor que el de Dopri8, 10 etapas contra 12, se podría decir que para valores muy grandes de τ el método de orden 7 utilizado por *Mathematica* puede llegar a ser un poco mejor que Dopri8, pero la mejora es mínima. Si nos fijamos en la gráfica correspondiente al método de orden 9, veremos que solo para valores muy pequeños de τ mejora los resultados que muestra Dopri8, pero teniendo en cuenta que tiene 3 etapas más que Dopri8 podríamos decir que dicha mejora se anula por el incremento del costo computacional. Resumiendo, de entre los cuatro métodos comparados en la Figura 4.9 es el método Dopri8 el que mejores resultados muestra. Sólo el método de orden 7 podría considerarse equiparable, o un poco mejor que Dopri8 pero para valores de τ grandes.

De la Observación 7 sabemos que para los métodos consistentes cuyo tablero de Butcher contiene exclusivamente valores positivos se tiene que $\kappa = 1$. Este es el caso de dos métodos que hemos incluido en la comparación: por un lado el *método clásico de Runge-Kutta* dado en (4.54), que es un método de orden 4 con 4 etapas internas, y por tanto la composición de 3 pasos de longitud $\frac{h}{3}$ del método se puede comparar con Dopri8. Y por otro lado el método que denotamos como *m4* dado en (4.56) que tiene 6 etapas, y

Figura 4.9: Curvas de las cotas teóricas del error local $D_n(\tau)$ del método de orden 7 (curva a trazos largos), método de orden 8 (curva a trazos cortos), Dopri 8 (curva continua) y el método de orden 9 (curva con trazos largos y cortos).



la composición de dos pasos de longitud $\frac{h}{2}$ del método tiene el mismo costo computacional que Dopri8.

En la figura 4.10 mostramos las funciones $D_n(\tau)$ de los métodos que tienen 12 etapas o que la composición de varios pasos menores generan métodos de 12 etapas:

- Cuatro pasos de longitud $\frac{h}{4}$ del método de orden 3 de *Mathematica*
- Tres pasos de longitud $\frac{h}{3}$ del método de orden 4 de *Mathematica*
- Tres pasos de longitud $\frac{h}{3}$ del método clásico de Runge-Kutta, *RK4*
- Dos pasos de longitud $\frac{h}{2}$ del método *m4*
- Dos pasos de longitud $\frac{h}{2}$ de Dopri5
- Dopri8

De entre todos ellos solo dos toman los valores más bajos para alguna franja de valores de $\tau = hL(y)$. Se trata de las dos curvas continuas y corresponden a Dopri8 (curva negra) y a m4 (curva gris). Para valores pequeños de τ Dopri8 es el mejor método, pero para valores grandes, es decir para pasos largos, m4 es el método que obtiene cotas del error local más pequeñas.

Todavía nos quedan dos métodos para comparar, el de orden 5 y el de orden 6. Mostramos la cota teórica del error local en la gráfica de la Figura 4.11 y se puede ver que para valores altos de $\tau = hL(y)$ el método de orden 5 puede ser considerado mejor que el de orden 6, ya que, aunque la curva de la cota sea similar para ambos, el costo computacional es menor para el método de orden 5. Este método tiene 7 etapas mientras que el método de orden 6 tiene 8 etapas. Para valores pequeños de τ la cota del error local para el método de orden 5 es mayor, pero la diferencia es muy pequeña, por lo que podemos decir que los dos métodos son muy similares. Hemos comparado el método compuesto por dos pasos de longitud $\frac{h}{2}$ del método de orden 5 con Dopri8 y con la composición de dos pasos de longitud $\frac{h}{2}$ del método *m4*. La comparación se puede ver en la gráfica de la Figura 4.12. La cota del error del método de orden 5 está representado por la curva a tramos. Su costo computacional es mayor que el de los otros dos, ya que tiene 14 etapas mientras que los otros tienen 12. Aun así, para valores pequeños de τ el método Dopri8 obtiene mejores resultados, mientras que para pasos largos m4 es mejor que el método de orden 5.

En la Figura 4.13 mostramos los esquemas de Runge-Kutta que ofrecen las cotas de los errores locales más pequeños para algún rango de valores de

Figura 4.10: Comparación de las funciones de las cotas del error local $D_n(\tau)$ del método de orden 3, el de orden 4, el método *clásico de Runge-Kutta*, m4, Dopri5 y Dopri8

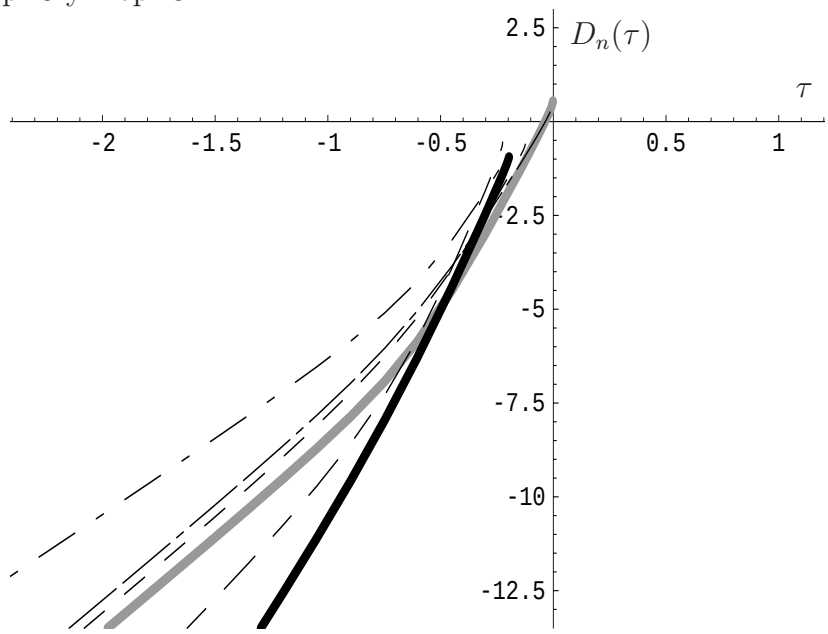


Figura 4.11: Cotas teóricas del error local de los métodos de orden 5 (curva con tramos cortos y largos) y de orden 6 (curva con tramos largos).

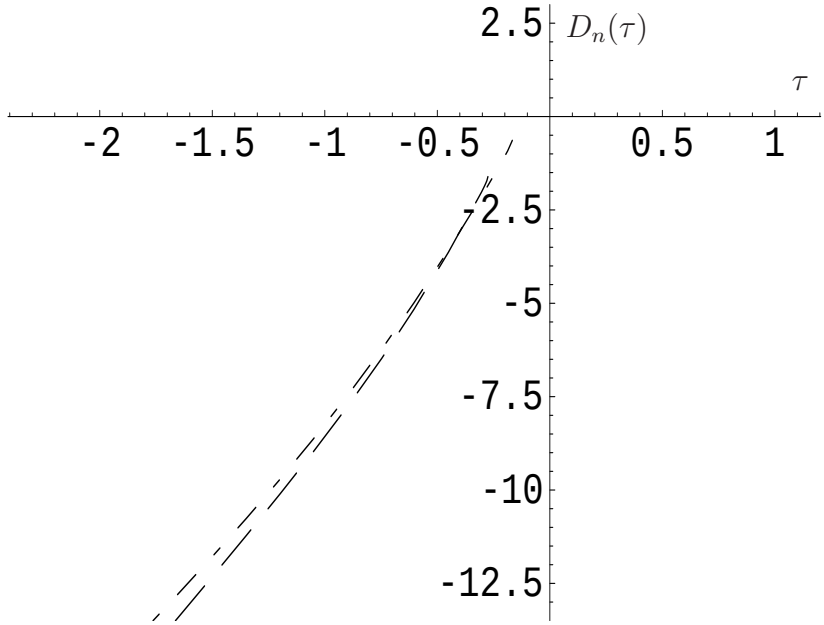
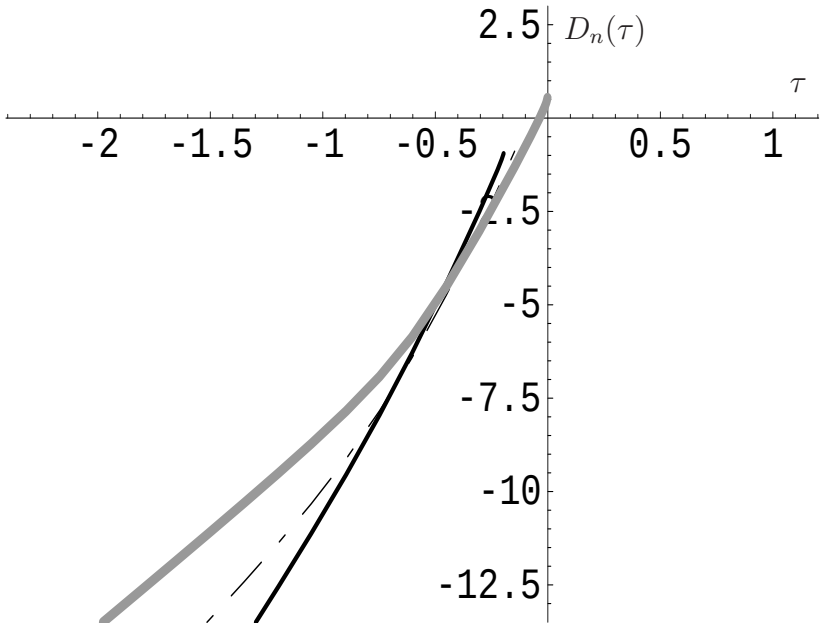


Figura 4.12: Comparación de la cota teórica del error local de Dopri8 (curva negra), m4 (curva gris) y el método de orden 5 (curva a tramos).

