

ACADEMIA DE CIENCIAS EXACTAS, FISICAS,  
QUIMICAS Y NATURALES DE ZARAGOZA

**LOS METODOS DE RUNGE-KUTTA  
EN LA RESOLUCION NUMERICA  
DE ECUACIONES DIFERENCIALES**

DISCURSO DE INGRESO LEIDO POR EL ACADEMICO ELECTO

**Ilmo. Sr. D. MANUEL CALVO PINILLA**

EN EL ACTO DE SU RECEPCION SOLEMNE  
CELEBRADO EL DIA 10 DE MARZO DE 1998.

Y

DISCURSO DE CONTESTACION POR EL

**Ilmo. Sr. D. MARIANO GASCA GONZALEZ**

ACADEMICO NUMERARIO



ZARAGOZA

1998

Depósito Legal: Z. 516 – 1998

*Imprime:*

Sdad. Coop. de Artes Gráficas  
LIBRERIA GENERAL  
Pedro Cerbuna, 23  
50009 Zaragoza

**LOS METODOS DE RUNGE-KUTTA  
EN LA RESOLUCION NUMERICA  
DE ECUACIONES DIFERENCIALES**

**POR EL**

**Ilmo. Sr. D. MANUEL CALVO PINILLA**

Excmo. Sr. Presidente  
Excmos. e Ilmos. Sres. Académicos  
Señoras y Señores

Quisiera comenzar mi intervención en este acto expresando mi más profundo agradecimiento a los miembros de esta Academia y de un modo especial a aquellos que promovieron mi elección, poniendo de manifiesto la gran satisfacción que he sentido al verme honrado con vuestra confianza.

Por otro lado y continuando con el capítulo de agradecimientos, quiero recordar en primer lugar a mis Profesores de Matemáticas en esta Facultad (algunos de los cuales pertenecen a esta Academia), cuyo ejemplo y dedicación fueron el estímulo más decisivo para mi vocación matemática. En particular, a mi maestro el Prof. D. Rafael Cid, el cual, al finalizar mis estudios de licenciatura, me inició en la investigación en el campo de la Mecánica Celeste dirigiendo mi tesis doctoral y guiando mis primeros trabajos en este tema y posteriormente durante toda mi vida profesional proporcionandome valiosos consejos y sugerencias.

Seguidamente quiero agradecer a mi amigo y compañero Mariano Gasca no sólo el que haya aceptado apadrinar mi ingreso en esta Academia, sino también su apoyo personal y profesional durante más de dos décadas. Acaso valga la pena recordar que Mariano fué el primer Catedrático de Análisis Numérico de nuestro país y que allá por los años setenta, durante su estancia en Bilbao, ya entré en contacto con él para recabar sus orientaciones en distintos problemas de Análisis Numérico y que desde entonces nuestra amistad y colaboración ha continuado ininterrumpidamente hasta este momento.

Asimismo he de reconocer que en mi quehacer investigador he sido una persona especialmente afortunada ya que durante toda mi carrera en las Universidades de La Laguna y Zaragoza he contado con unos colaboradores excepcionales: J.I. Montijano, S. González, T. Grande, L. Rández, y un largo etc. han sido y son el principal estímulo de mi actividad investigadora.

También quiero hacer patente un especial agradecimiento a mi entorno familiar por el constante apoyo y la comprensión que me han dispensado a lo largo de estos años, en particular a Maite, a Pilar y a Belen.

La admisión en la Academia de Ciencias de Zaragoza representa en mi caso un inmerecido honor y he de reconocer que la redacción de este discurso

ha sido debida principalmente a D. Nácere Hayek y a D. Horacio Marco, Presidentes de las Academias de Tenerife y Zaragoza respectivamente, que en los últimos meses han sido los catalizadores que me han animado a su conclusión definitiva.

Asimismo, me siento orgulloso de ocupar en esta Academia el lugar que ocupó D. Teodoro Ríos Balaguer, prestigioso arquitecto zaragozano que no solamente es conocido por las obras de consolidación que dirigió en la Basílica del Pilar, sino por el diseño y la restauración de muchos otros edificios en nuestra ciudad y fuera de ella. Sirvan de ejemplo los colegios del Sagrado Corazón, Escolapias y Teresianas de Zaragoza, así como la Residencia Universitaria de Jaca y las restauraciones de la Catedral de Tudela y la Iglesia de Santa María la Real de Sangüesa.

En cuanto al tema sobre el que voy a hablaros lo he elegido porque confluyen en él varias razones: Acaba de cumplirse un siglo de la introducción por parte de Runge [19] de los métodos de Runge-Kutta los cuales, debido a su sencillez y eficiencia, han jugado un papel fundamental en la resolución numérica de ecuaciones diferenciales. Puede decirse sin temor a equivocarse que en, cualquier curso elemental de Cálculo Numérico que incluya ecuaciones diferenciales, aparece inevitablemente un célebre método de cuarto orden debido a Kutta, que se suele llamar "el método de Runge-Kutta". Por otro lado, a pesar de ser un tema clásico en el que desde hace años todo parecía estar descubierto, las investigaciones han continuado y continúan en la actualidad no sólo con el descubrimiento de nuevas fórmulas con mejores propiedades de aproximación y estabilidad que permiten mejorar el software existente, sino también con el diseño de métodos Runge-Kutta especiales, adaptados a tipos particulares de problemas diferenciales, como pueden ser los asociados a flujos ortogonales, hamiltonianos etc. Asimismo me proporciona la oportunidad de referirme a algunas investigaciones llevadas a cabo junto a algunos de mis colaboradores en los últimos años.

Finalmente, unos breves comentarios sobre la presentación del tema. Cualquier problema de matemática aplicada, y en particular la resolución numérica de sistemas diferenciales, es susceptible de distintos enfoques según el punto de vista que se adopte. Así, para los científicos e ingenieros un integrador numérico es una herramienta que le ayuda a analizar e interpretar el comportamiento de sus problemas. En consecuencia no suelen dedicar su tiempo a analizar los detalles matemáticos del integrador y lo consideran

mas bien como una receta que hay que usar adecuadamente, ayudándose de ciertas dosis de intuición numérica. Otro punto de vista diferente es el de las personas relacionadas con las ciencias de la computación. Para ellos los aspectos algorítmicos o los relacionados con la arquitectura de los ordenadores son los más importantes. Por último, para los matemáticos, los integradores de sistemas diferenciales constituyen un elemento más dentro de su mundo matemático, cuyas propiedades hay que estudiar usando no sólo las herramientas teóricas adecuadas de Análisis, Algebra, Geometría, etc., sino también la experimentación numérica. En mi caso, he de confesar que por mi formación y dedicación durante los últimos años mi punto de vista está próximo a esta última línea. Sin embargo, puesto que buena parte de la audiencia no es exclusivamente matemática, he renunciado a dar una presentación rigurosa y he preferido hacer uso de la intuición siempre que ha sido posible.

## 1 Breves comentarios históricos sobre las ecuaciones diferenciales

Como es bien conocido, la aparición de las ecuaciones diferenciales está históricamente ligada a la del cálculo diferencial y suele asociarse a los nombres de Newton y Leibniz. Recordemos al respecto que Newton (1642–1727) en su tratado sobre el cálculo diferencial de 1671<sup>1</sup>, ya considera varios ejemplos de ecuaciones de primer orden. En particular, en (*Problema II, Solutio Casus II, Ex. 1*) considera el problema de valor inicial de primer orden consistente en calcular una función  $y(x)$  tal que

$$y' = 1 - 3x + y + x^2 + xy, \quad y(0) = 0,$$

y lo resuelve por desarrollo en serie usando un procedimiento iterativo, es decir dada una aproximación de Taylor  $y_a(x)$  a la solución buscada, sustituye ésta en el segundo miembro de la ecuación diferencial e integra el polinomio resultante para obtener una nueva aproximación mejorada de la anterior. De esta manera, tras aplicar dos veces el proceso partiendo de  $y_a(x) = 0$ , llega

---

<sup>1</sup>Methodus Fluxionum et Serierum Infinitarum, edit. Londini, 1736

a obtener la siguiente aproximación de orden seis

$$y = x - \frac{1}{3}x^3 + \frac{1}{6}x^4 - \frac{1}{30}x^5 + \frac{1}{45}x^6, \&c$$

Sin embargo, los estudios más relevantes de Newton en relación con las ecuaciones diferenciales se encuentran en sus "*Principia Mathematica*", publicados en 1687 y que fueron elaborados, al parecer, una veintena de años antes. En estos estudios, Newton introdujo el modelo gravitatorio basado en la fuerza inversamente proporcional al cuadrado de la distancia para describir el movimiento de los planetas del sistema solar, y por tanto planteó las ecuaciones diferenciales que han servido de base para todos los estudios posteriores de Mecánica Celeste.

Por lo que respecta a Leibniz (1646–1716), su relación con las ecuaciones diferenciales y en general con el cálculo diferencial suele datarse unos años más tarde que en el caso de Newton. Más concretamente, tras su estancia en París (1672–1676), entró en contacto con las ideas del pre-cálculo de Cavalieri, Pascal, Fermat, etc. y a raíz de esa época se produjeron sus contribuciones más decisivas en el ámbito matemático. Así, hacia 1676, ya envió una carta a Huygens<sup>2</sup> en la que aparecen las ecuaciones diferenciales obtenidas al resolver el problema geométrico de las tangentes inversas.

Como es bien sabido la controversia dialéctica entre Newton y Leibniz fué muy intensa en su época, ya que el primero acusó reiteradamente al segundo de plagiar sus descubrimientos, aunque en realidad parece que Leibniz desarrolló independientemente sus ideas, si bien en ciertos aspectos del cálculo diferencial las ideas de ambos tuvieron puntos en común.

En mi apreciación personal, la figura de Leibniz ha quedado a menudo oscurecida por la brillantez de la de Newton, pero creo que la originalidad de sus soluciones y la visión futurista dada por Leibniz a muchos problemas justificarían un mejor conocimiento de su obra, particularmente en el ámbito de la matemática constructivista, y por ello voy a permirme dedicarle una breve referencia.

Gottfried Wilhelm Leibniz nació en Leipzig en 1646 en el seno de una familia académica y ya desde niño fué un ávido devorador de libros de las

---

<sup>2</sup>*Methodus, qua innumerarum linearum construction ex data . . . Letter to Huygens, in: C.I. Gerhardt, Leibnizens math. Schriften, 1850, Band II, pp. 116-121*

mas variadas disciplinas. Todos sus biógrafos reconocen su gran memoria y su capacidad de trabajo. Estuvo en la Universidad de Leipzig entre los 14 y 22 años, presentando a dicha edad su doctorado en Jurisprudencia titulado “*Sobre casos difíciles en Leyes*” en el cual proponía definir todos los conceptos legales en función de un mínimo número de conceptos básicos y deducir asimismo las leyes de estos. Poco tiempo después, presentó una disertación en Filosofía titulada “*El arte de las Combinaciones*”, en la que presenta una especie de lenguaje universal que puede considerarse en cierto modo como precursor del álgebra de Boole, ya que asocia a cada concepto primitivo un número primo y posteriormente los conceptos compuestos irán asociados a la factorización correspondiente (p.e. si hombre  $\leftrightarrow$  3 y racional  $\leftrightarrow$  7, hombre racional  $\leftrightarrow$  21), no obstante resultaba imposible trasladar a este lenguaje todas las reglas usuales del conocimiento.

