Journal of Software Engineering

ISSN 1819-4311

Academic Journals Inc.

www.academicjournals.com

Journal of Software Engineering 4 (2): 147-155, 2010
ISSN 1819-4311 / DOI: 10.3923/jse.2008.1.9

A Novel and Efficient Computational Algorithm of STWS for Generalized Linear Non-Singular/Singular Time Varying Systems*

R. Ponalagusamy
Department of Mathematics, National Institute of Technology,
Tiruchirapalli-620 015, Tamilnadu, India

Abstract: A new computational algorithm of Single-Term Walsh Series (STWS) for solving 2nth order state space equation representing the generalized linear, non-singular or singular, time varying system has been proposed. It is noticed from the literature that a second order (or third order) state space system (r×r matrix) is solved by converting into its first order (or second order) state space system resulting that the size of original system becomes doubled (2r×2r matrix) and the matrix to be inverted has its size drastically increased. In general, suppose that the problem of 2nth order state space system with r unknown dependent variables is to be solved by the method of second order state space formulation via STWS developed by others, then the size of the matrix to be inverted becomes (nr×nr). It is further noted that for a differential equations of (2n-1)th order with r unknown dependent variables, its order should, first be made as 2n by differentiating the differential equations of (2n-1)th order with respect to the independent variable and the resultant matrix to be inverted is of dimensions (nr×nr) in the case of second order state space formulation via STWS. In contrast to the technique mentioned above developed by others, the present numerical algorithm solves the given state space system with any order without converting into its lower order which in turn, implies that the original size of system matrix and the matrix to be inverted are not altered. So, the proposed new numerical algorithm is computationally very effective in lesser computing time as well as storage space.

Key words: Numerical computation, STWS, non-singular/singular time varying systems, generalized linear system

INTRODUCTION

A good amount of work has been done in recent years on the application of Walsh Functions (WFs) to various problems such as non-linear dynamic systems, design and analysis of time varying systems and optimization and identification of systems. Chen and Shih (1978) have applied Walsh series analysis to the optimal control of time varying linear systems. Sivaramakrishnan and Srisailam (1985) have presented the WFs method to obtain the solution of second order state space systems. Sheih *et al*. (1978) have obtained the solution of state space equations using Block Pulse Function (BPF) techniques. Rao *et al*. (1980) and Rao (1983) have introduced the Single Term Walsh Series (STWS) approach to remove inconveniences of the WF and BPF methods. Campbell and Rose (1982) have shown that a singular second order system may be decomposed into a linear and a singular subsystem. Using the single term Walsh series method, vibrating mechanical system has been analyzed by Palanisamy and Arunachalam (1987). Thanushkodi *et al*. (1988) have solved first and third order linear differential equations with constant coefficients, by second order state space formulation via STWS. In many investigations (Palanisamy and Arunachalam, 1987; Thanushkodi *et al*., 1988; Thanushkodi *et al*., 1990), second (or third) order state space equations are transformed into their equivalent first (or second) order state space equations while analyzing second (or third) order state

**Originally Published in Journal of Software Engineering, 2008* space equations. Also, Murugesan *et al*. (2000) converted the observer design in a generalized state-space transistor circuit into a second order system. By doing this, the sizes of the original matrices (r×r) are doubled (2r×2r) and the matrix to be inverted has its size drastically enhanced which implied that the technique mentioned above is not computationally effective and consequently demands large computing time and storage space as well.

STWS method is being adapted by many investigators to compute numerical solutions of industrially applicable problems. Recently, Sepehrian and Razzaghi (2005) have used STWS method to solve the non-linear Volterra-Hammerstein integral equations in which the problem has been reduced to solving a system of non-linear algebraic equations. They have demonstrated the validity and applicability of the technique with the help of several examples. A general and efficient pole-placement solution for linear time-invariant systems with state-PID feedback is derived (Guo *et al*., 2006). Their results open a new area for the design and tuning of state-PID feedback types of controller. Boukas (2006) has dealt with the class of linear discrete-time systems with varying time delay. The problems of stability and stabilizability for this class of systems are discissed. A numerical comparison between the variational iteration method and the Adomain decomposition method for solving linear systems of fractional differential equations has been presented by Momani and Odibat (2007).

It is of immense importance to point out that suppose the problem of 2nth order state-space system with r unknown dependent variables is solved by the method of second-order state space formulation via STWS developed by Thanushkodi *et al*. (1988) and Murugesan *et al*. (2000), then the size of the matrix to be inverted becomes (nr×nr). To have a method that is computationally effective, a modest effort has been made to develop a new numerical algorithm that computes numerical solution of state space system of any order.