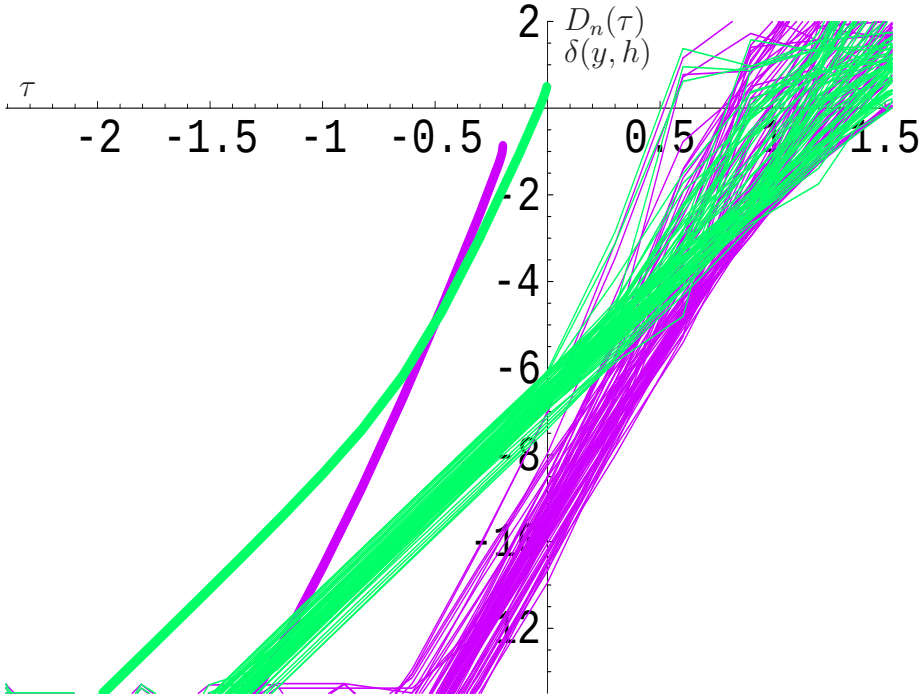


$\tau = hL(y)$. Se trata de Dopri8, cuya curva de error es la de color negro, y la composición de dos pasos del método m4, representado por la curva gris. Para valores grandes de τ la curva gris está por debajo de la curva negra, mientras que para valores pequeños la curva de Dopri8 esta por debajo de la curva gris. Además de las curvas correspondientes a la cota del error local mostramos en la misma figura los resultados numéricos obtenidos con esos dos métodos en la resolución del problema *Arenstorf*. En ellos se observa que, nuevamente, los peores resultados numéricos muestran una curva muy parecida a la curva obtenida para la cota teórica del error local, dada por $D_n(\tau)$.

Esta gráfica merece ciertos comentarios a modo de conclusiones generales que hemos observado en las diferentes pruebas y experimentos realizados.

Antes de nada, conviene destacar que los dos métodos que aparecen en la Figura 4.13 tienen un número de etapas muy distinto, el método Dopri8 tiene 12 etapas, mientras que el de orden 4 tiene 6. Evidentemente 12 etapas permiten un método de orden superior al que permiten 6 etapas, pero, a su vez, 6 etapas posibilitan una adecuación de la longitud de paso más dinámica que la que permiten 12 etapas, es decir, con el método *m4* cada seis evaluaciones de la función $f(y)$ podemos adecuar la longitud de paso, mientras que con el método Dopri8 podemos adecuarla cada 12 evaluaciones

Figura 4.13: La gráfica de la cota teórica del error local, $D_n(\tau)$, y los resultados numéricos del error local $\delta(y,h)$ (para $h = \frac{\tau}{\tilde{L}(y)}$) obtenidos en la resolución del problema *Arenstorf* con los métodos Dopri8, curvas oscuras, y m4, curvas más claras.



de $f(y)$. Todo ello indica que con el mismo coste computacional podemos resolver un problema de dos formas distintas: mas pasos con mayor adaptabilidad frente a menos pasos pero mayor orden. En el caso de la Figura 4.13 las gráficas corresponden a un paso del método Dopri8 frente a dos pasos de longitud $h/2$ del método *m4*, sin tener en cuenta la posibilidad de adaptación de la longitud de paso que permite el método de orden 4. No está claro cual de las dos estrategias es mejor cuando nos acercamos al limite de lo aceptable.

En relación a lo que hemos observado sobre el comportamiento de la función $D_n(\tau)$ hemos de resaltar unas cuantas cuestiones:

Por un lado, las cotas del error local dadas por $D_n(\tau)$ (donde la función $D_n(\tau)$ es obtenida según el teorema 1) son cotas que muestran, de algún modo, el comportamiento del método a la hora de resolver numéricamente los sistemas de ecuaciones diferenciales ordinarias.

Por otra parte, conviene que el dominio de definición de $D_n(\tau)$ sea lo más amplio posible, y de la Observación 7 se deduce que los métodos consistentes cuyos tableros de Butcher se componen de números positivos, así como aquellos métodos con algunos valores negativos, pero que cumplen que $\sum_{j=1}^{i} |a_{ij}| \leq 1$ son tales que $D_n(\tau)$ está bien definido para $\tau \in [0, 1)$. Entre estos métodos están los que aparecen en el trabajo de S.J. Ruuth [22]. En él podemos encontrar varios métodos que reúnen las condiciones que hemos mencionado y que a su vez muestran unas cotas de error teórico bastante buenas. No obstante, y aunque hayamos realizado varias pruebas y experimentos con dichos métodos, sobre todo con los métodos de orden 4 (compuesto por valores positivos) y el de orden 5 (con algunos valores negativos pero κ pequeño), los resultados obtenidos no mejoran los que hemos mostrado en la Figura 4.13 y por tanto no hemos incluido ninguna gráfica correspondiente a dichos métodos.

Otra cuestión a destacar es que el error local que se puede esperar en la resolución numérica de un problema depende no solo de h sino también de $L(y)$. Es decir, el valor $L(y)$, de alguna forma, indica la dificultad o la variabilidad en la resolución del problema, y en la medida en que crezca la dificultad hay que reducir la longitud del paso para poder mantener el error cometido en cada paso. En este sentido, parece razonable establecer una estrategia de variación de la longitud de paso h que mantenga $hL(y)$ aproximadamente constante para obtener errores locales de magnitud parecida a lo largo de la integración numérica.

4.4.5. Estimaciones numéricas de $L(y)$ y del error local

En general $L(y)$ depende del problema que se quiere resolver, y de alguna forma indica la variabilidad o la dificultad del problema a resolver. Sería muy útil si dispusiéramos de dicha información, ya que, como hemos comentado, la magnitud del error local depende de $hL(y)$. En (1.30) hemos mostrado la forma que tiene la expansión en serie de potencias de h de cada $f(Y_i)$ de (1.9). Si nos basamos en (1.30), sabemos que una combinación de las evaluaciones de f de las primeras k etapas de la aplicación de un paso del proceso de integración mediante un método de Runge-Kutta tiene la forma

$$\sum_{i=1}^{k} \lambda_i f(Y_i) = \sum_{t\in\mathcal{T}} \frac{h^{|t|-1}}{\sigma(u)} \sum_{i=1}^{k} \lambda_i c_i'(t) F(t)(y). \tag{4.57}$$

Dicho valor está acotado, teniendo en cuenta (4.26) y (4.25), por

$$\left|\left|\sum_{i=1}^{k} \lambda_i f(Y_i)\right|\right|_S \leq \sum_{t\in\mathcal{T}} h^{|t|-1} \sum_{i=1}^{k} |\lambda_i c_i'(t)\omega(t)|\, L_S(y)^{|t|-1}\, ||f(y)||_S\,, \tag{4.58}$$

y teniendo en cuenta (4.43) llegamos a

$$\left|\left|\sum_{i=1}^{k} \lambda_i f(Y_i)\right|\right| \leq \left|\left|S^{-1}\right|\right|\, ||S||\, \sum_{t\in\mathcal{T}} h^{|t|-1} \sum_{i=1}^{k} |\lambda_i c_i'(t)\omega(t)|\, L(y)^{|t|-1}\, ||f(y)||\,. \tag{4.59}$$

