Error Free Butcher Algorithms for Linear Electrical Circuits

K. Murugesan, N.P. Gopalan, and Devarajan Gopal

In this paper, an error-free Butcher algorithm is introduced to study the singular system of a linear electrical circuit for time invariant and time varying cases. The discrete solutions obtained using Runge-Kutta (RK)-Butcher algorithms are compared with the exact solutions of the electrical circuit problem and are found to be very accurate. Stability regions for the single term Walsh series (STWS) method and the RK-Butcher algorithm are presented. Error graphs for inductor currents and capacitor voltages are presented in a graphical form to show the efficiency of the RK-Butcher algorithm. This RK-Butcher algorithm can be easily implemented in a digital computer for any singular system of electrical circuits.

Keywords: Singular systems, single term Walsh series (STWS) and RK-Butcher algorithms.

Manuscript received May 10, 2004; revised Dec. 27, 2004.

K. Murugesan (phone: + 82 51 510 2207, email: kmnitt@yahoo.com) is with Department of Mathematics, Pusan National University, Busan, Korea.

N.P. Gopalan (email: gopalan@nitt.edu) is with Department of Computer Science and Engineering, National Institute of Technology, India.

Devarajan Gopal (email: gopaldev66@yahoo.co.in) is with Department of Computer Electronics and Telecommunication, Woosong University, Daejeon, Korea.

I. Introduction

Alexander and Coyle [1], Evans [2], Evans and Yaakub [3], [4], Lambert [5], and Murugesan and others [6]-[10] have applied RK methods to determine numerical solutions of problems modeled as initial value problems involving differential equations that arise in the fields of science and engineering. Though the RK method had been introduced at the turn of the 20th century, research in this area is still very active and its applications are enormous. This is because of the method's nature of extending accuracy in the determination of approximate solutions and its flexibility.

Runge-Kutta methods have also become very popular both as computational techniques and as subject for research, as discussed by Butcher [11], [12] and Shampine [13]. This method was derived by Runge around the year 1894 and extended by Kutta a few years later. They developed algorithms to solve differential equations efficiently and give solutions closer to the equations' exact solutions.

Runge-Kutta (RK) algorithms have always been considered superb tools for the numerical integration of ordinary differential equations (ODEs). The fact that RK methods are self-starting, easy to program, and show extreme accuracy and versatility in ODE problems has led to their continuous analysis and use in mathematical and engineering research. One of the most exciting developments in RK usage has been the discovery that, by a judicious re-arrangement of the interim values of the RK predictors, one can obtain a second predictor of one order less. These two equations are generally referred to as an RK pair. Fehlberg [14] was among the first to suggest on theoretical grounds that the difference between the two predictors would be directly proportional to the local truncation error (LTE). The unusual success of the Fehlberg approach was addressed in the popular text by Forsythe and others [15] and cited as the "state of the art" of the RK code. The LTE is then used as a test to see whether a step has been successful. If not, the step size is reduced (usually halved) until the LTE passes the tolerance requirement. The beauty of the RK pair is that it requires no extra function evaluations, which is the most time consuming aspect of all ODE solvers. This breakthrough has initiated a search for RK algorithms of higher and higher order for better error estimates.

Butcher [11] derived the best RK pair along with an error estimate by all statistical measures, which appeared as the RK-Butcher algorithms. Morris Bader [16], [17] introduced the RK-Butcher algorithms for finding the truncation error estimates and intrinsic accuracies, and for the early detection of stiffness in coupled differential equations that arises in theoretical chemistry problems. Recently, Murugesan and others [18] and Park and others [19] applied the RK-Butcher algorithm for finding the numerical solution of an industrial robot arm control problem and optimal control of singular systems. In this paper, we introduce the RK-Butcher algorithm for the singular system of a linear electrical circuit problem with more accuracy for time-invariant and time varying cases.

II. RK-Butcher Algorithms

The normal order of an RK algorithm is the approximate number of leading terms of an infinite Taylor series, which calculates the trajectory of a moving point, as discussed by Shampine and Gordon [20]. The remainder of the excluded infinite sum is referred to as the local truncation error (LTE). The RK algorithms are forward looking predictors; that is, they use no information from preceding steps to predict the future position of a point. For this reason, they require a minimum of input data and consequently are very easy to program and simple to use.

The general p-stage Runge-Kutta method for solving an initial value problem

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

with the initial condition $y(x_0) = y_0$ is defined by

$$y_{n+1} = y_n + h\sum_{i=1}^{p} b_i k_i ,$$

where

$$k_i = f\left(x_n + c_i h, y_n + h\sum_{j=1}^{p} a_{ij} k_j \right)$$

and

$$c_i = \sum_{j=1}^{p} a_{ij}, \quad i = 1,2,\ldots,p ,$$

with c and b as p dimensional vectors and $A(a_{ij})$ as the p×p matrix. Then, the RK-Butcher algorithm of (1) is of the following form.

$$\left.\begin{aligned}
&k_1 = hf\left(x_n, y_n\right)\\
&k_2 = hf\left(x_n + \frac{h}{4}, y_n + \frac{k_1}{4}\right)\\
&k_3 = hf\left(x_n + \frac{h}{4}, y_n + \frac{k_1}{8} + \frac{k_2}{8}\right)\\
&k_4 = hf\left(x_n + \frac{h}{2}, y_n - \frac{k_2}{2} + k_3\right)\\
&k_5 = hf\left(x_n + \frac{3h}{4}, y_n + \frac{3k_1}{16} + \frac{9k_4}{16}\right)\\
&k_6 = hf\left(x_n + h, y_n - \frac{3k_1}{7} + \frac{2k_2}{7} + \frac{12k_3}{7} - \frac{12k_4}{7} + \frac{8k_5}{7}\right)\\
&\text{5th order predictor :}\\
&\quad y_{n+1} = y_n + \frac{1}{90}\left(7k_1 + 32k_3 + 12k_4 + 32k_5 + 7k_6\right)\\
&\text{4th order predictor :}\\
&\quad y^*_{n+1} = y_n + \frac{1}{6}\left(k_1 + 4k_4 + k_6\right)\\
&\text{Local truncation error estimate (EE):}\\
&\quad EE = y_{n+1} - y^*_{n+1}
\end{aligned}\right\} \qquad (2)$$