WALSH SERIES

A function f(t) integrable in [0,1) can be approximated using Walsh series as

$$f(t) = \sum_{i=0}^{\infty} f_i \psi_i(t) \tag{1}$$

Where, $\Psi_i(t)$ is ith Walsh function and f_i is the corresponding coefficient. In practice, while approximating a function only the first k components are considered where k is an integral power of 2. If the coefficients of the Walsh functions are concisely written as k vectors then

$$F = [f_0, f_1, \ldots, f_{k-1}]^T \tag{2}$$

Where, T denotes transpose and

$$\Psi(t) = [\Psi_0(t), \Psi_1(t), \ldots, \Psi_{k-1}(t)]^T \tag{3}$$

Then,

$$f(t) \approx F^T \Psi(t) \tag{4}$$

The coefficients f_i are chosen to minimize the mean integral square error,

$$e = \int_0^1 [f(t) - F^T \psi(t)]^2 dt \tag{5}$$

The coefficients are given by

$$f_i = \int_0^1 f(t)\psi_i(t)dt \tag{6}$$

It has been proved that

$$\int_0^t f(t)dt \approx F^T E\psi(t) \tag{7}$$

Where, E is called the operational matrix of integration in terms of WFs. In STWS, E = 1/2 (Rao *et al*., 1980).

NUMERICAL ALGORITHM FOR GENERAL1ZED LINEAR STATE SPACE ENGINEERING SYSTEM

Let us consider a generalized (2nth order) non-singular/singular time varying system. The dynamics of this system is described by the 2nth order state space equation as

$$A\, x^{(2n)}(t) = B_1\, x^{(2n-1)}(t) + B_2\, x^{(2n-2)}(t) + \ldots + B_{2n-1}\, x^{(1)}(t) + B_{2n}\, x(t) + C\, u(t) \tag{8}$$

Where, x(t) is an r-state vector, A is an either non-singular or singular $r\times r$ matrix, B_1, B_2, …, B_{2n}, are r x r matrices, C is an r x m matrix, u(t) is an m-input vector, $x^{(1)}(t)$ is the first derivative of $x(t)(=\dot{x}(t)), x^{(2)}(t)$ is the second derivative of x (t) (= $\ddot{x}$ (t)), …, (t)) and $x^{(2n)}(t)$ is the 2nth derivative of x(t). The initial conditions are

$$x^{(2n-1)}(t) = x_0^{(2n-1)},\ x^{(2n-2)}(t) = x_0^{(2n-2)},\ \ldots,\ x^{(1)}(t) = x_0^{(1)} \text{ and } x(t) = x_0 \text{ at } t = 0 \tag{9}$$

It is of importance to note that for a given value of 2n and r, systems of the form in Eq. 8 along with Eq. 9 cover a variety of cases in many areas of science and technology such as electrical circuits, network theory, power systems, nuclear reactor plants, robotics, aircraft systems, neurological events, etc.With the STWS approach, the given function is expanded into single term Walsh series in the normalized interval $\tau \in (0, 1)$, which corresponds to $\tau \in (0, 1/k)$ by defining $t = \tau/k$, k being any integer. In the normalized interval, Eq. 8 becomes

$$A x^{(2n)}(\tau) = \frac{B_1}{k} x^{(2n-1)}(\tau) + \frac{B_2}{k^2} x^{(2n-2)}(\tau) + \ldots + \frac{B_{2n-1}}{k^{2n-1}} x^{(1)}(\tau) + \frac{B_{2n}}{k^{2n}} x(\tau) + \frac{C}{k^{2n}} u(\tau) \tag{10}$$

Expanding $x^{(2n)}(\tau)$, $x^{(2n-1)}(\tau)$, ..., $x^{(1)}(\tau)$, $x(\tau)$ and $u(\tau)$ by STWS at the ith interval

$$X^{(2n)}(\tau) = R_{2n}(i)\, \Psi_0(\tau),\ x^{(2n-1)}(\tau) = R_{2n-1}(i)\, \Psi_0(\tau), \ldots\ldots\ldots\ldots, x^{(1)}(\tau) = R_1(i)\, \Psi_0(\tau),$$
$$x(\tau) = R(i)\, \Psi_0(\tau),\ u(\tau) = P(i)\, \Psi_0(\tau) \tag{11}$$

$R_{2n}(i)$ is the block pulse value of the 2nth order rate vector, $R_{2n-1}(i)$ is the block pulse value of the (2n-1)th order rate vector, …, $R_1(i)$ is the block pulse value of the first order rate vector, R(i) is the block pulse value of state vector of $x(\tau)$ respectively and P(i) is the block pulse value of input vector $u(\tau)$.