Si elegimos de forma adecuada los valores λ_i para $i = 1, \ldots, k$, podemos anular los coeficientes $\lambda_i c_i'(t)$ de los árboles de menos vértices. El número de árboles cuyos coeficientes podemos anular depende del número de etapas que queramos utilizar para la obtención de la combinación lineal. En general, necesitamos por lo menos dos etapas para anular el árbol de un vértice, tres etapas para anular también el coeficiente del árbol de dos vértices, 5 etapas para anular también los de los árboles de tres vértices, etc. Es decir, una etapa más que el número de coeficientes que queramos anular, ya que debemos evitar el caso trivial de $\lambda_i = 0$ para todo i. Con la anulación de los coeficientes de los árboles con menos de m vértices, la cota (4.59) se puede escribir de la siguiente forma:

$$\left|\left|\sum_{i=1}^{k} \lambda_i f(Y_i)\right|\right| \leq \left|\left|S^{-1}\right|\right|\, ||S||\, ||f(y)||\, \sum_{t\in\mathcal{T}_l, l\geq m} (hL(y))^{l-1}\omega(t) \left|\sum_{i=1}^{k} \lambda_i c_i'(t)\right|, \tag{4.60}$$

donde $\mathcal{T}_l$ es el conjunto de árboles con l vértices. De la ecuación (4.60) llegamos a,

$$\lim_{h\to 0} \frac{\left|\left|\sum_{i=1}^{k} \lambda_i f(Y_i)\right|\right|}{||f(y)||\, h^{m-1}} \leq \left|\left|S^{-1}\right|\right| \, ||S||\, L(y)^{m-1} \sum_{t\in\mathcal{T}_m} \omega(t) \left|\sum_{i=1}^{k} \lambda_i c_i'(t)\right|, \quad (4.61)$$

de donde podemos obtener una estimación $\hat{L}_h(y)$ de $L(y)$, definida como

$$\hat{L}_h(y) = \frac{1}{h}\left(\frac{||\sum_{i=1}^{k} \lambda_i f(Y_i)||}{||f(y)|| \sum_{t\in\mathcal{T}_m} \omega(t) \left|\sum_{i=1}^{k} \lambda_i c_i'(t)\right|}\right)^{\frac{1}{m-1}} \quad (4.62)$$

y tenemos que

$$\hat{L}(y) := \lim_{h\to 0} \hat{L}_h(y) \leq \left(\left|\left|S^{-1}\right|\right| \, ||S||\right)^{\frac{1}{m-1}} L(y).$$

Hemos realizado varios experimentos con distintos métodos, entre los que se incluyen Dopri8 y Dopri5 aplicados a varios problemas como *Arenstorf* o el de *Kepler*, y en ellos hemos tomado distintos valores de m. Para cada m hemos tenido en cuenta las condiciones de los árboles a anular y a dichas condiciones les hemos añadido una condición más para evitar la solución trivial $\lambda_i = 0$ para $i = 1, \ldots, k$. En concreto, para el método Dopri8 las condiciones son:

- para $m = 2$, utilizando las dos primeras etapas del proceso de integración hemos calculado los valores λ_i que cumplen

  $$\sum_{i=1}^{2} \lambda_i = 0$$

  junto con la condición añadida $\lambda_2 = 1$, para evitar el caso trivial $\lambda_1 = \lambda_2 = 0$.

- para $m = 3$ hemos utilizado tres etapas, y los valores de λ_i han de cumplir

  $$\sum_{i=1}^{3} \lambda_i = 0,$$
  $$\sum_{i=1}^{3} \lambda_i c_i = 0,$$

  además de la condición añadida, $\lambda_3 = \frac{1}{3}$.

- y para $m = 4$ necesitamos cinco etapas, y las condiciones que se deben cumplir son

$$\sum_{i=1}^{5} \lambda_i = 0,$$
$$\sum_{i=1}^{5} \lambda_i c_i = 0,$$
$$\sum_{i=1}^{5} \lambda_i \sum_{j=1}^{i} a_{ij} c_j = 0,$$
$$\sum_{i=1}^{5} \lambda_i c_i^2 = 0,$$

  y hemos añadido la condición $\lambda_5 = \frac{1}{12}$.

Los experimentos realizados se han encaminado a mostrar lo que la teoría nos indica, es decir, por un lado hemos mirado que los valores obtenidos para la expresión dada en (4.61) realmente atienden a la forma

$$\frac{||\sum_{i=1}^{k} \lambda_i f(Y_i)||}{||f(y)||} \approx C(hL(y))^{m-1}$$

y por otro lado hemos comprobado que las estimaciones $\hat{L}(y)$ obtenidas para $L(y)$ en la ecuación (4.61) son valores válidos, es decir, toman valores inferiores a las cotas teóricas de $L(y)$, y que para distintos valores de h las estimaciones de $\hat{L}(y)$ toman valores muy parecidos.

A la hora de verificar que la expresión dada en (4.61) realmente depende de $(hL(y))^{m-1}$, hemos supuesto que

$$\frac{||\sum_{i=1}^{k} \lambda_i f(Y_i)||}{||f(y)||} = C(hL(y))^{\alpha},$$

donde C y α son valores constantes desconocidos. Y para cada par de valores de h hemos resuelto el sistema de dos ecuaciones y dos incógnitas, y se ha podido ver que para distintos valores de m el valor resultante para α es siempre un valor muy cercano a $m - 1$. Estas pruebas han sido realizadas para distintos problemas, resueltos con diversos métodos. No parece que tenga ningún interés mencionar todos los experimentos, ya que los resultados han sido muy claros.

Para ver en qué medida las estimaciones $\hat{L}(y)$ de $L(y)$ obtenidas a partir de la ecuación (4.61) son estimaciones válidas, hemos realizado más experimentos, aunque aquí sólo mostramos los resultados obtenidos con el método Dopri8 y el problema *Arenstorf*. Para cada punto de la solución del problema mostrado en la Figura 4.2 hemos calculado el valor de la cota teórica de $L(y)$, es decir $\tilde{L}(y)$ obtenido como se explica en el Ejemplo 8 (y que puede ser aproximado por (4.53)). Estas cotas se pueden ver en la Figura 4.14, en ella aparece en escala logarítmica el valor de $\tilde{L}(y)$ de cada uno de los 80 puntos de la órbita de la solución del problema *Arenstorf*. A partir de estos valores hemos obtenido 30 valores de h para cada punto, de forma que tengamos 30 valores de $h\tilde{L}(y) \in [1, 0{,}001]$ distribuidos uniformemente en el intervalo. Con cada h hemos computado un paso del método de integración y hemos calculado el valor de la expresión (4.62) correspondiente a la estimación numérica $\hat{L}_h(y)$ de $L(y)$ que surge de (4.61).

Como era de esperar, para los diferentes valores de h suficientemente pequeñas, todas las estimaciones $\hat{L}_h(y)$ son muy parecidas, por lo que hemos aproximado el valor $\hat{L}(y)$ como el valor $\hat{L}_h(y)$ con $h = \frac{0{,}001}{\tilde{L}(y)}$.

Los experimentos se han repetido para $m = 2, 3$ y 4. En la Figura 4.15 podemos ver los resultados obtenidos con $m = 2$. En ella se pueden ver dos curvas, por un lado, vemos los mismos puntos mostrados en la Figura 4.14, que corresponden a las cotas teóricas $\tilde{L}(y)$ de $L(y)$. Y bajo esa cota de $L(y)$, con valores inferiores, se pueden ver los valores correspondientes a las estimaciones numéricas $\hat{L}(y)$ de $L(y)$. Para cada punto de la Figura 4.2 tenemos, por tanto, 2 puntos; el que corresponde a la cota teórica $\tilde{L}(y)$ más la estimación numérica $\hat{L}(y)$ de $L(y)$. Para el caso $m = 2$, la estimación utiliza las dos primeras etapas del proceso de integración numérica, y se puede observar que, aunque solo estemos utilizando la información de un único árbol para obtener $\hat{L}(y)$, lo que significa que estamos teniendo en cuenta una única diferencial elemental de f, los resultados obtenidos aproximan en cierta medida la forma de la curva que siguen los valores de las cotas teóricas $\tilde{L}(y)$ de $L(y)$.