Además de las contribuciones anteriores, Leibniz fué autor de cartas y publicaciones de la temática más diversa, acaso valga la pena recordar que incluso se conserva una carta en la que hace una propuesta dirigida a católicos y protestantes para que armonizando sus puntos de vista se creasen los “Estados Unidos de Europa”. Sin embargo, es en el ámbito matemático donde sus ideas han tenido mayor relevancia, así los analistas lo recuerdan por ser el introductor en el calculo diferencial de notaciones que han llegado hasta nuestros días (“*Utile erit scribit f pro omnia*” 1675) y los analistas numéricos por su visión constructivista que le llevó a diseñar máquinas de cálculo, y a vaticinar que el sistema binario era el más adecuado para el cálculo aritmético, proponiendo un convertidor de sistema decimal a binario. Estas consideraciones nos muestran que las visiones futuristas de Leibniz han resultado plenamente confirmadas en la actualidad.

Enfín, tras tres siglos de andadura, las ecuaciones diferenciales han resultado ser por un lado una fuente inagotable de problemas en el campo de la matemática pura y por otro una herramienta indispensable para modelizar los mas variados fenómenos de la ciencia y la técnica. Recordemos al respecto que, la mayor parte de los problemas físicos se describen mediante ciertas variables de estado, que dependen (o pueden depender) de las coordenadas espaciales y del tiempo y la modelización del problema consiste en proponer unas reglas de variación de las variables de estado mediante un conjunto de ecuaciones diferenciales. Esta situación implica que las ecuaciones diferenciales hayan resultado ser un punto de encuentro de casi todas las ramas de



la matemática pura con las ciencias aplicadas, de tal manera que, ecuaciones particulares que han aparecido en el estudio de algunos modelos prácticos, han dado la pauta para el desarrollo de teorías con los mas sofisticados recursos matemáticos.

Sería inacabable hacer aquí una enumeración somera de importantes matemáticos cuyas contribuciones han estado ligadas al desarrollo de las ecuaciones diferenciales y por tanto renunciaremos a dicho empeño. Sin embargo insistiremos en que en los últimos años muchos problemas planteados en el ámbito de ecuaciones diferenciales siguen siendo fuente de inspiración de nuevas teorías matemáticas. Podemos recordar al respecto el famoso modelo de Lorenz [16], propuesto por este autor en 1979 a raíz de sus estudios numéricos de predicción meteorológica que, a pesar de su aparente simplicidad, llevó a la introducción de los llamados atractores extraños, posteriormente relacionados con las bifurcaciones y las cascadas de Feigenbaum.

Los estudios de ecuaciones diferenciales, debido a su amplitud y a su variedad temática, se suelen clasificar en ecuaciones diferenciales ordinarias (EDOs) y ecuaciones en derivadas parciales (EDPs). Mientras que en las primeras las funciones incógnitas dependen de una sola variable independiente, en las segundas depende de varias variables. Las ecuaciones de Maxwell, Navier–Stokes, etc., son ejemplos de estas últimas. Si bien hay aspectos comunes a la resolución numérica de ambos tipos de ecuaciones, en general la metodología es diferente y nosotros nos referiremos en lo sucesivo a las EDOs.

## 2 Los métodos Runge–Kutta explícitos

Como ya se ha comentado anteriormente, en el estudio de las ecuaciones diferenciales, la herramienta numérica ha jugado siempre un papel importante, debido a que la mayor parte de las ecuaciones que aparecen en los problemas prácticos no se pueden resolver exactamente y por tanto hay que recurrir a algún tipo de aproximación para tener idea del comportamiento de sus soluciones. Sin embargo, durante el siglo pasado y buena parte de éste, la impotencia de los medios de cálculo fué una barrera decisiva para la utilización de muchos métodos numéricos. Esta situación empieza a cambiar a partir de los años cuarenta–cincuenta con la aparición de los ordenadores

electrónicos y se ha acelerado especialmente durante los últimos veinte años con la disponibilidad de ordenadores personales que poseen una elevada velocidad de cálculo y una gran capacidad de almacenamiento de datos. Estas posibilidades han tenido un doble efecto en la matemática numérica: por una parte ha sido posible abordar la resolución efectiva de problemas que debido a su complejidad habían quedado anteriormente fuera de toda consideración y por otra parte, ha sido necesario desarrollar el software necesario para hacer frente a estos problemas con la máxima eficiencia posible.

Los códigos usuales para la resolución numérica de problemas de valor inicial en EDOs presuponen que el sistema a integrar está escrito en la forma normal, es decir, como sistema explícito de  $m$  ecuaciones diferenciales de primer orden

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \mathbf{y}(t_0) = \mathbf{y}_0 \in \mathbb{R}^m, \quad (1)$$

y calculan sucesivamente aproximaciones a la solución exacta  $\mathbf{y} = \mathbf{y}(t)$ , en un conjunto de puntos  $t_0 < t_1 < \dots$  del intervalo de integración  $\mathbf{y}_0, \mathbf{y}_1, \mathbf{y}_2, \dots$

La resolución numérica procede inicialmente en el modo paso a paso, es decir, supuesto que conocemos una aproximación  $\mathbf{y}_n$  a la solución en el punto  $t_n$ , el método numérico nos proporciona una nueva aproximación  $\mathbf{y}_{n+1}$  en el punto  $t_{n+1}$ . Más adelante veremos que también se puede aproximar la solución entre pares de puntos consecutivos.

En este contexto los métodos se suelen clasificar en dos grandes familias: los llamados métodos de un paso y los métodos multipaso. En los primeros se usa exclusivamente la información de la solución en un punto para obtener una aproximación de la solución en el punto siguiente. En los segundos se emplea la información calculada en varios puntos previos para avanzar la integración al punto siguiente. Hay que notar que desde el siglo pasado ha existido una dialéctica permanente entre las ventajas e inconvenientes de los métodos multipaso frente a los métodos de un paso. Por un lado se argumenta que en los métodos multipaso el emplear la información disponible de varios puntos previos -sin costo adicional- para mejorar la aproximación futura les proporciona una clara ventaja respecto a los de un paso. Por otro lado se afirma que los métodos de un paso reflejan más fielmente el carácter instantáneo del flujo de un sistema diferencial de primer orden, ya que el comportamiento depende exclusivamente del estado inicial. Además los métodos de un paso son más fácilmente implementables que los multipaso.

Los métodos lineales multipaso tienen una larga tradición en la resolución numérica de ODEs y están ligados históricamente a los nombres de Adams y Bashforth (1883) y Moulton (1926), que propusieron varias familias de métodos que llevan su nombre y se siguen empleando en códigos actuales. Asimismo, tras las importantes contribuciones de Dahlquist a partir de los años sesenta [9] recopiladas posteriormente en los clásicos libros de Henrici [15] y Grigorieff [12], existe una completa teoría que analiza el comportamiento de dichos métodos con paso fijo e incluso bastantes aspectos del paso variable. Hacia los años ochenta, en colaboración con varios colegas de Departamento se desarrollaron investigaciones sobre estos métodos, pero no pretendemos aquí comentarlas ya que nos llevarían fuera del objetivo de esta conferencia.

En el marco de los métodos de un paso, el más sencillo es el llamado de Euler explícito o de las tangentes descrito por éste autor en 1768,<sup>3</sup> en el cual la solución numérica avanza de  $(t_n, \mathbf{y}_n) \rightarrow (t_{n+1} = t_n + h, \mathbf{y}_{n+1})$  mediante la fórmula:  $\mathbf{y}_{n+1} = \mathbf{y}_n + hf(t_n, \mathbf{y}_n)$ . Desde un punto de vista geométrico, el método avanza la solución numérica a lo largo de la tangente al flujo en el punto considerado. Esta fórmula es bien simple de programar y solo requiere una evaluación de la función  $\mathbf{f}(t, \mathbf{y})$  en cada paso, pero su principal inconveniente reside en que para lograr buenas aproximaciones es necesario tomar pasos  $h$  muy pequeños y por tanto la integración avanza muy lentamente. En efecto, los errores locales, es decir, los introducidos en cada paso, son del orden de  $h^2$  por lo que el método es de primer orden.

Una idea que podría utilizarse para mejorar la aproximación sin perder el carácter unipaso podría ser usar la bien conocida fórmula de Taylor del cálculo infinitesimal, en la cual se aproxima una función alrededor de un punto usando los valores de la función y de sus derivadas en el punto considerado. En particular, si se cumplen ciertas condiciones suficientes de derivabilidad para la solución de (1), podemos aproximar la solución de la ecuación diferencial en  $t_{n+1} = t_n + h$ , a partir de los valores iniciales  $\mathbf{y}(t_n) = \mathbf{y}_n$ . Para ello necesitamos evaluar las derivadas  $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$ ,  $\mathbf{y}'' = \mathbf{f}_t + \mathbf{f}_y \mathbf{f}$  etc., en el punto  $(t_n, \mathbf{y}_n)$ . Sin embargo, salvo en algunos problemas muy concretos en los que es posible calcular fácilmente las derivadas sucesivas de  $\mathbf{y}' = \mathbf{f}$ , hay que entrar en manipulaciones de la ecuación diferencial que son particularmente

<sup>3</sup>*Institutiones Calculi Integralis, Caput VII, pp. 424*

engorrosas en el caso de sistemas diferenciales y la realidad es que el usuario típico prefiere evitar complicaciones y lo que desea es dar exclusivamente la ecuación diferencial + condiciones iniciales y recibir de una manera clara la información numérica o gráfica que le interese sobre la solución. Por tanto los métodos basados en la fórmula de Taylor no han tenido hasta el momento -salvo en algunos problemas concretos- demasiada difusión, aunque las posibilidades de los actuales manipuladores algebraicos les abren nuevas perspectivas futuras al menos para tipos particulares de sistemas diferenciales.

Otra alternativa para obtener métodos de un paso con orden mayor que uno, propuesta hace aproximadamente un siglo por Runge [19], consiste en utilizar reevaluaciones sucesivas de la función  $\mathbf{f}(t, \mathbf{y})$  en puntos adecuados, por ejemplo, éste autor observó que si formamos sucesivamente las evaluaciones de función:

$$\mathbf{g}_1 = \mathbf{f}(t_n, \mathbf{y}_n), \quad \mathbf{g}_2 = \mathbf{f}(t_n + h, \mathbf{y}_n + (h/2) \mathbf{g}_1),$$

la fórmula  $\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{g}_2$ , ya tiene orden dos, es decir que los errores locales son proporcionales a  $h^3$ . Por tanto, usando la idea de Runge, con dos evaluaciones de función por paso, se pueden deducir fórmulas de segundo orden.

Otras fórmulas de mayor orden fueron propuestas también por Runge en su trabajo de 1895 y por Heun y Kutta en trabajos posteriores. En particular está la bien conocida

$$\mathbf{y}_{n+1} = \mathbf{y}_n + (h/6) (\mathbf{g}_1 + 2\mathbf{g}_2 + 2\mathbf{g}_3 + \mathbf{g}_4),$$

donde

$$\begin{aligned} \mathbf{g}_1 &= \mathbf{f}(t_n, \mathbf{y}_n), \\ \mathbf{g}_2 &= \mathbf{f}(t_n + h/2, \mathbf{y}_n + (h/2) (\mathbf{g}_1)), \\ \mathbf{g}_3 &= \mathbf{f}(t_n + h/2, \mathbf{y}_n + (h/2) (\mathbf{g}_2)), \\ \mathbf{g}_4 &= \mathbf{f}(t_n + h, \mathbf{y}_n + h \mathbf{g}_3). \end{aligned}$$

que es de cuarto orden y que se conoce a menudo como “la fórmula de Runge-Kutta”. De ella ya dijo Runge en 1905: “Von den neueren Verfahren halte ich das folgende von Herrn Kutta angegebene für das beste” y parece que tenía razón, ya que por su simplicidad, su facilidad de programación y los satisfactorios resultados numéricos que proporciona, ha sido y es ampliamente

usada estando incluida en la actualidad en muchos manipuladores algebraicos ( MAPLE, MATLAB, Mathematica, etc ).