The formation of the Butcher array of (2) then takes the following form.

$$\begin{array}{c|cccccc}
0 & & & & & & \\
\frac{1}{4} & \frac{1}{4} & & & & & \\
\frac{1}{4} & \frac{1}{8} & \frac{1}{8} & & & & \\
\frac{1}{2} & 0 & -\frac{1}{2} & 1 & & & \\
\frac{3}{4} & \frac{3}{16} & 0 & 0 & \frac{9}{16} & & \\
0 & -\frac{3}{7} & \frac{2}{7} & \frac{12}{7} & -\frac{12}{7} & \frac{8}{7} & \\
\hline
 & \frac{7}{90} & 0 & \frac{32}{90} & \frac{12}{90} & \frac{32}{90} & \frac{7}{90} \\
 & \frac{1}{6} & 0 & 0 & \frac{4}{6} & 0 & \frac{1}{6}
\end{array}$$

This Butcher array plays a vital role in the study of stability regions and is presented in the later sections.

III. Study of Linear Electrical Circuit

Consider the physical model of an electrical circuit discussed by Chua and Lin [21] and Balachandran and Murugesan [22], as shown in Fig. 1.

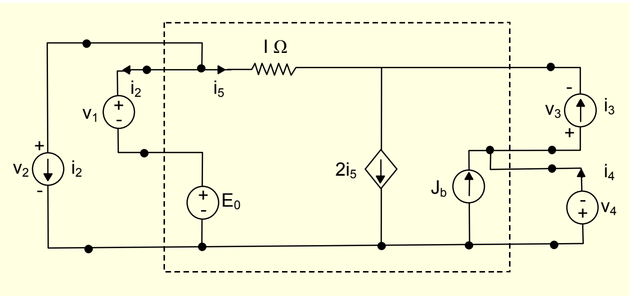

Fig. 1. Electrical circuit.

This electrical circuit is governed by the following hybrid equations [23].

$$\begin{bmatrix} i_1 \\ i_4 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_4 \\ i_2 \\ i_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -1 \\ 1 & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} E_a \\ J_b \end{bmatrix} \qquad (3)$$

Since $i_c = C\dot{v}_c$ and $v_l = L\dot{i}_l$, substituting $i_1 = 2\dot{v}_1$, $i_2 = 2\dot{v}_2$, $v_3 = 2\dot{i}_3$, and $v_4 = 2\dot{i}_4$ into (3) we then obtain

$$\begin{bmatrix} 2\dot{v}_1 \\ i_4 \\ v_2 \\ 2\dot{i}_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ 2\dot{i}_4 \\ 2\dot{v}_2 \\ i_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -1 \\ 1 & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} E_a \\ J_b \end{bmatrix}. \qquad (4)$$

After re-arranging the terms, we obtain the singular system of equations as

$$\begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i}_3 \\ \dot{i}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ i_3 \\ i_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -1 & 0 \\ 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} E_a \\ J_b \end{bmatrix}. \qquad (5)$$

$$\left.\begin{aligned} &\text{This is of the form} \\ &\qquad K\dot{x}(t) = Ax(t) + Bu(t) \\ &\text{with the initial condition} \\ &\qquad x(0) = x_0, \end{aligned}\right\} \qquad (6)$$

where K is an $n \times n$ matrix, but singular in nature; therefore it is called **singular systems**. It is also called “generalized state-space systems” or “descriptor systems.” A is an $n \times n$ matrix, B is an $n \times r$ matrix, $x(t)$ is an n-state vector, and $u(t)$ is an r-input vector.

In some cases, the variables have some inherent meaning such as voltage, current, position, velocity, or acceleration. Or, the coefficient matrices have some special structures that may be lost by manipulating a system of the form in (6) into an ordinary state-space system.

$$\left.\begin{aligned} &\text{By taking } E_a = 1 + t + \frac{t^2}{2} + \frac{t^3}{3} \\ &\text{and } J_b = 1 + t + t^2 \text{ in (5),} \end{aligned}\right\} \qquad (7)$$

we obtain the exact solutions of (5) as

$$\left.\begin{aligned} v_1(t) &= -\frac{93}{2}\left(1-\sqrt{5}\right)\exp\left(\frac{1+\sqrt{5}}{8}\right)t \\ &\quad -\frac{93}{2}\left(1+\sqrt{5}\right)\exp\left(\frac{1-\sqrt{5}}{8}\right)t \\ &\quad -27t + \frac{3t^2}{2} - \frac{t^3}{3} + 163 \\ v_2(t) &= -\frac{93}{2}\left(1-\sqrt{5}\right)\exp\left(\frac{1+\sqrt{5}}{8}\right)t \\ &\quad -\frac{93}{2}\left(1+\sqrt{5}\right)\exp\left(\frac{1-\sqrt{5}}{8}\right)t \\ &\quad -26t + 2t^2 + 164 \\ i_3(t) &= -93\exp\left(\frac{1+\sqrt{5}}{8}\right)t - 93\exp\left(\frac{1-\sqrt{5}}{8}\right)t \\ &\quad -14t + 2t^2 + 106 \\ i_4(t) &= -93\exp\left(\frac{1+\sqrt{5}}{8}\right)t - 93\exp\left(\frac{1-\sqrt{5}}{8}\right)t \\ &\quad -15t + t^2 + 105 \\ &\text{with} \\ &[v_1(0)\ v_2(0)\ i_3(0)\ i_4(0)]^T = [70\ \ 71\ \ -80\ \ -81]^T \end{aligned}\right\} \qquad (8)$$