En el caso de $m = 3$, las condiciones impuestas anulan los coeficientes de los árboles de uno y de dos vértices, por lo que la estimación de $L(y)$ se obtiene utilizando los dos árboles de 3 vértices, y para esta estimación se requieren por lo menos tres etapas del proceso de integración numérica. Los resultados obtenidos de esta forma se muestran en la Figura 4.16, y en ella volvemos a mostrar junto con el valor de $\tilde{L}(y)$ las estimaciones numéricas $\hat{L}(y)$ para cada punto de la Figura 4.2. En este caso la forma de la curva que muestran las estimaciones se aproxima más a la forma de la curva que muestran las cotas teóricas $\tilde{L}(y)$ de $L(y)$. Se puede volver a ver que las cotas

Figura 4.14: La gráfica muestra los valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ en escala logarítmica para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2

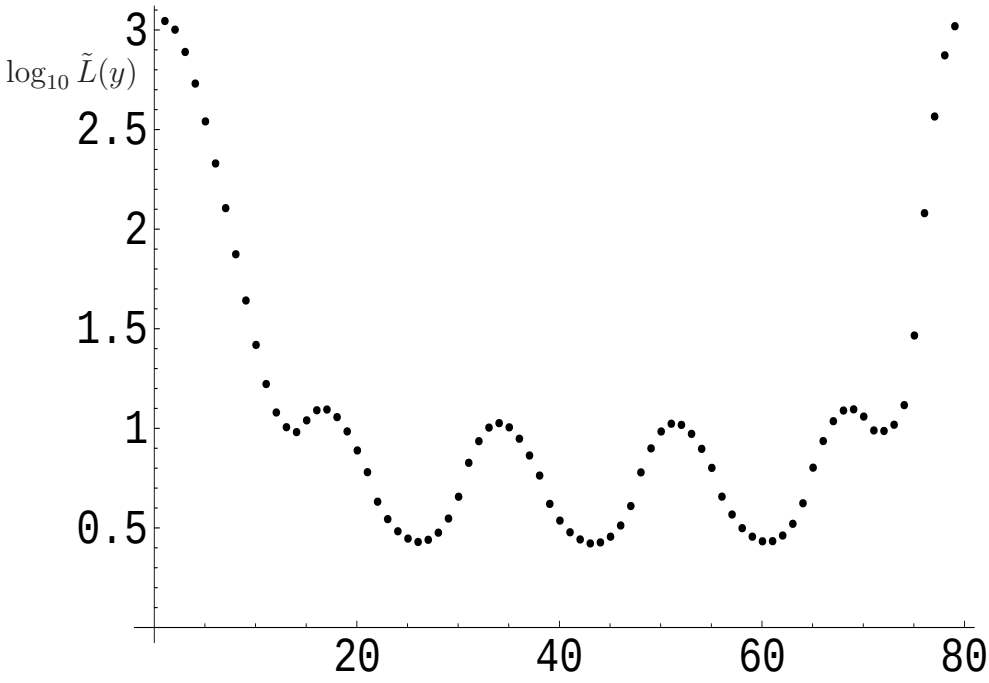


Figura 4.15: La gráfica muestra los valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2 junto con los valores $\hat{L}(y)$ estimados numéricamente para $m = 2$. Ambos valores se muestran en escala logarítmica.

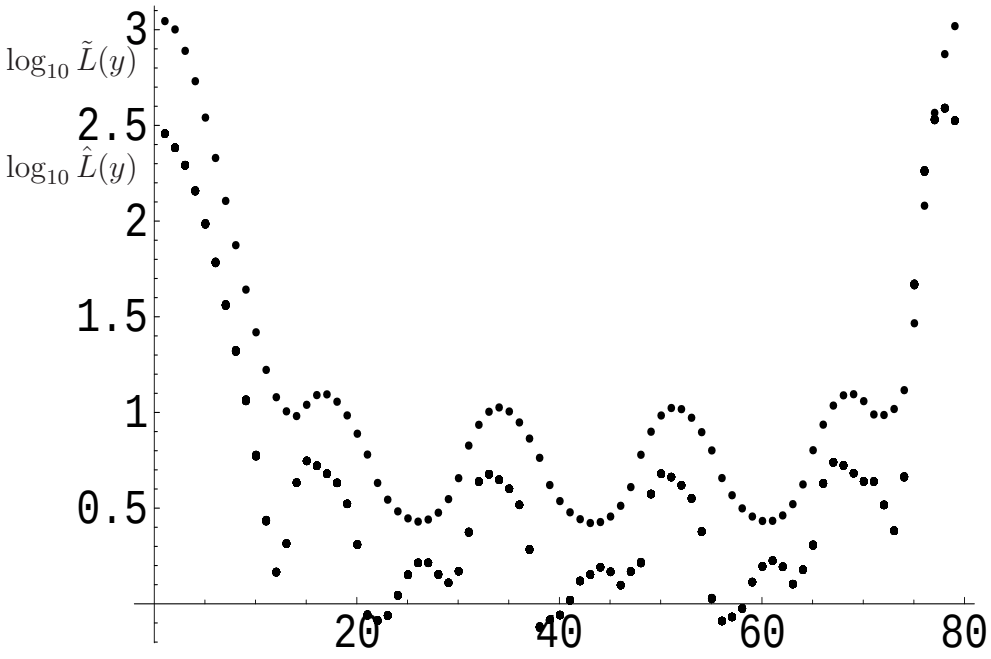
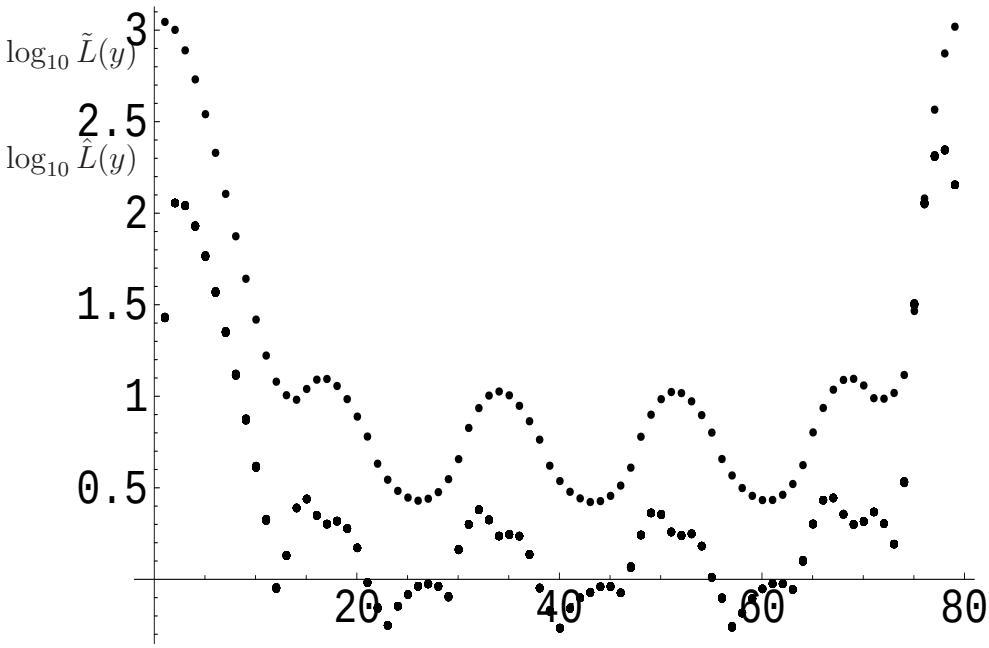


Figura 4.16: Valores de la cota teórica $\tilde{L}(y)$ de $L(y)$ en escala logarítmica para cada uno de los puntos de la solución del problema *Arenstorf* mostrados en la Figura 4.2 junto con los valores $\log_{10} \hat{L}(y)$ estimados numéricamente para $m = 3$.



teóricas vuelven a estar por encima de las estimaciones numéricas $\hat{L}(y)$ de $L(y)$.

Por último, mostramos los resultados obtenidos para $m = 4$ en la Figura 4.17. En este caso se han anulado los coeficientes de los árboles de uno, de dos y de tres vértices, por lo que en la estimación de $L(y)$ toman parte los cuatro árboles de cuatro vértices. Para la combinación lineal hemos utilizado las cinco primeras etapas del método Dopri8 y las gráficas nos muestran que la curva que forman las estimaciones $\hat{L}(y)$ en los diferentes puntos se aproxima mucho a la forma que toman las cotas teóricas $\tilde{L}(y)$ de $L(y)$, pero siempre por debajo de los valores de la cota.

En definitiva, podemos concluir diciendo que la estimación de los valores de $L(y)$ es factible y no es computacionalmente demasiado costoso. En la Figura 4.18 mostramos las estimaciones obtenidas tanto con $m = 3$ como con $m = 4$. En la figura mostramos para cada punto de la órbita de la solución del problema 3 valores: la cota teórica $\tilde{L}(y)$, la estimación $\hat{L}(y)$ de $L(y)$ obtenida con $m = 3$, y otra estimación $\hat{L}(y)$ de $L(y)$, esta última

Figura 4.17: Para el caso de $m = 4$ las estimaciones $\hat{L}(y)$ de $L(y)$ en los puntos mostrados en la Figura 4.2 siguen la forma de la curva mostrada por las cotas teóricas $\tilde{L}(y)$ de $L(y)$.

