

# The Scheme of 10<sup>th</sup> Order Implicit Runge-Kutta Method to Solve the First Order of Initial Value Problems

Z. Bahri<sup>1</sup>, L. Zakaria<sup>2,\*</sup>, Syamsudhuha<sup>1</sup>

**Abstract**—To construct a scheme of implicit Runge-Kutta methods, there are a number of coefficients that must be determined and satisfying consistency properties and Butcher's simplifying assumptions. In this paper we provide the numerical simulation technique to obtain a scheme of 10<sup>th</sup> order Implicit Runge-Kutta (IRK<sub>10</sub>) method. For simulation process, we construct an algorithm to compute all the coefficients involved in the IRK<sub>10</sub> scheme. The algorithm is implemented in a language programming (Turbo Pascal) to obtain all the required coefficients in the scheme. To show that our scheme works correctly, we use the scheme to solve Hénon-Heiles system.

**Keywords**—ODEs, 10<sup>th</sup> order IRK method, numerical technique, Hénon-Heiles system

## I. INTRODUCTION

Let a first order ordinary differential equation system (ODES)

$$y' = f(x, y(x)) \quad (1)$$

together with

$$y(x_0) = y_0 \quad (2)$$

In (1), where the "prime" indicates differentiation with respect to  $x$ ,  $y$  is a  $D$ -dimensional vector ( $y \in \mathbb{R}^D$ ), and  $f : \mathbb{R} \times \mathbb{R}^D \rightarrow \mathbb{R}^D$ . The ODES (1)-(2) is the well-known the first order of initial value problem. To solve problem (1)-(2) can be used analytical and/or numerical procedures. But, for solving the special problems (*Hamiltonian and Divergen Free systems*, for examples) and taking efficiency and effective calculations, some mathematicians recommended to use numerical approximations ([1-4,5]). One of numerical methods which can be used to solve (1-2) which is enough recognized and a lot of used is Runge-Kutta method.

**Definition :** Let  $b_i, a_{ij}$ , and  $c_i$  ( $i, j = 1, 2, \dots, s$ ) be real numbers. The method

$$y_{n+1} = y_n + \dagger \sum_{i=1}^s b_i f_i \quad (3)$$

$$f_i = f \left( t_n + c_i \dagger, y_n + \dagger \sum_{j=1}^s a_{ij} f_j \right)$$

<sup>1</sup>Department of Mathematics, Faculty of Mathematics and Natural Sciences, University of Riau, Kampus Bina Widya Simpang Baru Km. 12.5 Pekanbaru, Riau, Indonesia.

<sup>2</sup>Department of Mathematics, Faculty of Mathematics and Natural Sciences, University of Lampung, Soemantri Brojonegoro 1 Bandar Lampung, Lampung, Indonesia.

\*Correspondence to 2<sup>nd</sup> Author, email: lazakaria.1969@fmipa.unila.ac.id. Tel.: +62-721-704625; fax: +62-721-704625.

is called an  $s$ -stage Runge-Kutta method. When  $a_{ij} = 0$  for  $i \leq j$  the method is called an **explicit Runge-Kutta** method. If  $a_{ij} = 0$   $i < j$  and least one  $a_{ii} \neq 0$ , the method is called a

**diagonal implicit Runge-Kutta** method. If all of the diagonal elements are identical,  $a_{ij} = \gamma$  for  $i = 1, 2, \dots, s$ , the method is called a **singly diagonal implicit Runge-Kutta** method. In all other cases, the method is called an **implicit Runge-Kutta** method. The Runge-Kutta methods are often given in the form of a tableau containing their coefficients namely

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} \quad (4)$$

where  $c_i = \sum_{j=1}^s a_{ij}$ ;  $i = 1, 2, \dots, s$ .

Consistency in the Runge-Kutta methods is investigated by using a Taylor series expansion. Any  $s$ -stage Runge-Kutta process of the form (3) or (4) is consistent if  $\sum_{i=1}^s b_i = 1$ .

Butcher in [6] states that this condition is necessary and sufficient condition for the local truncation error of the method to have *asymptotic behavior*  $O(\dagger)$ .

There are some advantages to use Implicit Runge-Kutta (IRK) methods (see [6-8]). (1) IRK methods usually are required for systems whose solutions contain rapidly decaying components (see [6]); (2) Some methods may also be used in preserving the symplectic structure of Hamiltonian systems; (3) IRK methods (*Gauss quadratures*) have a number of big potency for the computing of integrate the geometry; (4) high order integrator to be used by the reason of doubled accuracy is recommended; (5) to evaluate the vector field "costly", all stage- $s$  in IRK can be evaluated by parallel and; (6) IRK can be solve the ordinary differential equation (ODE) or system (ODEs) in general.