The discrete solutions of (5) with $x(t)=[v_1(t)\ v_2(t)\ i_3(t)\ i_4(t)]^T$ are evaluated using the RK-Butcher algorithms represented in (2), and the results are compared with the solutions obtained by the STWS method by Balachandran and Murugesan [22] and are shown in Tables 1 through 4 along with the exact solutions calculated using (8). An error graph is presented for the variables $v_1(t)$, $v_2(t)$, $i_3(t)$ and $i_4(t)$ in Figs. 2 through 5 at various time intervals.

Table 1. Solutions of (2) and (8) for $v_1(t)$.

Solution number	Time t (s)	$v_1(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	70.000000	70.000000	0.000000	70.000000	0.000000
2	0.25	75.156776	75.156799	0.000003	75.156776	0.000000
3	0.50	80.904671	80.904674	0.000003	80.904671	0.000000
4	0.75	87.289886	87.289889	0.000003	87.289886	0.000000
5	1.00	94.365692	94.365698	0.000006	94.365692	0.000000
6	1.25	102.193207	102.193215	0.000008	102.193207	0.000000
7	1.50	110.842285	110.842396	0.000011	110.842285	0.000000
8	1.75	120.392502	120.392517	0.000015	120.392502	0.000000
9	2.00	130.934204	130.934219	0.000015	130.934204	0.000000

Table 2. Solutions of (2) and (8) for $v_2(t)$.

Solution number	Time t (s)	$v_2(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	71.000000	71.000000	0.000000	71.000000	0.000000
2	0.25	76.443237	76.443240	0.000003	76.443237	0.000000
3	0.50	82.571335	82.571338	0.000003	82.571342	0.000000
4	0.75	89.461761	89.461764	0.000003	89.461769	0.000000
5	1.00	97.199028	97.199034	0.000006	97.199028	0.000000
6	1.25	105.875496	105.875504	0.000008	105.875496	0.000000
7	1.50	115.592285	115.592296	0.000011	115.592285	0.000000
8	1.75	126.460213	126.460228	0.000015	126.460205	0.000000
9	2.00	138.600861	138.600876	0.000015	138.600861	0.000000

Table 3. Solutions of (2) and (8) for $i_3(t)$.

Solution number	Time t (s)	$i_3(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	-80.000000	-80.000000	0.000000	-80.000000	0.000000
2	0.25	-89.747978	-89.747979	0.000001	-89.747978	0.000000
3	0.50	-100.432671	-100.432674	0.000003	-100.432671	0.000000
4	0.75	-112.161095	-112.161101	0.000006	-112.161095	0.000000
5	1.00	-125.052383	-125.052391	0.000008	-125.052383	0.000000
6	1.25	-139.239059	-139.239072	0.000013	-139.239059	0.000000
7	1.50	-154.868408	-154.868422	0.000014	-154.868423	0.000000
8	1.75	-172.104084	-172.104099	0.000015	-172.104084	0.000000
9	2.00	-191.127686	-191.127701	0.000015	-191.127701	0.000000

Table 4. Solutions of (2) and (8) for $i_4(t)$.

Solution number	Time t (s)	$i_4(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	-81.000000	-81.000000	0.000000	-81.000000	0.000000
2	0.25	-91.060478	-91.060481	0.000003	-91.060478	0.000000
3	0.50	-102.182671	-102.182674	0.000003	-102.182671	0.000000
4	0.75	-114.473595	-114.473598	0.000003	-114.473595	0.000000
5	1.00	-128.052383	-128.052389	0.000006	-128.052383	0.000000
6	1.25	-143.051559	-143.051570	0.000011	-143.051559	0.000000
7	1.50	-159.618408	-159.618419	0.000011	-159.618423	0.000000
8	1.75	-177.916580	-177.916593	0.000013	-177.916580	0.000000
9	2.00	-198.127686	-198.127701	0.000015	-198.127701	0.000000

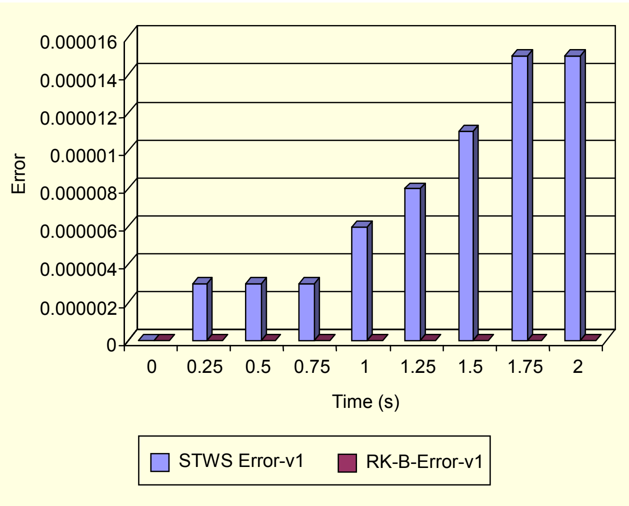

Fig. 2. Error graph for $v_1(t)$.