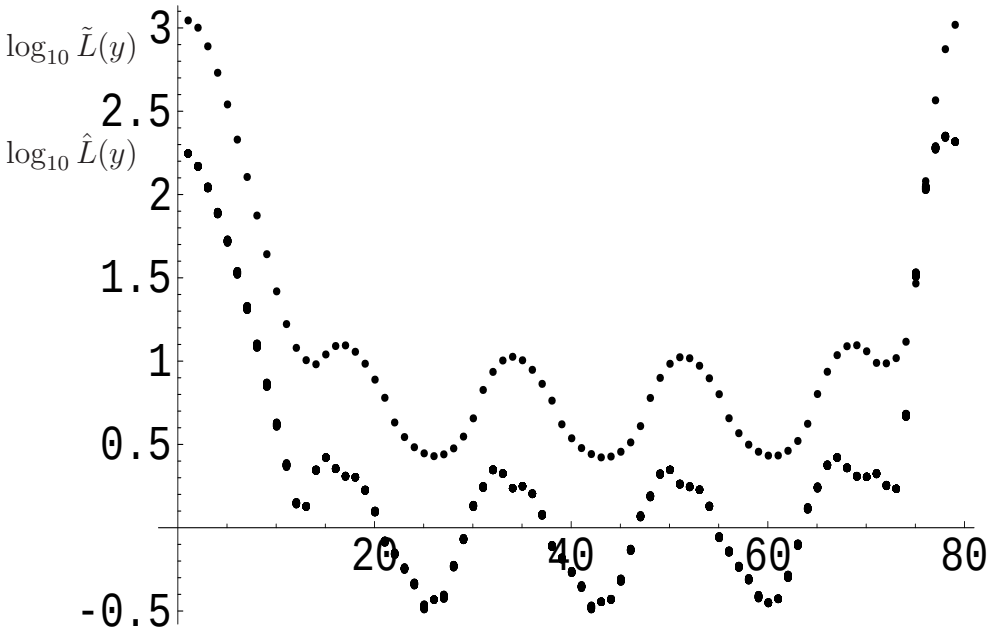
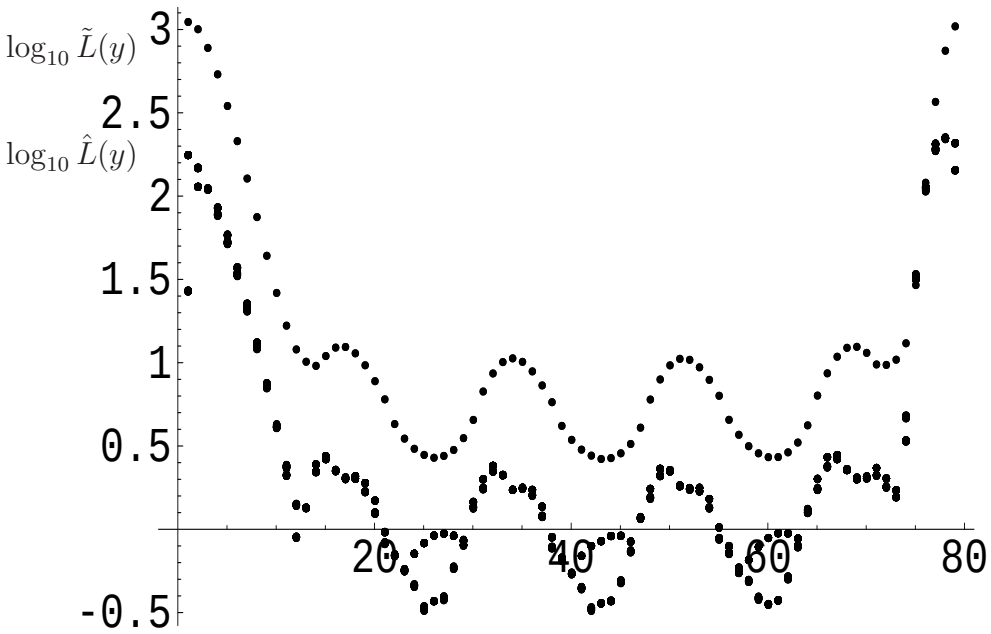


Figura 4.18: Cota teórica $\tilde{L}(y)$ de $L(y)$ junto con las estimaciones numéricas $\hat{L}(y)$ de $L(y)$ obtenidas para $m = 3$ y para $m = 4$ en cada uno de los puntos mostrados en la Figura 4.2.



obtenida con $m = 4$. Se puede observar que la diferencia entre los puntos que corresponden a las estimaciones es muy pequeña en la mayoría de los puntos.

Capítulo 5

Conclusiones y plan de trabajo futuro

5.1. Conclusiones

En los capítulos anteriores se ha estudiado tanto el error local (1.15) como global (1.16) de los métodos de Runge-Kutta.

En cuanto al error global podemos destacar, a modo de conclusiones, lo siguiente:

- Hemos dado una metodología para construir métodos que aporten estimaciones del error global cometido a lo largo de la integración numérica, para los que las condiciones a cumplir toman la forma (2.43).

- Hemos mostrado que, sin un elevado coste computacional, es posible obtener información útil sobre el error global mediante los métodos dados en (2.13)-(2.14).

- Se ha estudiado la propagación de los errores locales en el proceso de la integración, y en los Lemas 3 y 4 se han dado a conocer ciertas condiciones bajo las que se puede acotar el error global en función de las longitudes de cada paso.

- Hemos visto cómo puede ser utilizada la información sobre el error global para mejorar la eficiencia del proceso del cálculo de la solución numérica. Basándose en las condiciones del Lema 4, la información sobre el error global posibilita la utilización de tolerancias variables (3.12), lo que permite la elección de longitudes de paso más largas,

sin que por ello la cota del error global se vea incrementada sustancialmente.

En lo relativo al error local se pueden resaltar las siguientes conclusiones:

- Hemos acotado de forma rigurosa el desarrollo en serie de potencias de h del error local de los métodos de Runge-Kutta dada en (4.9) mediante la función $D_n(\tau)$ dada en (4.35), lo que nos ha permitido comparar la precisión de diferentes métodos de Runge-Kutta.

- Las cotas obtenidas para el error local de los métodos de Runge-Kutta nos han dado información sobre los criterios, tales como el dado en la Observación 7, que pueden ser utilizados para construir métodos de Runge-Kutta optimizados.

- El estudio de las cotas del error local nos ha llevado a entender mejor cómo cambia la escala de tiempo a lo largo de una solución del sistema de ecuaciones diferenciales ordinarias. Hemos mostrado que la escala de tiempo depende de $L(y)$ dado en (4.45) y que si en la integración se avanza manteniendo constante $hL(y)$, entonces los errores locales se mantienen en unos márgenes muy estrechos.

- Se ha dado una forma de obtener estimaciones numéricas de $L(y)$ (4.62) de forma que no suponga un incremento de coste computacional.

Todo ello nos da las bases para poder trabajar en el desarrollo de algoritmos de resolución de ecuaciones diferenciales ordinarias que exploten las ventajas que se puedan derivar de las aportaciones realizadas. La inclusión de dichas ventajas nos dará como resultado algoritmos de integración de EDOs que con menor coste computacional obtengan mejores resultados.

5.2. Mejoras en el proceso de la integración

Las mejoras que mostramos a continuación se basan en las aportaciones realizadas sobre el error local y global del proceso de integración mediante métodos de Runge-Kutta. En cuanto al error global, podemos utilizar dicha información de dos formas:

- Podemos controlar que el error global no exceda de un máximo tolerable, y en caso contrario, parar el proceso de integración.

- Podemos variar la política de adecuación de la longitud de paso introduciendo en ella la información acerca del error global.

Respecto al error local, tenemos la posibilidad de utilizar nuevas técnicas de adecuación de la longitud de paso basándonos en los resultados de la Sección 3.3, e incluso podemos elegir mejor los métodos que vayamos a utilizar dependiendo de la tolerancia que quiera utilizar el usuario.

Primeramente analizaremos el algoritmo básico de resolución de EDOs, y seguidamente mostraremos cómo se puede mejorar dicho proceso con la inclusión de las ventajas derivadas del estudio de los errores.

5.2.1. Proceso básico de resolución de una EDO mediante métodos de RK

Cualquier código de uso general para la resolución de EDOs debe pedir al usuario ciertos datos mínimos: por un lado debe especificar el problema que se debe resolver, dando las ecuaciones correspondientes al sistema (1.1), también debe indicar el intervalo (t_0, t_f) para el que se quiere resolver el problema, y el valor inicial $y(t_0)$, es decir, el valor que toma la solución para el valor inicial de la variable independiente. Además, deberá especificar de alguna forma la precisión requerida para el proceso de integración numérica, normalmente con un valor de la tolerancia al error local.

