

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Solution to a System of Second Order Robot Arm by Parallel Runge-Kutta Arithmetic Mean Algorithm

S. Senthilkumar and Abd Rahni Mt Piah
*Universiti Sains Malaysia, School of Mathematical Sciences,
Pulau Pinang
Malaysia*

1. Introduction

Enormous amount of real time robot arm research work is still being carried out in different aspects, especially on dynamics of robotic motion and their governing equations. Taha [5] discussed the dynamics of robot arm problems. Research in this field is still on-going and its applications are massive. This is due to its nature of extending accuracy in order to determine approximate solutions and its flexibility. Many studies [4-8] have reported different aspects of linear and non-linear systems. Robust control of a general class of uncertain non-linear systems are investigated by zhihua [10].

Most of the initial value problems (IVPs) are solved using Runge-Kutta (RK) methods which in turn are employed in order to calculate numerical solutions for different problems, which are modelled in terms of differential equations, as in Alexander and Coyle [11], Evans [12], Shampine and Watts [14], Shampine and Gordan [18] codes for the Runge-Kutta fourth order method. Runge-Kutta formula of fifth order has been developed by Butcher [15-17]. Numerical solution of robot arm control problem has been described in detail by Gopal et al.[19]. The applications of non-linear differential-algebraic control systems to constrained robot systems have been discussed by Krishnan and Mcclamroch [22]. Asymptotic observer design for constrained robot systems have been analyzed by Huang and Tseng [21]. Using fourth order Runge-Kutta method based on Heronian mean (RKHeM) an attempt has been made to study the parameters concerning the control of a robot arm modelled along with the single term Walsh series (STWS) method [24]. Hung [23] discussed on the dissipativity of Runge-Kutta methods for dynamical systems with delays. Ponalagusamy and Senthilkumar [25,26] discussed on the implementations and investigations of higher order techniques and algorithms for the robot arm problem. Evans and Sanugi [9] developed parallel integration techniques of Runge-Kutta form for the step by step solution of ordinary differential equations.

This paper is organized as follows. Section 2 describes the basics of robot arm model problem with variable structure control and controller design. A brief outline on parallel Runge-Kutta integration techniques is given in section 3. Finally, the results and conclusion on the overall notion of parallel 2-stage 3-order arithmetic mean Runge-Kutta algorithm and obtains almost accurate solution for a given robot arm problem are given in section 4.

2. Statement of the robot arm model problem and essential variable structure

2.1 Model of a robot arm

It is well known that both non-linearity and coupled characteristics are involved in designing a robot control system and its dynamic behavior. A set of coupled non-linear second order differential equations in the form of gravitational torques, coriolis and centrifugal represents dynamics of the robot. It is inevitable that the significance of the above three forces are dependent on the two physical parameters of the robot namely the load it carries and the speed at which the robot operates. The design of the control system becomes more complex when the end user needs more accuracy based on the variations of the parameters mentioned above. Keeping the objective of solving the robot dynamic equations in real time calculation in view, an efficient parallel numerical method is needed. Taha [5] discussed dynamics of robot arm problem represented by as

$$T = A(Q)\ddot{Q} + B(Q, \dot{Q}) + C(Q) \quad (1)$$

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

For a robot with two degrees of freedom, by considering lumped equivalent massless links, i.e. it means point load or in this case the mass is concentrated at the end of the links, the dynamics are represented by

$$\left. \begin{aligned} T_1 &= D_{11}\ddot{q}_1 + D_{12}\ddot{q}_2 + D_{122}(\ddot{q}_2)^2 + D_{112}(\dot{q}_1\dot{q}_2) + D_1, \\ T_2 &= D_{21}\ddot{q}_1 + D_{22}\ddot{q}_2 + D_{211}(\dot{q}_1)^2 + D_2, \end{aligned} \right\} \quad (2)$$

