Mecánica Computacional Vol. XVI
compilado por G. Etse y B. Luccioni
Tucumán, Septiembre de 1996

LOBATTO IMPLICIT SIXTH ORDER RUNGE-KUTTA METHOD FOR SOLVING ORDINARY DIFFERENTIAL EQUATIONS WITH STEPSIZE CONTROL

Andrés L. Granados M.
Universidad Simón Bolívar, Dpto. Mecánica.
Apdo. 89000, Caracas 1080A. Venezuela

RESUMEN

En este trabajo se ha desarrollado un método para resolver ecuaciones diferenciales ordinarias usando métodos Runge-Kutta implícitos. Los métodos Runge-Kutta aquí empleados están basados en dos cuadraturas del tipo Lobatto. La primera cuadratura conforma el método Runge-Kutta principal que es de sexto orden; la segunda cuadratura conforma un método de Runge-Kutta de tercer orden y está acoplado con el primero. Ambos métodos de Runge-Kutta implícitos constituyen la forma acoplada de Lobatto de tercer y sexto orden con cuatro etapas. La ventaja más importante de este método es que es implícito sólo en la segunda y tercera etapa, lo que reduce considerablemente los costos de cálculo en la computadora, dentro de sus pocas etapas. La notación de Butcher es usada para el análisis de los métodos tratados aquí. Con la finalidad de resolver en cada paso el sistema de ecuaciones no lineales originado en las variables $\mathbf{k}_2$ y $\mathbf{k}_3$, un método de Runge-Kutta explícito de cuatro etapas y cuarto orden es definido para los mismo puntos intermedios, como en el método de Runge-Kutta implícito de sexto orden. Este método explícito estima los valores iniciales de las mencionadas variables auxiliares, y, luego, un método iterativo del tipo "punto fijo" es usado para resolver el sistema de ecuaciones no lineales en cada paso. Con el método de Runge-Kutta de tercer orden puede ser calculado una estimación del error local de truncamiento, comparándolo con el método de sexto orden. Esto es usado para controlar el tamaño del paso cuando las tolerancias para los errores relativo y global absoluto son especificados. Se presenta un algoritmo para realizar este control de paso automáticamente. El método implícito, tal como se expone aquí, es realmente útil y ha demostrado ser eficiente para resolver sistemas de ecuaciones diferenciales ordinarias rígidas y de gran tamaño. Finalmente, son elaborados los criterios de convergencia y los análisis de estabilidad para los métodos Runge-Kutta implícitos aquí presentados.

ABSTRACT

A method for solving ordinary differential equations has been developed using implicit Runge-Kutta methods. The implicit Runge-Kutta methods used are based in two quadratures of Lobatto type. The first quadrature produces the principal Runge-kutta method which is of sixth order, while the second quadrature produces a Runge-Kutta method of third order which is embedded in the former. Both implicit Runge-Kutta methods constitute the Lobatto embedding form of third and sixth orders with four stages. The most important advantage of this method is that it is implicit only in the second and third stages, which reduces considerably the costs of computer calculations. The Butcher notation is used here for the analysis of the studied methods. In order to solve, for each step, the system of non-linear equations in the implicit auxiliary variables $\mathbf{k}_2$ and $\mathbf{k}_3$, an explicit Runge-Kutta method of four stages and fourth order is defined for the same intermediate points, such as the implicit sixth order Runge-Kutta method. This explicit method estimates the inicial values for the aforementioned auxiliary variables, and then, an iterative method of the type "fixed point" is used to solve the system of non-linear equations for each step. With the third order Runge-Kutta method, an estimation of the local truncation error may be calculated using a comparison with the sixth order method. This aspect is used to control the step size when tolerances for the relative and absolute global errors are specified. An algorithm is presented to do this step control automatically. The implicit method, as is exposed here, is really useful and has demonstrated to be efficient to solve huge and stiff systems of ordinary differential equations. Finally, convergence criteria and stability analysis are studied for the Runge-Kutta methods presented here.

INTRODUCTION

The principal aim of this work is the development of an algorithm for solving ordinary differential equations based on known implicit Runge-Kutta methods but where the selection of the methods and the application of iterative procedures have been accurately studied in order to produce the least number of numerical calculations and the

highest order of exactitude with a reasonable stability.

As it is well known, the implicit Runge-Kutta methods are more stable than explicit ones. However, the solution of the system of non-linear equations with the auxiliary variables "k", produced by the implicit dependence, generates an iterative process that can be extremely slow and can produce a restriction in the stepsize. This restriction can be overcome by the selection of a special implicit Runge-Kutta method based on the Lobatto quadrature. This method is implicit only in the second and the third stages. This characteristic makes the iterative process less restrictive respective to the size of the step, but, at the same time, reduces considerably the number of numerical calculations. This method conforms the principal implicit Runge-Kutta method.

In order to improve the speed of the iterative process, a good initial value for each auxiliary variable k is estimated by an algorithm using a fourth order, four stages explicit Runge-Kutta method. This method is new and its best characteristic is that the auxiliary variables are evaluated in the same intermediate points as in the principal implicit Runge-Kutta method of sixth order and four stages.

Additionally to the aforementioned two aspects, a third order, three stages implicit Runge-Kutta has been selected to control the step size in each jump of the method. This method is also based on the Lobatto quadrature and its best characteristic is that it is embedded in the principal implicit Runge-Kutta method. This makes the algorithm more efficient for the stepsize control purpose, and, as a consecuence, calculations necessary for solving the ordinary differential equations are the same to those necessary to control the size of the step.

Finally, convergence criteria for the iterative process and stability analysis are made for all the Runge-Kutta methods presented here.

IMPLICIT RUNGE-KUTTA METHODS

DIFFERENTIAL EQUATIONS

As it is well known, every system of ordinary differential equations of any order may be transformed, with a convenient change of variables, into a system of first order ordinary differential equations [1,2]. This is the reason why only this last type of differential equations will be studied.