Con todos estos datos, el proceso básico de la resolución de una ecuación diferencial ordinaria mediante métodos de Runge-Kutta explícitos realiza una iteración hasta llegar a la solución final. La iteración parte del valor inicial de la variable independiente y va dando pasos de longitud variable hasta llegar al valor final de la variable independiente. Podemos ver el esquema del algoritmo en la Figura 5.1, y en términos generales, el proceso, expresado en lenguaje C, debería tener la siguiente forma:

```
ivp_solve()
{
obten_val_inicial(&y0, &t0, &tf, &tol_local);
obtener_h_inicial(tol_local,&h);
for (t = t0;        //inicialización
     t < tf;        //condición final
     h = h_nuevo)           //actualización
  {
  ivp_step(t, h, y, &y_nuevo, &δ̄) ;
  post_step_funct(y, h, y_nuevo, δ̄...);
  if (aceptable(err_local,tol_local, &h_nuevo...))
    {
    t += h;
```

```
    y = y_nuevo;
    }
  }
devolver_resultados();
}
```

En este proceso simple faltarían por definir las siguientes funciones:

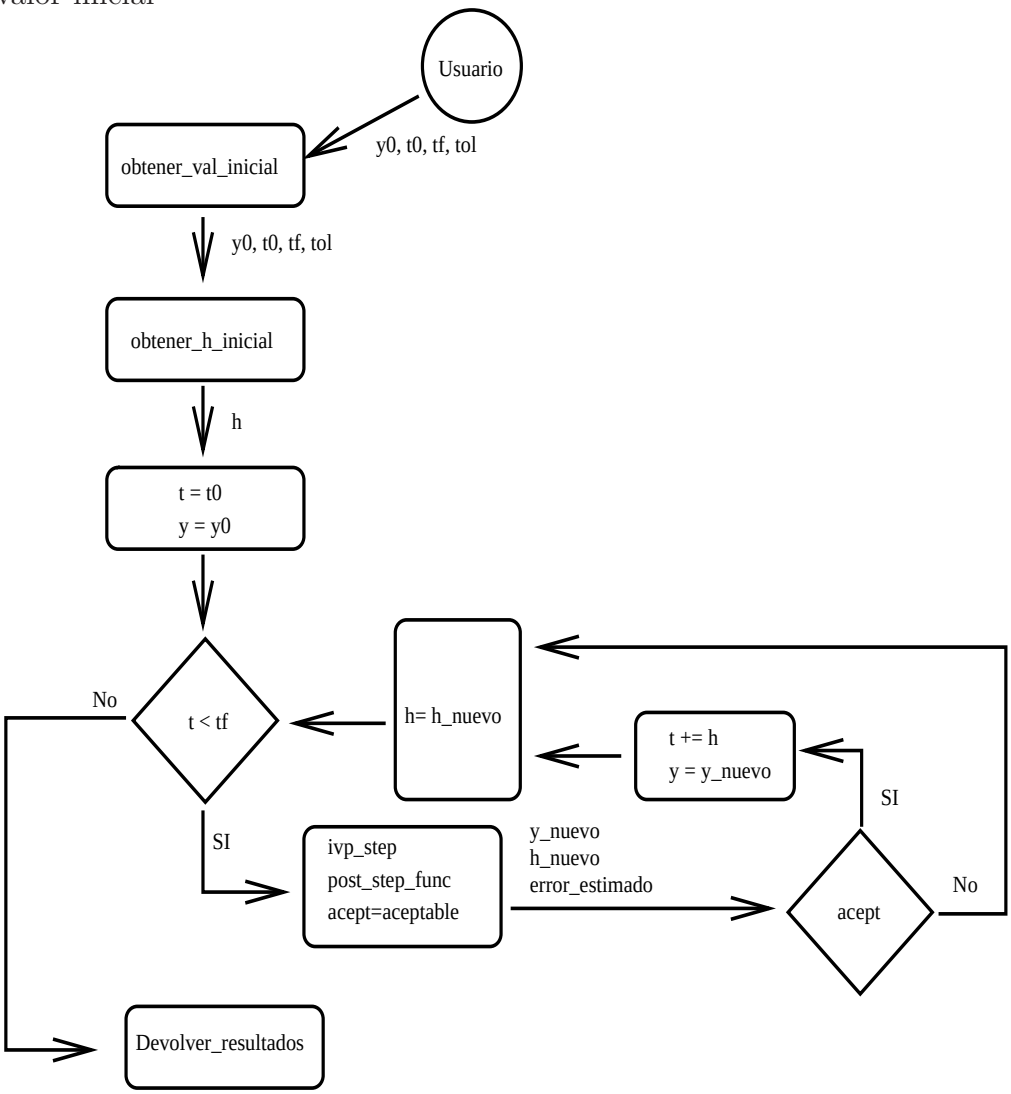
1. *obtener_h_inicial* debe calcular una longitud de paso inicial, acorde a la tolerancia, con la que comenzar el proceso iterativo.

2. *ivp_step* se encarga de dar un paso en el proceso de la integración, es decir, partiendo del valor y que es la solución numérica para $y(t)$ debe obtener el valor de la solución numérica y_{nuevo} que se pretende que sea la solución numérica para $y(t+h)$. Esta función, además, debe devolver $\bar{\delta}(y,h)$, que es la diferencia entre dos soluciones numéricas. El control del error local se suele realizar mediante el uso de otra solución numérica $\hat{y}$ obtenida mediante un método embebido o encajado en el tablero de Butcher, por lo que no supone un incremento significativo de coste computacional. La diferencia entre las dos soluciones numéricas, $\bar{\delta}(y,h) = \psi_h(y) - \hat{\psi}_h(y)$, es lo que se utiliza como estimación del error local, pero no es el error local. No obstante, se espera que, manteniendo dicha diferencia mas o menos constante (o por debajo de cierta tolerancia), indirectamente, se controle el error local $\delta(y,h)$.

$$\begin{aligned} y_{\text{nuevo}} &= \psi_h(y), \\ \bar{\delta} &= y_{\text{nuevo}} - \hat{\psi}_h(y). \end{aligned}$$

3. *aceptable*, esta función se encarga de comparar la diferencia entre las dos soluciones numéricas con la tolerancia al error local, para ello necesita hacer uso de alguna expresión parecida a las mostradas en la Sección 1.10 como pueden ser (1.45) ó (1.46) . El usuario debe tener la posibilidad de especificar la expresión que le interese, pero en su defecto el código debe aportar expresiones genéricas o estándar para que el usuario pueda elegir entre ellas. Según sea el valor de la expresión el paso se aceptará o se rechazará, pero en todo caso debe dar la longitud del nuevo paso que se debe realizar en el proceso de la integración. La nueva longitud de paso se calcula utilizando una expresión parecida a la mostrada en (1.47).

Figura 5.1: Algoritmo general del proceso de resolución de problemas de valor inicial



- Si esta función decide que el paso es aceptable el proceso debe avanzar y el nuevo paso que hay que computar debe partir de $t + h$, siendo y_{nuevo} el valor aceptado como solución numérica para $y(t + h)$, y calculará la solución numérica para un paso de longitud h_{nuevo},
- mientras que si decide que el paso no es aceptable, tiene que volverse a calcular el paso partiendo de nuevo de t, pero con la longitud de paso rectificada h_{nuevo}.

4. post_step_funct, esta otra función debe dar la posibilidad de que el usuario pueda realizar la acción que le interese tras el cómputo del paso. Bien sea el cálculo de soluciones en puntos internos al intervalo avanzado, bien sea el almacenamiento de los valores de cada paso para su posterior manipulación o estudio, etc. El usuario ha de tener la posibilidad de elegir o incluso especificar las acciones a realizar, y en su defecto el código debe aportar distintas posibilidades (entre las que se incluye el no hacer nada) para que el usuario haga uso de las mismas.

5.2.2. Control del error global

El proceso definido en la Sección 5.2.1 se guía por la tolerancia al error local aportada por el usuario. No tiene en cuenta el error global y en este sentido, habría que empezar a variar el punto de vista tradicional de resolución de problemas de valor inicial: el usuario, además de la tolerancia al error local, debería tener la opción de aportar una tolerancia al error global de la solución. Es decir, si la acumulación y propagación de los errores locales se hace intolerable se debería parar el proceso, o como mínimo habría que avisar al usuario de lo que ocurre para que decida si merece que continúe el proceso o no. Es decir, antes del cálculo de cada paso habría que introducir un control que nos dé dicha posibilidad, o al menos habría que dar la opción de activar el control y dejar en manos del usuario la posibilidad de hacer uso del nuevo sistema de control.

Esta posibilidad obliga a que la función *ivp_step* tenga que aportar, además del valor y_{nuevo} una segunda solución al problema $\bar{y}_{\text{nuevo}}$, o mejor, una estimación del error global. Una forma eficiente para la obtención de dicha estimación se basa en la utilización de los métodos presentados en (2.13-2.14). Por tanto, la función *ivp_step* deberá calcular:

$$y_{\text{nuevo}} = \psi_h(y, \bar{y}),$$

$$err_{\text{global}} = E_h(y, \bar{y}),$$
$$\bar{\delta} = y_{\text{nuevo}} - \hat{\psi}_h(y).$$

Aparte de cambiar la función que computa cada paso, hay que darle al usuario la opción de establecer la tolerancia respecto al error global. Y en la iteración que controla el proceso de integración se puede controlar que el error global del proceso sea aceptable, por lo que habría que cambiar la condición que controla la iteración, y en su lugar pondríamos:

```
.../...
err_global= 0;
for (t = t0;        //inicialización
     control_err_global(err_global, tol_global) &&
     t < tf;        //condición final
     h = h_nuevo)           //actualización
   {
   ivp_step(t_inicial, h, y_inicial,
            &y_nuevo, &err_global,δ̄);
   .../...
   }
```

La función *control_err_global* se encargaría de mirar si los resultados que va obteniendo el proceso son aceptables desde el punto de vista del error global; si hiciera falta debería preguntar al usuario si hay que detener o no la integración y finalmente deberá devolver un valor indicando si el proceso ha de continuar o no.