To construct a scheme of IRK methods, Butcher discovered the existence of  $s$ -stage methods of order  $2s$ , for all  $s$ . He used *simplifying assumptions* to find these methods. The simplifying assumptions are

$$A(i) = \sum_{j=1}^s a_{ij} c_j^{q-1} = \frac{1}{q} c_i^q \quad i=1,2,\dots,s \quad q=1,2,\dots, \quad (5)$$

$$B(p) = \sum_{i=1}^s b_i c_i^{q-1} = \frac{1}{q} \quad q=1,2,\dots,p$$

$$C(\dots) = \sum_{i=1}^s b_i c_i^{q-1} a_{ij} = \frac{b_j}{q} (1 - c_j^q) \quad j=1,2,\dots,s \quad q=1,2,\dots,$$

Based on consistency property and Butcher's simplifying assumptions, Hairer *et.al* (2007)[7] and Ismail (2009)[8] derive some IRK methods new analytical procedures.

In analytical procedure, we can use the idea of collocation to derive IRK methods ([4],[8]). Unfortunately, for the stage of *stage s = 3*, constructing an integrator IRK using analytical procedure will be difficult because there are some values of  $b_i, c_i$ , and  $a_{ij}$  ( $i, j = 1, 2, \dots, s$ ) must be determined. Therefore, the numerical technique or procedure is an alternative choice to use. In this procedure, we can use computer to determine the values of  $c_i$  ( $i = 1, 2, \dots, s$ ),  $b_i$  and  $a_{ij}$  ( $i, j = 1, 2, \dots, s$ ).

Notice that one important property of IRK method is symplecticness. An IRK method (4) is symplectic if it satisfy the following condition ([9])

$$b_i b_j - b_i a_{ij} - b_j a_{ji} = 0 \quad , \quad i, j \in \{1, 2, \dots, s\}.$$

## II. RESULT AND DISCUSSION

In this section, we describe how to get a class of 10<sup>th</sup> order IRK method using computer simulation to obtain all Butcher's coefficient values (4) based on Butcher's simplifying assumptions (5).

Setting  $k = i = j = s = 1, 2, 3, 4, 5$  into first equation in (5), we have the following systems

$$a_{11} + a_{12} + a_{13} + a_{14} + a_{15} = c_1 \quad (6.1)$$

$$a_{21} + a_{22} + a_{23} + a_{24} + a_{25} = c_2 \quad (6.2)$$

$$a_{31} + a_{32} + a_{33} + a_{34} + a_{35} = c_3 \quad (6.3)$$

$$a_{41} + a_{42} + a_{43} + a_{44} + a_{45} = c_4 \quad (6.4)$$

$$a_{51} + a_{52} + a_{53} + a_{54} + a_{55} = c_5 \quad (6.5)$$

$$a_{11}c_1 + a_{12}c_2 + a_{13}c_3 + a_{14}c_4 + a_{15}c_5 = \frac{1}{2}c_1^2 \quad (7.1)$$

$$a_{21}c_1 + a_{22}c_2 + a_{23}c_3 + a_{24}c_4 + a_{25}c_5 = \frac{1}{2}c_2^2 \quad (7.2)$$

$$a_{31}c_1 + a_{32}c_2 + a_{33}c_3 + a_{34}c_4 + a_{35}c_5 = \frac{1}{2}c_3^2 \quad (7.3)$$

$$a_{41}c_1 + a_{42}c_2 + a_{43}c_3 + a_{44}c_4 + a_{45}c_5 = \frac{1}{2}c_4^2 \quad (7.4)$$

$$a_{51}c_1 + a_{52}c_2 + a_{53}c_3 + a_{54}c_4 + a_{55}c_5 = \frac{1}{2}c_5^2 \quad (7.5)$$

$$a_{11}c_1^2 + a_{12}c_2^2 + a_{13}c_3^2 + a_{14}c_4^2 + a_{15}c_5^2 = \frac{1}{3}c_1^3 \quad (8.1)$$

$$a_{21}c_1^2 + a_{22}c_2^2 + a_{23}c_3^2 + a_{24}c_4^2 + a_{25}c_5^2 = \frac{1}{3}c_2^3 \quad (8.2)$$

$$a_{31}c_1^2 + a_{32}c_2^2 + a_{33}c_3^2 + a_{34}c_4^2 + a_{35}c_5^2 = \frac{1}{3}c_3^3 \quad (8.3)$$