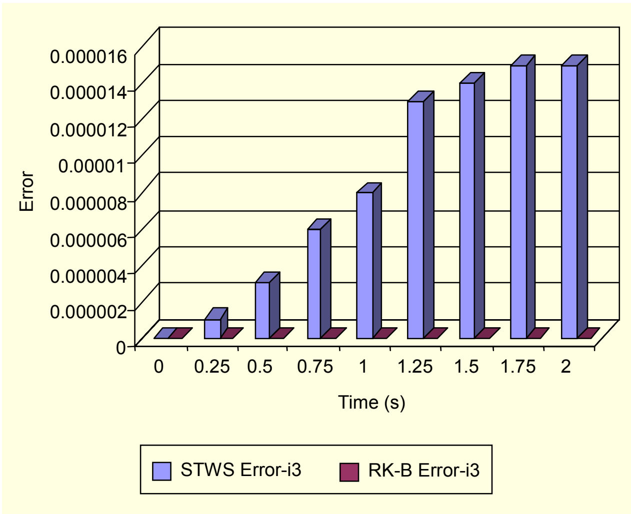

Fig. 4. Error graph for $i_3(t)$.

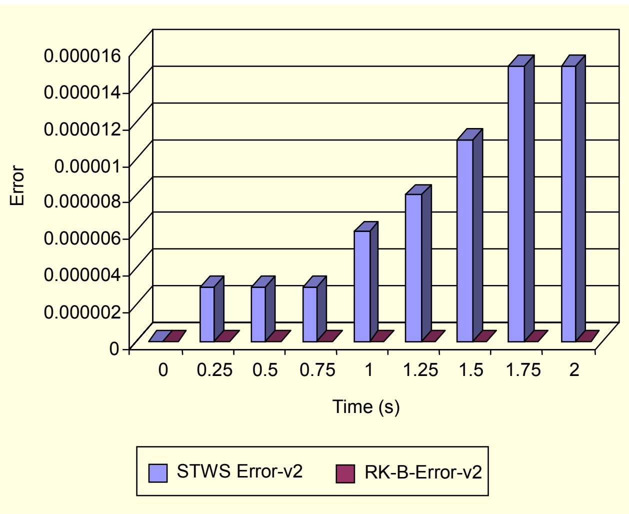

Fig. 3. Error graph for $v_2(t)$.

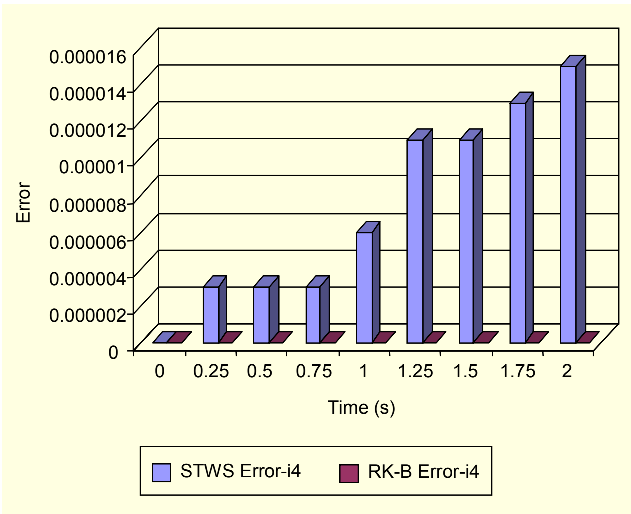

Fig. 5. Error graph for $i_4(t)$.

IV. Stability Analyses

Consider test equation $\dot{y} = \lambda y$, where λ is a constant, is complex in nature, and is used to determine the stability region of the method.

$$k_1 = f(y_n) = \lambda y_n$$

$$k_2 = f\left(y_n + \frac{hk_1}{4}\right) = \lambda y_n\left(1 + \frac{h\lambda}{4}\right)$$

$$k_3 = f\left(y_n + \frac{hk_1}{8} + \frac{hk_2}{8}\right) = \lambda y_n\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right)$$

$$k_4 = f\left(y_n - \frac{hk_2}{2} + hk_3\right) = \lambda y_n\left(1 - \frac{h\lambda}{2}\left(1 + \frac{h\lambda}{4}\right) + h\lambda\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right)\right)$$

$$k_5 = f\left(y_n + \frac{3hk_1}{16} + \frac{9hk_4}{16}\right) = \lambda y_n\left(1 + \frac{3h\lambda}{16} + \frac{9h\lambda}{16}\left(\begin{array}{l} 1 - \frac{h\lambda}{2}\left(1 + \frac{h\lambda}{4}\right) \\ + h\lambda\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right)\end{array}\right)\right)$$

$$k_6 = f\left(y_n - \frac{3hk_1}{7} + \frac{2hk_2}{7} + \frac{12hk_3}{7} - \frac{12hk_4}{7} + \frac{8hk_5}{7}\right)$$

$$= \lambda y_n\left(\begin{array}{l} 1 - \frac{3h\lambda}{7} + \frac{2h\lambda}{7}\left(1 + \frac{h\lambda}{4}\right) + \frac{12h\lambda}{7}\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right) \\ - \frac{12h\lambda}{7}\left(1 - \frac{h\lambda}{2}\left(1 + \frac{h\lambda}{4}\right) + h\lambda\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right)\right) \\ + \frac{8h\lambda}{7}\left(1 + \frac{3h\lambda}{16} + \frac{9h\lambda}{16}\left(\begin{array}{l} 1 - \frac{h\lambda}{2}\left(1 + \frac{h\lambda}{4}\right) \\ + h\lambda\left(1 + \frac{h\lambda}{8} + \frac{h\lambda}{8}\left(1 + \frac{h\lambda}{4}\right)\right)\end{array}\right)\right)\end{array}\right)$$