where

$$D_{11} = (M_1 + M_2)d_2^2 + 2M_2d_1d_2 \cos(q_2),$$

$$D_{12} = D_{21} = M_2d_2^2 + M_2d_1d_2 \cos(q_2),$$

$$D_{22} = M_2d_2^2,$$

$$D_{112} = -2M_2d_1d_2 \sin(q_2),$$

$$D_{122} = D_{211} = -M_2d_1d_2 \sin(q_2),$$

$$D_1 = [(M_1 + M_2)d_1 \sin(q_1) + M_2d_2 \sin(q_1 + q_2)]g$$

and

$$D_2 = [M_2d_2 \sin(q_1 + q_2)]g.$$

The values of the robot parameters used are $M_1= 2\text{kg}$, $M_2 = 5\text{kg}$, $d_1 = d_2 = 1$. For problem of set point regulation, the state vectors are represented as

$$X = (X_1, X_2, X_3, X_4)^T = (q_1 - q_{1d}, \dot{q}_1, q_2 - q_{2d}, \dot{q}_2)^T, \quad (3)$$

where

q_1 and q_2 are the angles at joints 1 and 2 respectively, and q_{1d} and q_{2d} are constants. Hence, equation (2) may be expressed in state space representation as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{D_{22}}{d} (D_{122}X_2^2 + D_{112}X_2X_4 + D_1 + T_1) - \frac{D_{12}}{d} (D_{211}X_4^2 + D_2 + T_2) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= \frac{-D_{12}}{d} (D_{122}X_2^2 + D_{112}X_2X_4 + D_1 + T_1) - \frac{D_{12}}{d} (D_{211}X_4^2 + D_2 + T_2). \end{aligned} \quad (4)$$

Here, the robot is simply a double inverted pendulum and the Lagrangian approach is used to develop the equations.

In [5] it is found that by selecting suitable parameters, the non-linear equation (3) of the two-link robot-arm model may be reduced to the following system of linear equations:

$$\begin{aligned} \dot{e}_1 &= x_2, \\ \dot{x}_2 &= B_{10}T_1 - A_{11}x_2 - A_{10}e_1, \\ \dot{e}_3 &= x_4, \\ \dot{x}_4 &= B_{20}^2T_2 - A_{21}^2x_4 - A_{20}^2e_3, \end{aligned} \quad (5)$$

where one can attain the system of second order linear equations:

$$\begin{aligned} \ddot{x}_1 &= -A_{11}\dot{x}_1 - A_{10}x_1 + B_{10}T_1, \\ \ddot{x}_3 &= -A_{21}^2x_3 - A_{20}^2x_1 + B_{210}^2T_2, \end{aligned}$$

with the parameters concerning joint-1 are given by

$$A_{10} = 0.1730, A_{11} = -0.2140, B_{10} = 0.00265,$$

and the parameters of joint-2 are given by

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

If we choose $T_1 = r$ (constant) and $T_2 = \lambda$ (constant), it is now possible to find the complementary functions of equation (4) because the nature of the roots of auxiliary equations (A. Es) of (4) is unpredictable. Due to this reason and for the sake of simplicity, we take $T_1 = T_2 = 1$.

considering $q_1 = q_2 = 0$, $q_{1d} = q_{2d} = 1$ and $\dot{q}_1 = \dot{q}_2 = 0$, the initial conditions are given by $e_1(0) = e_3(0) = -1$ and $e_2(0) = e_4(0) = 0$ and the corresponding exact solutions are,

$$\left. \begin{aligned} e_1(t) &= e^{0.107t} [-1.15317919 \cos(0.401934074t) + 0.306991074 \sin(0.401934074t)] + 0.15317919' \\ e_2(t) &= e^{0.107t} [0.463502009 \sin(0.401934074t) + 0.123390173 \cos(0.401934074t)] + e^{0.107t} [-1.15317919 \cos(0.401934074t) + 0.306991074 \sin(0.401934074t)] \\ e_3(t) &= 1.029908976 e^{-0.113404416t} - 6.904124484 e^{-0.016916839t} + 4.874215508, \\ e_4(t) &= -0.116795962 e^{-0.113404416t} + 0.116795962 e^{-0.016916839t} . \end{aligned} \right\} (6)$$