$$a_{41}c_1^2 + a_{42}c_2^2 + a_{43}c_3^2 + a_{44}c_4^2 + a_{45}c_5^2 = \frac{1}{3}c_4^3 \quad (8.4)$$

$$a_{51}c_1^2 + a_{52}c_2^2 + a_{53}c_3^2 + a_{54}c_4^2 + a_{55}c_5^2 = \frac{1}{3}c_5^3 \quad (8.5)$$

$$a_{11}c_1^3 + a_{12}c_2^3 + a_{13}c_3^3 + a_{14}c_4^3 + a_{15}c_5^3 = \frac{1}{4}c_1^4 \quad (9.1)$$

$$a_{21}c_1^3 + a_{22}c_2^3 + a_{23}c_3^3 + a_{24}c_4^3 + a_{25}c_5^3 = \frac{1}{4}c_2^4 \quad (9.2)$$

$$a_{31}c_1^3 + a_{32}c_2^3 + a_{33}c_3^3 + a_{34}c_4^3 + a_{35}c_5^3 = \frac{1}{4}c_3^4 \quad (9.3)$$

$$a_{41}c_1^3 + a_{42}c_2^3 + a_{43}c_3^3 + a_{44}c_4^3 + a_{45}c_5^3 = \frac{1}{4}c_4^4 \quad (9.4)$$

$$a_{51}c_1^3 + a_{52}c_2^3 + a_{53}c_3^3 + a_{54}c_4^3 + a_{55}c_5^3 = \frac{1}{4}c_5^4 \quad (9.5)$$

$$a_{11}c_1^4 + a_{12}c_2^4 + a_{13}c_3^4 + a_{14}c_4^4 + a_{15}c_5^4 = \frac{1}{5}c_1^5 \quad (10.1)$$

$$a_{21}c_1^4 + a_{22}c_2^4 + a_{23}c_3^4 + a_{24}c_4^4 + a_{25}c_5^4 = \frac{1}{5}c_2^5 \quad (10.2)$$

$$a_{31}c_1^4 + a_{32}c_2^4 + a_{33}c_3^4 + a_{34}c_4^4 + a_{35}c_5^4 = \frac{1}{5}c_3^5 \quad (10.3)$$

$$a_{41}c_1^4 + a_{42}c_2^4 + a_{43}c_3^4 + a_{44}c_4^4 + a_{45}c_5^4 = \frac{1}{5}c_4^5 \quad (10.4)$$

$$a_{51}c_1^4 + a_{52}c_2^4 + a_{53}c_3^4 + a_{54}c_4^4 + a_{55}c_5^4 = \frac{1}{5}c_5^5 \quad (10.5)$$

Then, from second equation in (5) we have the following systems

$$b_1 + b_2 + b_3 + b_4 + b_5 = 1 \quad (11.1)$$

$$b_1c_1 + b_2c_2 + b_3c_3 + b_4c_4 + b_5c_5 = \frac{1}{2} \quad (11.2)$$

$$b_1c_1^2 + b_2c_2^2 + b_3c_3^2 + b_4c_4^2 + b_5c_5^2 = \frac{1}{3} \quad (11.3)$$

$$b_1c_1^3 + b_2c_2^3 + b_3c_3^3 + b_4c_4^3 + b_5c_5^3 = \frac{1}{4} \quad (11.4)$$

$$b_1c_1^4 + b_2c_2^4 + b_3c_3^4 + b_4c_4^4 + b_5c_5^4 = \frac{1}{5} \quad (11.5)$$

Equations (11.1)-(11.5) are solved respect to with  $b_i$  ( $i = 1, 2, 3, 4, 5$ ), we have the following form

$$b_1 = \frac{\left\{ (12-15c_4+5(-3+4c_4)c_5-5c_3(3-4c_5+c_4(-4+6c_5)))+ \right.}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \left. \right\}}{\left\{ 5c_2(-3+c_4(4-6c_5))+4c_5+2c_3(2-3c_5+c_4(-3+6c_5)) \right\}} \quad (12.1)$$

$$b_2 = \frac{\left\{ (5c_4(3-4c_5)+3(-4+5c_5)+5c_3(3-4c_5+c_4(-4+6c_5)))+ \right.}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \left. \right\}}{\left\{ 5c_1(3-4c_5+c_4(-4+6c_5))+c_3(-4+c_4(6-12c_5)+6c_5) \right\}} \quad (12.2)$$