5.2.3. Tolerancia variable al error local

Si el proceso de integración de la EDO maneja información relativa al error global podemos hacer uso de las ventajas que aporta la utilización de tolerancias variables al error local de la forma (3.12), tal y como hemos mostrado en la Sección 3.3. Esta estrategia nos pide que el usuario aporte información sobre el valor de K definido en (3.13), al que habría que asignarle un valor por defecto predefinido como, por ejemplo $K = 0{,}2$, o como hemos comentado anteriormente, habría que buscar alguna forma automática para obtener dicha información.

En la Sección 3.4 hemos comentado la forma de introducir en el código el control de la tolerancia local. Hemos mostrado cómo limitar el cambio de la tolerancia para evitar cambios demasiado grandes, y a su vez, hemos

comentado que el cambio se realiza cada cierto número de pasos. Todo ello exige algunos cambios en el código:

- El usuario debe tener la posibilidad de activar o no el mecanismo, para ello utilizamos una variable *tol_variable* que indica si el mecanismo está activo ($tol\_variable > 0$) o no ($tol\_variable < 0$), y además, si está activo, el contenido de la variable indica cada cuantos pasos hay que actualizar la tolerancia.

- Ya que hay que controlar el número de pasos desde la última vez que se adecuó la tolerancia, cada paso que se avance debe incrementar un contador para controlarlo, pero cuando se actualice la tolerancia hay que reinicializar el contador.

- Cuando el contador de pasos para controlar la actualización de la tolerancia lo indique, hay que variar el valor de la tolerancia al error local tal y como se indica en (3.12). Y habrá que tener en cuenta el máximo cambio permitido cada vez, e incluso la máxima tolerancia permitida. Estos dos últimos valores deben tener unos valores por defecto, pero al usuario se le debe permitir la posibilidad de establecerlos a su gusto en la inicialización de la integración.

- La función *aceptable* encargada de aceptar o rechazar el paso, como ya hemos comentado, debe proporcionar la longitud de paso óptima para el siguiente paso a computar, y para ese cálculo deberá utilizar la tolerancia variable, y no la tolerancia inicial establecida por el usuario. No obstante, hay que guardar el valor establecido inicialmente por el usuario, ya que si queremos controlar la variación máxima permitida para la tolerancia habrá que conocer el valor dado por el usuario.

5.2.4. Control de la variación de la escala temporal del problema

A la hora de acotar el error local de cada paso de la integración mediante métodos de Runge-Kutta, en el Teorema 1 hemos dado la cota definida por (4.35). Esta cota depende de h y de $L(y)$ (4.18), y en la Subsección 4.4.5 hemos mostrado la forma de obtener estimaciones numéricas del valor $L(y)$. El procedimiento a seguir no supone casi ningún coste computacional adicional, ya que se basa en la reutilización de los valores $f(Y_i)$ de las etapas intermedias del paso y del propio valor $f(y)$. La estimación de $L(y)$ viene dada por (4.62), y junto con la longitud de paso h utilizada en cada paso estamos en condiciones de predecir en cierta medida la alteración que va a sufrir el error local del paso (1.15).

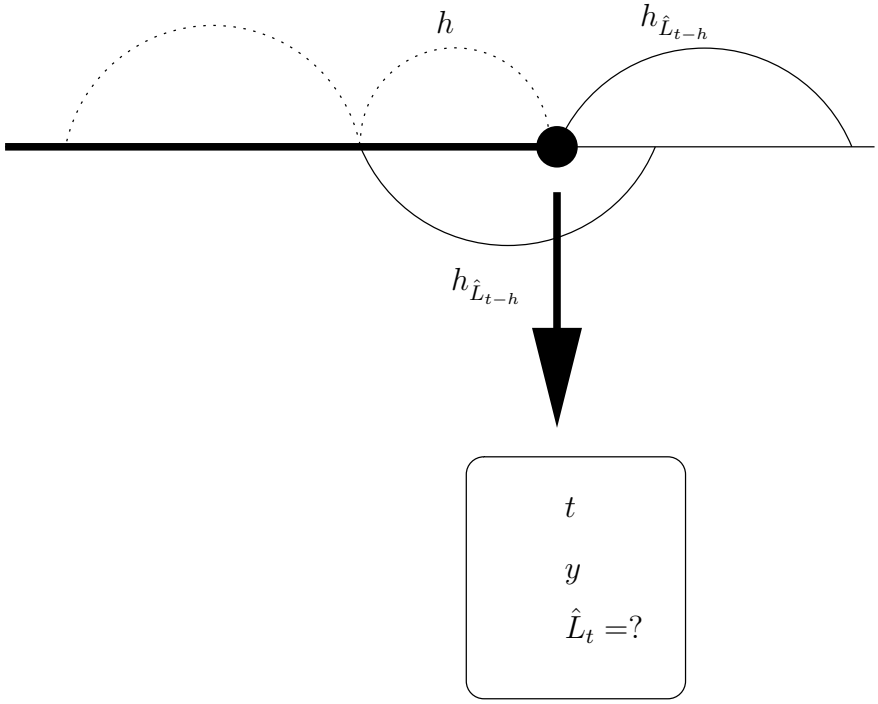
La estrategia estándar de adecuación de la longitud de paso, y por tanto la gran mayoría de las aplicaciones del mercado, trata de mantener el error local constante, por debajo de la tolerancia, y para ello hace variar la longitud del paso en función de la ecuación (1.47). No obstante, utiliza la diferencia entre dos soluciones numéricas como estimación del error local y se basa en la esperanza de que dicha diferencia tenga un comportamiento parecido al error local. En contraposición, hemos visto en la Figura 4.4 que en distintos momentos de la integración de un problema, si mantenemos fijo el valor de $hL(y)$ el error local obtenido se mantiene en unos márgenes muy estrechos, por lo que si conociéramos $L(y)$ tendría sentido adecuar la longitud de paso de forma que $hL(y)$ se mantenga constante a lo largo de la integración.

En (4.62) tenemos una forma de obtener una estimación $\hat{L}(y)$ de $L(y)$ que no supone un incremento computacional significativo, no mayor que el cálculo de una segunda solución numérica mediante un método embebido en el tablero de Butcher. Esta estimación puede ser utilizada para la adecuación de la longitud de paso en la integración, en lugar de utilizar la segunda solución numérica.

El valor de $\hat{L}(y)$ del que podemos disponer en cada momento es la estimación correspondiente al comienzo del último paso computado, es decir, si hasta este momento hemos aceptado la solución numérica para t, y para ese punto hemos aceptado el valor y como solución numérica, podemos disponer de la estimación $\hat{L}(y_{t-h})$ de $L(y_{t-h})$ que se haya realizado al computar ese paso. Con esa estimación $\hat{L}(y_{t-h})$ se puede predecir la longitud de paso que mantiene el error local dentro de unos márgenes muy estrechos para ese mismo paso, aunque nosotros vayamos a utilizar la estimación $\hat{L}(y_{t-h})$ para obtener la longitud del siguiente paso. En la Figura 5.2 podemos ver la situación que se da antes de dar un nuevo paso: hasta el momento se ha aceptado la solución numérica para el punto en el que la variable independiente toma el valor de t, el último paso aceptado ha comenzado en un punto anterior, y es justo en ese punto anterior donde se ha realizado la estimación $\hat{L}(y_{t-h})$, por lo que a ese paso previo le hubiera correspondido la longitud de paso $h_{\hat{L}_{t-h}}$.

Tanto en el método tradicional de adecuación de la longitud de paso, basada en la estimación del error local mediante la diferencia de dos soluciones numéricas, como en el método basado en la estimación de $L(y)$, la información disponible en cada momento se refiere al último paso computado, pero podemos esperar que el cambio en la escala de tiempo no se dé muy bruscamente, por lo que en el nuevo paso a computar se espera que la longitud de paso adecuada para el paso previo sea una longitud de paso

Figura 5.2: Proceso de adecuación de la longitud de paso: la longitud de paso que en el anterior paso hubiera mantenido el error local dentro de unos márgenes muy estrechos es la que se utilizará para avanzar en la integración



aceptable. No obstante, la adecuación de la longitud de paso basada en la estimación de $L(y)$ nos da la posibilidad de obtener la nueva estimación de $L(y)$ (correspondiente a la situación actual y) sin tener que esperar a que se calculen todas las etapas internas del paso, por lo que la corrección de la longitud de paso podría tener un coste computacional menor (en caso de que el paso en curso hubiera que rechazarse debido a un cambio sustancial del valor de $L(y)$).