3. A brief sketch on parallel Runge-Kutta numerical integration techniques

The system of second order linear differential equations originates from mathematical formulation of problems in mechanics, electronic circuits, chemical process and electrical networks, etc. Hence, the concept of solving a second order equation is extended using parallel Runge-Kutta numerical integration algorithm to find the numerical solution of the system of second order equations as given below. It is important to mention that one has to determine the upper limit of the step-size (h) in order to have a stable numerical solution of the given ordinary differential equation with IVP. We thus consider the system of second order initial value problems,

$$\dot{y}_j = f_j(x, y_j, \dot{y}_j), j = 1, 2, \dots, m \quad (7)$$

with $y_j(x_0) = y_{j0}$

$$\dot{y}_j(x_0) = \dot{y}_{j0} \quad \text{for all } j = 1, 2, \dots, m.$$

3.1 Parallel Runge-Kutta 2-stage 3-order arithmetic mean algorithm

A parallel 2-stage 3-order arithmetic mean Runge-Kutta technique is one of the simplest technique to solve ordinary differential equations. It is an explicit formula which adapts the Taylor's series expansion in order to calculate the approximation. A parallel Runge-Kutta 2-stage 3-order arithmetic mean formula is of the form,

$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf\left(x_n + \frac{1}{2}, y_n + \frac{k_1}{2}\right) = k_2^* \\ k_3 &= hf(x_n + k_1, y_n + k_1) = k_3^* . \end{aligned}$$

Hence, the final integration is a weighted sum of three calculated derivatives per time step is given by,

$$y_{n+1} = y_n + \frac{h}{6}[k_1 + 4k_2 + k_3].$$

Parallel 2-stage 3-order arithmetic mean Runge-Kutta algorithm to determine y_j and $\dot{y}_j, j=1,2,3,\dots,m$ is given by,

$$y_{j+1} = y_j + \frac{h}{6}[k_{1j} + 4k_{2j} + k_{3j}] \tag{8}$$

and

$$\dot{y}_{j+1} = \dot{y}_j + \frac{h}{6}[u_{1j} + 4u_{2j} + u_{3j}]$$

$$k_{1j} = \dot{y}_j,$$

$$k_{2j} = \dot{y}_j + \frac{hu_{1j}}{2} = k_{2j}^*$$

$$k_{3j} = \dot{y}_j + hu_{1j} = k_{3j}^* \tag{9}$$

$$u_{1j} = f(x_n, y_{jn}, \dot{y}_{jn}), \forall j=1,2,3,\dots,m \tag{10}$$

$$u_{2j} = f(x_n + \frac{h}{2}, y_{1n} + \frac{hk_{11}}{2}, y_{2n} + \frac{hk_{12}}{2}, \dots, y_{mn} + \frac{hk_{1m}}{2}, \dot{y}_{1n} + \frac{hu_{11}}{2}, \dot{y}_{2n} + \frac{hu_{12}}{2}, \dots, \dot{y}_{mn} + \frac{hu_{1m}}{2})$$

$$u_{3j} = f(x_n + h, y_{1n} + hk_{11}, y_{2n} + hk_{12}, \dots, y_{mn} + hk_{1m}, \dot{y}_{1n} + hu_{11}, \dot{y}_{2n} + hu_{12}, \dots, \dot{y}_{mn} + hu_{1m}).$$

The corresponding parallel 2-stage 3-order arithmetic mean Runge-Kutta algorithm array to represent equation (9) takes the form

| | |
|---------------|---------------|
| 0 | |
| $\frac{1}{2}$ | $\frac{1}{2}$ |
| 1 | 1 |
| | 1 4 1 |