Substituting $z = h\lambda$ we get

$$k_1 = f(y_n) = \lambda y_n$$

$$k_2 = \lambda y_n\left(1 + \frac{z}{4}\right)$$

$$k_3 = \lambda y_n\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right)$$

$$k_4 = \lambda y_n\left(1 - \frac{z}{2}\left(1 + \frac{z}{4}\right) + z\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right)\right)$$

$$k_5 = \lambda y_n\left(1 + \frac{3z}{16} + \frac{9z}{16}\left(1 - \frac{z}{2}\left(1 + \frac{z}{4}\right) + z\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right)\right)\right)$$

$$k_6 = \lambda y_n\left(\begin{array}{l} 1 - \frac{3z}{7} + \frac{2z}{7}\left(1 + \frac{z}{4}\right) + \frac{12z}{7}\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right) \\ - \frac{12z}{7}\left(1 - \frac{z}{2}\left(1 + \frac{z}{4}\right) + z\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right)\right) \\ + \frac{8z}{7}\left(1 + \frac{3z}{16} + \frac{9z}{16}\left(1 - \frac{z}{2}\left(1 + \frac{z}{4}\right) + z\left(1 + \frac{z}{8} + \frac{z}{8}\left(1 + \frac{z}{4}\right)\right)\right)\right)\end{array}\right)$$

Then, the 5th order predictor formula is

$$y_{n+1} = y_n + \frac{h}{90}\left(7k_1 + 32k_3 + 12k_4 + 32k_5 + 7k_6\right).$$

Substituting the values of k_1, k_2, k_3, k_4, k_5 and k_6, we then obtain

$$y_{n+1} = y_n + \frac{h\lambda y_n}{90}\left(90 + \frac{90}{2}z + \frac{30}{2}z^2 + \frac{30}{8}z^3 + \frac{30}{40}z^4 + \frac{30}{240}z^5\right).$$

If we divide both sides by y_n, the stability polynomial $Q(z) = y_{n+1}/y_n$ is then given as

$$Q(z) = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \frac{z^4}{4!} + \frac{z^5}{5!} + \frac{z^6}{6!}.$$

Figure 6 shows a comparative study of the stability regions of the STWS method and the RK-Butcher algorithm. In this stability region, the range for the real part of λ in STWS is $-3.284 < \mathrm{Re}(z) < 0.0$, whereas in the RK-Butcher algorithm it is $-2.780 < \mathrm{Re}(z) < 0.0$.

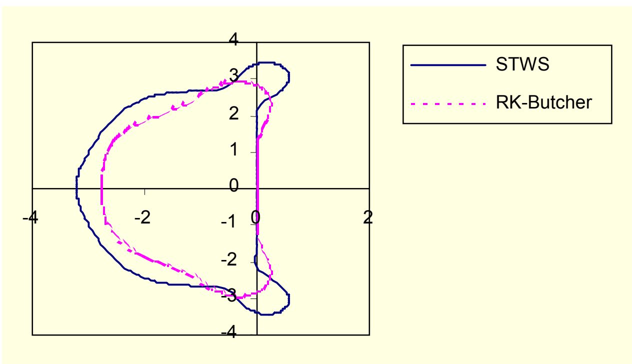

Fig. 6. Stability region for STWS and the RK-Butcher algorithms.

V. Study of Time-Varying Linear Electrical Circuit

Balachandran and Murugesan [22] applied the STWS method for the time–invariant electrical circuit problem. Here, we introduce the RK-Butcher algorithm for studying the time-varying electrical circuit, which is represented by a singular system.

Consider the electrical circuit depicted in Fig. 1 in section III. The following hybrid equation is obtained.

$$\begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i}_3 \\ \dot{i}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \\ i_3 \\ i_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -1 & 0 \\ 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} E_a \\ J_b \end{bmatrix} \tag{9}$$

This is of the form

$$K\dot{x}(t) = Ax(t) + Bu(t)\,. \tag{10}$$

In order to study the effectiveness of the time varying singular system in electrical circuits, a hypothetical system is formed by transforming the matrices K, A, and B, which are basically time independent in (9) with time-varying components.

Hence, the singular system of the time-varying electrical circuit is of the form

$$\begin{bmatrix} 2 & 2 & 0 & 0 \\ 0 & 0 & 2t & 2t \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{i}_3 \\ \dot{i}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -t & 0 \\ 0 & 0 & t & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & t & -t \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \\ i_3 \\ i_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -t & 0 \\ t & 0 \\ 0 & -t \end{bmatrix}\begin{bmatrix} t \\ \cos(t) \end{bmatrix}. \tag{11}$$

This is of the form $K(t)\dot{x}(t) = A(t)x(t) + B(t)u(t)$.

The exact solution of (11) is

$$\left.\begin{aligned} v_1 &= \frac{-t^3}{12} - t^2 - \frac{(4-t)\cos(t) + (4t+1)\sin(t)}{34} \\ &\quad + \frac{59(t-4)e^{\frac{t}{4}}}{17} + \frac{255}{17} \\ v_2 &= v_1 + t^2 \\ i_3 &= (t+4) + \frac{2}{17}[\sin(t) + 4\cos(t)] - \frac{59e^{\frac{t}{4}}}{17} \\ i_4 &= i_3 - \cos(t) \\ &\text{with initial conditions} \\ &[v_1(0)\;\; v_2(0)\;\; v_3(0)\;\; i_4(0)]^T = [1\;\; 1\;\; 1\;\; 0]^T. \end{aligned}\right\} \tag{12}$$

The discrete solutions of (11) have been determined using the RK-Butcher algorithm (with step-size h=0.25) in (2) and are compared with the exact solutions of (11) presented in (12) along with the solutions obtained by using the STWS method. These results are presented in Tables 5 through 8. This RK-Butcher algorithm yields more accurate results when compared to the STWS method. Errors between the exact and discrete solutions are also given in Tables 5 through 8. To exhibit the efficiency of the discussed methods, an error graph is presented for the variables $v_1(t)$, $v_2(t)$, $i_3(t)$ and $i_4(t)$ in Figs. 7 through 10 at various time intervals. From this, we can observe that the RK-Butcher algorithm gives more accurate results when compared to the STWS method discussed by Balachandran and Murugesan [22].