Let the following system of M first order ordinary differential equations be expressed as $dy^i/dx = f^i(x,\mathbf{y})$ with $i = 1,2,3,\ldots,M$, being $\mathbf{y}$ a M-dimensional function with each component depending on x. This may be symbolically expressed as

$$\frac{d\mathbf{y}}{dx} = \mathbf{f}(x,\mathbf{y}) \qquad \text{where} \qquad \mathbf{y} = \mathbf{y}(x) = (y^1(x), y^2(x), y^3(x), \ldots, y^M(x)) \tag{1}$$

When each function $f^i(x,\mathbf{y})$ depends only on each y^i the system is said to be *uncoupled*, otherwise it is said to be *coupled*. If the system of ordinary differential equations is uncoupled then every differential equation can be solved separately. When the system does not explicitly depends on x the system is said to be *autonomous*. When the conditions of the solution $\mathbf{y}(x)$ are known at a unique specific point, for example, $y^i(x_o) = y^i_o$ at $x = x_o$, or symbolically

$$x = x_o \qquad \mathbf{y}(x_o) = \mathbf{y}_o \tag{2}$$

Both expressions (1) and (2) are said to state an *Initial Value Problem*, otherwise they state a *Boundary Value Problem*.

In deed, the system of differential equations (1) is a particular case of a general autonomous system stated in the next form [2,3]

$$\frac{d\mathbf{y}}{dx} = \mathbf{f}(\mathbf{y}) \equiv \begin{cases} dy^i/dx = 1 & \text{if } i = 1 \\ dy^i/dx = f^i(\mathbf{y}) & \text{if } i = 2,3,\ldots,M+1 \end{cases} \tag{3}$$

but with an additional condition $y^1_o = x_o$ in (2).

RUNGE-KUTTA METHODS

Trying to make a general formulation, a Runge-Kutta method of order P and equiped with N stages is defined [3] with the expression

$$y^i_{n+1} = y^i_n + h(c_r k^i_r) \tag{4.a}$$

where the auxiliary M-dimensional variables $\mathbf{k}_r$ are calculated by

$$k^i_r = f^i[\, x_n + b_r h \,,\, \mathbf{y}_n + h\,(a_{rs}\mathbf{k}_s)\,] \tag{4.b}$$

for $i = 1,2,3,\ldots,M$ and $r,s = 1,2,3,\ldots,N$. Notice that index convention of sum has been used. Thus every time an index appears twice or more in a term, this should be summed up to complete its range (In this context, it is not important the number of factors with the same index within each term).

A Runge-Kutta method (4) has order P if, for a sufficiently smooth problem, the expressions (1) and (2) satisfy

$$\|\mathbf{y}(x_n + h) - \mathbf{y}_{n+1}\| \leq \Phi(\zeta)\, h^{P+1} = O(h^{P+1}) \qquad \zeta \in [x_n, x_n + h], \tag{5}$$

i.e., the Taylor series for the exact solution $\mathbf{y}(x_n + h)$ and for the numerical solution $\mathbf{y}_{n+1}$ coincide up to (and include) the term with h^P [7].

The Runge-Kutta method thus defined can be applied for solving initial value problems, and it is used recurrently. Given a point $(x_n, \mathbf{y}_n)$, it can be obtained the next point $(x_{n+1}, \mathbf{y}_{n+1})$ using the expressions (4), being $x_{n+1} = x_n + h$, where h is named the *stepsize* of the method. Every time that this is made, the method goes forward (or backward if h is negative) an integration step h in x, offering the solution in consecutive points, one for each jump. In this way, if the method begins with the initial conditions $(x_0, \mathbf{y}_0)$ stated by (2), it can calculate $(x_1, \mathbf{y}_1), (x_2, \mathbf{y}_2), (x_3, \mathbf{y}_3), \ldots, (x_n, \mathbf{y}_n)$, and continue this way, up to the desired boundary in x. In each integration or jump the method reinitiates with the information from the adjacent point inmediately preceding. This characteristic clasifies the Runge-Kutta methods within the group of one step methods. It should be notice, however, that the auxiliary variables k_r^i are calculated for each r up to N stages in each step. These calculations are no more than evaluations of the functions $f^i(x, \mathbf{y})$ for intermediate points $x + b_r h$ in the interval $[x_n, x_{n+1}]$ $(0 \leq b_r \leq 1)$

Let it now be introduced a condensed representation of the generalized Runge-Kutta method, formerly developed by Butcher [4,5] and that is presented systematically in the book of Lapidus and Seinfeld [6] and in the books of Hairer, Norsett and Wanner [7,8]. This last books has a numerous collection of Runge-Kutta methods using the Butcher's notation and an extensive bibliography. After the paper of Butcher [4] it became customary to symbolize the general Runge-Kutta method (4) by a tableau. In order to illustrate this representation, consider the expresions (4) applied to a method of four stages ($N = 4$). Accomodating the coefficients a_{rs}, b_r and c_r in a adequate form, they may be schematically represented as

$$\begin{array}{c|cccc} b_1 & a_{11} & a_{12} & a_{13} & a_{14} \\ b_2 & a_{21} & a_{22} & a_{23} & a_{24} \\ b_3 & a_{31} & a_{32} & a_{33} & a_{34} \\ b_4 & a_{41} & a_{42} & a_{43} & a_{44} \\ \hline & c_1 & c_2 & c_3 & c_4 \end{array} \tag{6}$$

The aforementioned representation allows for the basic distinction of the following types of Runge-Kutta methods, according to the characteristics of the matrix a_{rs}: If $a_{rs} = 0$ for $s \geq r$, then the matrix a_{rs} is lower tringular, excluding the principal diagonal, and the method is said to be completely *explicit*. If, $a_{rs} = 0$ for $s > r$, then the matrix a_{rs} is lower triangular, including the principal diagonal, and the method is said to be *semi-implicit* or simple-diagonally implicit. If the matrix a_{rs} is diagonal by blocks, the method is said to be *diagonally implicit*. if the first row of the matrix a_{rs} is filled with zeros, $a_{1,s} = 0$, and the method is diagonally implicit, then the method is called *Lagrange Method* [9] (the coefficients b_r may be arbitrary). If a Lagrange method has $b_N = 1$ and the last row of the matrix is the array $c_s = a_{N,s}$, then the method is said to be *stiffly accurate*. If, conversely, none of the previous conditions are satisfied, the method is said to be simply *implicit*. In the cases of implicit methods, it can be noticed that an auxiliary variable $\mathbf{k}_r$ may depend on itself and on any other variables not calculated before in the same stage. That is why the methods are named implicit in these cases.