Therefore, the final integration is a weighted sum of three calculated derivatives per time step given by,

$$y_{n+1} = y_n + \frac{h}{6}[k_1 + 4k_2 + k_3] \tag{11}$$

3.2 Parallel Runge-Kutta 2-stage 3-order geometric mean algorithm of type-I

The parallel 2-stage 3-order geometric mean Runge-Kutta formula of type-I is of the form,

$$k_1 = hf(x_n, y_n),$$

$$k_2 = hf\left(x_n + \frac{2}{3}, y_n + \frac{2k_1}{3}\right) = k_2^*,$$

$$k_3 = hf(x_n + k_1, y_n + k_1) = k_3^*,$$

Hence, the final integration is a weighted sum of three calculated derivatives per time step which is given by,

$$y_{n+1} = y_n + hk_1^{1/4} k_2^{3/4}.$$

Parallel 2-stage 3-order geometric mean Runge-Kutta algorithm of type-I to determine y_j and $\dot{y}_j, j=1,2,3,\dots,m$ is given by,

$$y_{j+1} = y_j + hk_{1j}^{1/4} k_{2j}^{3/4}, \quad (12)$$

and

$$\dot{y}_{j+1} = \dot{y}_j + hk_{1j}^{1/4} k_{2j}^{3/4},$$

$$k_{1j} = \dot{y}_j,$$

$$k_{2j} = \dot{y}_j + \frac{2hu_{1j}}{3} = k_{2j}^*,$$

$$k_{3j} = \dot{y}_j + hu_{1j} = k_{3j}^*, \quad (13)$$

$$u_{1j} = f(x_n, y_{jn}, \dot{y}_{jn}), \quad \forall j=1,2,3,\dots,m \quad (14)$$

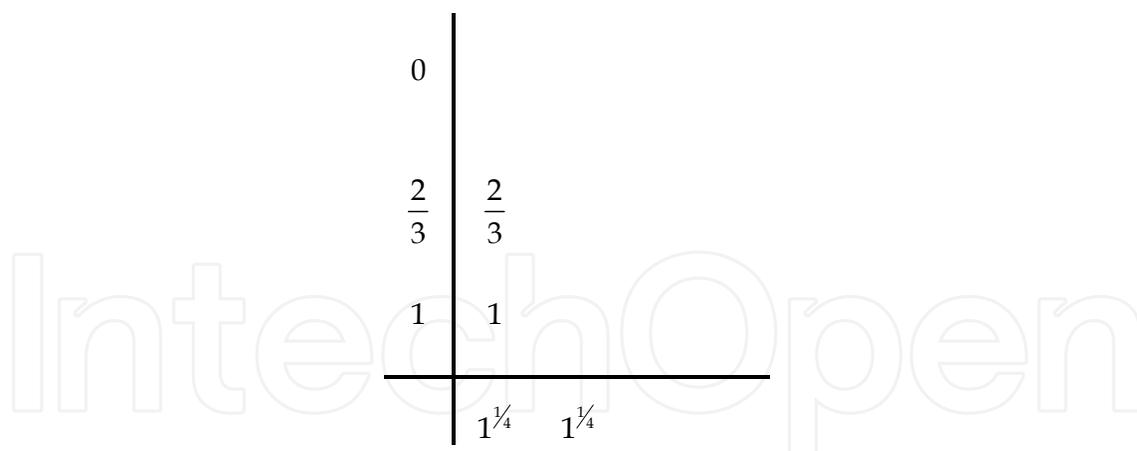
$$u_{2j} = f\left(x_n + \frac{2h}{3}, y_{1n} + \frac{2hk_{11}}{3}, y_{2n} + \frac{2hk_{12}}{3}, \dots, y_{mn} + \frac{2hk_{1m}}{3}, \dot{y}_{1n} + \frac{2hu_{11}}{3}, \dot{y}_{2n} + \frac{2hu_{12}}{3}, \dots, \dot{y}_{mn} + \frac{2hu_{1m}}{3}\right),$$

$$u_{3j} = f(x_n + h, y_{1n} + hk_{11}, y_{2n} + hk_{12}, \dots, y_{mn} + hk_{1m}, \dot{y}_{1n} + hu_{11}, \dot{y}_{2n} + hu_{12}, \dots, \dot{y}_{mn} + hu_{1m}).$$

parallel Runge-Kutta 2-stage 3-order geometric mean of type-I array represent equation (13) takes the form