$$b_3 = \frac{\left\{ 12-15c_4+5(-3+4c_4)c_5-5c_2(3-4c_5+c_4(-4+6c_5)))+ \right.}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \left. \right\}}{\left\{ 5c_1(-3+c_4(4-6c_5))+4c_5+2c_2(2-3c_5+c_4(-3+6c_5)) \right\}} \quad (12.3)$$

$$b_4 = \frac{\left\{ 5c_3(3-4c_5)+3(-4+5c_5)+5c_2(3-4c_5+c_3(-4+6c_5)))+ \right.}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4-c_5)} \left. \right\}}{\left\{ 5c_1(3-4c_5+c_3(-4+6c_5))+c_2(-4+c_3(6-12c_5)+6c_5) \right\}} \quad (12.4)$$

$$b_5 = \frac{\left\{ (12-15c_3+5(-3+4c_3)c_4-5c_2(3-4c_4+c_3(-4+6c_4)))+ \right.}{60(c_1-c_5)(c_2-c_5)(c_3-c_5)(c_4-c_5)} \left. \right\}}{\left\{ 5c_1(-3+c_3(4-6c_4))+4c_4+2c_2(2-3c_4+c_3(-3+6c_4)) \right\}} \quad (12.5)$$

Equations (6.1)-(10.5) solved respect to with  $a_{ij}$  ( $i, j = 1, 2, 3, 4$ ), we have the following form

$$a_{11} = \frac{\left\{ c_1(12c_4^4+60c_2c_3c_4c_5-15c_1^3(c_2+c_3+c_4+c_5))+20c_1^2(c_3c_4+(c_3+c_4)c_5)+ \right.}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \left. \right\}}{\left\{ c_2(c_3+c_4+c_5)-30c_1(c_3c_4c_5+c_2(c_3c_4+(c_3+c_4)c_5)) \right\}} \quad (13.1)$$

$$a_{12} = \frac{\left\{ c_1^2(3c_1^3-30c_3c_4c_5-5c_1^2(c_3+c_4+c_5))+10c_1(c_4c_5+c_3(c_4+c_5)) \right.}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \left. \right\}}{\left\{ c_2^2(3c_1^3-30c_3c_4c_5-5c_1^2(c_3+c_4+c_5))+10c_1(c_4c_5+c_3(c_4+c_5)) \right\}} \quad (13.2)$$

$$a_{13} = \frac{\{c_1^2(-3c_1^3+30c_2c_4c_5+5c_1^2(c_2+c_4+c_5)-10c_1(c_4c_5+c_2(c_4+c_5)))\}}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \quad (13.3)$$

$$a_{14} = \frac{\{c_1^2(3c_1^3-30c_2c_3c_5-5c_1^2(c_2+c_3+c_5)+10c_1(c_3c_5+c_2(c_3+c_5)))\}}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4+c_5)} \quad (13.4)$$

$$a_{15} = \frac{\left\{ \begin{array}{l} 12-15c_3+5(-3+4c_3)c_4-5c_2(3-4c_4+c_3(-4+6c_4)) \\ +5c_1(-3+c_3(4-6c_4))+4c_4+2c_2(2-3c_4+c_3(-3+6c_4)) \end{array} \right\}}{60(c_1-c_5)(c_2+c_5)(c_3+c_5)(c_4+c_5)} \quad (13.5)$$

$$a_{21} = \frac{\{c_2^2((-3c_2^3+30c_5c_3c_4+5c_2^2(c_3+c_4+c_5)-10c_2(c_5c_4+c_3(c_5+c_4))))\}}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \quad (14.1)$$

$$a_{22} = \frac{\left\{ \begin{array}{l} c_2(c_2(-12c_2^3+30c_3c_4c_5+15c_2^2(c_3+c_4+c_5))-20c_2(c_3c_4+(c_3+c_4)c_5)) \\ +5c_1((3c_2^3-12c_2c_3c_4c_5-4c_2^2(c_3+c_4+c_5)+6c_2(c_3c_4+(c_3+c_4)c_5))) \end{array} \right\}}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \quad (14.2)$$

$$a_{23} = \frac{\{c_2^2(5c_1(c_2^2+6c_4c_5-2c_2(c_4+c_5)))+c_2(-3c_2^2-10c_4c_5+5c_2(c_4+c_5))\}}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \quad (14.3)$$

$$a_{24} = \frac{\{c_2^2(3c_2^3+10c_3c_5-5c_2(c_3+c_5))-5c_1(c_2^2+6c_3c_5-2c_2(c_3+c_5))\}}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4+c_5)} \quad (14.4)$$