Table 5. Solutions of (2) and (12) for $v_1(t)$.

Solution number	Time t (s)	Time varying $v_1(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	1.000000	1.000000	0.000000	1.000000	0.000000
2	0.25	0.960697	0.960700	0.000003	0.960697	0.000000
3	0.50	0.842521	0.842524	0.000003	0.842521	0.000000
4	0.75	0.646642	0.646646	0.000004	0.646642	0.000000
5	1.00	0.376276	0.376284	0.000008	0.376276	0.000000
6	1.25	0.036505	0.036516	0.000011	0.036505	0.000000
7	1.50	-0.366008	-0.366021	0.000013	-0.366008	0.000000
8	1.75	-0.823387	-0.823400	0.000013	-0.823387	0.000000
9	2.00	-1.326949	-1.326964	0.000015	-1.326949	0.000000

Table 6. Solutions of (2) and (12) for $v_2(t)$.

Solution number	Time t (s)	Time varying $v_2(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	1.000000	1.000000	0.000000	1.000000	0.000000
2	0.25	1.023197	1.023200	0.000003	1.023197	0.000000
3	0.50	1.092521	1.092524	0.000003	1.092521	0.000000
4	0.75	1.209142	1.206145	0.000003	1.209142	0.000000
5	1.00	1.376276	1.376280	0.000004	1.376276	0.000000
6	1.25	1.599005	1.599011	0.000006	1.599005	0.000000
7	1.50	1.883992	1.884000	0.000008	1.883992	0.000000
8	1.75	2.239113	2.239124	0.000011	2.239113	0.000000
9	2.00	2.673051	2.673066	0.000015	2.673051	0.000000

Table 7. Solutions of (2) and (12) for $i_3(t)$.

Solution number	Time t (s)	Time varying $i_3(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	1.000000	1.000000	0.000000	1.000000	0.000000
2	0.25	1.040643	1.040643	0.000000	1.040643	0.000000
3	0.50	1.036691	1.036694	0.000003	1.036691	0.000000
4	0.75	0.988188	0.988191	0.000003	0.988188	0.000000
5	1.00	0.896933	0.896936	0.000003	0.896933	0.000000
6	1.25	0.766301	0.766308	0.000007	0.766301	0.000000
7	1.50	0.600964	0.600975	0.000011	0.600964	0.000000
8	1.75	0.406530	0.406543	0.000013	0.406530	0.000000
9	2.00	0.189110	0.189125	0.000015	0.189110	0.000000

Table 8. Solutions of (2) and (12) for $i_4(t)$.

Solution number	Time t (s)	Time varying $i_4(t)$				
		Exact solution	STWS solution	STWS error	RK-Butcher solution	RK-Butcher error
1	0.00	0.000000	0.000000	0.000000	0.000000	0.000000
2	0.25	0.071731	0.071734	0.000003	0.071731	0.000000
3	0.50	0.159109	0.159112	0.000003	0.159109	0.000000
4	0.75	0.256500	0.256503	0.000003	0.256500	0.000000
5	1.00	0.356631	0.356637	0.000006	0.356631	0.000000
6	1.25	0.450978	0.450986	0.000008	0.450978	0.000000
7	1.50	0.530227	0.530238	0.000011	0.530227	0.000000
8	1.75	0.584776	0.584789	0.000013	0.584776	0.000000
9	2.00	0.605257	0.6050272	0.000015	0.605257	0.000000

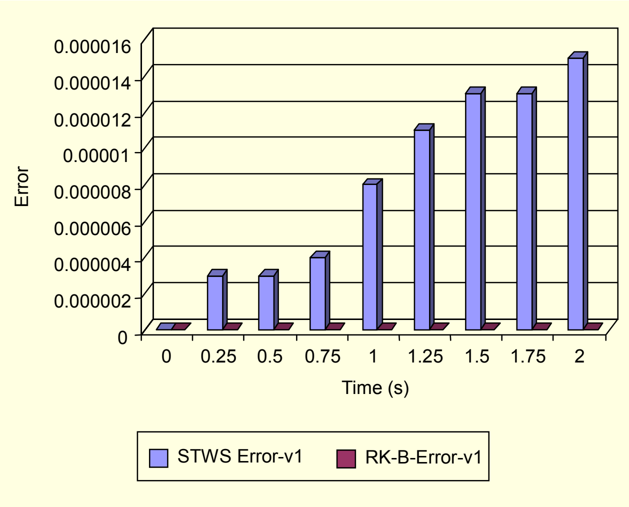

Fig. 7. Error graph of $v_1(t)$ for time varying cases.