Hence, the final integration is a weighted sum of three calculated derivatives per time step and the parallel Runge-Kutta 2-stage 3-order geometric mean of type-I formula is given by,

$$y_{n+1} = y_n + hk_1^{1/4} k_2^{3/4}. \quad (15)$$



3.3 Parallel 2-stage 3-order geometric mean runge-kutta formula of type-II

The parallel 2-stage 3-order geometric mean Runge-Kutta formula of type-II is of the form,

$$k_1 = hf(x_n, y_n),$$

$$k_3 = hf\left(x_n - \frac{k_1}{6}, y_n - \frac{k_1}{6}\right).$$

Hence, the final integration is a weighted sum of three calculated derivatives per time step given by,

$$y_{n+1} = y_n + hk_1^4 k_3^{-3}.$$

Parallel 2-stage 3-order geometric Mean Runge-Kutta algorithm of type-II to determine y_j and $\dot{y}_j, j=1,2,3,\dots,m$ is given by,

$$y_{j+1} = y_j + h[k_{1j}^4 k_{3j}^{-3}]. \tag{16}$$

and

$$\dot{y}_{j+1} = \dot{y}_j + h[u_{1j}^4 + u_{3j}^{-3}].$$

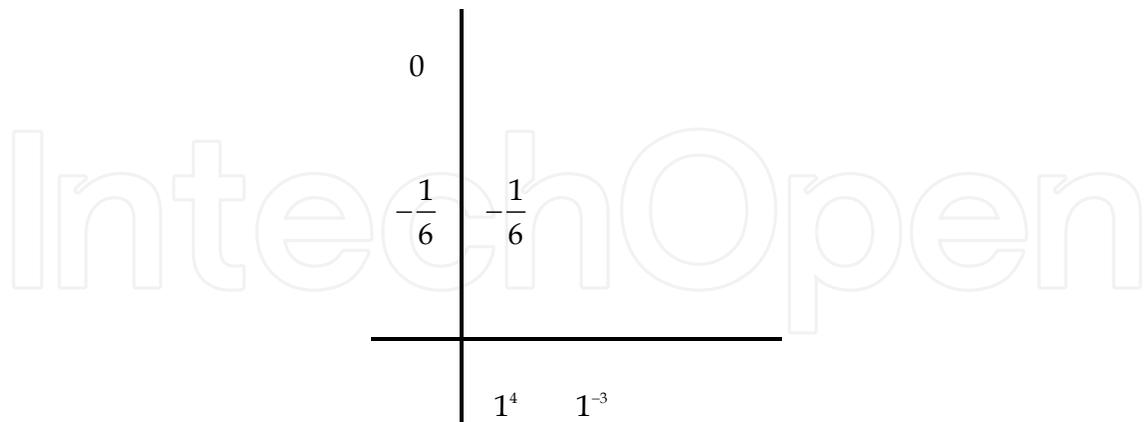
$$k_{1j} = \dot{y}_j,$$

$$k_{3j} = \dot{y}_j - \frac{hu_{ij}}{6}, \tag{17}$$

$$u_{1j} = f(x_n, y_{jn}, \dot{y}_{jn}), \quad \forall j=1,2,3,\dots,m \tag{18}$$

$$u_{3j} = f\left(x_n - \frac{h}{6}, y_{1n} + \frac{hk_{11}}{6}, y_{2n} + \frac{hk_{12}}{6}, \dots, y_{mn} + \frac{hk_{1m}}{6}, \dot{y}_{1n} + \frac{hu_{11}}{6}, \dot{y}_{2n} + \frac{hu_{12}}{6}, \dots, \dot{y}_{mn} + \frac{hu_{1m}}{6}\right).$$