Additionally, the condense representation above described permits to verify very easily certain properties that the coefficients a_{rs}, b_r, and c_r should fulfill. These properties are

$$0 \leq b_r \leq 1 \qquad a_{rs}\,\delta_s = b_r \qquad c_r\,\delta_r = 1 \tag{7}$$

The vector δ is unitary in every component, i.e., $\delta_r = 1\ \forall r = 1,2,3,\ldots,N$. Those mentioned properties may be interpreted in the following manner: The property (7.a) expresses that the Runge-Kutta is a one step method, and the functions $f^i(x, \mathbf{y}(x))$ in (4.b) should be evaluated for $x \in [x_n, x_{n+1}]$. The property (7.b) results in applying the Runge-Kutta method (4) to the system of differential equations (3), where $k_s^1 = 1\ \forall s = 1,2,3,\ldots,N$, and thus the sum of a_{rs} in each line r offers the value of b_r. The property (7.c) means that in the expression (4.a) the value of y_{n+1}^i is obtained from the value of y_n^i adn the projectios with h an average of the derivatives $dy^i/dx = f^i(x, \mathbf{y})$, calculated using weighted coefficients. Obviously, the sum of all c_r should be equal to unit.

The coefficients a_{rs}, b_r and c_r are determined applying the properties (7) and using some relations that are deduced in the following form:

Let the system of ordinary differential equations (3) be expressed according to (2) as an initial value problem. The Runge-Kutta method applied to this problem is formulated by (4)

If now the expansion in Taylor series is made to the component k_r^i of (4.b), around the point (x_n, y_n), being particularly $y_n = y(x_n)$ in this case, the result is

$$\begin{aligned} k_r^i =& h f^i [\delta_r] + h f_j^i [a_{rs} k_s^j] + \frac{h}{2} f_{jk}^i [a_{rs} k_s^j][a_{rt} k_t^k] + \frac{h}{6} f_{jkl}^i [a_{rs} k_s^j][a_{rt} k_t^k][a_{ru} k_u^l] \\ &+ \frac{h}{24} f_{jklm}^i [a_{rs} k_s^j][a_{rt} k_t^k][a_{ru} k_u^l][a_{rv} k_v^m] + O(h^6) \end{aligned} \tag{8.a}$$

where the following notation has been used

$$f^i = f^i(x_n) \qquad f_j^i = \frac{\partial f^i}{\partial y^j}\Big|_{y_n} \qquad f_{jk}^i = \frac{\partial^2 f^i}{\partial y^j \partial y^k}\Big|_{y_n} \quad \cdots \tag{8.b}$$

Here the fuctions are supposed to be C^∞ (Analytical Functions), and therefore the subindexes in (8) are permutable.

The variable k_s^j in the second term of the right side of (8.a) may again be expanded in Taylor series as

$$k_s^j = h f^j [\delta_s] + h f_k^j [a_{s\alpha} k_\alpha^k] + \frac{h}{2} f_{kl}^j [a_{s\alpha} k_\alpha^k][a_{s\beta} k_\beta^l] + \frac{h}{6} f_{klm}^j [a_{s\alpha} k_\alpha^k][a_{s\beta} k_\beta^l][a_{s\gamma} k_\gamma^m] + O(h^5) \tag{8.c}$$

In the same way k_α^k can be expanded as

$$k_\alpha^k = h f^k [\delta_\alpha] + h f_l^k [a_{\alpha\delta} k_\delta^l] + \frac{h}{2} f_{lm}^k [a_{\alpha\epsilon} k_\delta^l][a_{\alpha\epsilon} k_\epsilon^m] + O(h^4) \tag{8.d}$$

and thus sucessively

$$k_\delta^l = h f^l [\delta_\delta] + h f_m^l [a_{\delta\varphi} k_\varphi^m] + O(h^3) \qquad \text{up to} \qquad k_\varphi^m = h f^m [\delta_\varphi] + O(h^2) \tag{8.e,f}$$

If a recurrent backward substitution is finally made, it results in a long expresión

$$\begin{aligned} k_r^i =& h f^i [\delta_r] + h^2 [f_j^i f^j b_r] + h^3 [f_j^i f_k^j f^k a_{rs} b_s + \frac{1}{2} f_{jk}^i f^j f^k b_r^2] \\ &+ h^4 [f_j^i f_k^j f_l^k f^l a_{rs} a_{st} b_t + \frac{1}{2} f_j^i f_{kl}^j f^k f^l a_{rs} b_s^2 + f_{jk}^i f_l^j f^k f^l b_r a_{rs} b_s + \frac{1}{6} f_{jkl}^i f^j f^k f^l b_r^3] \\ &+ h^5 [f_j^i f_k^j f_l^k f_m^l f^m a_{rs} a_{st} a_{tu} b_u + \frac{1}{2} f_j^i f_k^j f_{lm}^k f^l f^m a_{rs} a_{st} b_t^2 + f_j^i f_{kl}^j f_m^k f^l f^m a_{rs} b_s a_{st} b_t \\ &+ \frac{1}{6} f_j^i f_{klm}^j f^k f^l f^m a_{rs} b_s^3 + f_{jk}^i f_l^j f^k f_m^l f^m b_r a_{rs} a_{st} b_t + \frac{1}{2} f_{jk}^i f_{lm}^j f^k f^l f^m b_r a_{rs} b_s^2 \\ &+ \frac{1}{2} f_{jk}^i f_l^j f_m^k f^l f^m a_{rs} b_s a_{rt} b_t + \frac{1}{2} f_{jkl}^i f_m^j f^k f^l f^m b_r^2 a_{rs} b_s + \frac{1}{24} f_{jklm}^i f^j f^k f^l f^m b_r^4] + O(h^6) \end{aligned} \tag{8.g}$$

This expression may be inserted for the components of $\mathbf{k}_r$ into the equation (4.a), and then may be compared with the next expansion in Taylor series of y_{n+1} (around y_n)

$$\begin{aligned} y_{n+1}^i =& y_n^i + h f^i + \frac{h^2}{2}(f_j^i f^j) + \frac{h^3}{6}(f_j^i f_k^j f^k + f_{jk}^i f^j f^k) + \frac{h^4}{24}(f_j^i f_k^j f_l^k f^l + f_j^i f_{kl}^j f^k f^l + 3 f_{jk}^i f_l^j f^l f^k + f_{jkl}^i f^j f^k f^l) \\ &+ \frac{h^5}{120}(f_j^i f_k^j f_l^k f_m^l f^m + f_j^i f_k^j f_{lm}^k f^l f^m + 3 f_j^i f_{kl}^j f_m^k f^l f^m + f_j^i f_{klm}^j f^k f^l f^m + 4 f_{jk}^i f_l^j f^k f_m^l f^m \\ &+ 4 f_{jk}^i f_{lm}^j f^k f^l f^m + 3 f_{jk}^i f_l^j f_m^k f^l f^m + 6 f_{jkl}^i f_m^j f^k f^l f^m + f_{jklm}^i f^j f^k f^l f^m) + O(h^6) \end{aligned} \tag{9}$$