Los cambios en el código no son muy grandes, pero afectarían a la propia estructura del cuerpo de la iteración, ya que tanto la decisión de aceptar o rechazar el paso como la longitud óptima del paso se pueden establecer antes de la finalización del mismo.

Por otra parte, nos interesa combinar la posibilidad de utilizar una tolerancia variable, comentado en la Subsección 5.2.3, con el nuevo método de control del error local basado en la estimación de $L(y)$. Para poder combinar ambas mejoras habrá que establecer una relación entre la tolerancia (inicialmente aportada por el usuario) y los valores $hL(y)$ correspondientes a dicha tolerancia, y el hecho de variar la tolerancia debe reflejarse en un

nuevo valor $hL(y)$.

5.2.5. Elección del método a utilizar en la integración

Cuando en la Subsección 4.4.4 hemos comparado los métodos de Runge-Kutta hemos utilizado la función $D_n(\tau)$ definida en (4.35) asociada a cada uno de los métodos. En la comparación hemos comentado el comportamiento de unos métodos es mejor que el de otros dependiendo del rango de valores de $\tau = hL(y)$, y en este sentido, creemos que es posible construir métodos de Runge-Kutta optimizados para ciertos rangos de valores de $\tau = hL(y)$. Si obtuviéramos un conjunto de estos métodos de forma que para cualquier valor de $\tau = hL(y)$ pudiéramos elegir el método que, a igual coste computacional, vaya a obtener unos resultados numéricos con el menor error local posible, estaríamos en condiciones de elegir el método con el que resolver el problema que nos plantee el usuario. Esta elección, en principio, seria una elección a mantener durante toda la integración del problema, pero en la medida que avancemos en la resolución numérica, podríamos ir obteniendo junto con la solución numérica, estimaciones del error global, lo que nos permitiría variar la tolerancia del error local, es decir, estaríamos aumentando la tolerancia, lo cual es equivalente a aumentar el valor de τ. Para el nuevo valor de $\tau = hL(y)$ es posible que otro método obtenga mejores resultados, lo que nos lleva a la necesidad de cambiar el método utilizado en la resolución del problema.

5.3. Plan de trabajo futuro

De las mejoras planteadas en la Subsección 5.2 se desprenden algunas líneas de trabajo que quedan por cubrir, y por otra parte, se puede plantear la extensión de estas aportaciones a otros métodos numéricos:

- Queda por cubrir la obtención de métodos de Runge-Kutta optimizados para diferentes rangos de valores de $hL(y)$.
- Habrá que obtener métodos de la forma (2.13)-(2.14) para la estimación del error global de los métodos optimizados.
- Todavía hay que trabajar en la implementación de la adecuación de la longitud de paso que tenga en cuenta los nuevos métodos, las estimaciones del error global y la estrategia de control del error local basada en estimaciones de $L(y)$.

- Es posible que el método de la variación de la escala de tiempo mediante $L(y)$ se pueda aplicar en los métodos simplécticos para problemas Hamiltonianos, más concretamente, para determinar para clases de problemas Hamiltonianos concretos, el cambio de escala temporal apropiado para una implementación eficiente de métodos simplécticos.

Creemos posible la construcción de métodos optimizados para rangos de valores de $hL(y)$. Para ello habría que optimizar los métodos de Runge-Kutta de forma que la función $D_n(\tau)$ correspondiente al método sea optimizado para el rango de τ elegido. Dicha optimización habrá que realizarla teniendo en cuenta los criterios derivados del estudio de las cotas del error local, tales como el comentado en la Observación 7 derivado de la Proposición 2, así como los valores que toma la función $D_n(\tau)$, definida según el Teorema 1, para los valores de τ pertenecientes al rango elegido. Las búsquedas de estos métodos optimizados requerirá la utilización de máquinas de elevada capacidad de cálculo.

Basándonos en los métodos optimizados habrá que construir los métodos Runge-Kutta embebidos con estimación del error global (2.13)-(2.15) con las condiciones de independencia adecuadas y, probablemente, teniendo en cuenta las condiciones (2.41)-(2.42), para que la optimización realizada sobre el método no se vea afectada por el segundo método $\bar{\psi}_h(y, \bar{y})$.

Una vez que tengamos los métodos optimizados para diferentes rangos de valores de $hL(y)$, junto con las estimaciones del error global aportadas por los métodos globalmente embebidos, habrá que modificar el algoritmo de resolución de ecuaciones diferenciales ordinarias para que la adecuación de la longitud de paso tenga en cuenta la propagación de los errores locales y la estimación de $L(y)$ dada por (4.62). En función de la propagación de los errores locales habría que activarse la posibilidad del uso de la tolerancia variable, y para ello habría que buscar algún medio de calcular automáticamente el valor K dado en (3.13). Además, el cambio de la tolerancia deberá permitir el cambio del método a utilizar en la resolución numérica, ya que es posible que para el valor de $hL(y)$ derivado de la nueva tolerancia, otro método tenga un comportamiento mejor que el método utilizado hasta ese momento.

Bibliografía

[1] G. Bennetin and A. Giorgilli. On the hamiltonian interpolation of near to the identity symplectic mappings with application to symplectic integration algorithms. *J. Stat. Phis.*, 74:1117–1143, 1994.

[2] P. Bogacki. and L.F. Shampine. An efficient runge-kutta (4,5) pair. Technical report, Math. Dept. Southern Methodist Univ., Dallas, Texas, 1989.

[3] J.C. Butcher. Coefficients for the study of runge-kutta integration processes. *J. Austral. Math. Soc.*-, (3):185–201, 1963.

[4] J.C. Butcher. *The Numerical Analysis of Ordinary Differential Equations*. John Willey and Sons, 1987.

[5] J.C. Butcher and J.M. Sanz-Serna. The number of conditions for a runge-kutta method to have effective order *p*. *Applied Numerical Mathematics*, 22:103–111, 1996.

[6] J.R. Dormand and P.J. Prince. A family of embedded runge-kutta formulae. *J. Comp. Appl. Math*, 6:19–26, 1980.

[7] J.R. Dormand. J.P. Gilmore. and P.J. Prince. Globally embedded runge-kutta schemes. *Annals of Numerical Mathematics*, 1:97–106, 1994.

[8] R.W. Brankin. I. Gladwell. and L.F. Shampine. Rksuite: a suite of runge-kutta codes for the initial value problem for odes. Softreport 92-s1, Department of Mathematics, Southern Methodist University, Dallas, Texas, U.S.A,, 1992.

[9] E. Hairer and C. Lubich. The life-span of backward error analysis for numerical integrators. *Numer. Math.*, 76(4):441–462, 1997.

[10] J.D. Lambert. *Numerical Methods for Ordinary Differential systems. The initial value problem*. John Willey and Sons, 1991.

[11] W.M. Lioen and J.J.B. de Swart. Test set for initial value problem solvers. release 2.1. http://www.cwi.nl/cwi/projects/IVPtestset, September 1999.

[12] E. Hairer. C. Lubich and G. Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Diferential Equations*, volume 31 of *Springer Series in Computational Mathematics*. Springer, 2002.

[13] J. Makazaga and A. Murua. New runge-kutta based schemes for odes with cheap global error estimation. *BIT Numerical Mathematics*, 43(55):595–610, 2003.

[14] R.H. Merson. An operational method for the study of integration processes. In *proceedings of a conference on Data processing and automatic computing machines*, pages 110–1 to 110–25. Weapons Research Establisshment, Salisbury, Australia, June 3rd-8th 1957.

[15] M. Calvo. D.J. Higham. J.L. Montijano and L. Rández. Stepsize selection for tolerance proportionality in explicit runge-kutta codes. *Advances in Computational Mathematics*, (7):361–382, 1997.

[16] A. Murua. *Métodos simplécticos desarrollables en P-series*. PhD thesis, Valladolid, 1994.

[17] A. Murua. Formal series and numerical integrators, part i: System of odes and symplectic integrators. *Applied Numerical Mathematics*, 29:221–251, 1999.

[18] A. Murua and J. Makazaga. Cheap one-step global error estimation for odes. *New Zealand journal of mathematics*, 29:211–221, 2000.

[19] Jitse Niesen. *On the Global Error of Discretization Methods for Ordinary Differential Equations*. PhD thesis, Unibersity of Cambridge, 2004.

[20] E. Hairer. S.P. Norset and G. Wanner. *Solving Ordinary Differential Equations I Nonstiff Problems*. Springer-Verlag, second revised edition, 1993.

[21] S. Reich. Backward error analysis for numerical integrators. *SIAM J. Numer. Anal.*, 36(5):1549–1570, 1999.

[22] Steven J. Ruuth. Global optimization of explicit strong-stability-preserving runge-kutta methods. *Mathematics of Computation*, 75(253):183–207, September 2005.

[23] Lawrence F. Shampine. *Numerical Solution of Ordinary Differential Equations*. Mathematics. Chapman & Hall, 1994.

[24] R.D. Skeel. Thirteen ways to estimate global error. *Numer. Math.*, 48:1–20, 1986.