The corresponding parallel Runge-Kutta 2-stage 3-order geometric mean algorithm of type-II array to represent Equation (17) takes the form:



Therefore, the final integration is a weighted sum of three calculated derivatives and the parallel Runge-Kutta 2-stage 3-order geometric mean algorithm formula is given by

$$y_{n+1} = y_n + hk_1^4 k_3^{-3}. \quad (19)$$

4. Results and conclusion

In this paper, the ultimate idea is focused on making use of parallel integration algorithms of Runge-Kutta form for the step by step solution of ordinary differential equations to solve system of second order robot arm problem. The discrete and exact solutions of the robot arm model problem have been computed for different time intervals using equation (5) and y_{n+1} . The values of $e_1(t)$, $e_2(t)$, $e_3(t)$ and $e_4(t)$ can be calculated for any time t ranging from 0.25 to 1 and so on.

To obtain better accuracy for $e_1(t)$, $e_2(t)$, $e_3(t)$ and $e_4(t)$ by solving the equations (5) and y_{n+1} .

| Sol. No. | Time | Exact Solution | Parallel RKAM Solution | Parallel RKAM Error |
|----------|------|----------------|------------------------|---------------------|
| 1 | 0.00 | -1.00000 | -1.00000 | 0.00000 |
| 2 | 0.25 | -0.99365 | -0.99533 | -0.00167 |
| 3 | 0.50 | -0.97424 | -0.97864 | -0.00440 |
| 4 | 0.75 | -0.94124 | -0.94943 | -0.00819 |
| 5 | 1.00 | -0.89429 | -0.90733 | -0.01303 |

Table 1. Solutions of equation (5) for $e_1(t)$

| Sol. | Time | Exact Solution | Parallel RKAM Solution | Parallel RKAM Error |
|------|------|----------------|------------------------|---------------------|
| 1 | 0.00 | 0.00000 | 0.00000 | 0.00000 |
| 2 | 0.25 | 0.05114 | 0.04598 | 0.00515 |
| 3 | 0.50 | 0.10452 | 0.09412 | 0.01044 |
| 4 | 0.75 | 0.15968 | 0.14389 | 0.01578 |
| 5 | 1.00 | 0.21610 | 0.19499 | 0.02110 |

Table 2. Solutions of equation (5) for $e_2(t)$

| Sol. No. | Time | Exact Solution | Parallel RKAM Solution | Parallel RKAM Error |
|----------|------|----------------|------------------------|---------------------|
| 1 | 0.00 | -1.00000 | -1.00000 | 0.00000 |
| 2 | 0.25 | -0.99965 | -0.99973 | -0.00008 |
| 3 | 0.50 | -0.99862 | -0.99871 | 0.00009 |
| 4 | 0.75 | -0.99693 | -0.99700 | 0.00007 |
| 5 | 1.00 | -0.99460 | -0.99462 | 0.00001 |

Table 3. Solutions of equation (5) for $e_3(t)$

| Sol. No. | Time | Exact Solution | Parallel RKAM Solution | Parallel RKAM Error |
|----------|------|----------------|------------------------|---------------------|
| 1 | 0.00 | 0.00000 | 0.00000 | 0.00000 |
| 2 | 0.25 | 0.00277 | 0.00285 | -0.00007 |
| 3 | 0.50 | 0.00545 | 0.00560 | -0.00015 |
| 4 | 0.75 | 0.00805 | 0.00879 | -0.00074 |
| 5 | 1.00 | 0.01056 | 0.01084 | -0.00028 |

