J. Appl. Math. & Informatics Vol. **26**(2008), No. 3 - 4, pp. 793 - 809
Website: http://www.kcam.biz

NUMERICAL SOLUTION FOR ROBOT ARM PROBLEM USING LIMITING FORMULAS OF RK(7,8)

S. SENTHILKUMAR

ABSTRACT. The aim of this article is focused on providing numerical solutions for system of second order robot arm problem using the RK-eight stage seventh order limiting formulas. The parameters governing the arm model of a robot control problem have also been discussed through RK-eight stage seventh order limiting algorithm. The precised solution of the system of equations representing the arm model of a robot has been compared with the corresponding approximate solutions at different time intervals. Results and comparison show the efficiency of the numerical integration algorithm based on the absolute error between the exact and approximate solutions. Based on the numerical results a thorough comparison is carried out between the numerical algorithms.

AMS Mathematics Subject Classification: 65L10, CR G1.7
Key-words: RK-eight stage seventh order limiting formulas, Ordinary Differential Equations, Singular Systems and Robot arm Problem.

CONTENTS

Received June 14, 2007. Revised February 8, 2008.

1. Introduction

Extensive research work is still being carried out on variety of aspects in the field of robot control, especially about the dynamics of a robotic motion and their governing equations. The dynamics of robot arm problem was initially discussed by Warwick and Pugh [22]. Research in this area is still active and its applications are enormous. This is because of its nature of extending accuracy in the determination of approximate solutions and its flexibility. Said Oucheriah [15] discussed 'Robust Tracking and Model Following the Uncertain Dynamic Delay Systems by Memoryless Linear Controllers'. David Lim and Homayoun Seraji [13] discussed 'Configuration Control of a Mobile Dexterous Robot'. Polycarpou and Loannou [17] discussed about a 'Robust Adaptive Non-linear Control Design'. Hariharan Krishnan and Harris Mcclamroch [12] presented with the Applications of Non-linear Differential Algebraic Control Systems to Constrained Robot Systems' and Zluhua Qu [23] analyzed 'Robust Control of a class of Non-linear Uncertain Systems'. Because of the-linear and coupled characteristics nature, the design of a robot control system is made complex.

The dynamics of a robot can be described by a set of coupled non linear equations in the form of gravitational torques, coriolis and centrifugal forces. The significance of these forces is dependent in the physical parameters of the robot, the load it carries and the speed at which the robot operates. If accuracy is required then compensation for these parameter variations and disturbances becomes much more serious. Therefore, the design of the control system becomes much more complex. The theory of Variable Structure System (VSS) is developed and applied to solve wide variety of applications in the control process essentially; it is a system with discontinuous feedback control. Operating such a system in sliding mode makes it insensitive to parameter variations and disturbances.

Runge-Kutta (RK) methods are being applied to compute numerical solutions for the problems, which are modelled as Initial Value Problems (IVPs) differential equations by Alexander and Coylc [1], Evans [8], Hung [11], Shampine and Watts [19][20]. Runge-kutta methods have become very popular, both as computational techniques as well as subject for research, which were discussed by Butcher [4][5][6] and Shampine [19][20]. This method was derived by Runge and extended by Kutta to solve differential equations efficiently which are equivalent of approximating the exact solutions by matching 'n'terms of the Taylor series expansion

Runge-Kutta algorithms are considered as a excellent tool for the numerical integration of Ordinary Differential Equations (ODEs) because of self-starting in behavior, easy programming, and illustrate extreme accuracy and versatility in ODE problems. The most exciting developments in the RK usage is that by judicious re-arrangement of interim values of the RK predictor to obtain a second predictor of one order less. The two equations are generally referred to as an RK pair. Fehlberg [9] was the first to propose 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 et al[10]. The LTE is then used as a test to see whether a 'step has been successful, and if not, the step size is reduced (usually halved) until the LTE passes the tolerance requirement. The benefit 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 initiated a search for the RK algorithms of higher and higher order and better error estimates.

Butcher [4][5][6] derived the best RK pair by all statistical measures appeared to be the RK algorithm and also an error estimate was given. The RK-Butcher algorithm is nominally considered sixth order since it requires six function evaluations, but in actual practice the 'working order' is five but still exceeds all the other algorithms examined including RK-Fehlberg, RK-Centrodial mean and RK-Arithmetic mean.

Morris Bader [2][3] introduced the RK-Butcher algorithm for finding the truncation error estimates and intrinsic accuracies and the early detection of' stiffness in coupled differential equations that arises in theoretical chemistry problems. For the purpose of solving the initial-value problem, well established Single Step methods of numerical integration techniques are used by Ponalagusamy and Senthilkumar [18].

It is well known that RK-eight stage explicit limiting formulas are of order at most six. However, by taking the limit as the first abscissa approaches zero, the formulas can achieve seventh order. Such formulas are called limiting formulas which require the evaluations of the second derivatives of the solution. The possible order of s-stage explicit Runge-Kutta methods is s-1 for $s = 5, 6, 7$ but, they can achieve sth order in the limiting case where distance between some pairs of abscissas approaches zero. Such formulas are known as s-stage sth order limiting formulas. Harumi Ono [16] discussed five stage five order and six stage sixth order limiting formulas and also presented five stage six stage formulas of orders numerically five and six. They are obtained by replacing the second derivatives involved in the limiting formulas with simplest numerical differentiation. The reason to perform is that the second derivatives in the limiting formulas does not need full significant figures carried in computation and the user can choose free parameters so as to minimize the error caused by numerical differentiation. Devarajan Gopal, et. al.,[?] have discussed about the numerical solution of system of second order robot arm problem using RK-fifth order. In this article, the robot arm problem is solved with different approach using the algorithms such as RK-Fifth order, RK-Sixth order and the RK-Eight stage seventh order limiting formulas [21] to yield higher accuracy with less error.

2. RK-Sixth order algorithm

The RK-Sixth order algorithm is an explicit method discussed by Ponalagusamy and Senthilkumar [18]. The increase of the state variable x is stored in the constant k_1. This result is used in the next iteration for evaluating k_2. The same procedure must be repeated to compute the values of $k_3, k4$, k5 and k_6 and k_7. 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, which was discussed by Shampine and Gordon [20]. The remainder of the infinite sum excluded is referred to as the LTE. 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 number of input data which are very easy to program and simple to use.