Thus, this gives the following relations that should be fulfilled by the coefficients a_{rs}, b_r and c_r

$$
\begin{array}{c|l}
h & c_r\delta_r = 1 \\
\hline
h^2 & c_r b_r = 1/2 \\
\hline
h^3 & \begin{array}{l} c_r a_{rs} b_s = 1/6 \\ c_r b_r^2 = 1/3 \end{array} \\
\hline
h^4 & \begin{array}{l} c_r a_{rs} a_{st} b_t = 1/24 \\ c_r a_{rs} b_s^2 = 1/12 \\ c_r b_r a_{rs} b_s = 1/8 \\ c_r b_r^3 = 1/4 \end{array} \\
\hline
h^5 & \begin{array}{l} c_r a_{rs} a_{st} a_{tu} b_u = 1/120 \\ c_r a_{rs} a_{st} b_t^2 = 1/60 \\ c_r a_{rs} b_s a_{st} b_t = 1/40 \\ c_r a_{rs} b_s^3 = 1/20 \\ c_r b_r a_{rs} a_{st} b_t = 1/30 \\ c_r b_r a_{rs} b_s^2 = 1/15 \\ c_r a_{rs} b_s a_{rt} b_t = 1/20 \\ c_r b_r^2 a_{rs} b_s = 1/10 \\ c_r b_r^4 = 1/5 \end{array}
\end{array}
\qquad (10)
$$

In these relations, b_r has been defined according to the property (7.*b*). Notice also that in the development of the aforementioned relations expansion in Taylor series were used only up to the term of fifth order (with h^5). Since for higher order the deduction is very tedious, the terms with h^6 were not included in the series, even though this term indicates the order of the method here studied. The relations (10) are valid for Runge-Kutta methods, whether implicit or explicit, from first order method (e.g. Euler method) to fifth order method (e.g. Fehlberg method [10]). In all the case the indexes r, s, t and u vary from 1 to number of stages N.

Gear [3], has presented a similar deduction for (8), but only for explicit methods. In Hairer et al. [7], relations similar to (10) appear, but only for explicit methods and only up to the term of order h^4. Also Hairer et al. deduce a theorem that expresses the equivalence of the implicit Runge-Kutta methods and the orthogonal collocation methods.[7,8]

Ralston in 1965 (see for example [11]) made a similar analysis to (8), to obtain the relations of the coefficients, but for an explicit Runge-Kutta method of fourth order and four stages, and found the following family of solutions of relations (10)

$$b_1 = 0 \qquad b_4 = 1 \qquad a_{rs} = 0 \quad (s \geq r) \qquad (11.a-c)$$

$$a_{21} = b_2 \qquad a_{31} = b_3 - a_{32} \qquad a_{32} = \frac{b_3(b_3 - b_2)}{2\,b_2(1 - 2\,b_2)} \qquad a_{41} = 1 - a_{42} - a_{43} \qquad (11.d-g)$$

$$a_{42} = \frac{(1 - b_2)[b_2 + b_3 - 1 - (2\,b_3 - 1)^2]}{2\,b_2(b_3 - b_2)[6\,b_2 b_3 - 4\,(b_2 + b_3) + 3]} \qquad a_{43} = \frac{(1 - 2\,b_2)(1 - b_2)(1 - b_3)}{b_3(b_3 - b_2)[6\,b_2 b_3 - 4\,(b_2 + b_3) + 3]} \qquad (11.h,i)$$

$$c_1 = \frac{1}{2} + \frac{1 - 2(b_2 + b_3)}{12\,b_2 b_3} \qquad c_2 = \frac{2\,b_3 - 1}{12\,b_2(b_3 - b_2)(1 - b_2)} \qquad (11.j,k)$$

$$c_3 = \frac{1 - 2\,b_2}{12\,b_3(b_3 - b_2)(1 - b_3)} \qquad c_4 = \frac{1}{2} + \frac{2\,(b_2 + b_3) - 3}{12\,(1 - b_2)(1 - b_3)} \qquad (11.l,m)$$

Notice that if $b_2 = 1/2$ and $b_3 = 1/2$, then the well known classical Runge-Kutta method of fourth order is obtained

$$
\begin{array}{c|cccc}
0 & 0 & 0 & 0 & 0 \\
1/2 & 1/2 & 0 & 0 & 0 \\
1/2 & 0 & 1/2 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 \\
\hline
 & 1/6 & 1/3 & 1/3 & 1/6
\end{array}
\qquad (12)
$$

LOBATTO QUADRATURES

The explicit Runge-Kutta methods are of direct appication, while the implicit Runge-Kutta methods require the resolution of a system of simultaneous equations with the variables $\mathbf{k}_r$ in each step of integration of the differential equations, as it is suggested by the expression (4.*b*). This system of equations is generally not linear, unless the function $\mathbf{f}(x, \mathbf{y})$ be linear, and can be solved applying the iterative method of fixed point, explained farther on.

The implicit Runge-Kutta method to be used here is a sixth order method ($P = 6$) with four stages ($N = 4$), developed on the bases of Lobatto quadrature [12] (for more details see [6] or [7]). The coefficients of this method expressed in the Butcher notation are

$$\begin{array}{c|cccc|c} 0 & 0 & 0 & 0 & 0 \\ (5-\sqrt{5})/10 & (5+\sqrt{5})/60 & 1/6 & (15-7\sqrt{5})/60 & 0 \\ (5+\sqrt{5})/10 & (5-\sqrt{5})/60 & (15+7\sqrt{5})/60 & 1/6 & 0 \\ \hline 1 & 1/6 & (5-\sqrt{5})/12 & (5+\sqrt{5})/12 & 0 \\ \hline & 1/12 & 5/12 & 5/12 & 1/12 \end{array} \tag{13}$$

This method will be named the principal implicit Runge-Kutta method.

Within the coefficients of the principal implicit Runge-Kutta method it can be detected that a part of them forms another implicit Runge-Kutta method embedded in the first one. This method is of third order ($P = 3$), has three stages ($N = 3$), and can be expressed as Butcher tableau (13) without last column and with last row of matrix a_{rs} as the coefficients c_r.

Both methods, the principal and the embedded, form what is named the Lobatto embedding form of third and sixth orders with four stages. Notice that these methods are implicit only in the auxiliary variables $\mathbf{k}_2$ and $\mathbf{k}_3$, and therefore the system of non-linear equations should be solved only in the mentioned variables. The other variables are of direct solution.