Table 4. Solutions of equations (5) for $e_4(t)$

Similarly, by repeating the same computation process for parallel Runge-Kutta 2-stage 3-order geometric mean algorithm of type-I and type-II respectively, yield the required results. It is pertinent to pinpoint out that the obtained discrete solutions for robot arm model problem using the 2-parallel 2-processor 2-Stage 3-order arithmetic mean Runge-Kutta algorithm gives better results as compared to 2-parallel 2-processor 2-stage 3-order geometric mean Runge-Kutta algorithm of type-I and 2-parallel 2-processor 2-stage 3-order geometric mean Runge-Kutta algorithm of type-II. The calculated numerical solutions using 2-parallel 2-processor 2-stage 3-order arithmetic mean Runge-Kutta algorithm is closer to the exact solutions of the robot arm model problem while 2-parallel 2-processor 2-stage 3-order geometric mean Runge-Kutta algorithm of type-I and type-II gives rise to a considerable error. Hence, a parallel Runge-Kutta 2-stage 3-order arithmetic mean algorithm is suitable for studying the system of second order robot arm model problem in a real time environment. This algorithm can be implemented for any length of independent variable on a digital computer.

5. Acknowledgement

The first author would like to extend his sincere gratitude to Universiti Sains Malaysia for supporting this work under its post-doctoral fellowship scheme. Much of this work was carried out during his stay at Universiti Sains Malaysia in 2011. He wishes to acknowledge Universiti Sains Malaysia's financial support.

6. References

- [1] K. Murugesan, N.P.Gopalan and D. Gopal,(2005), "Error free Butcher algorithms for linear electrical circuits", *ETRI Journal*, Vol.27, No.2, pp.195-205.
- [2] J.Y. Park, D.J.Evans, K. Murugesan, S. Sekar, and V.Murugesan,(2004), "Optimal control of singular systems using the RK-Butcher algorithm", *International Journal of Computer Mathematics*, Vol.81, No.2, pp.239-249.
- [3] J.Y. Park, K. Murugesan, D.J.Evans, S. Sekar, and V.Murugesan,(2005), "Observer design of singular systems (transistor circuits) using the RK-Butcher algorithm", *International Journal of Computer Mathematics*, Vol.82, No.1, pp.111-123.
- [4] S.Sekar, V.Murugesan and K.Murugesan (2004), "Numerical strategies for the system of second order IVPs Using the RK-Butcher algorithms", *International Journal of Computer Science and Applications*, Vol.1, No.2, pp.96-117.
- [5] Z.Taha, "Approach to variable structure control of industrial robots", pp.53-59, Compiled by K.Warwick and A.Pugh, (1988), *Robot Control: Theory and applications*, (I E E Control Engineering Series, Peter Peregrinus Ltd, North-Holland.
- [6] S.Oucheriah, (1999), "Robust tracking and model following of uncertain dynamic delay systems by memory less linear controllers", *IEEE Transactions on automatic control*, Vol.44. No.7, pp. 1473-1481.
- [7] D.Lim and H. Seraji, (1997), "Configuration control of a mobile dexterous Robot: Real time implementation and experimentation", *International Journal of Robotics Research*, Vol. 16, No.5. pp. 601-618.