The general s-stage RK method for solving an Initial Value Problem is

$$y' = f(x,y) \tag{2.1}$$

with the initial condition $y(x_0) = y_0$ is defined by where $y_{n+1} = y_n + h\sum_{i=1}^{s} b_i k_i$ where $k_i = f(x_n) + c_i h, y_n + h\sum_{i=1}^{s} a_{ij} k_i$ and $c_i = \sum_{i=1}^{s} a_{ij}, i = 1,2,..,s$ with c and b are $s-$dimensional vectors and $A(a_{ij})$ be the $s \times s$ matrix. Then the Sixth-order array is of the form,

$$\begin{array}{llllll}
c_1 & a_{11} & & & & \\
c_2 & a_{21} & a_{22} & & & \\
c_3 & a_{31} & a_{32} & a_{33} & & \\
\vdots & \vdots & \vdots & \vdots & & \\
c_s & a_{s1} & a_{s2} & ...a_{s-1} & a_{ss} & \\
 & b_1 & b_2... & b_{s-1} & b_s &
\end{array}$$

then the RK-sixth-order algorithm of the above Equation 2.1 is of the form

$$\begin{cases}
k_1 = hf(x_n, y_n), \\
k_2 = hf(x_n + \frac{1}{2}, y_n + \frac{k_1}{2}), \\
k_3 = hf(x_n + \frac{2}{3}, y_n + \frac{2k_1}{9} + \frac{4k_2}{9}), \\
k_4 = hf(x_n + \frac{1}{3}, y_n + \frac{7k_1}{36} + \frac{2k_2}{9} - \frac{k_3}{12}), \\
k_5 = hf(x_n + \frac{5}{6}, y_n - \frac{35k_1}{144} - \frac{55k_2}{36} + \frac{35k_3}{48} + \frac{15k_4}{8}), \\
k_6 = hf(x_n + \frac{1}{6}, y_n - \frac{k_1}{360} - \frac{11k_2}{36} - \frac{k_3}{8} + \frac{k_4}{2} + \frac{k_5}{10}) \\
k_7 = hf(x_n + h, y_n - \frac{41k_1}{260} + \frac{22k_2}{13} + \frac{43k_3}{156} - \frac{118k_4}{39} + \frac{32k_5}{195} + \frac{80k_6}{39})
\end{cases} \tag{2.2}$$

The formation of the RK-sixth-order array equation (2.2) takes form as follows:

$$y_{n+1} = y_n + [\frac{13k_1}{200} + \frac{11k_3}{40} + \frac{11k_4}{40} + \frac{4k_5}{25} + \frac{4k_6}{25} + \frac{13k_7}{200})$$

0							
$\frac{1}{2}$	$\frac{1}{2}$						
$\frac{2}{3}$	$\frac{2}{9}$	$\frac{4}{9}$					
$\frac{1}{3}$	$\frac{7}{36}$	$\frac{2}{9}$	$\frac{-1}{12}$				
$\frac{5}{6}$	$\frac{-35}{144}$	$\frac{-55}{36}$	$\frac{-35}{48}$	$\frac{15}{8}$			
$\frac{1}{6}$	$\frac{-1}{360}$	$\frac{-11}{36}$	$\frac{-1}{8}$	$\frac{1}{2}$	$\frac{1}{10}$		
1	$\frac{-41}{260}$	$\frac{22}{13}$	$\frac{43}{156}$	$\frac{-118}{39}$	$\frac{32}{195}$	$\frac{80}{39}$	
	$\frac{13}{200}$	0	$\frac{11}{40}$	$\frac{11}{40}$	$\frac{4}{25}$	$\frac{4}{25}$	$\frac{13}{200}$

3. Stability analysis

We take the test equation $\dot{y} = \lambda y$ where λ is a complex constant and it is used to determine the stability region of the RK-Sixth order method.

$$
\begin{aligned}
k_1 &= hf(x_n, y_n) = \lambda y_n, \\
k_2 &= hf(x_n + \frac{1}{2}, y_n + \frac{hk_1}{2}) = \lambda y_n(1 + \frac{\lambda h}{2}), \\
k_3 &= hf(x_n + \frac{2}{3}, y_n + \frac{2k_1}{9} + \frac{4k_2}{9}) = \lambda y_n[(1 + \frac{2\lambda h}{9} +)\frac{4\lambda h}{9}(1 + \frac{\lambda h}{2})] \\
k_4 &= hf(x_n + \frac{1}{3}, y_n + \frac{7k_1}{36} + \frac{2k_2}{9} - \frac{k_3}{12}), \\
&= \lambda y_n[1 + \frac{7\lambda h}{36} + \frac{2\lambda h}{9}(1 + \frac{\lambda h}{2}) - \frac{\lambda h}{12}(1 + \frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2}))] \\
k_5 &= hf(x_n + \frac{5}{6}, y_n - \frac{35k_1}{144} - \frac{55k_2}{36} + \frac{35k_3}{48} + \frac{15k_4}{8}), \\
&= \lambda y_n[1 - \frac{35\lambda h}{144} - \frac{55\lambda h}{36}(1 + \frac{\lambda h}{2}) + \frac{35\lambda h}{48}(1 + \frac{2\lambda h}{9}) + \\
&\quad \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2})) + (\frac{15\lambda h}{8}(1 + \frac{7\lambda h}{36} + \frac{2\lambda h}{9}(1 + \frac{\lambda h}{2}) - 1 + \frac{2\lambda h}{9} \\
&\quad + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2}))) \\
k_6 &= hf(x_n + \frac{1}{6}, y_n - \frac{k_1}{360} - \frac{11k_2}{36} - \frac{k_3}{8} + \frac{k_4}{2} + \frac{k_5}{10}) \\
&= \lambda y_n[1 + \frac{\lambda h}{360} - \frac{11\lambda h}{36}(1 + \frac{\lambda h}{2}) + \frac{\lambda h}{48}(1 + \frac{2h\lambda}{9}) + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2}) + \\
&\quad \frac{\lambda h}{2}(1 + \frac{7\lambda h}{36} + \frac{2\lambda h}{9}(1 + \frac{\lambda h}{2}) - \frac{\lambda h}{12}(1 + \frac{2\lambda h}{9}) + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2}) + \\
&\quad \frac{\lambda h}{10}(1 - \frac{35\lambda h}{144} - \frac{55\lambda h}{36}(1 + \frac{\lambda h}{2}) + \frac{35\lambda h}{48}(1 + \frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2})) \\
&\quad +(\frac{15\lambda h}{8}(1 + \frac{7\lambda h}{36} + \frac{2\lambda h}{9}(1 + \frac{\lambda h}{2}) - \frac{\lambda h}{12}(1 + \frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1 + \frac{\lambda h}{2}))))]
\end{aligned}
$$

$$\begin{aligned}
k_7 &= hf(x_n+h, y_n - \frac{41k_1}{260} + \frac{22k_2}{13} + \frac{43k_3}{156} - \frac{118k_4}{39} + \frac{32k_5}{195} + \frac{80k_6}{39}) \\
&= \lambda y_n[1 + \frac{41\lambda h}{260} - \frac{22\lambda h}{13}(1+\frac{\lambda h}{2}) + \frac{43\lambda h}{156}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})) - \\
&\frac{118\lambda h}{39}(1+\frac{7\lambda h}{36} + \frac{2\lambda h}{9}(1+\frac{\lambda h}{2}) - \frac{\lambda h}{12}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2}))) + \\
&\frac{32\lambda h}{195}(1-\frac{35\lambda h}{144}) - \frac{55\lambda h}{36}(1+\frac{\lambda h}{2}) + \frac{35\lambda h}{48}(1+\frac{2\lambda h}{9}) + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2}) + \\
&\frac{15\lambda h}{8}(1+\frac{7\lambda h}{36}) + \frac{2\lambda h}{9}(1+\frac{\lambda h}{2})) - \\
&\frac{\lambda h}{12}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})) - \frac{80\lambda h}{39}(1+\frac{\lambda h}{360} - \\
&\frac{11\lambda h}{36}(1+\frac{\lambda h}{2}) - \frac{\lambda h}{48}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})) + \\
&\frac{\lambda h}{2}(1+\frac{7\lambda h}{36}) + \frac{2\lambda h}{9}(1+\frac{\lambda h}{2}) - \frac{\lambda h}{12}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})) + \\
&\frac{\lambda h}{10}(1-\frac{35\lambda h}{144}) - \frac{55\lambda h}{36}(1+\frac{\lambda h}{2}) \\
&+\frac{35\lambda h}{48}(1+\frac{2\lambda h}{9} + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})))) + \frac{15\lambda h}{8}(1+\frac{7\lambda h}{36} \\
&+\frac{2\lambda h}{9}(1+\frac{\lambda h}{2} - \frac{\lambda h}{12}(1+\frac{2\lambda h}{9}) + \frac{4\lambda h}{9}(1+\frac{\lambda h}{2})))] + \\
&\lambda y_n[1 + \frac{h\lambda}{360} - \frac{11h\lambda}{36}(1+\frac{h\lambda}{2}) + \frac{h\lambda}{48}(1+\frac{2h\lambda}{9} + \frac{4h\lambda}{9}(1+\frac{h\lambda}{2}))]
\end{aligned}$$

On substituting $z = h\lambda$, we have

$$\begin{cases}
k_1 = hf(x_n, y_n) = \lambda y_n, \\
k_2 = hf(x_n+\frac{1}{2}, y_n+\frac{hk_1}{2}) = \lambda y_n(1+\frac{z}{2}), \\
k_3 = hf(x_n+\frac{2}{3}, y_n+\frac{2k_1}{9}+\frac{4k_2}{9}) = \lambda y_n(1+\frac{2zh}{9}+)\frac{4zh}{9}(1+\frac{z}{2}) \\
k_4 = hf(x_n+\frac{1}{3}, y_n+\frac{7k_1}{36}+\frac{2k_2}{9}-\frac{k_3}{12}), \\
= \lambda y_n(1+\frac{7zh}{36}+)\frac{2zh}{9}1+\frac{z}{2}-\frac{zh}{12}1+\frac{2zh}{9}+\frac{4zh}{9}1+\frac{zh}{2} \\
k_5 = hf(x_n+\frac{5}{6}, y_n-\frac{35k_1}{144}-\frac{55k_2}{36}+\frac{35k_3}{48}+\frac{15k_4}{8}), \\
= \lambda y_n(1-\frac{35zh}{144}-)\frac{55zh}{36}(1+\frac{zh}{2})+\frac{35zh}{48}(1+\frac{2zh}{9}) \\
\quad +\frac{4zh}{9}(1+\frac{zh}{2})+\frac{15zh}{8}(1+\frac{7zh}{36})+\frac{2zh}{9}(1+\frac{zh}{2})-1+\frac{2zh}{9}+\frac{4zh}{9}1+\frac{zh}{2} \\
k_6 = hf(x_n+\frac{1}{6}, y_n-\frac{k_1}{360}-\frac{11k_2}{36}-\frac{k_3}{8}+\frac{k_4}{2}+\frac{k_5}{10}) \\
= \lambda y_n(1+\frac{zh}{360}-)\frac{11zh}{36}1+\frac{zh}{2}+\frac{zh}{48}(1+\frac{2zh}{9}) \\
\quad +\frac{4zh}{9}(1+\frac{zh}{2})+\frac{zh}{2}(1+\frac{7zh}{36})+\frac{2zh}{9}(1+\frac{zh}{2})- \\
\quad \frac{zh}{12}(1+\frac{2zh}{9})+\frac{4zh}{(}1+\frac{zh}{2})+\frac{zh}{10}(1-\frac{35zh}{144})-\frac{55zh}{36}(1+\frac{zh}{2})+\frac{35zh}{48}(1+\frac{2zh}{9})+ \\
\quad \frac{4zh}{9}(1+\frac{zh}{2})+\frac{15zh}{8}(1+\frac{7zh}{36})+\frac{2zh}{9}(1+\frac{zh}{2})-\frac{zh}{12}(1+\frac{2zh}{9})+\frac{4zh}{9}(1+\frac{zh}{2})
\end{cases} \tag{3.1}$$

$$\begin{cases} k_7 = hf(x_n + h, y_n - \frac{41k_1}{260} + \frac{22k_2}{13} + \frac{43k_3}{156} - \frac{118k_4}{39} + \frac{32k_5}{195} + \frac{80k_6}{39}) \\ = \lambda y_n(1 + \frac{41zh}{260} - \frac{22zh}{13}(1 + \frac{zh}{2}) + \frac{43zh}{156}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) \\ -\frac{118zh}{39}(1 + \frac{7zh}{36}) + \frac{2zh}{9}(1 + \frac{zh}{2}) - \frac{zh}{12}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) + \\ \frac{32zh}{195}(1 - \frac{35zh}{144}) - \frac{55zh}{36}(1 + \frac{zh}{2}) + \frac{35zh}{48}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) \\ +\frac{15zh}{8}(1 + \frac{7zh}{36}) + \frac{2zh}{9}(1 + \frac{zh}{2}) - \frac{zh}{12}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) \\ +\frac{80zh}{39}(1 + \frac{zh}{360}) - \frac{11zh}{36}(1 + \frac{zh}{2}) + \frac{zh}{48}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) + \\ \frac{zh}{2}(1 + \frac{7zh}{36}) + \frac{2zh}{9}(1 + \frac{zh}{2}) - \frac{zh}{12}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) + \\ \frac{zh}{10}(1 - \frac{35zh}{144}) - \frac{55zh}{36}(1 + \frac{zh}{2}) + \frac{35zh}{48}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) + \\ \frac{15zh}{8}(1 + \frac{7zh}{36}) + \frac{2zh}{9}(1 + \frac{zh}{2}) - \frac{zh}{12}(1 + \frac{2zh}{9}) + \frac{4zh}{9}(1 + \frac{zh}{2}) \end{cases}$$

Therefore, the final integration is a weighted sum of the six calculated derivatives and the RK-sixth-order predictor formula is given by,

$$y_{n+1} = y_n + h[\frac{13k_1}{200} + \frac{11k_3}{40} + \frac{11k_4}{40} + \frac{4k_5}{25} + \frac{4k_6}{25} + \frac{13k_7}{200}) \tag{3.2}$$

Substituting the values of $k_1, k_2, k_3, k_4, k_5, k_6$ and k_7 into equation (3.2) we get,

$$y_{n+1} = y_n + \frac{h\lambda y_n}{2160}(2160 + 2160z + 1080z^2 + 360z^3 + 90z^4 + 18z^5 + 3z^6 - z^7) \tag{3.3}$$

From equation (3.3),the stability of the polynomial $Q(z) = \frac{y_{n+1}}{y_n}$ becomes

$$Q(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24} + \frac{z^5}{120} + \frac{z^6}{720} - \frac{z^7}{2160} \tag{3.4}$$

In a similar manner, the stability polynomial for the test equation $\dot{y} = \lambda y$ (λ is a complex constant) using the RK-fifth order method has been obtained as

$$Q_1(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24} + \frac{z^5}{120} + \frac{z^6}{640} \tag{3.5}$$

4. RK-Eight stage seventh order limiting formulas

Let us consider an initial value problem $\frac{dx}{dt} = f(t, x), x(t_0) = x_0$ where f and x are vectors and f is assumed to be differentiable sufficiently often for the definition to be meaningful. The parameters of an s-stage explicit Runge-Kutta method are represented in the following Butcher array [5].

$$\begin{array}{llll} c_1 & a_{11} & & \\ c_2 & a_{21} & a_{22} & \\ c_3 & a_{31} & a_{32} & a_{33} \end{array}$$

$$\begin{array}{ccccc} \vdots & \vdots & \vdots & \vdots & \\ c_s & a_{s1} & a_{s2} & \ldots a_{s-1} & a_{ss} \\ & b_1 & b_2 \ldots & b_{s-1} & b_s \end{array}$$

and x_i is used to denote the x ordinate at the abscissa c_i namely,

$$x_i = x_n + h \sum_{i=1}^{i-1} a_{ij} f_j$$

where

$$f_1 = t_n, x_n, f_i = f(t_n + c_i h, x_i), i = 2, .., s$$

Using them, the method can be written as

$$x_{n+1} = x_n + h \sum_{i=1}^{s} b_i f_i$$

Many RK-eight stage sixth order formulas that uses are known [21] and [16] and their properties are precisely reported [21]. An eight stage limiting formula that uses the values of the second derivatives at the point (t_n, x_n) is of the form

$$\begin{aligned} f_1 &= f(t_n, x_n), \\ f_2 &= D(f(t_n, x_n))u(f(1)) \\ x_3 &= x_n + h(a_{31} f_1 + h\alpha_3 f_2) \\ f_3 &= f(t_n + c_3 h, x_3)) \\ x_i &= x_n + h(a_{i1} f_1 + \sum_{j=3}^{i-1} a_{ij} f_j + h\alpha_i f_2) \\ f_i &= f(t_n + c_i h, x_i)), i = 4, 5, ..., 8 \end{aligned}$$

$$x_{n+1} = x_n + h(b_1 f_1 + \sum_{j=3}^{8} b_i f_i + h\beta_2 f_2) \qquad (4.1)$$

where $D(f(t_n, x_n))$ and $u(f_1)$ denote the Jacobian matrix of f at the point (t_n, x_n) and the vector $(1, f_1^1, f_1^2, ..., f_1^n)^T$ respectively (the superscripts denote the component numbers). The parameters of this limiting formula can be written in the following array analogous to Butcher array.

$$\begin{array}{cccccc} c_3 & a_{31} & & & & \alpha_3 \\ c_4 & a_{41} & a_{43} & & & \alpha_4 \\ c_5 & a_{51} & a_{53} & a_{54} & & \alpha_5 \\ \cdots & \cdots & \cdots & \cdots & & \\ \cdots & \cdots & \cdots & \cdots & & \\ c_8 & a_{81} & a_{83} & a_{84} & a_{87} & \alpha_8 \\ & b_1 & b_3 & b_4 & b_7 \ldots & \beta_2 \end{array}$$

4.1. Order conditions. We restrict ourselves to the case that $C_8 = 1, b_3 = 0$ and the following simplifying assumptions hold:

$$\alpha_3 = \frac{c_3^2}{2} \tag{4.2}$$

$$\sum_{j=3}^{i-1} a_{ij}c_j + \alpha_i = \frac{c_i^2}{2}, \quad (i = 4, 5, 8) \tag{4.3}$$

$$\sum_{j=3}^{i-1} a_{ij}c_j^2 = \frac{c_i^3}{3}, \quad (i = 4, 5, 8) \tag{4.4}$$

Comparing the Taylor series expansion for (4.1) with that of the true value $x(t_n + h)$ and matching the coefficients of each elementary differential, after tedious computations, one may get the following equations of conditions for seventh order accuracy:

$$a_{31} = c_3, , a_{i1} + \sum_{j=3}^{i-1} a_{ij} = c_i, \quad i = 4, 5, ..., 8 \tag{4.5}$$

$$\sum_{i=j+1}^{8} b_i a_{ij} = b_j(1 - c_j), \quad i = 4, 5, ..., 7 \tag{4.6}$$

$$\sum_{i=4}^{8} b_i a_{i3} = 0 \tag{4.7}$$

$$\sum_{i=6}^{8} b_i \sum_{i=5}^{i-1} a_{ij} \sum_{k=4}^{i-1} a_{jk}a_{k3} = 0 \tag{4.8}$$

$$\sum_{i=5}^{8} b_i \sum_{j=4}^{i-1} a_{ij}c_j a_{j3} = 0 \tag{4.9}$$

$$b_1 + \sum_{i=4}^{8} b_i = 1 \tag{4.10}$$

$$\sum_{i=4}^{8} b_i c_i + \beta_2 = \frac{1}{2} \tag{4.11}$$

$$\sum_{i=4}^{8} b_i c_i^2 = \frac{1}{3} \tag{4.12}$$

$$\sum_{i=5}^{8} b_i \sum_{j=4}^{i-1} a_{ij}c_j^2 = \frac{1}{12} \tag{4.13}$$

$$\sum_{i=5}^{8} b_i \sum_{j=4}^{i-1} a_{ij}c_j^3 = \frac{1}{20} \tag{4.14}$$

$$\sum_{i=6}^{8} b_i \sum_{j=5}^{8} a_{ij} \sum_{k=4}^{i-1} a_{ij} c_k^2 = \frac{1}{60} \tag{4.15}$$

$$\sum_{i=5}^{8} b_i \sum_{j=4}^{i-1} a_{ij} c_j^4 = \frac{1}{30} \tag{4.16}$$

$$\sum_{i=6}^{8} b_i \sum_{j=5}^{i-1} a_{ij} \sum_{k=4}^{i-1} a_{jkj} c_k^3 = \frac{1}{120} \tag{4.17}$$

$$\sum_{i=7}^{8} b_i \sum_{j=6}^{i-1} a_{ij} \sum_{k=5}^{i-1} a_{jk} \sum_{l=4}^{k-1} a_{kl} c_l^2 = \frac{1}{360} \tag{4.18}$$

$$\sum_{i=5}^{8} b_i \sum_{j=4}^{i-1} a_{ij} c_j^5 = \frac{1}{42} \tag{4.19}$$

$$\sum_{i=6}^{8} b_i \sum_{j=5}^{i-1} a_{ij} \sum_{k=4}^{i-1} a_{jk} c_k^4 = \frac{1}{210} \tag{4.20}$$

$$\sum_{i=7}^{8} b_i \sum_{j=6}^{i-1} a_{ij} \sum_{k=5}^{i-1} a_{jk} \sum_{l=4}^{k-1} a_{kl} c_k^3 = \frac{1}{840} \tag{4.21}$$

$$\sum_{i=6}^{8} b_i \sum_{j=5}^{i-1} a_{ij} c_j \sum_{k=4}^{j-1} a_{jk} c_k^3 = \frac{1}{168} \tag{4.22}$$

Solving Euqations (4.5-4.22), the values of the parameters are obtained. For more detail, refer the book written by Mitsui [14]. The RK-Eight stage Seventh order limiting algorithm is an explicit method which is given as follows [16].

$$\begin{cases}
k_1^{ij} = \tau f'(x_{ij}(n\tau)) \\
k_2^{ij} = \tau f'(x_{ij}(n\tau) + \frac{7k_1^{ij}}{20}) \\
k_3^{ij} = \tau f'(x_{ij}(n\tau) + \frac{22979k_1^{ij}}{100842} + \frac{33275k_2^{ij}}{201684}) \\
k_4^{ij} = \tau f'(x_{ij}(n\tau) + \frac{25760306k_1^{ij}}{57421875} - \frac{11585024k_2^{ij}}{2296875} - \frac{2139752k_3^{ij}}{390625}) \\
k_5^{ij} = \tau f'(x_{ij}(n\tau) + \frac{119094452k_1^{ij}}{855999375} - \frac{22528k_2^{ij}}{25725} + \frac{37929472k_3^{ij}}{25788125} - \frac{19000k_4^{ij}}{288827}) \\
k_6^{ij} = \tau f'(x_{ij}(n\tau) - \frac{34346067405574k_1^{ij}}{580779387890625} \\
\quad - \frac{605933632k_2^{ij}}{91822828125} - \frac{306126104994304k_3^{ij}}{5780137717578125} - \frac{49440496k_4^{ij}}{3893987515} + \frac{1168031718k_5^{ij}}{76431573125}) \\
k_7^{ij} = \tau f'(x_{ij}(n\tau) - \frac{1484329913137k_1^{ij}}{501007140480} - \frac{235840k_2^{ij}}{196049} + \\
\quad \frac{2356095077990864k_3^{ij}}{260700167892285} + \frac{170270078125k_4^{ij}}{378771628384} \\
\quad + \frac{16765288525k_5^{ij}}{1576329984} + \frac{935180524328125k_6^{ij}}{256363606146816})
\end{cases} \tag{4.23}$$

Therefore, the final integration is a weighted sum of the seven calculated derivatives which is given below.

$$\begin{cases} x_{ij}((n+1)\tau) = & x_{ij}(n\tau) + (\frac{8835k_1^{ij}}{108416} + \frac{24748509184k_3^{ij}}{60419933937} \\ & -\frac{6640625k_4^{ij}}{86062944} + \frac{951125k_5^{ij}}{2363904} + \frac{57826519140625k_6^{ij}}{21093765402112}) \end{cases} \tag{4.24}$$

where f(.) is computed according to given function.

5. Robot arm model and essential of variable structure

5.1. Robot arm model. The dynamics of robot arm problem was discussed by Warwick and Pugh [22]. It can be represented in the following form.

$$T = A(Q)\ddot{Q} + BQ, \dot{Q} + C(Q) \tag{5.1}$$

where $A(Q)$ is the coupled inertia matrix,$B(Q,\dot{Q})$ is the matrix of coriolis and centrifugal forces. $C(Q)$ is the gravity matrix, T is the input torques applied at various joints.

For a robot with two degrees of freedom, assuming lumped equivalent massless links, the dynamics are represented by

$$T_1 = D_{11}\ddot{q_1} + D_{12}\ddot{q_{21}} + D_{122}(\dot{q_2})^2 + D_{112}(\dot{q_1}\dot{q_2}) + D_1 \tag{5.2}$$

$$T_2 = D_{21}\ddot{q_1} + D_{22}\ddot{q_2} + D_{122}(\dot{q_1})^2 + D_2 \tag{5.3}$$

where $D_{11} = (M_1 + M_2)d_2^2 + 2M_2d_1d_2\cos(q_2)$, $D_{12} = (M_2)d_2^2 + M_2d_1d_2\cos(q_2)$,
$D_{12} = D_{21}$, $D_{22} = M_2d_2^2$, $D_{112} = -2M_2d_1d_2\sin(q_2)$,
$D_{122} = -M_2d_1d_2\sin(q_2)$, $D_{211} = D_{122}$,
$D_1 = \{(M_1 + M_2)d_1\sin(q_1) + M_2d_2\sin(q_1 + q_2)g\}$, $D_1 = (M_2d_2\sin(q_1 + q_2)g$.

$$X = (X_1, X_2, X_3, X_4)^T = (q_1 - q_{1d}, \dot{q_1}, q_2 - q_{2d}, \dot{q_2})^T \tag{5.4}$$

where q_1 and q_2 are the angles at joints 1 and 2 respectivey and q_{1d} and q_{2d} are constants . Hence, equations (5.2) and (5.3) may be written in state space representation as

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = \frac{D_{22}}{d}(D_{122}x_2^2 + D_{112}X_2X_4 + D_1 + T_1) - \frac{D_{12}}{d}(D_{211}X_4^2 + D_2 + T_2) \\ \dot{x_3} = x_4 \\ \dot{x_4} = \frac{-D_{12}}{d}(D_{122}x_2^2 + D_{112}X_2X_4 + D_1 + T_1) - \frac{D_{12}}{d}(D_{211}X_4^2 + D_2 + T_2) \end{cases} \tag{5.5}$$

A synthesis of the control law would be very difficult because of the nonlinear and interactive nature of the canonical equations (5.5). Hence they should be reduced to a linear form.

5.2. Reduction of robot dynamics to a second-order linear systems. Although the physical and mathematical structure of the complete dynamic robot model are analytically coupled and non-linear, the transient responses of the robot dynamics appear to resemble the transient responses of linear systems. Consequently, each joint of the robot can be characterized as a single-input single-output system (SISO). The input is the actuator torque (or force) and the output is the joint position. Hence the mathematical model of a robot can be regarded as a 'black box'. The input into this black box is the transient response of a linear model to a step input. The output is the motive forces, or torques, required by the robot to reproduce responses similar to the linear model. Samples of the input and output of the black box have been fed into an identification program which will match a low-order decoupled linear time-invariant model of the form

$$G(s) = \frac{Y(s)}{U(s)} = \frac{B_m S^m + B_{m-1} S^{m-1} + \ldots + B_1 S + B_0}{S^n + A_{n-1} S^{n-1} + \ldots + A_1 S + A_0} \tag{5.6}$$

The model orders m and n are selected to give the lowest possible order that will characterize the structure of the mathematical model of the robot. It is found that the non-linear model equations (5.5) of the two-link-robot-arm model can be reduced to the following system of linear equations.

For two degrees of freedom robot, under the assumption of lumped equivalent masses and mass-less links, the dynamics are represented in terms of systems of non-linear equations and by applying the method of reduction, it has been represented in terms of the following system of linear equation as:

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = B_{10}T_1 - A_{11}x_2 - A_{10}x_1 \\ \dot{x_3} = x_4 \\ \dot{x_4} = B_{20}^2 T_2 - A_{21}^2 x_4 - A_{20}^2 x_3 \end{cases} \tag{5.7}$$

which is of the form

$$k\dot{x}(t) = Ax(t) + Bu(t) \tag{5.8}$$

where the values of the parameters concerning the joint-1 are given by

$$A_{10} = 0.1730, A_{11} = -0.240, B_{10} = 0.00265$$

and the values of parameters concerning the joint -2 are given by

$$A_{20} = 0.0438, A_{21} = 0.3610, B_{20} = 0.0967$$

and by choosing $T_1 = 1$ and $T_2 = 1$ with initial conditions,

$$[e_1(0) \quad x_2(0) \quad e_3(0) \quad x_4(0)]^T = [-1 \quad 0 \quad -1 \quad 0]^T$$

and the corresponding exact solution is given by

$$e_1(t) = \begin{cases} e^{0.107t}(-1.15317919 \cos(0.401934074t)+ \\ 0.306991074 \sin(0.401934074t)) + 0.15317919 \end{cases} \tag{5.9}$$

TABLE 1. Solutions of Equations (3.2) and (5.6) for $x_1(t)$

Sol. No.	Time	Exact Solution	RKAM Solution	RKAM Error
1	0.00	-1.0000000000	-1.0000000000	0.0000000000
2	0.25	-0.9936586212	-0.9953327587	-0.0016741375
3	0.50	-0.9742424794	-0.9786473848	-0.0044049054
4	0.75	-0.9412482065	-0.9494398199	-0.0081916134
5	1.00	-0.8942959125	-0.9073316683	-0.0130357558

$$x_2(t) = e^{0.107t}(0.463502099\sin(0.401934074t) + 0.123390173 \cos(0.401934074t)) + e^{0.107t}(-1.15317919 \cos(0.401934074t) + 0.306991074\sin(0.401934074t))$$

$$e_3(t) = \begin{cases} 1.029908976e^{-0.113404416t} - \\ 6.904124484e^{-0.016916839t} + 4.874215508 \end{cases} \tag{5.10}$$

$$x_4(t) = -0.116795962e^{-0.113404416t} + 0.116795962e^{-0.016916839t}$$

6. Numerical solutions of system of second order Cartesian robot arm model problem: A study and comparison

The discrete and exact solutions of the robot arm model problem are calculated for various time intervals using Equations (3) and (39) and are presented in Tables 1-4. The values of $e_1(t), x_2(t), e_3(t)$ and$x_4(t)$ are calculated for time t arranging from 0.25 to 1. The absolute error between the exact and discrete solutions for the RK methods based on RK-Fifth-order, RK-Sixth-order and RK-eight stage seventh order limiting formulas are calculated. For time $t = 0.0, 0.25, 0.05, 0.75$ and 1.0 the values are tabulated in Tables 1-4 respectively.

7. Discussions and conclusion

The present article sheds some light on different numerical integration algorithms involved in robot arm model problem. It is pertinent to pin-point out here that the obtained discrete solutions for the Robot Arm model problem using the RK-eight stage seventh order limiting formulas guarantees more accurate values compared to the RK-Fifth-order method and RK-Sixth-order algorithm. From the Tables 1-4, we observe that the solution obtained by the RK-eight stage seventh order limiting formulas match well with the exact solutions of the robot arm model problem but the RK-Fifth Order and RK-Sixth-order algorithm method yields a little error. Hence, RK-eight stage seventh order limiting formula is more suitable for studying the system of second order robot arm model problem and this algorithm can be implemented for any length of independent variable on a digital computer.

RK-ButcherSolution	RK-Butcher Error	RK-Sixth Order Solution
-1.0000000000	0.0000000000	-1.0000000000
-0.9953327583	-0.0016741371	-0.9953327581
-0.9786473825	-0.00440 49031	-0.9743245899
-0.9494352670	-0.0081870605	-0.9451413296
-0.9073036550	-0.0130077425	-0.9030901483

RK-Sixth Order Error	RK-Eight stage Seven Order Solution	RK- Eight stage Seven Order Solution Error
0.0000000000	-1.0000000000	0.0000000000
-0.0016741370	-0.9953327574	-0.0016741362
-0.0000821105	-0.9743245890	-0.0000821096
-0.0038931231	-0.9451413284	-0.0038931219
-0.0087942358	-0.90309 01472	-0.0087942347

TABLE 2. Solutions of Equations (3.2) and (5.6) for $x_2(t)$

Sol.	Time	Exact Solution	RKAM Solution	RKAM Error
1	0.00	0.0000000000	0.0000000000	0.0000000000
2	0.25	0.0511410611	0.0459864489	0.0051546122
3	0.50	0.1045249896	0.0941265859	0.0104450
4	0.75	0.1596829669	0.1438971697	0.0157857972
5	1.00	0.2161001218	0.1949942351	0.0211058867

RK-Butcher Solution	RK-Butcher Error	RK-Sixth Order Solution
0.0000000000	0.0000000000	0.0000000000
0.0459864413	0.0051546198	0.0459863413
0.0941265909	0.0104450	0.0940922750
0.1439868497	0.0156961172	0.1439950296
0.1950883237	0.0210117981	0.1966502520

RK-Sixth Order Error	RK-Eight stage Seven Order Solution	RK- Eight stage Seven Order Solution Error
0.0000000000	0.0000000000	0.0000000000
0.0051546098	0.0459863401	0.005154721
0.0051546101	0.0940922739	0.0104327157
0.0156879373	0.1439950284	0.0156879385
0.0194498698	0.1966502509	0.0194498709

TABLE 3. Solutions of Equations (3.2) and (5.6) for $x_3(t)$

Sol. No.	Time	Exact Solution	RKAM Solution	RKAM Error
1	0.00	-1.0000000000	-1.0000000000	0.0000000000
2	0.25	-0.9996516946	-0.9997351600	-0.0000834654
3	0.50	-0.9986216177	-0.9987198532	0.0000982355
4	0.75	-0.9969317452	-0.9970073822	0.0000756370
5	1.00	-0.9946034264	-0.9946209249	0.0000174985

RK-Butcher Solution	RK-Butcher Error	RK-Sixth Order Solution
-1.0000000000	0.0000000000	-1.0000000000
-0.9997057337	0.0000540391	-0.9997056599
-0.9986904291	0.0000688114	-0.9986904092
-0.9969779638	0.0000462186	-0.9969669638
-0.99461 95155	0.00001 60891	-0.99461 60056

RK-Sixth Order Error	RK-Eight stage Seven Order Solution	RK- Eight stage Seven Order Solution Error
0.0000000000	-1.0000000000	0.0000000000
-0.0000539653	-0.9997056585	-0.0000539639
-0.0000687915	-0.9986904080	-0.0000687903
-0.0000352186	-0.9969669627	-0.0000352175
0.0000125792	-0.9946160047	-0.0000125783

TABLE 4. Solutions of Equations (3.2) and (5.6) for $x_4(t)$

Sol. No.	Time	Exact Solution	RKAM Solution	RKAM Error
1	0.00	0.0000000000	0.0000000000	0.0000000000
2	0.25	0.0027718839	0.0028505791	-0.0000786952
3	0.50	0.0054545872	0.0056069156	-0.0001523284
4	0.75	0.0080506523	0.0087939398	-0.0007432875
5	1.00	0.0105625499	0.0108477411	-0.0002851912

RK-Butcher Solution	RK-Butcher Error	RK- Sixth Order Solution
0.0000000000	0.0000000000	0.0000000000
0.0028505764	-0.0000786925	0.0028499524
0.00560 68988	-0.0001523116	0.0056062955
0.0082717292	-0.0002210769	0.0082711480
0.0108475497	-0.0002849998	0.0108471859

REFERENCES

1. Alexander, R.K. and Coyle, J.J. *Runge-Kutta methods for differential- algebric systems* SIAM Journal of Numerical Analysis,Vol. 27, No.3, pp. 736- 752,1990.
2. Bader, M. *A comparative study of new truncation error estimates and intrinsic accuracies of some higher order Runge-Kutta algorithms* Computational Chemistry,Vol. 11, pp. 121-124,1987.
3. Bader, M. *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, pp. 215-219,1998.
4. Butcher. J.C. *On Runge proccesses of higher order* Journal of Australian Mathematical Society ,Vol.4. p.179,1964.
5. Butcher, J.C. *The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods*John Wiley and Sons, U.K.1987.

RK-Sixth Order Error	RK-Eight stage Seven Order Solution	RK- Eight stage Seven Order Solution Error
0.0000000000	0.0000000000	0.0000000000
-0.0000780685	0.0028499512	-0.0000780673
-0.0001517083	0.0056062943	-0.0001517071
-0.0002204957	0.0082711469	-0.0002204946
-0.000284636	0.0108471847	-0.0002846348

6. Butcher, J.C. *On order reduction for Runge-Kutta methods applied to differential-algebraic systems and to stiff systems of ODEs*SIAM J. of Numerical Analysis,Vol.27, pp. 447-456,1990.
7. Devarajan Gopal, V. Murugesh, K. Murugesan *CNumerical solution of second-order robot arm control problem using Runge-Kutta-Butcher algorithm* International Journal of Computer Mathematics, 83(3):345-356, 2006.
8. Evans. D.J. *A new 4th Order Runge-Kutta method for initial value problems with error control*Int. J. of Computer Mathematics,Vol.139, pp. 217- 227,1991
9. Fehlberg, E. *Klassische Runge-Kutta-Formeln vierter und niedriger Ordnung mit Schrittweitenkontolle und ihre Anwendung auf Warmeleitungsprobleme*Computing,Vol. 6, pp. 61-71,1970.
10. Forsythe, G.E., Malcolm. M.A. and Moler. CD. *Computer Methods for Mathematical Computations* Prentice-Hall, Englewood Cliffs, NJ,p. 135,1977
11. Hung. C. *Dissipativity of Runge-Kutta methods for dynamical systems with delays* IMA J. of Numerical Analysis,Vol.20, pp. 153-166,2000
12. Krishnan .H and Harris Mcclamroch, N.*Tracking in non-linear differential- algebraic control systems with applications to constrained Robot systems* Automatica, Vol.30, No. 12, pp. 1885-1897,1994.
13. Lim, D. and Seraji. H. *Configuration control of a mobile dexterous Robot: real time implementation and experimentation* Int. J. of Robotics Research,Vol. 16, No.5. pp. 601-618,1997.
14. Mitsui,T and Shinohara,Y, *Numerical Analysis of Ordinary Differential Equations and its Applications*published by world scientific publishers, ISBN 981-02-2229-71-15,1995.
15. Oucheriah, S. , A. *Robust tracking and model following of uncertain dynamic delay systems by memorless linear controllers* IEEE Transactions on Automatic Control,Vol.44. No.7, pp. 1473-1481,1999.
16. Ono.H, Journal of Information Processing,12:251-160, 1989.
17. Polvcarpou, MM. and Loannou. PA. *A Robust adaptive non-linear control design* Auromarica,Vol.32. No. 3. pp. 423-427,1996.
18. Ponalagusamy.R and S.Senthilkumar *Investigation on Raster CNN Simulation by Numerical Integration Algorithms*Journal of Combinatorial Mathematics and Combinatorial Computation,[Accepted in Press], 2007.
19. Shampine. L.F. and Watts. H.A. *The art of a Runge-Kutta code. Part I* Mathematical Software,Vol.3. pp. 257-275,1977.
20. Shampine. L.F. and Gordon. M.K. *Computer solutions of. ordinary differential equations*W.H.Freeman. San Francisco. CA. p. 23,1975.
21. Tanaka,M, K.Kasuga, S,Yamashita and H.Yamazaki *Trans. Information Processing Society of Japan*34:,62-74,(Japanese),1993.
22. Warwick. K. and Pugh, A. *Robot control-theory and applications*Peter Peregrinus Ltd, North-Holland, 1990.
23. Zhihua. Qu. *Robot Control of a class of non-linear uncertain systems* IEEE Transactions on Automatic Control. Vol.37, No. 9, pp.1437-1442,1992

SUKUMAR SENTHILKUMAR received his B.Sc Degree in Mathematics from Madras University, M.Sc, Degree in Mathematics from Bharathidasan University in 1996, M.Phil Degree in Mathematics from Bharathidasan University in 1999 and M.Phil Computer Science and Engineering from Bharathiar University in 2000. Also, he received PGDCA and PGDCH in Computer Applications and Computer Hardware from Bharathidasan University in 1996 and 1997 respectively. He is Currently Pursuing Doctoral Degree in the Department of Mathematics at National Institute of Technology [REC], Tiruchirappalli, Tamilnadu, India. He has published research papers in referred national and international journals. He served as a Lecturer, in the Department of Computer Science at Asan Memorial College of Arts and Science, Chennai, Tamilnadu, India. His Current Research Interests include Cellular Neural Networks, Digital Image Processing, Numerical Analysis and Techniques and other related areas.
Department of Mathematics,National Institute of Technology, Tiruchirappalli-620 015,Tamilnadu, INDIA
email: ssenthilkumar1974@yahoo.co.in