Desde un punto de vista algo más general un método de Runge-Kutta explícito es un método de un paso que avanza la solución desde  $(t_n, \mathbf{y}_n)$  a  $(t_{n+1} = t_n + h, \mathbf{y}_{n+1})$ , usando una fórmula del tipo

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h[b_1 \mathbf{g}_1 + \dots + b_s \mathbf{g}_s], \quad (2)$$

donde las llamadas etapas  $\mathbf{g}_1, \dots, \mathbf{g}_s$  se calculan sucesivamente desde las ecuaciones

$$\begin{aligned} \mathbf{g}_1 &= \mathbf{f}(t_n, \mathbf{y}_n), \\ \mathbf{g}_2 &= \mathbf{f}(t_n + c_2 h, \mathbf{y}_n + h a_{21} \mathbf{g}_1), \\ \mathbf{g}_3 &= \mathbf{f}(t_n + c_3 h, \mathbf{y}_n + h a_{31} \mathbf{g}_1 + h a_{32} \mathbf{g}_2), \\ &\vdots \\ \mathbf{g}_s &= \mathbf{f}(t_n + c_s h, \mathbf{y}_n + h a_{s1} \mathbf{g}_1 + \dots + h a_{s,s-1} \mathbf{g}_{s-1}). \end{aligned} \quad (3)$$

El número  $s$  de evaluaciones de función en el algoritmo anterior se llama “número de etapas” y frecuentemente es considerado como una medida del costo computacional de la fórmula considerada. Los parámetros reales  $b_i$  y  $a_{ij}$  se llaman respectivamente pesos externos y pesos internos de la fórmula y los  $c_i = a_{i1} + \dots + a_{i,i-1}$ .

Tras la introducción de los primeros métodos RK hacia finales del siglo pasado por Runge y las posteriores contribuciones de Kutta y Heun a comienzos del siglo actual, el desarrollo de nuevos métodos queda prácticamente paralizado. Hay que notar que, con los medios de cálculo existentes en la primera mitad del siglo, las fórmulas de cuarto orden implementadas con paso fijo eran la base de los métodos de máxima precisión. Sin embargo a medida que los científicos e ingenieros van disponiendo de ordenadores electrónicos se lanzan a realizar experimentos numéricos más complejos y se observa que es necesario disponer de integradores más potentes que los existentes. Mostraremos esta situación con un ejemplo sencillo: Se trata del modelo que describe el movimiento de un vehículo espacial en el campo gravitatorio creado por la Tierra y la Luna, que en Mecánica Celeste se llama problema restringido de los tres cuerpos. Para simplificar vamos a suponer que la Luna describe una órbita circular plana alrededor de la Tierra y que el vehículo se mueve en el plano Tierra-Luna. Tomando un sistema de coordenadas cartesianas con origen en el c.d.m. de la Tierra y la Luna, de tal

manera que la Tierra aparezca fija en el punto  $(-\mu, 0)$  y la Luna en el punto  $(1 - \mu = \mu', 0)$  con  $\mu = 0.012277$ , las coordenadas  $(x, y)$  del vehículo espacial verifican las ecuaciones

$$x'' = x + 2y' - \mu' \frac{(x + \mu)}{\Delta_1^3} - \mu \frac{(x - \mu')}{\Delta_2^3}, \quad y'' = y - 2x' - \mu' \frac{y}{\Delta_1^3} - \mu \frac{y}{\Delta_2^3},$$

donde:  $\Delta_1 = \sqrt{(x + \mu)^2 + y^2}$  y  $\Delta_2 = \sqrt{(x - \mu')^2 + y^2}$  representan las distancias del vehículo espacial a la Tierra y Luna respectivamente.

Es bien conocido desde Poincaré que este sistema posee diferentes familias de órbitas periódicas, pero salvo casos excepcionales no se pueden expresar explícitamente mediante funciones elementales. En particular para las condiciones iniciales

$$x(0) = 0.094, \quad x'(0) = 0, \quad y(0) = 0, \quad y'(0) = -2.00115851063,$$

tenemos una órbita (de la llamada familia de Arenstorf [1] ) de periodo  $T = 17.0652165601 \dots$ . En la Fig.1 se muestran las gráficas de las trayectorias obtenidas para diferentes métodos: El método de Euler (con paso fijo), el clásico método RK de cuarto orden (también con paso fijo) y el metodo DOPRI54 [10] de orden cinco que selecciona automáticamente el paso.

Método:	Euler (paso fijo)	RK4 (paso fijo)	DOPRI (paso var)
Número de Pasos	24.000	6.000	75

Puede observarse que el método de Euler desde el inicio de la integración, que está próximo a una singularidad del sistema diferencial, se pierde completamente y los resultados numéricos son totalmente equivocados, el método RK sigue más estrechamente a la solución exacta, pero los errores son también inaceptables y finalmente el método DOPRI54 que es de orden cinco llega al final del periodo con errores del orden de la milésima.

Este sencillo experimento nos ilustra acerca de las dificultades que podemos encontrar en la integración numérica de ecuaciones diferenciales cuando se usan fórmulas con paso fijo y si el orden de la fórmula es bajo: Si estamos recorriendo una parte del intervalo de integración en el cual la solución varía muy rápidamente (es decir hay derivadas grandes) convendrá tomar pasos

pequeños para que los errores cometidos en cada paso no nos invaliden los resultados que se obtengan en el resto de la integración. Por el contrario, si estamos recorriendo un subintervalo de variación lenta de la solución, será posible avanzar con pasos más grandes, ya que los errores introducidos serán menores que el error máximo admisible. Un ejemplo típico de esta situación se presenta en la integración del bien conocido problema de los dos cuerpos con excentricidades próximas a la unidad, es decir, con órbitas elípticas muy excéntricas próximas a parabólicas. De acuerdo con la ley de las áreas en las proximidades del perigeo, la velocidad es muy grande y por tanto los pasos deben ser pequeños y todo lo contrario ocurre en el apogeo.

### 3 Métodos Runge-Kutta adaptativos

Hacia el comienzo de los años sesenta se fué haciendo patente que un integrador numérico de uso general no podía consistir exclusivamente en una fórmula (incluso de alta precisión) que se aplica reiteradamente con un paso fijo  $h$  y nos va calculando sucesivamente aproximaciones a la solución de la ecuación diferencial dada en los puntos de una red uniforme del intervalo de integración  $t_0, t_0 + h, t_0 + 2h, \dots$ . Experimentos numéricos como el que se acaba de mostrar nos indican que para obtener resultados fiables es necesario, a veces, tomar pasos de integración muy pequeños y esto conlleva dos dificultades: por una parte el proceso es muy lento y por otra los errores de redondeo –inevitables en todo cálculo numérico– pueden contaminar el resultado de tal forma que éste resulte totalmente inválido. En resumen, un integrador numérico que avanza la integración con “velocidad” constante puede no ser fiable para algunos problemas y sería más lógico que la velocidad de avance de la solución se fuera adaptando automáticamente a la naturaleza de ésta, de tal manera que si en una parte del intervalo de integración hay alguna componente que varía rápidamente los pasos sean pequeños y por el contrario si la solución varía lentamente los pasos sean largos. De esta manera surgieron los llamados métodos adaptativos en los que, además de una fórmula de avance de la solución, se debe incluir un algoritmo de selección del paso. Respecto a este último aspecto una posibilidad evidente es dejar dicha selección en manos del usuario para que la lleve a cabo teniendo en cuenta su conocimiento de la solución, sin embargo en la mayor

parte de los casos el usuario no tiene la información necesaria para ello y además en general prefiere no ocuparse de este aspecto que le supone tener que intervenir en el programa de ordenador, por tanto de acuerdo con los requerimientos prácticos, la investigación se orientó a buscar mecanismos que seleccionen automáticamente el paso y que sean asimismo fiables. Diversos estudios teóricos muestran que la selección automática del paso se puede basar en diferentes magnitudes, de todas ellas las más usadas están basados en el error local y en el defecto de la solución numérica. En el contexto de los métodos numéricos el error local en un punto  $(t_n, \mathbf{y}_n)$  es el error cometido al avanzar un paso desde dicho punto y obviamente dicho error depende de la longitud del paso, por tanto se trata de ajustar dicha longitud, de manera que el error se mantenga inferior a una tolerancia de error prefijada por el usuario. A la vista de esto se trata de idear procedimientos que proporcionen una buena estimación del error local y que no requieran un excesivo coste computacional.

Los primeros estimadores del error local para métodos adaptativos se basaron en la vieja idea de extrapolación propuesta por Richardson [18] en 1927. Puesto que el error local  $\mathbf{EL} = \mathbf{EL}(h; t_n, \mathbf{y}_n)$  al integrar con un método de orden  $p$  y paso  $h$  desde el punto  $(t_n, \mathbf{y}_n)$ , es asintóticamente equivalente a  $\mathbf{C}(t_n, \mathbf{y}_n)h^{p+1}$  con una cierta función  $\mathbf{C}$ , es decir,  $\mathbf{EL}(h; t_n, \mathbf{y}_n) = \mathbf{C}(t_n, \mathbf{y}_n)h^{p+1} + \mathcal{O}(h^{p+2})$ , se pueden dar dos pasos consecutivos de tamaño  $h$  y simultáneamente uno de paso  $2h$  y, comparando los dos resultados tras unas sencillas operaciones, estimar asintóticamente el error local para dicho paso  $h$ ,  $\mathbf{EL}(h)$ . Este procedimiento es conceptualmente sencillo y además la experiencia le confiere un alto grado de fiabilidad, por lo cual fue muy usado en los primeros métodos adaptativos, en conjunción con varias fórmulas conocidas de órdenes cuatro y cinco. El principal inconveniente del estimador de extrapolación radica en su elevado coste computacional, ya que cada dos pasos consecutivos hay que dar un paso adicional doble que es el que realmente permite estimar el error local y verificar si éste se mantiene por debajo de la tolerancia prefijada, lo que supone en principio un cincuenta por ciento de coste adicional respecto a una integración a paso fijo. Además, si el error local es menor que la tolerancia el paso se acepta y la integración continuará con un nuevo paso, que dependerá del tamaño del error local frente a la tolerancia, pero si el error local es mayor que la tolerancia hay que reajustar el paso y volver a repetir el proceso con un nuevo paso más pequeño, por tanto



los pasos fallados suponen un notable coste extra. En los primeros códigos, por razones de sencillez, los aumentos o disminuciones de paso se reducían exclusivamente a multiplicar o dividir por dos el paso previamente usado ya que en tal caso para la reducción se puede usar una parte de la información del paso fallado, pero luego de haber visto que de esta manera se limita notablemente la flexibilidad del código y se han propuesto variaciones más amplias compatibles con la seguridad del código.

Debido al alto coste computacional del procedimiento de extrapolación se abrió paso a una nueva idea para dotar a los métodos RK de una estimación del error local. La idea, que consiste en usar lo que se llama un par encajado de fórmulas RK, es conceptualmente muy simple, ya que se trata de usar las mismas evaluaciones de función empleadas en un paso  $\mathbf{g}_i, i = 1, 2, \dots, s$  (3) para calcular dos aproximaciones a la solución  $\hat{\mathbf{y}}_{n+1}, \mathbf{y}_{n+1}$  de distintos órdenes (generalmente consecutivos) pongamos  $p$  y  $p + 1$ ; en tal caso la diferencia entre estas dos aproximaciones  $\hat{\mathbf{y}}_{n+1} - \mathbf{y}_{n+1}$ , es una estimación asintóticamente correcta del error local de la fórmula de orden más bajo. Así pues, las mismas evaluaciones de función  $\mathbf{g}_i$  (que suponen la mayor parte del costo computacional) se usan simultáneamente para obtener la solución numérica y estimar su error. Basándose en esta idea, varios autores como Merson (1957), Ceschino (1961), Zonneveld (1963), propusieron métodos de cuarto orden con ciertos estimadores del error local, que han sido utilizados en distintos programas de uso general y se han empleado durante más de dos décadas. Sin embargo, la obtención directa de pares de fórmulas de mayor orden presentaba tales dificultades que la hacían prácticamente inabordable.

El panorama empezó a cambiar tras la aparición de un memorable trabajo de Butcher [3] en 1963, que ha supuesto no solamente la mayor contribución al estudio teórico de los métodos RK, sino que también ha tenido, como veremos más adelante, importantes implicaciones prácticas. En dicho trabajo se introducen unas series formales, posteriormente llamadas por Hairer y Wanner [13] B-series, cuyos términos están en correspondencia biunívoca con los árboles de raíz de la teoría de grafos y por tanto pueden generarse recurrentemente, de tal manera que, las soluciones analítica y numérica, pueden expresarse en forma de B-series, lo que permite analizar su orden de convergencia y el error local. Del estudio de Butcher [3] se deducen las ecuaciones independientes que deben satisfacer los coeficientes de una fórmula, para que tenga un orden dado. Estas ecuaciones son no lineales y su número para un

orden dado  $p$  es el que se recoge en la siguiente tabla

Orden	1	2	3	4	5	6	7	8	9	10
Número Ecuaciones	1	2	4	8	17	37	85	200	486	1205

Tabla 1: Número de ecuaciones de condición para alcanzar un orden dado

Como puede apreciarse el número de condiciones crece muy rápidamente con el orden, y así para obtener un par de fórmulas de órdenes 4–5, es necesario manejar  $8+17=25$  condiciones y elegir los parámetros disponibles de manera que se satisfagan las ecuaciones de condición. Además, como tales ecuaciones son no lineales y resultan bastante complicadas, podemos hacernos una idea de las dificultades que se presentan para obtener pares de ordenes medio–altos. No obstante, introduciendo algunas relaciones entre los coeficientes (llamadas hipótesis simplificadoras), es posible reducir el número de condiciones y hacerlas más manejables y así se han propuesto distintos pares, entre los cuales los más conocidos son los de Fehlberg [11], varias familias de Verner [23], Dormand & Prince [10], [17] y Calvo, Montijano & Rández [5]. Cronológicamente, los primeros pares fueron debidos a Fehlberg y entre ellos cabe citar un par 4–5 que, popularizado por Shampine [21], [22], es bien conocido por los usuarios de muchas librerías como integrador para precisiones medias. Asimismo hay que mencionar que un par de Fehlberg de ordenes 7–8 es bien conocido (particularmente entre los astrónomos) como integrador de alta precisión. Por otra parte Verner es autor de varias familias de fórmulas de distintos ordenes, que han ido acumulando sucesivamente mejoras a las fórmulas de Fehlberg. Las fórmulas de Dormand & Prince [10], [17] se encuentran entre las más eficientes de la actualidad y buena prueba de ello es que sirven como punto de comparación obligado a la hora cualquier nueva fórmula que se proponga. En particular, el par propuesto por nosotros [5] en 1990, es una mejora del par 5–4 de Dormand & Prince y en la actualidad está en fase de prueba para su incorporación en la librería NAG.

Como se ha comentado anteriormente la teoría de Butcher no solo fija las condiciones para que una fórmula (o par de fórmulas) alcancen un orden determinado, sino que nos proporciona expresiones del error local en forma de series asintóticas en potencias del paso de integración. Así, para una fórmula de orden  $p$  el error local se puede escribir en la forma

$$\text{EL}(h) = h^{p+1} \sum_j \dot{C}_j D_{p+1,j} + \mathcal{O}(h^{p+2}) \quad (4)$$

donde los coeficientes  $C_j$  dependen solo del método numérico y los  $D_{p+1,j}$  de la ecuación diferencial-considerada y análogamente para los términos de orden  $p+2$  y siguientes. De acuerdo con esto, no sólo interesa que una fórmula posea el mayor orden posible, sino también que los coeficientes del error de los términos de orden  $p+1$  (y si es posible también  $p+2$ ) sean lo más pequeños posible. En tal caso es claro que la minimización debe trasladarse exclusivamente a los coeficientes  $C_j$ , que dependen exclusivamente del método numérico y no de la ecuación diferencial. Los métodos de Fehlberg, Verner, Dormand & Prince y los nuestros, todos ellos fueron diseñados teniendo en cuenta de una u otra forma, no sólo el orden, sino también sus desarrollos asintóticos del error local de la fórmula de avance y del estimador. Naturalmente este factor añadido (y otros que no vamos a comentar aquí) suponen una complicación adicional para la obtención de fórmulas adaptativas de calidad, pero se ha visto que hay que tenerlas en cuenta para progresar en la mejora de la calidad de los métodos.

## 4 Los códigos Runge-Kutta actuales

Cualquiera que haya pasado por la experiencia de elaborar personalmente en su ordenador un programa para resolver algún problema de cálculo numérico, sabe muy bien que, aunque el método usado sea el más idóneo para la resolución de su problema, una implementación deficiente puede arruinar todas las buenas propiedades del método numérico. Esta norma general no es una excepción en el ámbito de la resolución numérica de ecuaciones diferenciales y por tanto la producción de software RK de calidad no se puede considerar como una mera transcripción de un método adaptativo, como los mencionados anteriormente, al lenguaje de máquina.

Dentro de la gran variedad de códigos RK que podemos encontrarnos en la actualidad, la mayor parte de ellos son realmente códigos de uso particular y solamente unos pocos pueden considerarse como códigos de uso general. Entendemos por códigos de uso particular aquellos que han sido producidos por personas o grupos de trabajo para resolver ciertos problemas diferenciales que se presentan en su ámbito de trabajo. Naturalmente tales códigos no suelen estar muy documentados, ya que con frecuencia son usados por la misma persona que los ha producido y, en cualquier caso, su utilización se suele lim-

itar a un círculo restringido de usuarios con propósitos comunes. Además, en estos códigos solamente se calculan aquellas magnitudes de la solución que interesan a sus usuarios. Por el contrario los códigos de uso general, puesto que son de dominio público, como mínimo deben estar bien documentados, para facilitar su uso y asimismo, deben incluir el mayor número de opciones para dar respuesta a la diversidad de problemas de los usuarios.

Aunque existen excelentes códigos de uso particular elaborados por investigadores, dentro de sus ámbitos de aplicación, en mi opinión éstos no son demasiado frecuentes, ya que, a menudo, son producidos por no especialistas y por tanto no recogen las avances de la investigación existente en el tema. Así, es posible encontrar aún códigos basados en una fórmula RK (típicamente la de cuarto orden) actuando a paso fijo y sin ningún control de error y estabilidad, con la pretensión de que pueda aplicarse a cualquier problema. Esto es debido a que resulta muy sencillo escribir un programa de este tipo, pero hay que tener en cuenta que los “números” que se obtengan para ciertos problemas pueden tener escasa fiabilidad. De acuerdo con estas consideraciones, pensamos que la mayoría de los códigos de uso particular no dan, en general, una visión actualizada de los códigos RK y por eso nos centraremos en códigos de uso general que están producidos por especialistas con amplia experiencia numérica y que podemos encontrar en librerías de uso general.

Existen varias colecciones de códigos RK de uso general para la resolución numérica de problemas de valor inicial (no stiff) en EDOs ( una lista de “netlib” está disponible en **mail netlib @ornl.gob** tecleando **send index**). Entre ellas podemos mencionar las siguientes: RKSUITE debida a Brankin, Gladwell y Shampine [2] y que en sus versiones de F77 y F90 forma parte de la librería NAG. En la librería IMSL tenemos el código DVERK producido por Jackson, Hull y Enright y que utiliza un par 5–6 debido a Verner [23]. En MAPLE (versión 5.4) hay varias subrutinas disponibles: La RKF45, recomendada para precisión media, está basada en un par 4–5 de Fehlberg y fue producida por Shampine hace ya bastantes años. La DVERK78 que utiliza un par 7–8 de Verner [23] que fue obtenido por este autor para sustituir al del mismo orden de Fehlberg y eliminar el problema de las cuadraturas en la estimación del error local. IMSL también incluye otros códigos llamados “clásicos” basados en fórmulas de orden más bajo. En MATLAB, dentro del “ODE SUITE”, tenemos los códigos “*ode 23*” y “*ode45*”, el primero está

basado en un par de órdenes 2-3 debido a Bogacki & Shampine y el segundo en el ya comentado par 4-5 de Dormand & Prince [10]. Puesto que en MATLAB las posibilidades gráficas juegan un papel muy importante, los códigos están provistos de interpolantes que evalúan la solución en los puntos necesarios para conseguir gráficas suficientemente lisas.

Con objeto de dar una visión más cercana de los diversos aspectos que hay que tener en cuenta para elaborar un código RK de uso general, vamos a comentar someramente algunos detalles del RKSUITE, producido recientemente por Brankin, Gladwell y Shampine [2], que además de ser autores de la máxima solvencia, han aprovechado su experiencia previa en códigos anteriores.

En primer lugar, este código implementa tres métodos adaptativos basados en pares de órdenes 2-3, 4-5 y 7-8; los dos primeros son debidos a Bogacki & Shampine y el último es el bien conocido de Prince & Dormand. Todos ellos actúan como métodos adaptativos con paso variable avanzando con la solución de mayor orden y estimando el error local y controlando los pasos de integración, tal como se indicó en el párrafo anterior. El usuario puede elegir el método que estime más idóneo de acuerdo con la precisión (y/o estabilidad) de su problema. El par 7-8 es el más eficiente y se aconseja para alta precisión, el par 4-5 para precisión media y el par 2-3 para precisión baja o bien cuando se detectan problemas de estabilidad. Hay que notar que esta práctica de incluir varios métodos adaptativos de distintos órdenes en un mismo paquete está cada vez más extendida, ya que permite al usuario disponer de una mayor flexibilidad y por tanto adaptarse mejor a las necesidades de cada problema. Para ayudar a la elección de uno u otro método, la documentación del código proporciona indicaciones más precisas sobre el método a utilizar, de acuerdo con el error que se desee obtener.

El código está dirigido a obtener aproximaciones a la solución de PVI para sistemas de  $N$ - ecuaciones diferenciales que se suponen dadas en la forma normal (1) en un conjunto de puntos prefijado dentro del intervalo de integración:  $a \leq \tau_1 < \tau_2 > \dots \tau_q \leq b$ , por tanto debemos pensar que tanto  $y$  como  $f$  son vectores de  $N$ - componentes.

Hay un primer aspecto importante de todo código numérico, y en particular de RKSUITE, que consiste en la forma práctica de controlar los errores. Podemos imaginar que puede haber sistemas diferenciales cuyas variables de

estado, las componentes  $y_1, \dots, y_N$  componentes del vector  $\mathbf{y}$  en el curso de la integración son todas grandes (pongamos del orden  $10^6$ ), o acaso todas pequeñas (del orden  $10^{-6}$ ), o bien están mezcladas e incluso pueden variar bastante su tamaño a lo largo de la integración. Por este motivo el exigir que el error absoluto de cada una de las componentes se mantenga por debajo de una cantidad común a todas (por ejemplo  $10^{-6}$ ), puede resultar en unos casos una exigencia inaceptable y en otros carente de sentido. Esto lleva a la necesidad de introducir un error relativo  $TOL$  común a todas las componentes, por tanto, cuando se piensa en exigir una tolerancia  $TOL = 10^{-6}$ , debemos entender que éste es el error relativo en cada una de las componentes o bien que estas tienen seis cifras significativas exactas. Por otro lado, y para dar mayor flexibilidad al código, con vistas a controlar individualmente los errores globales de cada una de las componentes, el código introduce un vector de errores absolutos  $THRES[I], I = 1, \dots, N$ . Así pues, suponiendo en una integración numérica definidos  $TOL$  y  $THRES[I], I = 1, \dots, N$ , la estimación del error  $EST[I]$  en el paso de  $(t_n, \mathbf{y}_n) \rightarrow (t_{n+1}, \mathbf{y}_{n+1})$ , exige que cada una de las componentes de la estimación del error local satisfaga:  $|EST[I]| \leq TOL * (\|\mathbf{y}_n\| + \|\mathbf{y}_{n+1}\|)/2 + THRES[I]$ .

Como es habitual en la resolución numérica de EDOs, los códigos de RKSUITE pueden funcionar en el modo “paso a paso” o en el modo intervalar. En el modo paso a paso el código avanza la integración desde el punto actual  $t_n$  hasta el siguiente de la red  $t_n + h$ , de acuerdo con su selección de paso compatible con los requerimientos de error. En la modalidad intervalar, se dan los pasos necesarios para recorrer un intervalo prefijado, típicamente  $[\tau_n, \tau_{n+1}]$ , es decir se sale de  $\tau_n$  y se aterriza tras uno o varios pasos en  $\tau_{n+1}$ .

Otros aspectos de importancia en los códigos de paso variable son los algoritmos de cambio de paso, es decir la previsión del paso siguiente a un paso aceptado o de reducción tras uno fallado y la elección del paso inicial. En los códigos considerados se usa como estimador el llamado error por paso, es decir, la diferencia entre dos soluciones de ordenes consecutivos, y las fórmulas de cambio de paso son las derivadas del comportamiento asintótico de éste con factores de corrección, para minimizar el número de pasos fallados.

Por lo que respecta al paso inicial, siempre existe la posibilidad de que pueda ser elegido por el usuario, sin embargo, para ello se requiere un cierto conocimiento del comportamiento de la solución y tal situación no siempre ocurre. En RKSUITE, los autores han conseguido un algoritmo que, tras

ciertas evaluaciones adicionales, selecciona automáticamente el paso inicial en la escala del problema y compatible con el error local dado por el usuario, de tal manera que si el usuario pone el paso inicial nulo el código ya entiende que debe aplicar el algoritmo para su cálculo automático.

El código incluye la posibilidad de estimar el error global producido con cualquiera de los tres métodos adaptativos, usando la extrapolación de Richardson que se comentará más adelante. Entonces, suponiendo que en la definición del problema se ha especificado un cierto error global, en paralelo con la integración numérica elegida, se va estimando el error global en cada uno de los pasos y se retiene en memoria el máximo error global de todos los pasos, así como el del último paso realizado. También RKSUITE dispone de una subrutina de interpolación que permite usar esta facilidad en los pares de órdenes 2-3 y 4-5, para todas o parte de las componentes de la solución numérica.

No vamos a entrar en detalles de programación, solamente mencionar que hay una serie de módulos, por ejemplo hay uno de definición que contiene los datos del problema numérico que no varían a lo largo de la integración (la definición de la función  $f$  de la ecuación diferencial va aparte). También hay otro módulo que consiste en la aplicación de un paso del método adaptativo y luego están, además del vector de trabajo, otros adicionales como los de interpolación o de control del error global.

En el proceso de integración hay una serie de controles para detectar las anomalías que se puedan producir y que los autores clasifican en dos tipos: Los llamados errores fatales que paran totalmente la integración de tal manera que para continuar hay que reinicializar toda la integración. Entre éstos están, aparte de los de control de la programación, los debidos a pasos muy pequeños, que prácticamente paralizan el avance de la integración y también si el control del error global ha sido violado. Hay otro grupo de errores, llamados no fatales, que continúan la integración, pero advierten de circunstancias especiales. Entre ellos se encuentran las dificultades de estabilidad (hay una prueba para estimar si el problema resulta stiff y por tanto el método usado no es el más adecuado) y cuando se pide la solución en muchos puntos y tal circunstancia deteriora la eficiencia del método adaptativo. En estos casos el usuario puede consultar STAT para conocer más detalles sobre las anomalías detectadas.

## 5 Nuevas tendencias en los códigos Runge–Kutta generales

En el cálculo numérico, y en particular en la resolución numérica de EDOs, el desarrollo de nuevos métodos o la mejora de los existentes suele estar motivado por problemas concretos de interés práctico y apoyado por el análisis numérico, que es el complemento imprescindible para investigar y analizar los nuevos métodos y sus propiedades. De acuerdo con esto, las necesidades de cálculo planteadas por los usuarios de EDOs, junto a las investigaciones realizadas en el tema, determinan las tendencias de los futuros códigos RK.

Veremos seguidamente algunas de las innovaciones que, en nuestra opinión, van a introducirse (o se han introducido) en las nuevas generaciones de códigos RK.

En primer lugar se ha producido un importante esfuerzo investigador para obtener nuevas familias de métodos adaptativos que posean conjuntamente las mejores propiedades de aproximación numérica, estabilidad y coste computacional, ya que es obvio que cualquier mejora del método adaptativo usado, implica una mejora del código basado en éste. Por tanto cabe predecir una sustitución lenta y progresiva de muchas de las fórmulas actuales por otras más eficaces. De hecho las fórmulas de Fehlberg y Verner que han marcado una generación de códigos RK, poco a poco se van sustituyendo por otras de Dormand & Prince, Bogacki & Shampine etc.

Existe una creciente preocupación por los aspectos de robustez, fiabilidad y eficacia de los nuevos códigos. Esto se detecta en el interés por diseñar algoritmos cada vez más perfeccionados para la estimación del paso inicial, así como para los cambios y predicciones de paso. Igualmente existen algoritmos cada vez más sensibles para detectar circunstancias especiales de la integración, como puede ser la stiffness que hace inadecuados a los métodos RK explícitos

Por otro lado, parece aceptarse en la actualidad que todo integrador numérico de uso general tenga la posibilidad de proporcionar lo que se llama “salida densa”, es decir que sea capaz de calcular una solución continua en la parte que interese del intervalo de integración. Recordemos al respecto que los métodos adaptativos solamente calculan aproximaciones a la solución en los puntos de una red del intervalo de integración que es generada au-



tomáticamente por el propio método, a medida que la integración avanza y que la distancia entre dos puntos consecutivos de esta red no tiene que ser necesariamente pequeña, ya que, por razones de eficiencia, la integración avanza con los mayores pasos posibles compatibles con la tolerancia prefijada por el usuario.

La inclusión de la mencionada salida densa en el software de ODEs está motivada entre otras por las siguientes razones:

1. Dar una representación gráfica de la solución numérica para lo cual puede ser necesario calcularla en muchos puntos del intervalo de integración.
2. Estimar el defecto de la solución numérica aproximada  $\mathbf{y}_h(t)$  respecto de la ecuación diferencial  $\delta(t) = \mathbf{y}'_h(t) - \mathbf{f}(\mathbf{y}_h(t))$  que, como han mostrado algunos investigadores (Enright, Higham, etc ), puede servir para deducir estimaciones de los errores local y global.
3. Localizar singularidades en la solución de la ecuación diferencial o en alguna de sus derivadas.
4. Localizar puntos especiales de soluciones como pueden ser sus cortes con una hipersuperficie sección (secciones de Poincaré) para el cálculo de órbitas periódicas.
5. Integración numérica de ecuaciones diferenciales con argumentos retardados, p.c.  $\mathbf{y}'(t) = \mathbf{f}(\mathbf{y}(t), \mathbf{y}(t - r))$ , donde  $r = r(t) > 0$  es el retardo, en las cuales para el cálculo de  $\mathbf{f}$  hay que evaluar la solución en puntos pasados que no son, necesariamente, puntos previos de la red.

Una posibilidad trivial para que un método RK adaptativo calcule la solución en puntos conocidos a priori, es ir modificando adecuadamente los pasos de integración para “aterrizar” justamente en dichos puntos, pero como ya observaron Shampine y otros, esta forma de proceder no sólo hace muy ineficaz al método, sino que además puede introducir errores acumulados importantes en la integración numérica. De acuerdo con esto, pronto se vió claro que lo más conveniente es dejar que el integrador avance con sus pasos naturales y recurrir, cuando sea necesario, a algún tipo de interpolación entre puntos de la red. Así, los primeros interpolantes usados eran del tipo

Hermite o Hermite-Birkhoff sobre dos o más puntos consecutivos de la red. Con esta forma de proceder se ve fácilmente que, si se quiere que el error de interpolación no domine al error propio del método, es necesario, para métodos de orden  $\geq 4$ , usar más de dos puntos consecutivos de la red, lo que hace perder la deseable propiedad uni-paso del método. Por tanto, hacia comienzos de los ochenta, algunos autores como Horn y Shampine, comenzaron a estudiar fórmulas adaptativas que tenían la posibilidad de producir aproximaciones polinómicas continuas entre puntos consecutivos de la red, sin necesidad de evaluaciones adicionales de función, o bien con un número mínimo de evaluaciones adicionales de la forma

$$\mathbf{y}_h(t_n + \theta h) = \mathbf{y}_n + \sum_{j=1}^n b_j(\theta) \mathbf{g}_j, \quad \theta \in [0, 1],$$

donde  $b_j(\theta)$  son funciones polinómicas escalares de  $\theta$  que hacen el papel de los pesos externos de la fórmula RK para cada  $\theta \in [0, 1]$ . En particular, Horn (1983) observó que para un par de Fehlberg de órdenes 4–5 era posible construir una extensión continua de orden cuatro y Shampine vió que, añadiendo una evaluación adicional al par 5–4 de Dormand & Prince, era posible construir una extensión continua de orden cinco. En esta dirección han apuntado bastantes investigaciones sobre los métodos continuos de la última década: Tomando como base un método adaptativo discreto de eficiencia probada, añadir nuevas evaluaciones de función en puntos apropiados para obtener, cuando así se requiera, una solución continua entre pares de puntos consecutivos de la red. Shampine, Dormand & Prince y nosotros mismos hemos propuesto [6] diversas extensiones del bien conocido par 5–4 de Dormand & Prince. En particular, tras experimentos numéricos realizados por distintos autores, nuestra extensión es considerada por el momento como la mejor entre las de la fórmula considerada y solamente requiere dos evaluaciones adicionales de función por paso. Para el par 8–7 de Dormand & Prince, varias extensiones continuas de orden siete han sido propuestas con tres evaluaciones adicionales de función.

Por otra parte, otros investigadores como Owren, Zennaro y Verner, han abordado la construcción de métodos adaptativos continuos desde un punto de vista global, es decir, determinando simultáneamente el método continuo y el estimador del error local sobre un conjunto mínimo de etapas. En tal sentido se han dado condiciones algebraicas sobre los coeficientes del método

que garantizan la existencia de fórmulas RK continuas y se ha determinado (hasta cierto orden) el número mínimo de etapas necesarias para la compatibilidad de estas ecuaciones. En la tabla siguiente se da el número mínimo de etapas necesarias para tener una fórmula discreta y continua hasta orden seis

Orden	1	2	3	4	5	6
RK Discreta	1	2	3	4	6	7
RK Continua	1	2	4	6	8	11

Tabla 2: Número mínimo de etapas para alcanzar un orden dado

Se observa que la diferencia entre las etapas del caso discreto y continuo crece muy rápidamente con el orden, lo cual nos lleva a pensar que para fórmulas de orden alto el coste adicional es notable. Además, la construcción explícita de tales fórmulas con error local pequeño, es extremadamente complicada, por lo cual, si bien las ideas de Owren y Zennaro son atractivas, el problema global es complicado y hasta el momento no se han obtenido métodos globales con claras ventajas respecto a los anteriores, por consiguiente, aún no son de utilidad práctica.

Otro aspecto importante en el que se esperan innovaciones importantes está basado en el hecho de que todo usuario de software de ODEs, además de una solución aproximada de su problema está interesado en disponer de la máxima información sobre los errores en dicha solución. También es claro que un método numérico no nos puede proporcionar en general el valor exacto de dicho error, ya que ello implicaría la posibilidad de conocer la solución exacta de todo problema diferencial y esto es una utopía inalcanzable. Sin embargo cualquier información sobre el tamaño o comportamiento del error puede resultar muy valiosa para el usuario. Notemos al respecto que los métodos adaptativos comentados anteriormente proceden de manera que los errores locales, es decir, los errores introducidos en cada paso, se mantienen por debajo de la tolerancia prefijada por el usuario, pero en el curso de la integración numérica hay que contar con la propagación de dichos errores, que depende de los comportamientos de estabilidad de la propia ecuación diferencial y del método numérico, por tanto ya se intuye que los estudios generales en esta dirección son complicados. No obstante lo anterior, todos los investigadores reconocen unánimemente que los progresos en éste sentido son fundamentales para mejorar la seguridad y robustez del software.

Stetter hace aproximadamente dos décadas introdujo la llamada “proporcionalidad a la tolerancia” como medio de controlar el error global. Un método numérico posee dicha propiedad si, para cada tolerancia  $TOL$  (pequeña), los errores globales conseguidos en la integración numérica con dicha tolerancia en cada punto  $t_n$  de la red son, aproximadamente, de la forma  $C(t_n) TOL$ , donde  $C(t)$  es una función que puede depender del método usado y también del problema diferencial integrado, pero es independiente de  $TOL$ . Esto implica que, para  $TOL$  pequeñas, el error global en cada punto es una función aproximadamente lineal de la tolerancia y por tanto, si se hacen dos integraciones con tolerancias distintas, pongamos  $TOL$  y  $TOL/10$  los dígitos de la solución numérica que sean comunes a las dos integraciones pueden considerarse como exactos. Esta práctica de hacer dos integraciones con  $TOL$  distintas y comparar resultados ya había sido usada empíricamente durante mucho tiempo, para asegurar los resultados numéricos cuando no se conoce muy bien la fiabilidad de éstos y el concepto de proporcionalidad a la tolerancia de Stetter le ha dado realmente carta de existencia.

Naturalmente surgió la cuestión de si un método numérico adaptativo (o su extensión continua), en el que se incluye una técnica de cambio de paso, es capaz de producir soluciones numéricas con dicha proporcionalidad a la tolerancia. Esto fué estudiado inicialmente por Stetter y más tarde por Higham y nosotros mismos [8], identificando los métodos adaptativos y las técnicas de cambio de paso que implican la proporcionalidad a la tolerancia y puede decirse que el nuevo software propuesto en la actualidad goza en general de esta importante propiedad. Hay que notar que en esta línea Enright, Higham y otros han propuesto nuevas técnicas de cambio de paso para pares de métodos encajados o continuos, basadas en el defecto de la solución numérica y que garantizan una proporcionalidad a la tolerancia más estricta que la anteriormente comentada, pero como el defecto de la solución numérica es costoso de calcular con precisión, no han tenido todavía un interés práctico generalizado.

Como acabamos de comentar, en los métodos adaptativos que verifican la proporcionalidad a la tolerancia, tras concluir dos integraciones completas del mismo problema con dos  $TOL$  distintas, podemos deducir una estimación del error global. Otra posibilidad, acaso más conveniente, sería diseñar algoritmos que asociados a un método adaptativo nos proporcionen, junto a la solución numérica, una estimación del error local en la misma a medida

que la integración avanza. Esta idea ha estado por supuesto en la mente de muchos investigadores y se han propuesto varias técnicas para la estimación del error global simultánea con la solución. El principal inconveniente es su alto costo computacional, así como la complicación adicional que conlleva su inclusión en el software existente, por ello, salvo algunas excepciones (GERK, RKSUITE), no se ha incluido hasta el momento esta opción. Sin embargo, las crecientes exigencias de calidad nos hacen pensar que en el futuro software tal opción deberá ser incluida, por eso comentaremos algunas de las técnicas más prometedoras al respecto (hay que notar que en 1986, Skeel ya recopiló trece técnicas para estimar el error global).

Desde luego siempre podemos recurrir a la clásica extrapolación de Richardson [18], que además puede usarse en conjunción con cualquier método adaptativo, para ello en paralelo con la secuencia de pasos generada por el método se lleva cabo una integración con pasos mitad, lo cual da lugar a dos soluciones numéricas  $\mathbf{y}_n$  e  $\hat{\mathbf{y}}_n$ , respectivamente en cada punto  $t_n$  de la red inicial. Entonces, como demostró Henrici, bajo ciertas condiciones sobre la variación de pasos, la expresión  $(\mathbf{y}_n - \hat{\mathbf{y}}_n)/(2^p - 1)$ , donde  $p$  es el orden del método, es una estimación asintóticamente correcta del error global en la solución  $\mathbf{y}_n$ . Los experimentos numéricos con extrapolación indican que es una técnica muy fiable, pero tiene el inconveniente de su alto costo computacional.

Otra técnica interesante fué propuesta por Zadunaisky hacia 1964 en el contexto de la integración numérica de problemas de Mecánica Celeste y posteriormente se ha extendido a EDOs arbitrarias e incluso la idea básica ha sido usada en el contexto de otros problemas de Análisis Numérico. Se trata de construir artificialmente un problema diferencial "próximo" al que se está resolviendo cuya solución exacta sea conocida. Entonces cabe esperar que, si se aplica el mismo método numérico a los dos problemas (por continuidad), los errores de ambos sean similares y como los del problema próximo son conocidos podemos hacer así una estimación de los del problema dado. Mas concretamente, si estamos integrando el problema diferencial (1), y tenemos una solución numérica continua  $\mathbf{y}_h(t)$  (p.e. obtenida con cualquier método que posea salida densa), dicha solución numérica será solución exacta de un problema perturbado de (1)  $\mathbf{y}'_h(t) = \mathbf{f}(\mathbf{y}_h(t)) + \delta(t)$ ,  $\mathbf{y}_h(t_0) = \mathbf{y}_0$  donde  $\delta(t)$  es el defecto de  $\mathbf{y}_h(t)$ , que cabe esperar sea pequeño. Entonces este segundo problema es próximo al problema original y por tanto los errores globales en la integración numérica de ambos cabe esperar que sean similares.

Hay que notar que la construcción del problema próximo requiere una solución numérica continua  $\mathbf{y}_h(t)$  de éste y por tanto depende de la extensión continua o del interpolante del método adaptativo considerado. En las primeras aplicaciones de la teoría de Zadunaisky se usaron interpolantes tipo Hermite o Hermite-Birkhoff, basados en la información de la solución numérica en varios puntos consecutivos de la red, pero como se ha indicado anteriormente tales interpolantes aunque tienen un costo computacional mínimo no resultan adecuados porque destruyen el carácter unipaso del método y los experimentos numéricos indican que no son muy fiables. En vista de ello parece necesario usar una extensión continua del tipo de las mencionadas en la salida densa que tienen en general un mayor costo computacional, pero son más fiables.

El esquema de Zadunaisky ha sido fundamentado teóricamente por Stetter y colaboradores y ha sido objeto de bastantes experimentos numéricos. Se ha visto que, en algunos casos, la estimación calculada del error es muy precisa, pero que en otros los resultados son decepcionantes. En mi apreciación personal será necesario un mayor refinamiento de esta técnica antes de que pueda usarse en códigos de tipo general.

Mencionaremos por último la teoría llamada por Skeel “solución de la corrección”, que es posiblemente la que goza de una mayor consideración en la actualidad. Para su aplicación hay que partir de nuevo de una extensión continua  $\mathbf{y}_h(t)$  de la solución discreta producida por un método adaptativo discreto. Entonces el error global o “corrección” de dicha solución aproximada, dado por  $\varepsilon(t) = \mathbf{y}(t) - \mathbf{y}_h(t)$ , es solución del problema diferencial  $\varepsilon'(t) = \mathbf{F}(t, \varepsilon(t)) = f(\mathbf{y}_h(t) + \varepsilon(t)) - \mathbf{y}'_h(t)$ ,  $\varepsilon(t_0) = 0$ , y por tanto, su resolución numérica, dará un valor aproximado de  $\varepsilon(t)$  y de la corrección. Se trata pues de avanzar en paralelo la solución numérica  $\mathbf{y}_h(t)$  y la corrección  $\varepsilon(t)$ , pero a diferencia de la técnica de Zadunaisky, la ecuación de la corrección puede resolverse con otro método distinto del usado en el problema original. Además, para que la estimación sea más segura, los errores cometidos en la resolución numérica de la ecuación de la corrección deben ser más pequeños que los de la resolución de la ecuación original, lo que obliga a buscar fórmulas para  $\varepsilon(t)$  que sean de orden superior a las de  $\mathbf{y}_h(t)$ . Dormand y sus colaboradores han observado que, debido a la forma especial de la ecuación diferencial de  $\varepsilon(t)$ , es posible diseñar pares de métodos adaptativos especiales que resuelven la ecuación de la corrección con uno o dos

órdenes más que la solución de  $\mathbf{y}_h$  y con un costo computacional bastante ajustado. En este sentido se han propuesto métodos de distintos órdenes y nuestro grupo de trabajo [7] ha realizado experimentos numéricos con resultados positivos, pero estas investigaciones aún no han calado en los actuales códigos RK generales.

## 6 Métodos de Runge–Kutta implícitos

Los métodos RK considerados en los párrafos anteriores se llaman explícitos porque el avance de un paso se lleva a cabo realizando exclusivamente (aparte de operaciones elementales) un cierto número de evaluaciones de la función  $\mathbf{f}(t, \mathbf{y})$  de la ecuación diferencial en argumentos conocidos. Debido a esta simplicidad, así como el carácter unipaso, los métodos RK explícitos han sido candidatos naturales para la resolución numérica de ecuaciones diferenciales.

Sin embargo existen algunos problemas diferenciales llamados “stiff”, que además aparecen bastante en la práctica, cuya resolución mediante fórmulas explícitas resulta prácticamente imposible. Aunque está fuera de nuestro propósito el explicar la naturaleza de los problemas stiff y analizar las dificultades que experimentan los códigos RK explícitos con tales problemas (ver por ejemplo [14]) daremos algunas ideas someras sobre tales problemas. Un problema se llama “stiff” cuando existe alguna variedad en el espacio fásico ampliado  $(t, \mathbf{y})$  fuertemente estable, es decir tal que el flujo fuera de la variedad tiende rápidamente hacia ésta. En el caso unidimensional, podemos considerar el prototipo de ecuación lineal  $y' = \lambda(y - \varphi(t)) + \varphi'(t)$ , donde  $\varphi$  es una función de variación lenta y  $\lambda$  una constante negativa muy grande. En este problema todas las soluciones tienden rápidamente hacia la variedad  $y = \varphi(t)$ , que es además una solución particular del problema considerado. Si un problema stiff ( p.e. el anterior con  $\lambda = -10^6$  ) se integra con un código RK explícito, se observa que para preservar la estabilidad el código toma pasos muy pequeños y la integración prácticamente no avanza.

Esta dificultad ha llevado a la consideración de fórmulas implícitas cuyas propiedades de estabilidad son muy superiores a las explícitas, como p.e. el método implícito de Euler, en el cual la solución numérica  $\mathbf{y}_{n+1}$  de la ecuación diferencial en el punto  $t_{n+1} = t_n + h$  en función de su valor en el punto precedente  $(t_n, \mathbf{y}_n)$ , está dada por:  $\mathbf{y}_1 = \mathbf{y}_0 + hf(t_1, \mathbf{y}_1)$ . Por tanto

en cada paso de integración tenemos que resolver un sistema algebraico para calcular la nueva solución  $\mathbf{y}_{n+1}$ .

Un método de RK general de  $s$  etapas es un algoritmo de la forma (2) donde las etapas  $\mathbf{g}_1, \dots, \mathbf{g}_s$  verifican las ecuaciones implícitas

$$\mathbf{g}_i = \mathbf{f}(t_n + c_i h, \mathbf{y}_n + h \sum_{j=1}^s a_{ij} \mathbf{g}_j)$$

siendo  $b_i, a_{ij}$  los pesos externos e internos respectivamente y  $c_i$  los nodos de la fórmula considerada. En particular, si  $s = 1, a_{11} = 1, c_1 = 1, b_1 = 1$  tenemos el anterior método implícito de Euler. Es claro que si  $a_{ij} = 0$  para todo  $j \geq i$  la fórmula es explícita, pero en otro caso las ecuaciones de las etapas son implícitas.

Tales métodos surgieron inicialmente como generalización teórica de los métodos explícitos y Kuntzmann y Butcher, hacia principios de los sesenta, propusieron fórmulas de  $s$  etapas y orden  $2s$  en las cuales los nodos son los ceros de los polinomios ortogonales de Legendre en el intervalo  $[0, 1]$  y los pesos externos los de la fórmula de cuadratura gaussiana asociada. Esto mostraba una relación entre fórmulas de cuadratura y ciertas familias de métodos RK implícitos que fue explotada sistemáticamente por Butcher y Ehle [4], para deducir familias implícitas de los tipos Radau y Lobatto, en las cuales los pesos externos  $b_i$  y los nodos  $c_i$  tienen los valores de las correspondientes fórmulas de cuadratura y los coeficientes  $a_{ij}$  se determinan verificando adecuadas condiciones adicionales.

El principal inconveniente de las fórmulas implícitas radica en la resolución de las ecuaciones de las etapas. Si bien se pueden dar condiciones teóricas que garantizan la existencia de una única solución, las dificultades viene a la hora de calcularla de modo efectivo, ya que hay que usar métodos iterativos que aseguren no solo la convergencia, sino también que la velocidad de convergencia sea la mayor posible. Se ha visto que para los problemas stiff la iteración funcional, que es la más sencilla de aplicar, no es en general convergente salvo para pasos muy pequeños y hay que recurrir a métodos tipo Newton y quasi-Newton en los cuales entran en juego los jacobianos, que son costosos de calcular, así como las factorizaciones de los sistemas lineales correspondientes.

En resumen, existen métodos RK implícitos que poseen buenas propiedades de estabilidad, pero su efectividad depende esencialmente de la resolución de



las ecuaciones algebraicas implícitas de las etapas, por tanto ha habido un importante esfuerzo investigador en los últimos años para encontrar métodos implícitos (como son SIRK, DIRK y otros ) con buenas propiedades de estabilidad y tales que la resolución de las ecuaciones algebraicas no sea demasiado costosa.

Posteriormente, una extensa lista de investigadores entre los que se cuentan Hairer, Wanner, Norsett, Crouzeix, Butcher, y nuestro grupo de trabajo, hemos analizado las propiedades de estabilidad de distintas familias de métodos implícitos con vistas a su utilidad para la resolución de los mencionados problemas stiff, pero esta es una larga historia cuyo último capítulo aún no se puede dar por concluido.

## 7 Métodos de Runge–Kutta para ecuaciones especiales

Los métodos considerados en las secciones anteriores presuponen sistemas diferenciales de primer orden escritos en forma normal  $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$ , ya que, salvo en el caso de ecuaciones implícitas, todas las demás se pueden escribir fácilmente en la forma normal. Por este motivo los métodos numéricos diseñados para sistemas normales de primer orden son los más útiles por su amplia aplicabilidad.

Otro enfoque más particular consiste en considerar sistemas diferenciales de tipos especiales y diseñar métodos adecuados para cada uno de estos tipos. En tal caso se podría aprovechar la restricción en la clase de problemas diferenciales para mejorar los integradores en algunos aspectos interesantes de la clase considerada. Estas ideas han estado desde hace bastantes años en la mente de muchos investigadores y existe una variada propuesta de métodos RK especiales, adaptados a clases particulares de problemas de interés práctico. Comentaremos brevemente algunos de los tipos más conocidos.

Existen bastantes sistemas diferenciales que, en su formulación natural, aparecen como un sistema diferencial de segundo orden  $\mathbf{y}'' = \mathbf{f}(t, \mathbf{y}, \mathbf{y}')$ . Pensemos, por ejemplo, en las ecuaciones de la Mecánica Clásica en las cuales las aceleraciones (típicamente las derivadas segundas  $\mathbf{y}''$ ), son proporcionales

a las fuerzas  $\mathbf{f}(t, \mathbf{y}, \mathbf{y}')$ . Incluso en muchos casos estas fuerzas no dependen de las velocidades y los sistemas toman la forma  $\mathbf{y}'' = \mathbf{f}(t, \mathbf{y})$ , que se llama ecuación especial de segundo orden. La integración numérica de estos sistemas diferenciales se puede hacer introduciendo las velocidades  $\mathbf{v} = \mathbf{y}'$  y escribiendo el sistema dado en la forma normal equivalente  $\mathbf{y}' = \mathbf{v}$ ,  $\mathbf{v}' = \mathbf{f}(t, \mathbf{y})$ , a la cual son aplicables los métodos standard. Sin embargo podría considerarse la resolución directa del sistema de segundo orden sin pasar a la forma modificada. Esta posibilidad ya fué considerada por Nyström hacia 1925 proponiendo fórmulas en las cuales la solución numérica avanza desde  $(t_0, \mathbf{y}_0, \mathbf{y}'_0) \rightarrow (t_0 + h, \mathbf{y}_1, \mathbf{y}'_1)$ , con un algoritmo de la forma

$$\mathbf{y}_1 = \mathbf{y}_0 + h\mathbf{y}'_0 + h^2 \sum_{i=1}^s \bar{b}_i \mathbf{f}_i, \quad \mathbf{y}'_1 = \mathbf{y}'_0 + h \sum_{i=1}^s b_i \mathbf{f}_i,$$

donde  $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_s$  se calculan recurrentemente por

$$\begin{aligned} \mathbf{f}_1 &= \mathbf{f}(t_0, \mathbf{y}_0) \\ \mathbf{f}_2 &= \mathbf{f}(t_0 + hc_2, \mathbf{y}_0 + hc_2 \mathbf{y}'_0 + h^2 \bar{a}_{21} \mathbf{f}_1) \\ \mathbf{f}_3 &= \mathbf{f}(t_0 + hc_3, \mathbf{y}_0 + hc_3 \mathbf{y}'_0 + h^2 (\bar{a}_{31} \mathbf{f}_1 + \bar{a}_{32} \mathbf{f}_2)) \\ &\vdots \end{aligned}$$

y  $\bar{b}_i, b_i, \bar{a}_{ij}, c_i$ , son parámetros escalares constantes que caracterizan la fórmula y  $s$  el número de etapas. Puede apreciarse que estas fórmulas poseen una estructura similar a las de RK explícitas, por ello se llaman de Runge-Kutta-Nyström (RKN).

Se puede ver fácilmente que para cualquier fórmula RK existe una fórmula RKN que produce los mismos resultados numéricos y que existen algoritmos RKN que no se pueden obtener con fórmulas RK, lo cual nos indica que tenemos un mayor margen de maniobra dentro de las fórmulas RKN y por tanto cabe esperar que, a igualdad de coste computacional, existan fórmulas RKN con mejores propiedades que las de RK. Este hecho ya fue detectado por Nyström que, en su trabajo original, propuso fórmulas de orden cuatro con tres etapas y de orden cinco con cuatro etapas, mientras que en el contexto de las fórmulas RK son necesarias al menos cuatro y seis etapas respectivamente. Por otro lado otra ventaja adicional es que la implementación de las fórmulas RKN requiere aproximadamente la mitad de memoria que las RK, y esto puede ser importante para problemas de dimensión alta.

El estudio del orden de las fórmulas RKN sufrió un impulso notable a raíz de la publicación del ya mencionado trabajo de Butcher. Fueron Hairer & Wanner los que adaptaron la teoría de Butcher a las ecuaciones de segundo orden introduciendo unos árboles especiales y las correspondientes series formales (llamadas de Nyström), que permiten analizar de un modo sistemático el orden de las fórmulas RKN.

La implementación efectiva de fórmulas RKN con paso variable ha seguido las pautas marcadas por las fórmula RK, es decir, se han construido métodos adaptativos en los cuales se estima el error local por diferencia entre dos soluciones de distintos órdenes, y dicha estimación del error es usada para controlar el error local a lo largo de la integración y para la previsión y cambio de paso. Así hay, entre otras, familias de métodos adaptativos RKN de varios órdenes debidas a Fehlberg, Filippi & Graff, Sharp & Fine y Dormand, El-Mikkawy & Prince que en el caso de éstos últimos llegan a alcanzar orden doce. Algunas de estas han sido implementadas en códigos disponibles en varias librerías.

Entre las ecuaciones especiales cuya integración numérica ha sido objeto de una investigación mas exhaustiva en los últimos años están sin duda algunas que aparecen en los sistema hamiltonianos. Estos sistemas se presentan en la modelización de los más variados fenómenos de la Física y vienen dados por un sistema de ecuaciones diferenciales de la forma

$$\frac{d\mathbf{p}}{dt} = -\frac{\partial\mathcal{H}(\mathbf{p}, \mathbf{q})}{\partial\mathbf{q}}, \quad \frac{d\mathbf{q}}{dt} = \frac{\partial\mathcal{H}(\mathbf{p}, \mathbf{q})}{\partial\mathbf{p}},$$

donde  $\mathbf{q}$  y  $\mathbf{p}$  son los  $N$ -dimensional vectores de las coordenadas y momentos generalizados respectivamente y  $\mathcal{H}$  es una función escalar de las coordenadas y momentos que se llama función hamiltoniana. La dimensión  $N$  se llama número de grados de libertad y el  $2N$  dimensional espacio de coordenadas y momentos se llama espacio fásico. La función hamiltoniana a veces depende explícitamente del tiempo, pero para simplificar la presentación no incluiremos aquí tal dependencia.

En muchas aplicaciones de Mecánica la función hamiltoniana representa la energía del sistema y tiene la forma  $\mathcal{H} = T(\mathbf{p}) + V(\mathbf{q})$ , donde  $T$  es la energía cinética expresada como una forma cuadrática definida positiva que, en su forma más simple es diagonal  $T(\mathbf{p}) = (1/2)\mathbf{p}^T\mathbf{p}$ . Los hamiltonianos del tipo anterior se llaman separables.

La característica más importante de las soluciones de sistema hamiltonianos es que producen un flujo hamiltoniano simpléctico. Es decir la transformación que pasa de los momentos y coordenadas en un instante, pongamos  $t = 0$ , a otro instante cualquiera  $t$  es una transformación canónica o simpléctica. No vamos a entrar aquí en detalles sobre transformaciones canónicas, ya que son bien conocidas por muchos colegas de la casa y en particular por los relacionados con la Mecánica Celeste. Baste decir que un flujo simpléctico tiene algunas propiedades muy especiales. En dimensión  $N = 1$ , es decir espacio de fases plano, un flujo es simpléctico si y solo si conserva las áreas, lo que implica que la imagen de una figura plana se va deformando pero conserva el área. En particular, esto impide la existencia de ciclos límites en el plano fásico. En dimensión superior se conserva la suma de las áreas de las proyecciones sobre los planos coordenados  $(p_i, q_i)$  de una superficie bidimensional y esta propiedad implica la inexistencia de equilibrios asintóticamente estables, es decir, equilibrios a los cuales tienden las soluciones próximas.

De acuerdo con lo anterior parece natural requerir que los integradores de sistemas hamiltonianos avancen la solución numérica con transformaciones canónicas con objeto de respetar el carácter simpléctico del flujo. Los métodos numéricos que poseen dicha propiedad se llaman simplécticos y su desarrollo ha tenido lugar desde los años ochenta, debido, no sólo a investigadores de análisis numérico, sino también a físicos con intereses computacionales. Existe una importante línea de investigación, que se inició con Feng y sus colaboradores, en la cual los métodos simplécticos son producidos por funciones generatrices dependientes del paso de integración como parámetro. Puesto que las funciones generatrices pueden ser de diferentes tipos esto se traduce en los correspondientes métodos simplécticos. El principal inconveniente de estas fórmulas es que la función generatriz, dada en potencias del paso de integración, hay que aproximarla sucesivamente hasta un cierto orden y su cálculo depende de la función hamiltoniana a integrar, por tanto cada problema a integrar requiere un tratamiento diferenciado para construir su función generatriz.

Desde otro punto de vista, nuestro colega de la Universidad de Valladolid el Prof. Sanz Serna [20] abordó, junto a sus colaboradores, un detallado estudio de los métodos RK simplécticos que tiene como punto de partida un importante artículo suyo de 1988, en el cual se dan las condiciones sobre los

coeficientes de un método RK para que sea simpléctico. Estas condiciones expresadas mediante ecuaciones algebraicas sencillas permiten verificar inmediatamente si el método es simpléctico. La primera consecuencia práctica de este trabajo es que todo método simpléctico para funciones hamiltonianas arbitrarias es implícito. En particular las fórmulas de Gauss-Legendre de orden maximal son simplécticas. Estas fórmulas son al menos teóricamente buenas candidatas para la integración simpléctica, pero al ser implícitas hay que resolver eficientemente las ecuaciones de las etapas para conseguir métodos útiles desde el punto de vista computacional. Este problema motivó la tesis doctoral de nuestra colaboradora Maria Pilar Laburta en la cual se proponen distintos algoritmos para deducir valores iniciales, que permitan resolver las ecuaciones no lineales con el mínimo número de iteraciones y se hace una amplia experimentación numérica para optimizar los integradores simplécticos de Gauss-Legendre de dos y tres etapas.

Si bien la integración simpléctica general requiere métodos RK implícitos, el grupo del Prof. Sanz Serna observó que para hamiltonianos separables era posible encontrar métodos RK simplécticos de tipo explícito, usando distintas fórmulas para las coordenadas y para los momentos. Tales métodos se llaman "particionados" y existe un amplio catálogo de distintos órdenes, que se han usado para aplicaciones concretas. También, como los sistemas hamiltonianos separables se pueden reducir a sistemas de segundo orden, la investigación se ha extendido a la obtención de fórmulas RKN simplécticos. En particular, Mari Paz Calvo de Valladolid ha obtenido fórmulas de Nyström de varios órdenes y ha analizado su comportamiento en distintos experimentos numéricos.

A pesar de este vigoroso desarrollo de fórmulas simplécticas, la elaboración de un software de calidad tropieza con varios inconvenientes: Por un lado el ya mencionado de que en el caso de sistemas hamiltonianos generales las fórmulas son implícitas y su implementación se complica bastante. Por otro que aplicación de fórmulas simplécticas con paso variable destruye el carácter simpléctico global del flujo y se pierde el principal objeto de tales métodos, por tanto deben ser aplicados con paso fijo y esto puede ser no conveniente en ciertos problemas. En nuestra opinión estos inconvenientes han frenado bastante el desarrollo de software de uso general.

Finalmente, mencionar que los estudios teóricos acerca de integradores simplécticos han producido resultados colaterales muy interesantes, p.e. el

análisis regresivo del error permite considerar la solución numérica como solución exacta de un problema hamiltoniano perturbado, lo que permite conectar con las teorías de perturbaciones y aplicar p.e. la teoría de Kolmogorov-Arnold-Moser.

## References

- [1] R.F. ARENSTORF, *Periodic solutions of the restricted three body problem representing analytical continuations of Keplerian elliptic motions*, Amer. J. Math. Vol. LXXXV, pp. 27-35, (1963).
- [2] R.W. BRANKIN, I. GLADWELL AND L.F. SHAMPINE, *RKSUITE: a suite of Runge-Kutta codes for the initial value problem for ODEs*, SMU Math Report 92-S1, (1992). (also available in the NAG Fortran 77 library and the IMSL Math. library).
- [3] J.C. BUTCHER, *Coefficients for the study of Runge-Kutta processes*, J. Austr. Math. Soc., Vol 3, pp. 202-206, (1963).
- [4] J.C. BUTCHER, *The Numerical Analysis of Ordinary Differential Equations*, J. Wiley & Sons, (1987).
- [5] M. CALVO, J.I. MONTIJANO AND L. RANDEZ, *A new pair of embedded Runge-Kutta formulas of orders 5 and 6*, Comput. Math. Appl., 20, pp. 15-24, (1990).
- [6] M. CALVO, J.I. MONTIJANO AND L. RANDEZ, *New Continuous Extensions for the Dormand and Prince RK Method*, in Computational Ordinary Differential Equations, Eds Cash and Gladwell, Clarendon Press, Oxford, (1992).
- [7] M. CALVO, D.J. HIGHAM, J.I. MONTIJANO AND L. RANDEZ, *Global error estimation with adaptive Runge-Kutta methods*, IMA J. Num. Anal., 16, pp. 47-63, (1996).
- [8] M. CALVO, D.J. HIGHAM, J.I. MONTIJANO AND L. RANDEZ, *Stepsize selection for tolerance proportionality in explicit Runge-Kutta methods*, Advances in Comp. Math., 7, pp. 361-382, (1997).
- [9] G. DAHLQUIST, *Stability and error bounds in the numerical integration of ordinary differential equations*, Trans. of the Royal Inst. of Techn., Stockholm, Nr. 130, (1959).

- [10] J.R. DORMAND AND P.J. PRINCE, *A family of embedded Runge-Kutta formulae*, J. Comp. Appl. Math. Vol 6, pp. 19–26, (1980).
- [11] E. FEHLBERG, *Classical fifth-, sixth-, seventh-, and eighth order Runge-Kutta formulas with step size control*, NASA Technical Report 315, (1969); extract published in Computing Vol. 6, pp. 61–71, (1970).
- [12] R.D. GRIGORIEFF, *Numerik gewöhnlicher Differentialgleichungen 2.*, Teubner Studienbücher, Stuttgart, (1977).
- [13] E. HAIRER, S.P. NØRSETT AND G. WANNER, *Solving Ordinary Differential Equations I, Nonstiff problems*, Springer Berlin, Second Edition (1993).
- [14] E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations II.*, Springer Berlin, Second Edition (1996).
- [15] P. HENRICI, *Discrete variable methods in ordinary differential equations*, John Wiley & Sons, Inc., New York, (1962).
- [16] E.N. LORENZ, *On the prevalence of aperiodicity in simple systems*, Lecture Notes in Mathematics, vol 755, pp. 53–75, (1979).
- [17] P.J. PRINCE AND J.R. DORMAND, *High Order embedded Runge-Kutta formulae*, J. Comp. Appl. Math. Vol. 7, pp. 67–75, (1981).
- [18] L.F. RICHARDSON, *The deferred approach to the limit*, Phil. Trans., A, Vol 226, pp. 299–349, (1927).
- [19] C. RUNGE, *Ueber die numerische Auflösung von Differentialgleichungen*, Math. Ann., Vol 46, pp. 167–178, (1895).
- [20] J.M. SANZ-SERNA AND M.P. CALVO, *Numerical Hamiltonian Problems*, Chapman & Hall, London (1994).
- [21] L.F. SHAMPINE AND H.A. WATTS, *Solving nonstiff ordinary differential equations. The state of the art*, Appl. Math. Comput. Vol 5, pp. 93–121, (1979).
- [22] L.F. SHAMPINE, H.A. WATTS AND S.M. DAVENPORT, *The art of writing a Runge-Kutta code II*, SIAM Rev., Vol 18, pp. 376–410, (1976).
- [23] J.H. VERNER, *Some Runge-Kutta formula pairs*, SIAM J. Num. Anal. 28, pp. 496–511, (1991).

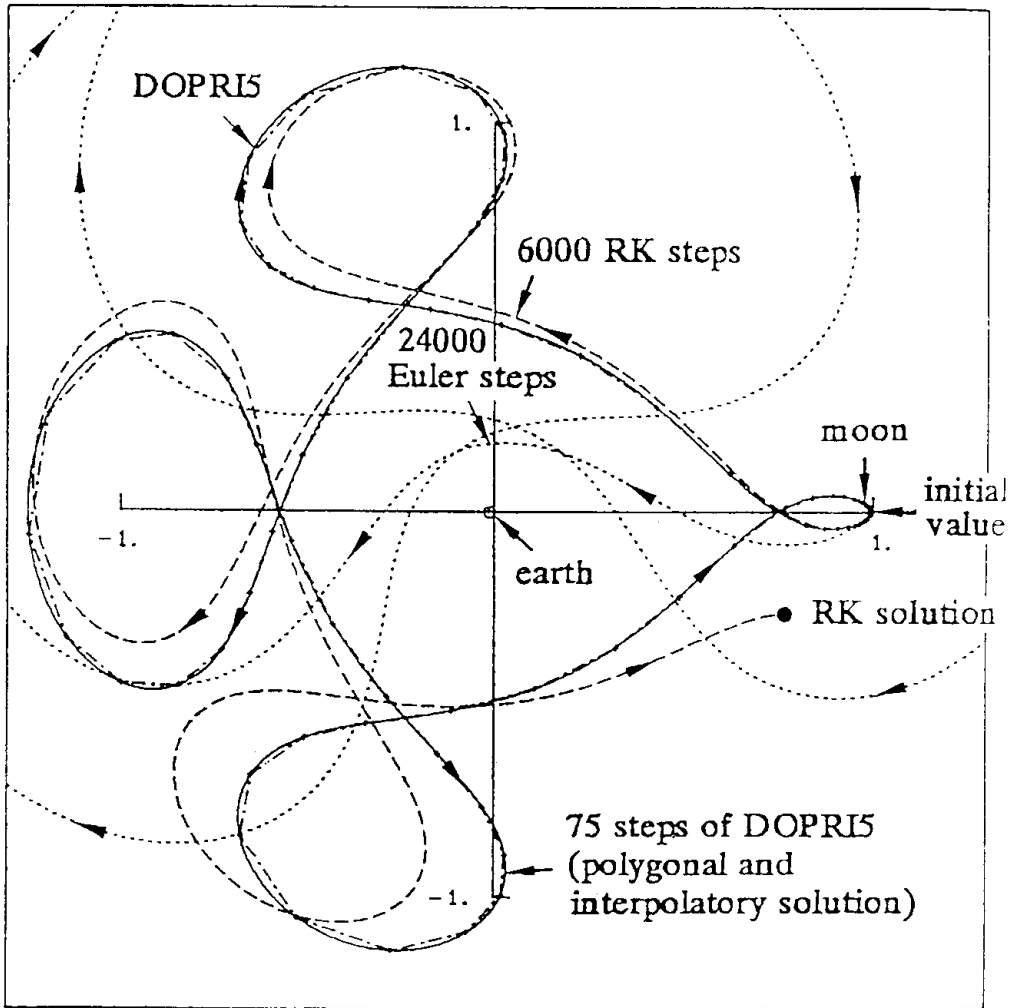


FIGURA 1