$$a_{25} = \frac{\{c_2^2(5c_1(c_2^2+6c_3c_4-2c_2(c_3+c_4)))+c_2(-3c_2^2-10c_3c_4+5c_2(c_3+c_4))\}}{60(c_1-c_5)(c_2+c_5)(c_3+c_5)(c_4+c_5)} \quad (14.5)$$

$$a_{31} = \frac{\{c_3^2(5c_2(c_3^2+6c_4c_5-2c_3(c_4+c_5)))+c_3(-3c_3^2-10c_4c_5+5c_3(c_4+c_5))\}}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \quad (15.1)$$

$$a_{32} = \frac{\{c_3^2(c_3(3c_3^2+10c_4c_5-5c_3(c_4+c_5))-5c_1(c_3^2+6c_4c_5-2c_3(c_4+c_5)))\}}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \quad (15.2)$$

$$a_{33} = \frac{\left\{ \begin{array}{l} c_3(c_3(c_3(12c_3^2+20c_4c_5-15c_3(c_4+c_5))-5c_2(3c_3^2+6c_4c_5-4c_3(c_4+c_5)))+ \\ 5c_1(2c_2(2c_3^2+6c_4c_5-3c_3(c_4+c_5))+c_3(-3c_3^2-6c_4c_5+4c_3(c_4+c_5))) \end{array} \right\}}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \quad (15.3)$$

$$a_{34} = \frac{\{c_3^2(c_3(-10c_1c_2+5(c_1+c_2)c_3-3c_3^2))+5(6c_1c_2-2(c_1+c_2)c_3+c_3^2)c_5\}}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4+c_5)} \quad (15.4)$$

$$a_{35} = \frac{\{c_3^2(c_3(-10c_1c_2+5(c_1+c_2)c_3-3c_3^2))+5(6c_1c_2-2(c_1+c_2)c_3+c_3^2)c_4\}}{60(c_1-c_5)(c_2+c_5)(c_3+c_5)(c_4+c_5)} \quad (15.5)$$

$$a_{41} = \frac{\{((c_4^2(c_4(-10c_2c_3+5(c_2+c_3)c_4-3c_4^2))+5(6c_2c_3-2(c_2+c_3)c_4+c_4^2)c_5))\}}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \quad (16.1)$$

$$a_{42} = \frac{\{c_4^2(c_4(10c_1c_3-5(c_1+c_3)c_4+3c_4^2))-5(6c_1c_3-2(c_1+c_3)c_4+c_4^2)c_5\}}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \quad (16.2)$$

$$a_{43} = \frac{\{c_4^2(c_4(10c_1c_2-5(c_1+c_2)c_4+3c_4^2))-5(6c_1c_2-2(c_1+c_2)c_4+c_4^2)c_5\}}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \quad (16.3)$$

$$a_{44} = \frac{\left\{ \begin{array}{l} c_4(c_4(30c_1c_2c_3-20(c_2c_3+c_1(c_2+c_3))c_4+15(c_1+c_2+c_3)c_4^2-12c_4^3)+ \\ 5(-12c_1c_2c_3+6(c_2c_3+c_1(c_2+c_3))c_4-4(c_1+c_2+c_3)c_4^2+3c_4c_4^2)c_5 \end{array} \right\}}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4+c_5)} \quad (16.4)$$

$$a_{45} = \frac{\{c_4^2(30c_1c_2c_3-10(c_2c_3+c_1(c_2+c_3))c_4+5(c_1+c_2+c_3)c_4^2-3c_4c_4^2)\}}{60(c_1-c_5)(c_2+c_5)(c_3+c_5)(c_4+c_5)} \quad (16.5)$$

$$a_{51} = \frac{\{c_5^2(30c_2c_3c_4-10(c_3c_4+c_2(c_3+c_4))c_5+5(c_2+c_3+c_4)c_5^2-3c_5c_5^2)\}}{60(c_1-c_2)(c_1-c_3)(c_1-c_4)(c_1-c_5)} \quad (17.1)$$

$$a_{52} = \frac{\{c_5^2(-30c_1c_3c_4+10(c_3c_4+c_1(c_3+c_4))c_5-5(c_1+c_3+c_4)c_5^2+3c_5^3)\}}{60(c_1-c_2)(c_2-c_3)(c_2-c_4)(c_2-c_5)} \quad (17.2)$$

$$a_{53} = \frac{\{c_5^2(-30c_1c_2c_4+10(c_2c_4+c_1(c_2+c_4))c_5-5(c_1+c_2+c_4)c_5^2+3c_5^3)\}}{60(c_1-c_3)(c_2-c_3)(c_3-c_4)(c_3-c_5)} \quad (17.3)$$

$$a_{54} = \frac{\{c_5^2(-30c_1c_2c_3+10(c_2c_3+c_1(c_2+c_3))c_5-5(c_1+c_2+c_3)c_5^2+3c_5^3)\}}{60(c_1-c_4)(c_2+c_4)(c_3+c_4)(c_4+c_5)} \quad (17.4)$$

$$a_{55} = \frac{\left\{ \begin{array}{l} c_5(-60c_1c_2c_3c_4+30(c_2c_3c_4+c_1(c_3c_4+c_2(c_3+c_4)))c_5-20(c_3c_4+ \\ c_2(c_3+c_4)+c_1(c_2+c_3+c_4))c_5^2+15(c_1+c_2+c_3+c_4)c_5^3-12c_5^4) \end{array} \right\}}{60(c_1-c_5)(c_2+c_5)(c_3+c_5)(c_4+c_5)} \quad (17.5)$$

In (12.1)-(17.5), if we set  $c_i$  ( $i = 1, 2, 3, 4, 5$ ) with a specific value, then we obtain  $b_i$  ( $i = 1, 2, 3, 4, 5$ ) and  $a_{ij}$  ( $i, j = 1, 2, 3, 4, 5$ ).

In IRK methods,  $c_i$  ( $i = 1, 2, \dots, s$ ) satisfies  $\sum_{i=1}^s c_i = \frac{s}{2}$ .

Therefore,  $\sum_{i=1}^{s=5} c_i = 2 \frac{1}{2}$  for 10<sup>th</sup> order IRK.

To make our computer simulation work efficient, we design an algorithm to find  $c_i$  ( $i = 1, 2, 3, 4, 5$ ) which must

fulfill  $\sum_{i=1}^5 b_i = 1$  and  $\sum_{i=1}^5 c_i = 2 \frac{1}{2}$ . The algorithm is written in *pseudo code* as below :

#### Determining The Butcher\_Coefficients\_10<sup>th</sup>\_order\_IRK {Name of Algorithm}

{This algorithm is used to the determination simulation assess the Butcher's coefficients for the 10<sup>th</sup> order IRK method with the choice assess  $0 < c_i < 1, i = 1, 2, 3, 4, 5$ . at random}

#### {Early Condition

The  $c_i$  ( $i = 1, 2, 3, 4, 5$ ) values selected random processing by using facility Randomize. Then, using the values,  $b_i$  ( $i = 1, 2, 3, 4, 5$ ), and  $a_{ij}$  ( $i, j = 1, 2, \dots, 5$ ) values will be obtained.

The values satisfy  $\sum_{i=1}^4 b_i = 1$ ,

$$c_i = \sum_{j=1}^5 a_{ij}, \text{ and } \sum_{i=1}^5 c_i = 2.5, \quad i = 1..5. \quad (15.5)$$

#### {Final Condition

$c_i$  ( $i = 1, 2, 3, 4, 5$ ),  $b_i$  ( $i = 1, 2, 3, 4, 5$ ), and  $a_{ij}$  ( $i, j = 1, 2, 3, 4, 5$ ) values as Butcher's coefficients for the 10<sup>th</sup> order IRK method are obtained.}

#### DECLARATION

Label 10 I : integer

c1, c2, c3, c4, c5, b1, b2, b3, b4, b5, a11, a12, a13, a14, a15, a21, a22, a23, a24, a25, a31, a32, a33, a34, a35, a41, a42, a43, a44, a45, a51, a52, a53, a54, a55: real;

function f1(c1, c2, c3, c4, c5: real): real

begin

f1 setting the rhs. Of equation 12.1 here

RETURN (f1)

function f2(c1, c2, c3, c4, c5: real): real

begin

f2 setting the rhs. Of equation 12.2 here

RETURN (f2)

function f3(c1, c2, c3, c4, c5: real): real

```

begin
  f3  setting the rhs. Of equation 12.3 here
RETURN (f3)
function f4(c1,c2,c3,c4,c5:real):real
begin
  f4  setting the rhs. Of equation 12.4 here
RETURN (f4)
function f5(c1,c2,c3,c4,c5:real):real
begin
  f5  setting the rhs. Of equation 12.5 here
RETURN (f5)

```

## DESCRIPTION

```

BEGIN
10:
  Randomize
  Repeat
  c1  Random  c2  Random
  c3  Random  c4  Random
  c5  Random

```

until ((abs(c1+c2+c3+c4+c5 - 2.5) <= 0.0000001) and (c1>0) and (c2>0) and (c3>0) and (c4>0) and (c5>0) and (c1<>c2) and (c2<>c3) and (c3<>c4) and (c4<>c5) )

```

b1  f1(c1,c2,c3,c4,c5)  b2  f2(c1,c2,c3,c4,c5)
b3  f3(c1,c2,c3,c4,c5)  b4  f4(c1,c2,c3,c4,c5)
b5  f4(c1,c2,c3,c4,c5)
if ( ((abs(b1+b2+b3+b4+b5 - 1) > 0.0000001) or (b1<0)
or (b2<0) or (b3<0) or (b4<0) or (b5<0)) ) then goto 10
else

```

```

BEGIN
  a11  setting the rhs. Of equation 13.1 here
  a12  setting the rhs. Of equation 13.2 here
  a13  setting the rhs. Of equation 13.3 here
  a14  setting the rhs. Of equation 13.4 here
  a15  setting the rhs. Of equation 13.5 here
  a21  setting the rhs. Of equation 14.1 here
  a22  setting the rhs. Of equation 14.2 here
  a23  setting the rhs. Of equation 14.3 here
  a24  setting the rhs. Of equation 14.4 here
  a25  setting the rhs. Of equation 14.5 here
  a31  setting the rhs. Of equation 15.1 here
  a32  setting the rhs. Of equation 15.2 here
  a33  setting the rhs. Of equation 15.3 here
  a34  setting the rhs. Of equation 15.4 here
  a35  setting the rhs. Of equation 15.5 here
  a41  setting the rhs. Of equation 16.1 here
  a42  setting the rhs. Of equation 16.2 here
  a43  setting the rhs. Of equation 16.3 here
  a44  setting the rhs. Of equation 16.4 here
  a45  setting the rhs. Of equation 16.5 here

  a51  setting the rhs. Of equation 17.1 here
  a52  setting the rhs. Of equation 17.2 here
  a53  setting the rhs. Of equation 17.3 here
  a54  setting the rhs. Of equation 17.4 here
  a55  setting the rhs. Of equation 17.5 here

```

```

BEGIN
Write(c1, c2, c3, c4, c5,b1,b2, b3,b4,b5,
      ,a11,a12,a13,a14,a15,a21,a22,a23,a24,a25
      ,a31,a32,a33,a34,a35,a41,a42,a43,a44,a45
      ,a51,a52,a53,a54,a55)
  end
END.

```

Implementing the algorithm to TURBO PASCAL programming, we have a the 10<sup>th</sup> order of implicit Runge-Kutta methods that given in the form of a tableau containing their coefficients below

Table 1. The Butcher's coefficients for the 10<sup>th</sup> order IRK

0.00062327669	0.00063421625	0.00001105145	-0.00000999963	0.00000344788	0.00000094280
0.62262155069	0.09617192074	-0.08577858789	0.19691976268	0.44098804083	-0.02567958566
0.68561704247	0.09620099283	-0.05393518544	0.22869146117	0.44069256651	-0.02603279259
0.30589831341	0.10069140845	-0.57368596565	0.50477982176	0.31903127918	-0.04491823033
0.88523974386	0.09527844353	-0.20783782461	0.51158700582	0.44878101607	0.03743110305

## III. NUMERICAL EXPERIMENT

In this section, we present an application of our scheme to solve Henon-Heiles problem. The Hénon-Heiles problem is typically non integrable system and have simultaneously and quasi periodic solutions. The Hénon-Heiles system is defined by (see [8])

$$H = T + V, \quad T = \frac{1}{2}(p_1^2 + p_2^2), \quad V = \frac{1}{2}(q_1^2 + q_2^2) + (q_1^2 q_2) - \frac{1}{3}q_2^3 \quad (18)$$

We apply the 10<sup>th</sup> order IRK method (**IRK<sub>10</sub>**) to solve problem (18). The results of using **IRK<sub>10</sub>** is displayed in the figure 1b. To show them, we used *MATHEMATICA*.

The figures displayed are *Poincaré* sections (i.e. sections of various orbits), for an energy  $E$  and time-steps  $\dagger$ . To obtain our *Poincaré* sections, we use the idea of linear interpolation [10]. The *Poincaré* sections is the integration results as points of intersection of the flow with the  $q_1 = 0$ -plane. Here we set 62 initial points.

In all graphs, the blue dots indicate that the points are moving up to  $q_1 = 0$ -plane whereas the red dots indicate that the points are moving down the plane. Fig. 1.b shows that the scheme of 10<sup>th</sup> IRK method can be used to solve Hénon-Heiles problem (18).

In Fig. 1.a and Fig. 1.b, we show a poincaré section for Hénon-Heiles (18) using the 10<sup>th</sup> order IRK method whose tableau are given in Table.1 and a standard 4<sup>th</sup> order explicit Runge-Kutta methods, respectively. We set an energy  $E=1.25$ ,  $\dagger = 0.01$ , and the number of iterations = 100,000 for producing the figures.

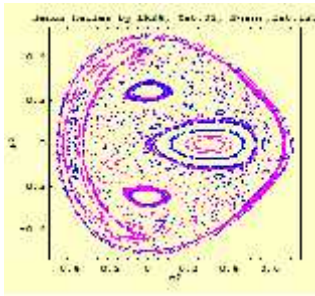


Fig. 1a. An Hénon-Heiles's *Poincaré* sections was produced by using a standard 4<sup>th</sup> order ERK method.

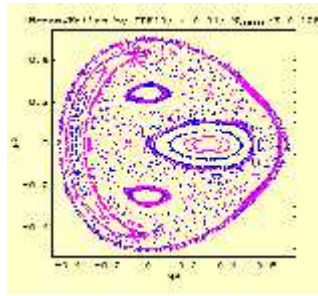


Fig. 1b. An Hénon-Heiles's *Poincaré* sections was produced by using an IRK<sub>10</sub> method.

#### IV. CONCLUSION

Based on theoretical surveying, algorithm designing and implementing, and numerical problem solving before, we conclude that we can construct a scheme of 10<sup>th</sup> order of IRK methods via simulating to choose  $c_i$  ( $i = 1..5$ ) values that

satisfy  $\sum_{i=1}^{s=5} c_i = 2 \frac{1}{2}$ ,  $\sum_{i=1}^5 b_i = 1$ , and  $c_i = \sum_{j=1}^5 a_{ij}$ ,  $i = 1..5$ .

Although, we have not exact values for  $c_i$ ,  $b_i$ , and  $a_{ij}$ , ( $i, j = 1, 2, 3, 4$ ), but our schemes can solve a problem of the first order of ordinary differential equation system like classical methods (Gauss-Legendre for example).

#### ACKNOWLEDGMENT

The first authors sincerely wish to acknowledge the continuing help, advice, and influence of the second and the third authors. They are constantly interest and enthusiasm in all parts of this paper. Thanks are also due to Riau University (staff and service employees) through Computer Laboratory FMIPA UR for funding and servicing all of our research needs. Finally, we thanks to our family for their moral support. We are deeply indebted to them for this.

#### REFERENCES

- [1] L. Zakaria, "A Numerical Technique to Obtain Scheme of 8<sup>th</sup> Order Implicit Runge-Kutta Method to Solve the First Order of Initial Value Problems," in *Proceeding of IndoMS International Conference on Mathematics and Applications (IICMA)*, Yogyakarta, 2009, pp 425-434
- [2] G. R. W. Quispel and C. Dyt, "Volume-preserving integrators have linear error growth," *Physics Letters A*, **242**, 1998, 25-30.
- [3] J.M. Sanz-Serna and M.P. Calvo, *Numerical Hamiltonian Problems*, Chapman & Hall, New York, 1994, pp. 27-54
- [4] G.S. Turner, "Three-Dimensional Reversible Mappings," Ph.D. dissertation, Dept. Math., La Trobe Univ., Australia. 1994
- [5] L. Zakaria, "Applying Linear Interpolating To Show Poincaré Section of The Hénon-Heiles," *Mathematics and Its Learning Journal. Special Edition*, Malang State University, 2002, 1003-1008.
- [6] J.C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*. Wiley, New York, 1987, pp 51-316
- [7] E. Hairer, R.I. McLachlan, and A. Razakarivony, "Achieving Brouwer's law with implicit Runge-Kutta methods," Research Report, 2007
- [8] F. Ismail, "Sixth Order Singly Diagonally Implicit Runge-Kutta Nyström Method with Explicit First Stage for Solving Second Order Ordinary Differential Equations," *European Journal of Scientific Research*, **26**, 470-479, 2009.
- [9] H. Yoshida, "Construction of higher order symplectic integrators," *Physics Letters A*, **150**, 1990, 262-268.
- [10] F.M. Lasagni, "Canonical Runge-Kutta methods," *ZAMP*, **39**, 1988, 952-953