- [8] M.M.Polvcarpou and P.A.Loannou, (1996), "A Robust adaptive non-linear control Design", *Automatica (Journal of IFAC)*, Vol.32. No. 3. pp. 423-427.
- [9] D.J.Evans and B.B.Sanugi, (1989), "A parallel Runge-Kutta integration method", *Parallel Computing*, Vol.11, pp.245-251.
- [10] Zhihua. Qu, (1992), "Robot control of a class of non-linear uncertain systems", *IEEE Transactions on Automatic Control*, Vol.37, No. 9, pp. 1437-1442.
- [11] R.K.Alexander and J.J.Coyle,(1990), "Runge-Kutta methods for differential-algebraic systems", *SIAM Journal of Numerical Analysis*, Vol. 27, No.3, pp. 736-752.
- [12] D.J.Evans, (1991), "A new 4th order Runge-Kutta method for initial value problems with error control", *International Journal of Computer Mathematics*, Vol.139, pp. 217-227.
- [13] C.Hung,(2000), "Dissipativity of Runge-Kutta methods for dynamical systems with delays", *IMA Journal of Numerical Analysis*, Vol.20, pp. 153-166.
- [14] L.F.Shampine and H.A.Watts, (1977), "The art of a Runge-Kutta code. Part-I", *Mathematical Software*, Vol.3. pp. 257-275.
- [15] J.C. Butcher, (1964), "On Runge processes of higher order", *Journal of Australian Mathematical Society*, Vol.4. p.179.
- [16] J.C. Butcher,(1987), "The numerical analysis of ordinary differential equations: Runge-Kutta and general linear methods", *John Wiley & Sons, U.K.*
- [17] J.C.Butcher,(1990), "On order reduction for Runge-Kutta methods applied to differential-algebraic systems and to stiff systems of ODEs", *SIAM Journal of Numerical Analysis*, Vol.27, pp. 447-456.
- [18] L.F. Shampine and M.K.Gordon,(1975), "Computer solutions of ordinary differential equations", *W.H.Freeman. San Francisco. CA.* p. 23.
- [19] D. Gopal, V. Muruges and K.Murugesan,(2006), "Numerical solution of second-order robot arm control problem using Runge-Kutta-Butcher algorithm", *International Journal of Computer Mathematics*, Vol. 83, No. 3, pp.345-356.
- [20] Z.Taha, (1987), *Dynamics and Control of Robots*, Ph.D Thesis, University of Wales.
- [21] H.P. Huang and Tseng,W.L,(1991), "Asymtotic observer design for constrained robot systems", *IEE Proceedings Pt-D*,138,3, pp.211-216,1991.
- [22] H.Krishnan and N.Haris Mcclamroch,(1994), "Tracking in non-linear differential algebra control systems with applications to constrained robot systems", *Automatica*, Vol.30,No.12,pp.1885-1897.
- [23] C.Hung, (2000), "Dissipativity of Runge-Kutta methods for dynamical systems with delays", *IMA Journal of Numerical Analysis*, Vol.20, pp.153-166.
- [24] D. P.Dhayabaran, E.C. H.Amirtharaj, K. Murugesan and D.J. Evans, (2006), "Numerical solution robot arm model using STWS RKHEM methods", *LNCS, computational methods*, pp.1695-1699.
- [25] R.Ponalagusamy and S.Senthilkumar,(2010), "System of second order robot arm problem by an efficient numerical integration algorithm", *International Journal of Computational Materials Science and Surface Engineering*, Vol.3, pp. 237-250.

- [26] R.Ponalagusamy and S.Senthilkumar,(2009), "Investigation on numerical solution for robot arm problem", Journal of Automation, Mobile Robotics and Intelligent Systems, Vol.3, No.3, pp.34-40.

IntechOpen

IntechOpen



Robot Arms

Edited by Prof. Satoru Goto

ISBN 978-953-307-160-2

Hard cover, 262 pages

Publisher InTech

Published online 09, June, 2011

Published in print edition June, 2011

Robot arms have been developing since 1960's, and those are widely used in industrial factories such as welding, painting, assembly, transportation, etc. Nowadays, the robot arms are indispensable for automation of factories. Moreover, applications of the robot arms are not limited to the industrial factory but expanded to living space or outer space. The robot arm is an integrated technology, and its technological elements are actuators, sensors, mechanism, control and system, etc.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

S. Senthilkumar and Abd Rahni Mt Piah (2011). Solution to a System of Second Order Robot Arm by Parallel Runge-Kutta Arithmetic Mean Algorithm, Robot Arms, Prof. Satoru Goto (Ed.), ISBN: 978-953-307-160-2, InTech, Available from: <http://www.intechopen.com/books/robot-arms/solution-to-a-system-of-second-order-robot-arm-by-parallel-runge-kutta-arithmetic-mean-algorithm>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen