

GENERALIZACIÓN A \mathbb{R}^n DE ALGUNOS MÉTODOS DE INTERPOLACIÓN CONOCIDOS EN ECUACIONES DIFERENCIALES

VERNOR ARGUEDAS TROYO* — ROBERTO MATA MONTERO**

Recibido: 9 Octubre 1997 / Versión revisada: 11 Mayo 1998

Resumen

Presentamos el método de integración Runge-Kutta, de varios niveles de un paso en \mathbb{R}^n , así como cuasi algoritmos para los métodos implícito y explícito. Se estudia el problema del control del error en \mathbb{R}^n y se dan ejemplos de métodos numéricos mediante su respectiva tabla o esquema paramétrico.

Palabras clave: Métodos de Runge-Kutta n -dimensionales, explícitos e implícitos, interpolación, cuasi algoritmos, esquema paramétrico.

Abstract

We present the Runge-Kutta method of several one-step levels in \mathbb{R}^n . as well as algorithms in pseudo-code for the implicit and explicit methods. We study the problem of error control in \mathbb{R}^n and we give numerical examples in tables or parameter schemata.

Keywords: n -dimensional Runge-Kutta methods, explicit and implicit methods, interpolation, pseudo-algorithms, parametric schema.

AMS Subject Classification: 34-04, 65L06

1. Introducción

En los últimos años ha habido una tendencia, a la búsqueda de métodos numéricos, que permitan una alta precisión o que reduzcan el tiempo de máquina, vía simplificación de los algoritmos.

*Escuela de Matemática, Universidad de Costa Rica, 2060 San José, Costa Rica

**Sede Regional de Guanacaste, Universidad de Costa Rica, Liberia, Costa Rica

En este sentido, la teoría de Runge-Kutta ha tenido un desarrollo muy grande; así como los métodos de extrapolación, interpolación y mixtos.

Es importante saber diferenciar los tipos de interpolación mencionados en libros y artículos y que pueden ser clasificados en:

- i) Interpolación estándar o finita, como la de Newton, Hermite y otros.
- ii) Interpolación finita con polinomios mónicos.
- iii) La interpolación con polinomios mónicos de grado mínimo.

2. Fundamentos teóricos

Un método Runge-Kutta, en forma matricial, es descrito de la siguiente manera:

Sean $f : [a, b] \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ y $y : [a, b] \rightarrow \mathbb{R}^n$, donde f posee $(p+1)$ derivadas parciales continuamente diferenciables y:

$$f = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} \quad x \in \mathbb{R}^n, x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \quad y'(t) = \begin{pmatrix} y'_1(t) \\ y'_2(t) \\ \vdots \\ y'_n(t) \end{pmatrix} \quad (1)$$

Emplearemos la norma:

$$\|x\|_\infty = \sup |x_i|, \text{ para } i = 1, \dots, n$$

Dado el problema de valor inicial:

$$\begin{aligned} y'(x) &= f(x, y(x)), \\ y(x_0) &= \eta_0, \eta_0 \in \mathbb{R}^n, \end{aligned}$$

con:

$$\begin{aligned} y_{n+\alpha} &:= y(x_n + \alpha h) \in \mathbb{R}^n; \\ x_n &:= a + hn \leq b - h; \end{aligned}$$

Definimos, de acuerdo con la literatura usual, el método de integración de un paso y p niveles, el cual puede ser implícito o explícito, de la forma [1]:

$$\begin{aligned} \eta_0 &= \eta_0(h) \text{ (usualmente } y_0 = \eta_0 \text{)} \\ \eta_{n+\alpha_i} &= \eta_n + h \sum_{j=1}^p \beta_{ij} f(x_n + \alpha_j h, \eta_{n+\alpha_j}) ; i = 1, \dots, p \\ \eta_{n+\alpha_{s+1}} &= \eta_n + h \sum_{j=1}^p \gamma_j f(x_n + \alpha_j h, \eta_{n+\alpha_j}) ; n = 0, 1, \dots \end{aligned}$$

con α_j, β_{ij} y las γ_j constantes del método, y con:

$$\alpha_i = \sum_{j=1}^p \beta_{ij}, \quad i = 1, \dots, p;$$

$$\alpha_{p+1} = \sum_{i=1}^p \gamma_i$$

Donde $y_{n+\alpha_i}$ es la solución exacta y $\eta_{n+\alpha_i}$ es la solución aproximada.

Esta notación es la que usamos en [2].

Observemos que el usar $\| \cdot \|_\infty$, nos permite trabajar con la función $f : [a, b] \rightarrow \mathbb{R}^n$, de una manera muy sencilla, usando la función componente. De paso digamos que estamos usando n veces el método Runge-Kutta; y la tolerancia, así como la aceptación del $h > 0$, es para todo el sistema de ecuaciones diferenciales.

El método de integración anterior, puede ser representado por el siguiente esquema paramétrico.

α_1	β_{11}				
α_2	β_{21}	β_{22}			
\vdots	\vdots	\vdots			
α_p	β_{p1}	β_{p2}	β_{p3}	\cdots	β_{pp}
α_{p+1}	γ_1	γ_2	γ_3	\cdots	γ_p

3. Cuasi algoritmos

A fin de lograr una mayor comprensión, del método de integración de varios niveles de un paso y de facilitar su empleo, presentamos a continuación, cuasi algoritmos para los métodos implícito y explícito.

3.1. Cuasi algoritmo para el método implícito [10]

Calcula la solución aproximada al problema de valor inicial:

$y'(x_n) = f(x_n, y(x_n))$, con $y(x_n) \in V$ dado, V espacio vectorial de dimensión finita; para $f : [a, b] \times V \rightarrow V$, con h variable, $h > 0$ y una tolerancia $\epsilon > 0$.

for $i = 1$ thru p

$\eta_{n+\alpha_i} \leftarrow 0$

$\alpha_i = \sum_{j=1}^p \beta_{ij}$

$\eta_{n+\alpha_i} \leftarrow \eta_n + h \sum_{j=1}^p \beta_{ij} f(x_n + \alpha_j h, \eta_{n+\alpha_j})$

{Observemos que el método se llama a sí mismo. Lo que estamos es buscando un “cero inteligente”, lo cual haremos por algún método numérico como gradiente descendiente, Newton, punto fijo, despeje, o alguna modificación de esos métodos.}

repeat.

{ Tomar $\alpha_{p+1} = \sum_{i=1}^p \gamma_i$ }

for $n = 0$ thru p

$\eta_{n+\alpha_{p+1}} \leftarrow 0$

$\eta_{n+\alpha_{p+1}} \leftarrow \eta_n + h \sum_{j=1}^p \gamma_j f(x_n + \alpha_j h, \eta_{n+\alpha_j})$

repeat.

end algoritmo.

En [2] realizamos un trabajo de este tipo, cuando, para el esquema trapezoidal implícito en una dimensión:

$$\eta_i(x_i + h) = \eta_i(x_i) + \frac{h}{2} [f(x_i, \eta_i(x_i)) + f(x_i + h, \eta_i(x_i + h))],$$

tomamos su equivalente:

$$\eta_i(x_i + h) - \frac{h}{2} \cdot f(x_i + h, \eta_i(x_i + h)) = \eta_i(x_i) + \frac{h}{2} \cdot f(x_i, \eta_i(x_i)) \quad (2)$$

donde:

$$\eta_i(x_i) + \frac{h}{2} \cdot f(x_i, \eta_i(x_i)) := \beta \in \mathbb{R}$$

y definimos:

$$\varphi(u) := u - \frac{h}{2} \cdot f(x_i + h, u)$$

Luego, si $\varphi(u) = \beta$ admite una solución única y el sistema no lineal es computacionalmente estable y fácil de resolver, ponemos $u := \eta_i(x_i + h)$ y retornamos a (2).

Aquí, la idea del método es ejemplarizada con:

$$\begin{aligned} f(x_i, y(x_i)) &= x_i \cdot \text{sen } y(x_i) \\ y'(x_i) &= x_i \cdot \text{sen } y(x_i), \text{ con } y(0) = 1 \text{ y } h = 0,01 \\ x &= (x_1, \dots, x_n) \end{aligned}$$

Para $x_i \in [0, 1,5]$, con $i = 1, \dots, n$. (por satisfacer f las condiciones de Lipshitz en este intervalo).

Al aplicar la regla del trapecio, obtenemos:

$$\begin{aligned} \eta_i(x_i + h) - \frac{h}{2} [(x_i + h) \cdot \text{sen}(\eta_i(x_i + h))] &= \eta_i(x_i) + \frac{h}{2} x_i \cdot \text{sen}(\eta_i(x_i)) \\ \eta_i(x_i + h) - \frac{10^{-2}}{2} \cdot \text{sen}(\eta_i(x_i + h)) &= 1 + \frac{0,1}{2} \cdot 0 = 1 \\ u - \frac{10^{-2}}{2} \cdot \text{sen}(u) - 1 &= 0 \end{aligned}$$

y, con la ayuda de MATLAB, queda $u = 1$.

3.2. Cuasi algoritmo para el método explícito [10]

Calcula la solución aproximada al problema de valor inicial:
 $y'(x_n) = f(x_n, y(x_n))$, con $y(x_n) \in V$, para $f : [a, b] \times V \rightarrow V$, con h variable, $h > 0$ y una tolerancia $\epsilon > 0$.

{ Tomar $y_{n+1} = 0$ }

for $i = 1$ thru p

$x_{n,i} \leftarrow 0$

$y_{n,i} \leftarrow 0$

$y_i \leftarrow 0$

repeat.

{ Tomar $f_0 = f(x_n, y_n)$ }

for $j = 1$ thru p

$x_{n,j} \leftarrow x_n + \alpha_j h$

$y_0 \leftarrow h\gamma_0 f_0$

$y_{n,j} \leftarrow y_n + h \sum_{k=0}^{j-1} \beta_{jk} f_k$

$f_j \leftarrow f(x_{n,j}, y_{n,j})$

$y_j \leftarrow y_{j-1} + h\gamma_j f_j$

repeat.

$y_{n+1} \leftarrow y_n + y_p$

end algoritmo.

Un ejemplo muy ilustrativo de los métodos explícitos, es el método Runge-Kutta-Fehlberg, que fuera presentado en 1970 por Fehlberg [5]. El mismo es generalizado para \mathbb{R}^n , considerando (1), para:

$$k_1 = \begin{pmatrix} k_{11} \\ k_{12} \\ \vdots \\ k_{1n} \end{pmatrix} \quad \dots \quad k_n = \begin{pmatrix} k_{n1} \\ k_{n2} \\ \vdots \\ k_{nn} \end{pmatrix}$$

con lo que se obtiene la técnica de usar el método de Runge-Kutta, con error de truncamiento local de orden cinco:

$$\hat{y}_{n+1} = y_n + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6$$

para estimar el error local en el método de Runge-Kutta de orden cuatro:

$$y_{n+1} = y_n + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5$$

donde:

$$\begin{aligned}
 k_1 &= hf(x_n, y_n) \\
 k_2 &= hf\left(x_n + \frac{h}{4}, y_n + \frac{1}{4}k_1\right) \\
 k_3 &= hf\left(x_n + \frac{3h}{8}, y_n + \frac{3}{32}k_1 + \frac{9}{32}k_2\right) \\
 k_4 &= hf\left(x_n + \frac{12h}{13}, y_n + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\right) \\
 k_5 &= hf\left(x_n + h, y_n + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\right) \\
 k_6 &= hf\left(x_n + \frac{h}{2}, y_n - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\right)
 \end{aligned}$$

teniendo presente que:

$$hf(x_n, y_n) = \begin{pmatrix} hf_1(x_n, y_n) \\ hf_2(x_n, y_n) \\ \vdots \\ hf_n(x_n, y_n) \end{pmatrix}$$

y así sucesivamente para las demás expresiones.

Una ventaja clara de este método es que sólo se requieren seis evaluaciones de f por paso, mientras que los métodos Runge-Kutta arbitrarios de orden cuatro y cinco usados conjuntamente, requerirían diez evaluaciones de f por paso, cuatro evaluaciones para el método de orden cuatro y seis evaluaciones adicionales para el método de orden quinto.

El siguiente algoritmo usa el método de Runge-Kutta-Fehlberg con control de error, generalizado para \mathbb{R}^n . El paso 9 se añade para eliminar modificaciones grandes en el tamaño de paso. Esto se hace para evitar las pérdidas de tiempo ocasionadas por los tamaños de paso muy pequeños, en regiones con irregularidades en la derivada de y , y para evitar tamaños de paso grandes, que puedan resultar en la omisión de regiones sensibles cercanas. En algunos casos el procedimiento del incremento del tamaño de paso se omite del algoritmo y el procedimiento del decrecimiento del tamaño de paso se modifica para que se incorpore solamente cuando sea necesario tener al error bajo control.

Algoritmo de Runge-Kutta-Fehlberg:

{Para aproximar la solución del problema de valor inicial $y'(x_n) = f(x_n, y(x_n))$, donde $y(x_n) \in \mathbb{R}^n$, $a \leq x_n \leq b$, $y(a) = \alpha$, con $\alpha = (\alpha_1, \dots, \alpha_n)$, $y' : [a, b] \rightarrow \mathbb{R}^n$, $h \in \mathbb{R}^n$ y la norma $\| \cdot \|_\infty$; con el error de truncamiento local dentro de cierta tolerancia dada.}

ENTRADA puntos extremos a,b; condición inicial α ; tolerancia TOL; tamaño de paso máximo hmáx; tamaño de paso mínimo hmín.

SALIDA x , y , h , donde y aproxima a $y(t)$. Se emplea la función f por medio de las componentes f_1, \dots, f_n y se usa el tamaño de paso h o un mensaje de que el tamaño de paso mínimo fue excedido.

Paso 1 Tomar:

$$\begin{aligned}x &= a \\y &= \alpha \\h &= \text{hm}áx;\end{aligned}$$

SALIDA (x, y) .

Paso 2 Mientras que $(x < b)$ seguir los Pasos 3 - 11.

Paso 3 Tomar:

$$\begin{aligned}k_1 &= hf(x, y) \quad (\text{se calcula para cada } f_i, \text{ con } i = 1, \dots, n); \\k_2 &= hf(x + \frac{1}{4}h, y + \frac{1}{4}k_1); \\k_3 &= hf(x + \frac{3}{8}h, y + \frac{3}{32}k_1 + \frac{9}{32}k_2); \\k_4 &= hf(x + \frac{12}{13}h, y + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3); \\k_5 &= hf(x + h, y + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4); \\k_6 &= hf(x + \frac{1}{2}h, y - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5).\end{aligned}$$

Paso 4 Tomar:

$$R = \left\| \frac{1}{360}k_1 - \frac{128}{4275}k_3 - \frac{2197}{75240}k_4 + \frac{1}{50}k_5 + \frac{2}{55}k_6 \right\|_{\infty} / h \quad (\text{se escoge } R = O(h), \text{ donde: } \\O(h) = \left\| \hat{y}_{n+1} - y_{n+1} \right\|_{\infty} / h, \text{ el cual es explicado posteriormente.})$$

Paso 5 Tomar $\delta = 0,84(TOL/R)^{1/4}$.

Paso 6 Si $R \leq TOL$ entonces seguir los pasos 7 y 8.

Paso 7 Tomar:

$$\begin{aligned}x &= x + h; \text{ (aproximaci3n aceptada).} \\y &= y + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5.\end{aligned}$$

Paso 8 SALIDA (x, y, h) .

Paso 9 Si $\delta \leq 0,1$ entonces tomar $h = 0,1h$

si no; si $\delta \geq 4$ entonces tomar $h = 4h$

si no tomar $h = \delta h$.

Paso 10 Si $h > \text{hm}áx$ entonces tomar $h = \text{hm}áx$.

Paso 11 Si $h < \text{hm}ín$ entonces

SALIDA (“h m3nimo excedido”);

(Procedimiento terminado sin 3xito).

PARAR.

Paso 12 (El procedimiento se complet3). [En la norma $\| \quad \|_{\infty}$]

PARAR.

Observemos que en el proceso anterior se est3n manipulando los vectores de la manera usual.

4. Control del error

A continuación, algunas definiciones que nos van a permitir una mayor comprensión, de las apreciaciones que haremos sobre el control del error.

Si se tiene el problema de valor inicial:

$$\begin{aligned} y'(x) &= f(x, y(x)) \\ y(x_0) &= y_0 \\ \text{con } f &: [a, b] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n, \\ &\text{como en (1)} \end{aligned} \tag{3}$$

y tenemos el método de un paso, para n dimensiones:

$$\begin{aligned} \eta_0 &:= y_0 \\ \text{para } i &= 1, \dots \\ \eta_{i+1} &:= \eta_i + h\phi(x_i, \eta_i; h) \\ x_{i+1} &:= x_i + h \end{aligned}$$

donde ϕ es la función generatriz del método, con:

$$x \in R_h := \{x_0 + ih/i = 1, 2, \dots\}$$

para el método $\eta(x; h)$, definido por $\eta(x; h) := \eta_i$, cuando $x = x_0 + ih$

Llamamos *error local de truncamiento del método* η_{i+1} , a:

$$\left\| \frac{y(x_i + h) - y(x_i)}{h} - \eta_i(x_i) \right\|_\infty$$

El método η se llama de orden $p > 0$, si su error de truncamiento local satisface:

$$\left\| \frac{y(x_i + h) - y(x_i)}{h} - \eta_i(x_i) \right\|_\infty \leq O(h^p), \forall f \in C^p$$

Observemos que y denota a la solución exacta de (3).

El *error global en el punto x , para el método η* , se define como: [3]

$$e(x; h) := \eta(x; h) - y(x)$$

Si definimos:

$$h_n \in H_x := \left\{ \frac{x - x_0}{n} / n = 1, 2, \dots \right\}$$

tendremos que el método η es *convergente* si:

$$\lim_{n \rightarrow \infty} e(x; h_n) = 0, \text{ para todo } x \in [a, b]$$

Un teorema clásico garantiza que un método de orden p es convergente de orden p , cuando ϕ satisface las condiciones de Lipschitz, con respecto a y y y' [3].

Un análisis del error nos conduce a las siguientes consideraciones:
dado el problema de valor inicial:

$$\begin{aligned} y'(x_n) &= f(x_n, y(x_n)) \\ \text{con } f &: [a, b] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n \\ a \leq x_n &\leq b, \text{ con } y(a) = \alpha, \end{aligned}$$

haremos una estimación del error local para el método de Euler:

$$\begin{aligned} y_0 &= \alpha, \\ y_{n+1} &= y_n + hf(x_n, y_n), \end{aligned}$$

el cual tiene error de truncamiento local de orden $O(h)$ dado por:

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

Ahora bien, el método de Euler modificado,

$$\begin{aligned} \hat{\eta}_0 &= \alpha \\ \hat{\eta}_{n+1} &= \hat{\eta}_n + \frac{h}{2}[f(x_n, \hat{\eta}_n) + f(x_{n+1}, \hat{\eta}_n + hf(x_n, \hat{\eta}_n))], \end{aligned}$$

tendrá error de truncamiento local $\hat{\tau}_{n+1}$ de orden $O(h^2)$.

Si $\eta_n \approx y(x_n) \approx \hat{\eta}_n$, entonces:

$$\begin{aligned} y(x_{n+1}) - \eta_{n+1} &= y(x_{n+1}) - \eta_n - hf(x_n, \eta_n) \\ &\approx y(x_{n+1}) - y(x_n) - hf(x_n, y(x_n)) \\ &= h\tau_{n+1} \end{aligned}$$

Así que:

$$\begin{aligned} \tau_{n+1} &\approx \frac{1}{h}[y(x_{n+1}) - \eta_{n+1}] \\ &= \frac{1}{h}[y(x_{n+1}) - \hat{\eta}_{n+1}] + \frac{1}{h}(\hat{\eta}_{n+1} - \eta_{n+1}) \\ &\approx \hat{\tau}_{n+1} + \frac{1}{h}(\hat{\eta}_{n+1} - \eta_{n+1}) \end{aligned}$$

Puesto que τ_{n+1} es $O(h)$ y $\hat{\tau}_{n+1} = O(h^2)$, la parte más importante de τ_{n+1} se debe a $(\hat{\eta}_{n+1} - \eta_{n+1})/h$, con lo que concluimos que :

$$\tau_{n+1} \approx \frac{1}{h}(\hat{\eta}_{n+1} - \eta_{n+1})$$

se puede usar para aproximar el error de truncamiento local del método de Euler. Esta es la idea subyacente, en la búsqueda de pares de métodos ϕ, τ , en donde uno es de orden n y el otro es de orden $n + 1$. (Casualmente, esto es lo que se utiliza en el algoritmo de Runge-Kutta-Fehlberg.)

La estimación del error de truncamiento local de un método de diferencia puede usarse ventajosamente para aproximar el tamaño de paso óptimo “ n ”, para controlar el error global, como veremos a continuación.

Supongamos que hay dos métodos canónicos de diferencia disponibles, para aproximar la solución del problema de valor inicial:

$$\begin{aligned} y'(x_n) &= f(x_n, y(x_n)) \\ \text{con } f &: [a, b] \times \mathbb{R}^n \longrightarrow \mathbb{R}^n \\ a \leq x_n &\leq b, \text{ con } y(a) = \alpha \end{aligned}$$

y que uno de estos métodos,

$$\begin{aligned} \eta_0 &= \alpha \\ \eta_{n+1} &= \eta_n + h_n \phi(x_n, h_n, \eta_n), \end{aligned}$$

tiene un error de truncamiento local τ_{n+1} de orden $O(h^n)$, mientras que el otro método:

$$\begin{aligned} \hat{\eta}_0 &= \alpha \\ \hat{\eta}_{n+1} &= \hat{\eta}_n + h_n \hat{\phi}(x_n, h_n, \hat{\eta}_n) \end{aligned}$$

tiene un error de truncamiento local $\hat{\tau}_{n+1}$ de orden $O(h^{n+1})$.

El mismo análisis que se usó para el método de Euler, nos permite obtener:

$$\tau_{n+1} \approx \frac{1}{h} (\hat{\eta}_{n+1} - \eta_{n+1})$$

Sin embargo, τ_{n+1} es de orden $O(h^n)$; así que existe una constante k tal que:

$$\tau_{n+1} \approx kh^n.$$

Luego, de las relaciones anteriores se tiene que:

$$kh^n \approx \frac{1}{h} (\hat{\eta}_{n+1} - \eta_{n+1}).$$

Ahora consideremos el error de truncamiento, reemplazando a h por qh , donde q es positivo pero acotado superiormente y no cerca de cero. Si denotamos este error de truncamiento como $\tau_{n+1}(qh)$, entonces:

$$\tau_{n+1}(qh) \approx k(qh)^n = q^n(kh^n) \approx \frac{q^n}{h} (\hat{\eta}_{n+1} - \eta_{n+1}).$$

Para acotar $\tau_{n+1}(qh)$ por ϵ , escojamos q tal que:

$$\frac{q^n}{h} |\hat{\eta}_{n+1} - \eta_{n+1}| \approx |\tau_{n+1}(qh)| \leq \epsilon;$$

es decir, tal que:

$$q \leq \left(\frac{\epsilon h}{|\hat{\eta}_{n+1} - \eta_{n+1}|} \right)^{1/n}.$$

Puesto que el usuario especifica una norma $\| \cdot \|$ y una tolerancia de error τ , la búsqueda de h termina cuando formemos una aproximación que denominamos “est”, para el error local, y que cumpla que:

$$\| est \| \leq \tau$$

Si el “est” no es aprobado, reducimos el valor de h . Si es aprobado, entonces y_{n+1} es aceptado y se escoge un h conveniente para el próximo paso.

5. Aplicaciones

Como ejemplos de aplicaciones, presentamos a continuación el esquema paramétrico de algunos métodos de uno y de varios pasos; todos en n dimensiones.

5.1. Método Runge-Kutta de cuarto orden, de un paso

Este método queda definido por la siguiente tabla:

$$\begin{array}{c|ccc} 1/2 & 1/2 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 \\ 1 & 0 & 0 & 1 \\ \hline 5/6 & 1/3 & 1/3 & 1/6 \end{array}$$

con $c_0 = 1/6$

5.2. Método Runge-Kutta de orden tres, de un paso

Su esquema paramétrico es el siguiente:

$$\begin{array}{c|cc} 1/2 & 1/2 & 0 \\ 1 & -1 & 2 \\ \hline -1/2 & -2/3 & 1/6 \end{array}$$

con $c_0 = 1/6$

5.3. Método de Heun

Este es un método Runge-Kutta de orden dos, dado mediante el arreglo:

$$\begin{array}{c|c} 2/3 & 2/3 \\ \hline 3/4 & 3/4 \end{array}$$

donde $c_0 = 1/4$.

5.4. Método Adams-Bashforth de tres pasos

Este método tiene un error de truncamiento local $\tau_{n+1} = \frac{3}{8}y^{(4)}(\mu_n)h^3$, y es dado por la tabla:

1	1	
2	2	0
$\frac{7}{12}$	$-\frac{16}{12}$	$\frac{23}{12}$

con $c_0 = \frac{5}{12}$.

5.5. Método Runge-Kutta-Ralston

Este es un método Runge-Kutta explícito de orden cuatro, con error de truncamiento de orden cinco, dado por:

0.4	0.4		
0.45573726	0.29697760	0.15875966	
1	0.21810038	-3.05096470	3.83286432
0.82523972	-0.55148053	1.20553547	0.17118478

con $c_0 = 0,17476028$.

Podríamos seguir ampliando la lista de esquemas; sin embargo, queremos terminar brindando los siguientes ejemplos obtenidos de [6], y que fueron dados a conocer inicialmente, en [7] y [8].

5.6. El esquema regular

$1/2 - \sqrt{3}/6$	$1/4$	$1/4 - \sqrt{3}/6$
$1/2 + \sqrt{3}/6$	$1/4 + \sqrt{3}/6$	$1/4$
	$1/2$	$1/2$

Otro esquema regular es dado por:

0	0	0	0
1/2	5/24	1/3	-1/24
1	1/6	2/3	1/6
	1/6	2/3	1/6

5.7. El esquema irregular

$1/2 - \sqrt{15}/10$	5/36	$2/9 - \sqrt{15}/15$	$5/36 - \sqrt{15}/30$
1/2	$5/36 + \sqrt{15}/24$	2/9	$5/36 - \sqrt{15}/24$
$1/2 + \sqrt{15}/10$	$5/36 + \sqrt{15}/30$	$2/9 + \sqrt{15}/15$	5/36
	5/18	4/9	5/18

con:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1/2 & 1/2 & 0 \\ \hline & 0 & 1 \end{array}$$

6. Conclusiones

La generalización de los métodos Runge-Kutta a \mathbb{R}^n fue canónica, así como el análisis de la propagación del error. En el estudio de los métodos implícitos, debemos continuar investigando, pues queda mucho por hacer al respecto.

Referencias

- [1] Albrecht, P. (1987) “A new theoretical approach to Runge-Kutta methods”, *SIAM Journal Numerical Analysis* **24**(2).
- [2] Arguedas, V.; Mata, R. (1992) “Un teorema sobre el orden de los métodos de Runge-Kutta”, *Ciencias Matemáticas*, U.C.R. **3**(1).
- [3] Burden, R.; Faires, D. (1985) *Análisis Numérico*. Editorial Iberoamérica, México.
- [4] Collatz, L. (1960) *The Numerical Treatment of Differential Equations*. Springer-Verlag, Heidelberg.
- [5] Fehlberg, E. (1970) “Klassische Runge-Kutta Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungs-probleme”, *Computing* **6**: 61–71.
- [6] Hairer, E.; Iserles, A.; Sanz-Serna, J.M. (1990) “Equilibria of Runge-Kutta methods”, *Numerische Mathematik* **58**: 243–254.
- [7] Kansy, K. (1973) “Elementare Fehlerdarstellung für Ableitungen bei der Hermite-Interpolation”, *Numer. Math.* **21**: 350–354.
- [8] Mata, R. (1990) *Algunos Aspectos Teóricos de los Métodos Runge-Kutta*. Tesis de Licenciatura, Universidad de Costa Rica.
- [9] Press, W.; Flannery, B.; Teukolsky, S.; Vetterling, W. (1987) *Numerical Recipes*, USA.
- [10] Press, W.; Flannery, B.; Teukolsky, S.; Vetterling, W. (1987) *Numerical Recipes, Example Book (Fortran)*, USA.
- [11] Shampine, L.F.; Gordon, M.K. (1975) *Computer Solution of Ordinary Differential Equations: the Initial Value Problem*. W. H. Freeman, San Francisco.
- [12] Shampine, L.F.; Watts, H.A. (1976) “Practical solution of ordinary differential equations by Runge-Kutta methods”, Rept. SAND 76-0585, Sandía National Laboratories, Albuquerque, N. M.

- [13] Shampine, L. F. (1985) “Interpolation for Runge-Kutta methods”, *SIAM Journal Numerical Analysis* **22**(5).
- [14] Stoer, J.; Burlisch, R. (1990) *Numerische Mathematik 2*. Tercera edición, Springer-Verlag, Berlin.
- [15] Werner, H.; Schabach, R. (1972) *Praktische Mathematik II*. Springer-Verlag, Berlin.