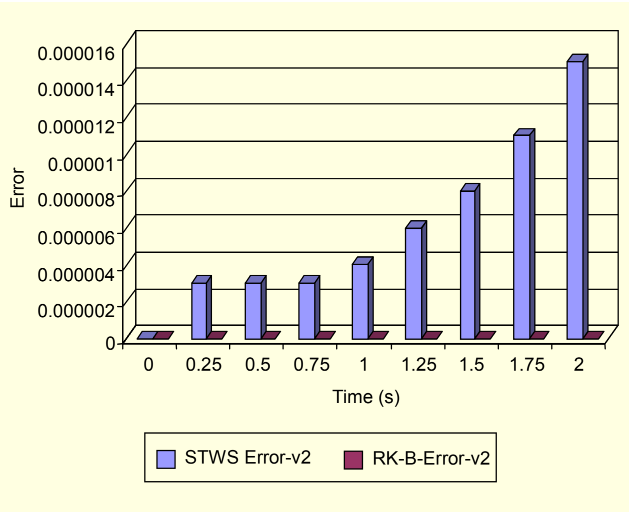

Fig. 8. Error graph of $v_2(t)$ for time varying cases.

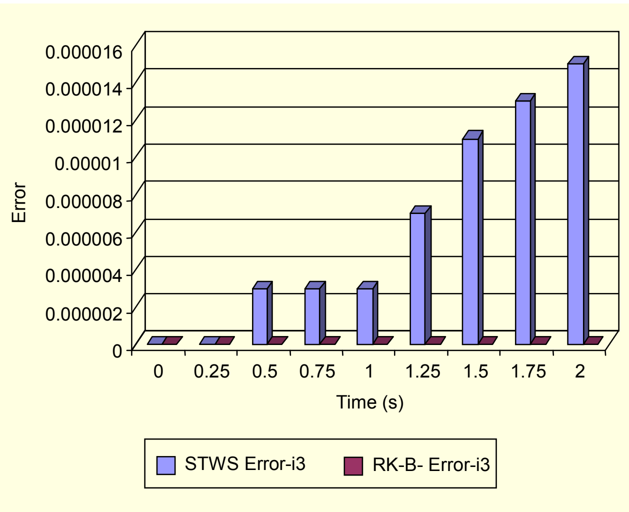

Fig. 9. Error graph of $i_3(t)$ for time varying cases.

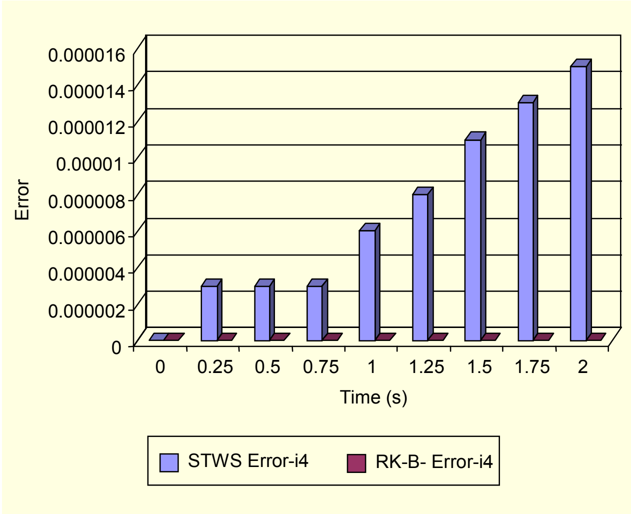

Fig. 10. Error graph of $i_4(t)$ for time varying cases.

VI. Conclusions

The discrete solutions obtained using the RK-Butcher algorithm give more accurate values when compared to the STWS method discussed by Balachandran and Murugesan [22]. From Tables 1 through 8, we observe that the solutions obtained by the RK-Butcher algorithm match well with the exact solutions of the electrical circuit problem irrespective of whether they are time-invariant or time varying cases, but the STWS method yields a little error. From the error graphs presented in Figs. 2 through 5 and 7 through 10, we can observe that the RK-Butcher algorithm yields much less error (almost no error) when compared to the STWS method.

From Fig. 6, we can predict that the RK-Butcher algorithm is useful to smaller time steps, and the STWS method is useful for larger time steps to solve the same electrical circuit problem. Moreover, this RK-Butcher method is highly stable because it is based on the Taylor series method [11], [12], which is also highly stable.

Acknowledgments

One of the authors, Dr. K. Murugesan, would like to thank Professor Jong Yeoul Park, Department of Mathematics, Pusan National University, Pusan 609-735, Republic of Korea, for his kind help during his stay in Pusan National University. Also, he would like to thank the Korean Science and Engineering Foundation (KOSEF), Korea, for the financial assistance provided to him under the Brain Pool Program.

References

[1] R.K. Alexander and J.J. Coyle, "Runge-Kutta Methods for

Differential-Algebric Systems," *SIAM J. of Numerical Analysis*, vol. 27, no. 3, 1990, pp. 736-752.

[2] D.J. Evans, "A New 4th Order Runge-Kutta Method for Initial Value Problems with Error Control," *Int.l J. Computer Mathematics*, vol. 139, 1991, pp. 217-227.

[3] D.J. Evans and A.R. Yaakub, "A New Fifth Order Weighted Runge-Kutta Formula," *Int. J. Computer Mathematics*, vol. 59, 1996, pp. 227-243.

[4] D.J. Evans and A.R. Yaakub, "Weighted Fifth Order Runge-Kutta Formulas for Second Order Differential Equations," *Int. J. Computer Mathematics*, vol. 70, 1998, pp. 233-239.

[5] J.D. Lambert, *Numerical Methods for Ordinary Differential Systems. The Initial Value Problem*, John Wiley & Sons, Chichester, UK, 1991.

[6] K. Murugesan, D. Paul Dhayabaran, and D.J. Evans, "Analysis of Different Second Order Systems via Runge-Kutta Method," *Int. J. Computer Mathematics*, vol. 70, 1999, pp. 477-493.

[7] K. Murugesan, D. Paul Dhayabaran, and D.J. Evans, "Analysis of Different Second Order Multivariable Linear System Using Single Term Walsh Series Technique and Runge-Kutta Method," *Int. J. of Computer Mathematics*, vol. 72, 1999, pp. 367-374.

[8] K. Murugesan, D. Paul Dhayabaran, and D.J. Evans, "Analysis of Non-Linear Singular System from Fluid Dynamics Using Extended Runge-Kutta Methods," *Int. J. Computer Mathematics,* vol. 76, 2000, pp. 239-266.

[9] K. Murugesan, D. Paul Dhayabaran, E.C. Henry Amirtharaj, and D.J. Evans, "A Comparison of Extended Runge-Kutta Formulae Based on Variety of Means to Solve System of IVPs," *Int.J. of Computer Mathematics*, vol. 78, 2001, pp. 225-252.

[10] K. Murugesan, D. Paul Dhayabaran, E.C. Henry Amirtharaj and D.J. Evans, "A Fourth Order Embedded Runge-Kutta RKACeM(4,4) Method Based on Arithmetic and Centrodial Means with Error Control," *Int. J. Computer Mathematics,* vol. 79, no. 2, 2002, pp. 247-269.

[11] J.C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*: *Runge-Kutta and General Linear Methods*, John Wiley & Sons, UK, 1987.

[12] J.C. Butcher, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, UK, 2003.

[13] L.F. Shampine, *Numerical Solution of Ordinary Differential Equations*, Chapman & Hall, New York, USA, 1994.

[14] E. Fehlberg, "Klassische Runge-Kutta-Formeln Vierter und Niedriger Ordnung Mit Schrittweitenkontolle und Ihre Anwendung auf Warmeleitungsprobleme," *Computing,* vol. 6, 1970, pp. 61-71.

[15] G.E. Forsythe, M.A. Malcolm, and C.D. Moler, *Computer Methods for Mathematical Computations,* Englewood Cliffs. NJ: Prentice-Hall, New Jersey, p.135. 1977.

[16] M. Bader, "A Comparative Study of New Truncation Error Estimates and Intrinsic Accuracies of Some Higher Order Runge-Kutta Algorithms," *Computational Chemistry*, vol. 11, 1987, pp. 121- 124.

[17] M. Bader, "A New Technique for the Early Detection of Stiffness in Coupled Differential Equations and Application to Standard Runge-Kutta Algorithms," *Theoretical Chemistry Accounts*, vol. 99, 1998, pp. 215-219.

[18] K. Murugesan, S. Sekar, V. Murugesh, and J.Y. Park, "Numerical Solution of an Industrial Robot Arm Control Problem Using the RK-Butcher Algorithm," *Int. J. Computer Applications in Technology*, vol. 19, no. 2, 2004, pp. 132-138.

[19] J.Y. Park, David J. Evans, K. Murugesan, S. Sekar, and V. Murugesh, "Optimal Control of Singular Systems Using the RK-Butcher Algorithm," *Int. J. Computer Mathematics*, vol. 81, no. 2, 2004, pp. 239-249.

[20] L.F. Shampine and M.K. Gordon, *Computer Solution of Ordinary Differential Equations-The Initial Value Problem*, W.H. Freeman & Co, San Francisco, USA, 1975.

[21] L.O. Chua and P.M. Lin, *Computer-Aided Analysis of Electronic Circuits,* Prentice-Hall, New Jersey, USA, 1975.

[22] K. Balachandran and K. Murugesan, "Analysis of Electronic Circuits Using the Single-Term Walsh Series Approach," *Int. J. of Electronics*, vol. 69, no. 3, 1990, pp. 327-332.

[23] H.C. Yan, *Investigation of Generalized State-Space Systems*, Ph.D. thesis, Clarkson University, Potsdam, USA, 1986.

K. Murugesan was born in 1965 in Chinna Kangayam Palayam, Erode district, Tamil Nadu, India. He obtained the BS and MS degrees in mathematics in 1985 and 1987 and the M.Phil. and PhD degrees in mathematics specializing in computer applications in 1989 and 1992. Since 1991, he has worked as a Senior Lecturer in the Department of Mathematics and Computer Applications, National Institute of Technology, Tiruchirappalli, Tamil Nadu, India. He worked as a Visiting Research Professor in the Department of Mathematics, Pusan National University, Busan, Korea under the Brain Pool Program. His research interests are in numerical methods and computer applications, and he has published more than 30 papers in various international journals related to numerical methods and computer applications.

N.P. Gopalan received the M.Sc. from Madras University in 1978 and the PhD in applied mathematics from Indian Institute of Science, Bangalore, in 1983. Currently, he is the Head of the Department of Computer Science and Engineering in the National Institute of Technology, Tiruchirappalli, Tamil Nadu, India. His research interests include algorithms, combinatorics, data mining, and distributed, parallel and grid computing.

Devarajan Gopal was born in Tamilnadu, India. He received the BE in electronics and communication from College of Engineering, Guindy, Madras University, the M.Tech. in communication and radar engineering from Indian Institute of Technology, New Delhi, and the PhD at Cochin University of Science and Technology, India. He joined the Defence Research and Development Organization in the Government of India and served as a Scientific Officer / Scientist for about two decades. During this period he was involved in radar, sonar and sonobuoy systems design and development. He received the prestigious National Scientific Award "VASVIK" for his contribution to "Airborne Sonobuoy System and Modular Data Bus" in 1991. He was associated with different universities in India and Ethiopia for curriculum revision at the undergraduate and graduate levels. He has held various positions at Madras University and Addis Ababa University. Currently, he is working as a Professor in the Department of Digital Information and Communication at Woosong University, Korea. His fields of interest are in digital signal processing, computer networks, and broadband communication.