In order to apply an iterative process to solve the system of non-linear equations, initial estimations of the values of the auxiliary variables are required. The best way to carry out the latter is to obtain these values from an explicit Runge-Kutta method, where the auxiliary variables $\mathbf{k}_r$ are evaluated in the same intermediate point in each step, i.e. an explicit method having the same coefficient b_r of the implicit method. Observing the method (13), it is clear that the aforementioned explicit method is rapidly obtained from the relations (11) assuming $b_1 = 0$, $b_2 = (5-\sqrt{5})/10$, $b_3 = (5+\sqrt{5})/10$ and $b_4 = 1$. Notice that these coefficients are consistent with the characteristics of an explicit method. This last aspect makes the implicit method (13) ideal for the desired purpose.

Thus, if b_2 and b_3 are substituted in the relations (11), it is obtained that $a'_{21} = (5-\sqrt{5})/10$, $a'_{31} = -(5+3\sqrt{5})/20$, and $a'_{32} = (3+\sqrt{5})/4$. These are the coefficients of the new explicit Runge-Kutta method that in Butcher's notation can be expressed as

$$\begin{array}{c|cccc|c} 0 & 0 & 0 & 0 & 0 \\ (5-\sqrt{5})/10 & (5-\sqrt{5})/10 & 0 & 0 & 0 \\ (5+\sqrt{5})/10 & -(5+3\sqrt{5})/20 & (3+\sqrt{5})/4 & 0 & 0 \\ 1 & 1/6 & (5-\sqrt{5})/12 & (5+\sqrt{5})/12 & 0 \\ \hline & 1/12 & 5/12 & 5/12 & 1/12 \end{array} \tag{14}$$

The obtained explicit Runge-Kutta method is not reported in the speciality literature and does not correspond to any known quadrature, but pertains to the family of solutions (11) of the fourth order, four stages explicit Runge-Kutta methods. This method will be used to obtain the initial estimations $\mathbf{k}'_2$ and $\mathbf{k}'_3$ for the iterative process in the following form

$$\mathbf{k}_{2,(0)} = h\,\mathbf{f}(x_n + b_2 h\,,\, \mathbf{y}_n + a'_{21}\mathbf{k}_1) \qquad \mathbf{k}_{3,(0)} = h\,\mathbf{f}(x_n + b_3 h\,,\, \mathbf{y}_n + a'_{31}\mathbf{k}_1 + a'_{32}\mathbf{k}_2) \tag{15}$$

Once these initial estimations are obtained, a rapid convergence to the solution of the system of non-linear equations (4.*b*), with the coefficients (13), can be expected.

ITERATIVE PROCESS

As it was mentioned before, the system of non-linear equations (4.*b*), that is originated by any implicit Runge-Kutta method, can be solved in the auxiliary variables $\mathbf{k}_r$ applying the iterative method of fixed point (see Gear[3])

$$k^i_{r,(m+1)} = h\,f^i(x_n + b_r h\,,\, \mathbf{y}_n + a_{rs}\mathbf{k}_{s,(m)}) \tag{16}$$

which is the easiest method to be used due to the form of the mentioned system of equations. Here $m = 0, 1, 2, 3, \ldots$ is the number of iteration in the iterative process.

The global error in the iterative process is defined as $\varepsilon^i_{r,(m)} = k^i_{r,(m)} - k^i_r$, where k^i_r is the exact solution of the system of non-linear equations.

The local error in the iterative process is defined as $\epsilon^i_{r,(m)} = k^i_{r,(m+1)} - k^i_{r,(m)}$, and it is stopped when $c_r\|\epsilon_{r,(m)}\| < \epsilon_{max}$, where ϵ_{max} is the tolerance imposed to the local error to find the solution y^i_{n+1} of the differential equations in one step, and the norm of the local error $\epsilon_{r,(m)}$ is supposed to be euclidean.

If now expression (4.*b*) is substracted from expression (16), then $k^i_{r,(m+1)} - k^i_r = h\,[f^i(x_n + b_r h\,,\, \mathbf{y}_n + a_{rs}\mathbf{k}_{s,(m)}) - f^i(x_n + b_r h\,,\, \mathbf{y}_n + a_{rs}\mathbf{k}_s)]$. If the Lipschitz condition (with $h > 0$ for convenience) is applied, it results $|k^i_{r,(m+1)} - k^i_r| \leq h\,l^i_j\,|a_{rs}|\,|k^j_{s,(m)} - k^j_s|$ with $|\varepsilon^i_{r,(m+1)}| \leq h\,l^i_j\,|a_{rs}|\,|\varepsilon^j_{s,(m)}|$, where l^i_j is the maximum of the absolute value of each element in the jacobean matrix of $\mathbf{f}$. This is $|f^i_j| \leq l^i_j$. Thus, if it is satisfied that $\varepsilon_{(m)} = \max_{1\leq j\leq M}\left(\max_{1\leq s\leq N}|\varepsilon^j_{s,(m)}|\right)$, then $|\varepsilon^i_{r,(m+1)}| \leq h\,l^i_j\,|a_{rs}|\,|\varepsilon^j_{s,(m)}| \leq h\,l^i_j\,\delta_j\,|a_{rs}|\,\delta_s\varepsilon_{(m)}$, where

$$\max_{1\leq i\leq M}\max_{1\leq r\leq N}\left(|\varepsilon^i_{r,(m+1)}|\right) \leq \max_{1\leq i\leq M}\left[\max_{1\leq r\leq N}\left(h\,l^i_j\,\delta_j\,|a_{rs}|\,\delta_s\varepsilon_{(m)}\right)\right] \tag{17}$$

Thus

$$\varepsilon_{(m+1)} \leq h\,\mathrm{L}\,\mathrm{A}\,\varepsilon_{(m)} \qquad \text{where} \qquad \mathrm{L} = \max_{1\leq i\leq M}\left(l^i_j\,\delta_j\right) \qquad \mathrm{A} = \max_{1\leq r\leq N}\left(|a_{rs}|\,\delta_s\right) \tag{18}$$

The expression (18) means that for a high number of iterations, $m \to \infty$, the global error $\varepsilon_{(m)} \to 0$ when $h \leq 1/(\mathrm{L}\,\mathrm{A})$ and the iterative process is convergent locally (also globally) in the form

$$c_r\|\epsilon_{r,(m+1)}\| < c_r\|\epsilon_{r,(m)}\| \tag{19}$$

The expression (18) is the limit of the stepsize, for the iterative method of fixed point to be convergent, when the system of non-linear equation is being solved in the implicit Runge-Kutta method. This is the only restriction of the implicit Runge-Kutta methods, compared to the explicit methods which are much less stable.

STEPSIZE CONTROL

ERROR ANALYSIS

The implicit Runge-Kutta method of sixth order with four stages that is defined by the coefficients (13), in fact represents two different embedded methods, one of third order within the other of sixth order, i.e. the coefficients (13) include both sixth and third order methods. This aspect is relevant to control the stepsize, because solving the system of non-linear equations (4.*b*) for the same coefficients, it can be obtained two solutions of different order in the local truncation error, reducing to a minimum the number of numerical calculations to be made. Fehlberg [10] reported this aspect to control the stepsize in his explicit Runge-Kutta methods of fourth and fifth orders in a embedded form.

Let $\mathbf{y}_n$ and $\bar{\mathbf{y}}_{n+1}$ be the solutions of the system of differential equations (1), offered by the Runge-Kutta methods type Lobatto of sixth and third orders, respectively, embedded in only one formulation as it was described before. This is

$$y^i_{n+1} = y^i_n + \frac{1}{12}(k^i_1 + 5k^i_2 + 5k^i_3 + k^i_4) \qquad \bar{y}^i_{n+1} = y^i_n + \frac{1}{12}[2k^i_1 + (5-\sqrt{5})k^i_2 + (5+\sqrt{5})k^i_3] \tag{20}$$

The auxiliary variables $\mathbf{k}_1$, $\mathbf{k}_2$, $\mathbf{k}_3$ and $\mathbf{k}_4$ are the same for both expressions and are obtained using the equation (4.*b*) with the coefficients (13).

It will be denoted as E^i_{n+1} the difference of the equation (20.*a*) minus the equation (20.*b*). This is

$$E^i_{n+1} = y^i_{n+1} - \bar{y}^i_{n+1} = \frac{1}{12}[-k^i_1 + \sqrt{5}(k^i_2 - k^i_3) + k^i_4] \tag{21}$$

If $\mathbf{y}(x_n)$ is the exact solution of the differential equation (1) in the value $x = x_n$, the local truncation errors of the numerical solutions (20) are defined respectively as $e^i_n = y^i_n - y^i(x_n) = O(h^7_{n-1})$, where $\bar{e}^i_n = \bar{y}^i_n - y^i(x_n) = O(h^4_{n-1})$, and then

$$E^i_{n+1} = y^i_{n+1} - \bar{y}^i_{n+1} = e^i_{n+1} - \bar{e}^i_{n+1} = O(h^4_n) \tag{22}$$

Remember that the local truncation error is of order $P+1$ if the Runge-Kutta method is of order P.

If the expression (22) is organized in the following form

$$E^i_{n+1} = \left[\frac{y^i_{n+1} - y^i(x_{n+1})}{y^i(x_{n+1})}\right] y^i(x_{n+1}) - [\bar{y}^i_{n+1} - y^i(x_{n+1})] \tag{23}$$

it is obtained

$$E^i_{n+1} = e^i_{(r),n+1} y^i(x_{n+1}) - \bar{e}^i_{n+1} \qquad \text{where} \qquad e^i_{(r),n+1} = \left[\frac{y^i_{n+1} - y^i(x_{n+1})}{y^i(x_{n+1})}\right] \tag{24}$$

is the relative local truncation error.

If now it is assumed that $y^i(x_{n+1})$ is approximated by y^i_{n+1} in the denominator of (24), it can be applied the Cauchy-Schwartz and the triangular desequalities to the expression (23), and this results

$$|E^i_{n+1}| \leq |e^i_{(r),n+1}|\ |y^i(x_{n+1})| + |\bar{e}^i_{n+1}| \leq e_{(r),max}\ |y^i_{n+1}| + \bar{e}_{max} \tag{25}$$

where $e_{(r),max}$ and $\bar{e}_{max}$ are respectively the tolerances for the relative and absolute local truncation errors for the implicit Runge-Kutta methods of sixth and third orders. The expression (25) also means that, for the solution of the differential equations in one step be accepted, it should be verified that

$$Q^i_n = \frac{|E^i_{n+1}|}{e_{(r),max}\ |y^i_{n+1}| + \bar{e}_{max}} \leq 1 \tag{26}$$

being the tolerances for the relative and absolute local truncation errors proposed by the user of the algorithm.

CONTROL ALGORITHM

Let h_{n+1} be the stepsize in the next step that tends to make $Q^i_n \cong 1$. Taken into account the order of the difference E^i_{n+1} defined by (22), the parameter Q_n may be redefined as

$$Q_n = \left(\frac{h_n}{h_{n+1}}\right)^4 \qquad \text{where} \qquad Q_n = \max_{1 \leq i \leq M} (Q^i_n) \tag{27}$$

and thus, solving for h_{n+1}, it results

$$h_{n+1} = h_n \left(\frac{1}{Q_n}\right)^{1/4} = h_n S_n \qquad \text{with} \qquad S_n = \left(\frac{1}{Q_n}\right)^{1/4} \tag{28, 29}$$

Here, it is convenient to mention that Shampine et al. [13] use expressions similar to (28) and (29) to control the size of the step of integration in the Runge-Kutta method of fourth and fifth orders with five stages developed by Fehlberg [10], but with some modifications, in order to guarantee that S_n always be bound in the interval $[S_{min}, S_{max}]$, and that h_{n+1} always be greater than a limit value h_{min}. Additionally, the mentioned authors multiply S_n of (29) by a coefficient C_q less than unit, to make h_{n+1} tends to h_n, and thus to make $Q_n \cong 1$, but a little lower. All the aforementioned modifications are resumed in continuation as

$$S_n = C_q \left(\frac{1}{Q_n}\right)^{1/4} \qquad C_q = 0.9 \sim 0.99 \tag{30}$$

$$S'_n = \max(\min(S_n, S_{max}), S_{min}) \qquad h_{n+1} = h_n S'_n \qquad h'_{n+1} = \max(h_{n+1}, h_{min}) \tag{31}$$

While in [13] the exponent is 1/5 in the expression (30), here the exponent is 1/4, and they also recommend for the coefficients and limits the values $C_q = 0.9$, $S_{min} = 0.1$ and $S_{max} = 5$. The value of the minimum stepsize, h_{min}, is determined by the precision of the computer to be used. In this work they are recommended the same values for the coefficients in the expressions (30) to (31).

The procedure to calculate the optimal value of the size of the integration step, that permits to satisfy the tolerances $e_{(r),max}$ and $\bar{e}_{max}$, is described in continuation:

- Estimated an initial stepsize h_n, the implicit Runge-Kutta method type Lobatto is used to calculate the auxiliary variables k_1^i, k_2^i, k_3^i and k_4^i with the expression (4.*b*), using the coefficients (13) and with the iterative process (16), using the initial values (15).
- The expressions (20) permit to find the solutions y_{n+1}^i and $\bar{y}_{n+1}^i$ of the methods of sixth and third orders, respectively.
- The definition (21) permits to calculate the difference E_{n+1}^i between both methods.
- With the equation (26) it can be calculated the parameters Q_n^i, and with the equation (27) it can be obtained the maximum of them.
- The relations (30) to (31) determine the value of the size of the next step h_{n+1}.
- If $Q_n \leq 1$, the integration with the step h_n (or the application of the Runge-Kutta method from x_n to x_{n+1}) is accepted and the step h_{n+1} is considered the step for the next integration (or the next application of the Runge-Kutta method from x_{n+1} to x_{n+2}).
- If $Q_n > 1$, the integration with the stepsize h_n is rejected and it is repeated all the algorithm but with $h_n = h'_{n+1}$ obtained from (31).

This procedure, sometimes increases the stepsize, and other times decreases the stepsize, in a optimal form, in order to guarantee that the relative error $e_{(r),n+1}^i$ of the sixth order Runge-Kutta method be less than the tolerance $e_{(r),max}$, and the error $\tilde{e}_{n+1}^i$ of the third order Runge-Kutta method be less than the tolerance $\tilde{e}_{max}$. In any case, the solution of the Runge-Kutta method will be y_{n+1}^i, i.e. the solution with the implicit Runge-Kutta method of sixth order.

ANALYSIS OF THE METHODS

PRECISION

The order of precision of any Runge-Kutta method comes from the comparison between this and the expansion in Taylor series of $\mathbf{y}(x_{n+1})$ around $\mathbf{y}(x_n)$. From this comparison was obtained the relations (10) that should be fulfilled by the coefficients of the Runge-Kutta methods. For a Runge-Kutta method of order P, it should be satisfied the relations (10) up to the term of the Taylor series that contains h^P. The remainder terms in the Taylor series are they which determine the order of the local truncation error. Thus, a method of order P has a local truncation error of order $P+1$, i.e. that depends on h^{P+1}. The global truncation error always is one order less than the local truncation error [3], i.e. that depends on h^P. All these criterions can be applied to the implicit Runge-Kutta type Lobatto of third and sixth order. In this form it is obtained that, the method of third order satisfy the relations (10) up to term with h^3 and the local and global truncation errors depend, respectively, on h^4 and h^3. The method of sixth order satisfy the relations (10) up to term with h^6 (the relations for this last term does not appear for the reasons explained there) and the local and global truncation depend, respectively, on h^7 and h^6. The dependence of an error respect to h^P is indicated as $O(h^P)$ and is said that the order of precision is P, according to the definition $x_{n+1} = x_n + h$. Also it is satisfied that $O(h^P) + O(h^Q) = O(h^P)$ when $Q \geq P$.

CONVERGENCY

In the section of Iterative Process it was indicated that the implicit Runge-Kutta methods generated a system of equations of the type (4.*b*), in general of non-linear characteristics, where the unknowns were the auxiliary variables $\mathbf{k}_r$. As it was said before, this system of non-linear equations may be solved using an iterative method of fixed point, which converges for the treated problem in particular, if it is satisfied (18) and the local convergence is established according to (19).

For the specific case when it is being solved the problem with only one differential equation of the form

$$\frac{dy}{dx} = f(y) \qquad f(y) = \lambda y \tag{32}$$

it is obtained that $L = |\lambda|$. For the implicit Runge-Kutta methods type Lobatto of third and sixth orders, it is obtained that $A = (5+\sqrt{5})/10$ and $A = 1$, respectively. Thus, to assure the convergence of the iterative process stated in these cases, particularly to the linear problem (32), it should be satisfied the following two conditions (assuming h positive)

$$h \leq \frac{5-\sqrt{5}}{2|\lambda|} \cong \frac{1.38}{|\lambda|} \quad \text{(Implicit 3rd order method)} \qquad h \leq \frac{1}{|\lambda|} \quad \text{(Implicit 6th order method)} \tag{33}$$

respectively. Notice that the condition (33.*b*) is the most restrictive of the two.

STABILITY

The stability of the Runge-Kutta methods is generally studied on the bases of their performance for solving the specific problem (32) with only one linear ordinary differential equation. In Appendix C it can be found the analysis of the stability for the more general case of a system of linear ordinary differential equations.

Thus, if the expression (4.*b*) is applied, taken into account that the problem being solved is (32), it is obtained that $k_r = h\,f(y_n + a_{r_s}k_s) = h\,\lambda\,(y_n + a_{r_s}k_s) = h\,\lambda\,y_n + h\,\lambda\,a_{r_s}\,k_s$. Moreover, if k_r is substituted as $\delta_{r_s}k_s$ and the therms are regrouped, it results $\delta_{r_s}k_s = h\,\lambda\,y_n + h\,\lambda\,a_{r_s}\,k_s$, where $\delta_{r_s}k_s - h\,\lambda\,a_{r_s}\,k_s = h\,\lambda\,y_n$, and thus, if k_s is factorized, then $[\delta_{r_s} - h\,\lambda\,a_{r_s}]\,k_s = h\,\lambda\,y_n\,\delta_r$ with $r, s = 1, 2, 3, \ldots, N$.

Thes latter expression represents a system of N linear equations with N unknowns k_s. If this system of linear equations is solved and if the solutions k_r are substituted in the equation (4.*a*), then, it is found a relation for y_{n+1} depending only on y_n and on the coefficients of the used Runge-Kutta method. The mentioned relation is of the form $y_{n+1} = \mu_1(h\lambda)\,y_n$, where $\mu_1(h\lambda)$ is found applying, for example, to the implicit sixth order Runge-Kutta method of Lobatto type, the procedure above explained. In this case the result is

$$\mu_1(z) = \left[\frac{1 + \frac{2}{3}z + \frac{1}{6}z^2 + \frac{1}{30}z^3 + \frac{1}{360}z^4}{1 - \frac{1}{3}z + \frac{1}{30}z^2}\right] \tag{34}$$

where the function $\mu_1(z)$ was deduced from the coefficients (13). For the case of the implicit third order Runge-Kutta method type Lobatto resumed in the inner coefficients (13), the result is $\tilde{y}_{n+1} = \tilde{\mu}_1(h\lambda)\,y_n$, where

$$\tilde{\mu}_1(z) = \left[\frac{1 + \frac{2}{3}z + \frac{1}{6}z^2 + \frac{1}{36}z^3}{1 - \frac{1}{3}z + \frac{1}{36}z^2}\right] \tag{35}$$

The functions $\tilde{\mu}_1(z)$ and $\mu_1(z)$ are denominated characteristic roots of the Runge-Kutta methods of third and sixth orders, respectively. The characteristic roots are also known as *stability functions*. The Runge-Kutta methods, to which these roots pertain, are considered stables if their absolute values are less than unit, for a determined real value of $z = h\lambda$. Notice that, if $|\tilde{\mu}_1(h\lambda)|$ or $|\mu_1(h\lambda)|$ is less than the unit, then it is satisfied that $|\tilde{y}_{n+1}|$ or $|y_{n+1}|$ is less than $|y_n|$ and the stability is guaranteed.

The functions (34) and (35) may be represented graphically as $\tilde{\mu}_1(h\lambda)$ and $\mu_1(h\lambda)$ vs. $h\lambda$, and the region of stability for each one may be observed. The methods are stable if it is satisfied the following two conditions (assuming that h is positive and λ is negative).

$$h \leq \frac{6.8232}{|\lambda|} \quad \text{(Implicit 3rd order method)} \qquad h \leq \frac{9.6485}{|\lambda|} \quad \text{(Implicit 6th order method)} \tag{36}$$

Therefore, the mentioned methods are not A-stable. When comparing the conditions (33) with the conditions (36), the condition (33.*b*) continues being the most restrictive of all.

The function $\mu_1(z)$ constitutes an aproximation of Padé [2] for the function $y = e^z$ (see Lapidus and Seinfeld [6]), and, additionally, is always positive and less than unit in the interval $[-9.648495252\ ,\ 0.0]$. Notice that the function $\mu_1(z)$ approximates well to the function $y = e^z$ for the range $z > -4$.

The function $\tilde{\mu}_1(z)$, however, is not an approximation of Padé, has only one root in the point $z = -2.706010973\ldots$, and is, in absolute value, less or equal to the unit within the interval $[-6.823183583\ ,\ 0.0]$. This can be observed graphically.

The conditions (36) reveal that the implicit Runge-Kutta methods type Lobatto of third and sixth orders are more stable than the explicit Runge-Kutta methods type Fehlberg of fourth and fifth orders, having these last methods the following stability conditions

$$h \leq \frac{2.785}{|\lambda|} \quad \text{(Explicit 4th order method)} \qquad h \leq \frac{3.15}{|\lambda|} \quad \text{(Explicit 5th order method)} \tag{37}$$

The only limitation of the implicit Runge-Kutta methods comes from the convergency conditions (33), which are based on the way used to solve the system of equations (4.*b*), and on the assumption that the differential equation has a linear form (32). These convergency conditions permit an increase of the stepsize h much less than the stability conditions (36). This aspect brakes the advance of the implicit Runge-Kutta methods in a notable manner, but the difficulty may be compensated partially in two forms. First, estimating conveniently the initial values of the auxiliary variables $\mathbf{k}_r$ for the iterative process. As it was treated before, this can be made using an explicit Runge-Kutta method with the expressions (15). Secondly, it may be used the fixed point method in a efficient manner, similar to the Gauss-Seidel method, i.e. when an unknown is approximately calculated, it is substituted inmediately in the next equation, and thus on. These two modifications of the implicit Runge-Kutta methods here used may unprove the performance of the method, in the sense of that the stepsize h is liberated from the convergency conditions (33) and thus it may be permitted to increase much more.

A more sofisticated solution to the problem mentioned before can be developed, if it is used the Newton-Rapson method to solve the system of non-linear equations (4.*b*). This may increase the number of numerical calculations to be performed by the algorithm, but the convergence will be more rapid.

REFERENCES

[1] Gerald, C. F. **Applied Numerical Analysis**. 2^{nd} Edition. Addison-Wesley, New York, 1970.

[2] Burden R. L.; Faires, J. D. **Numerical Analysis**. 3^{rd} Edition. PWS. Boston, 1985.

[3] Gear, C. W. **Numerical Initial Value Problems in Ordinary Differential Equations**. Prentice-Hall, Englewood Cliffs, New Jersey, 1971.

[4] Butcher, J. C. "On the Runge-Kutta Processes of High Order". **J. Austral. Math. Soc.**, Vol. **IV**, Part 2, 1964, pp. 179-194.

[5] Butcher, J. C. **The Numerical Analysis of Ordinary Differential Equations, Runge-Kutta and General Linear Methods**. John Wiley, New York, 1987.

[6] Lapidus, L.; Seinfeld, J. H. **Numerical Solution of Ordinary Differential Equations**. Academic Press, New York, 1971.

[7] Hairer, E.; Norsett, S. P.; Wanner, G. **Solving Ordinary Differential Equations I. Nonstiff Problems**. Springer-Verlag, Berlin, 1987.

[8] Hairer, E.; Wanner, G. **Solving Ordinary Differential Equations II: Stiff and Differential- Algebraic Problems**. Springer-Verlag, Berlin, 1991.

[9] van der Houwen, P. J.; Sommeijer, B. P. "Iterated Runge-Kutta Methods on Parallel Computers". **SIAM J. Sci. Stat. Comput.**, Vol.**12**, No.5, pp.1000-1028, (1991).

[10] Fehlberg, E. "Low-Order Classical Runge-Kutta Formulas with Stepsize Control". **NASA** Report No. TR R-315, 1971.

[11] Ralston, A.; Rabinowitz, P. **A First Course in Numerical Analysis**. 2nd Edition. McGraw-Hill, New York, 1978.

[12] Lobatto, R. **Lessen over Differentiaal- en Integraal-Rekening**. 2 Vol. La Haye, 1851-52.

[13] Shampine, L. F.; Watts, H. A.; Davenport, S. M. "Solving Non-Stiff Ordinary Differential Equations - The State of the Art". **SANDIA** Laboratories, Report No. SAND75-0182, 1975. **SIAM Review**, Vol. **18**, No. 3, 1976, pp. 376-411.