The following recursive relationship for both block pulse and discrete values are obtained as

$$R_{2n}(i) = \left[A - \frac{B_1}{2k} - \frac{B_2}{(2k)^2} - \cdots - \frac{B_{2n}}{(2k)^{2n}} \right]^{-1} Q(i)$$

$$\begin{aligned} &R_{2n-1}(i) = (1/2)\, R_{2n}(i) + x^{(2n-1)}(i-1),\ R_{2n-2}(i) = (1/2)\, R_{2n-1}(i) + x^{(2n-2)}(i-1), \ldots\ldots, \\ &R_1(i) = (1/2)\, R_2(i) + x^{(1)}(i-1),\ R(i) = 1/2\, R_1(i) + x(i-1), \\ &x^{(2n-1)}(i) = R_{2n}(i) + x^{(2n-1)}(i-1),\ x^{(2n-2)}(i) = R_{2n-1}(i) + x^{(2n-2)}(i-1), \ldots\ldots, \\ &x^{(1)}(i) = R_2(i) + x^{(1)}(i-1),\ x(i) = R_1(i) + x(i-1) \end{aligned} \tag{12}$$

Where:

$$\begin{aligned} Q(i) = \Bigg[&\frac{B_1}{k} x^{(2n-1)}(i-1) + \frac{B_2}{k^2}\left\{ \frac{1}{2} x^{(2n-1)}(i-1) + x^{(2n-2)}(i-1) \right\} + \frac{B_3}{k^3}\left\{ \frac{1}{4} x^{(2n-1)}(i-1) + \right. \\ &\left. \frac{1}{2} x^{(2n-2)}(i-1) + x^{(2n-3)}(i-1) \right\} + \cdots + \frac{B_{2n}}{k^{2n}}\left\{ \frac{1}{2^{2n-1}} x^{(2n-1)}(i-1) + \right. \\ &\left. \frac{1}{2^{2n-2}} x^{(2n-2)}(i-1) + \cdots + \frac{1}{2} x^{(1)}(i-1) + x(i-1) \right\} + \frac{C}{k^{2n}} P(i) \Bigg] \end{aligned} \tag{13}$$

and i = 1,2,3, … the interval number. The x(i) give the discrete values of the state vector and the R(i) give the block-pulse function (BPF) values of the state for any length of time. Higher accuracy can be obtained for increased value of k. This is the main advantage of the present method.

ILLUSTRATIVE EXAMPLES

To compare the results and advantage of the new method and software discussed in this study, two examples are considered. Finally, the presently computed results are compared with that of first order and second order state formulations.

Example 1

Consider the example taken from Sivaramakrishnan and Srisailam (1985).

$$x^{(3)}(t) + 6\, x^{(2)}(t) + 11\, x^{(1)}(t) + 6\, x(t) = 12u(t) \tag{14}$$

with $x(0) = 5$, $x^{(1)}(0) = -6$ and $x^{(2)}(0) = 14$, excited by an unit step input u(t).
After differentiating the above equation once and defining the state variables as below,

$$y_1 = x,\ y_1^{(1)} = x^{(1)},\ y_2 = y_1^{(2)} = x^{(2)},\ y_2^{(1)} = y_1^{(3)},\ y_2^{(2)} = y_1^{(4)} \tag{15}$$

the second-order state space form gets transferred to

$$\begin{bmatrix} y_1^{(2)}(t) \\ y_2^{(2)}(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -6 & -6 \end{bmatrix} \begin{bmatrix} y_1^{(1)}(t) \\ y_2^{(1)}(t) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & -11 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} u^{(1)}(t) \tag{16}$$

with the following initial conditions:

$$\begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 5 \\ 14 \end{bmatrix} \text{and} \begin{bmatrix} y_1^{(1)}(0) \\ y_2^{(1)}(0) \end{bmatrix} = \begin{bmatrix} -6 \\ -36 \end{bmatrix}$$

Taking $z_1^{(1)} = z_2$ and $z_3^{(1)} = z_4$, the first-order state-space form of Eq. 16 may be obtained as

$$\begin{bmatrix} z_1^{(1)}(t) \\ z_2^{(1)}(t) \\ z_3^{(1)}(t) \\ z_4^{(1)}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -6 & -11 & -6 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{bmatrix} u^{(1)}(t) \tag{17}$$

with the following initial conditions

$$\begin{bmatrix} z_1(0) \\ z_2(0) \\ z_3(0) \\ z_4(0) \end{bmatrix} = \begin{bmatrix} 5 \\ -6 \\ 14 \\ -36 \end{bmatrix}$$

It is of importance to note that the present algorithm computes the results of Eq. 14, the Eq. 16 is numerically solved by the second-order state space STWS (Thanushkodi *et al*., 1988; Palanisamy and Arunachalam, 1987) and the Eq. 17 has been solved by the first-order state space STWS (Sivaramakrishnan and Srisailam, 1985). And these computed results are tabulated (Table 1 and 2). A comparative study reveals that the results computed by the present algorithm tally exactly with that of Thanushkodi *et al*. (1990) and Sivaramakrishnan and Srisailam (1985) with the added advantage of considerable reduction in the size of the matrix to be inverted. In the present case it is a mere (1×1) matrix compared to an (2×2) matrix involved in the second-order formulation and a matrix of size (4×4) in the case of first-order state space formulation.

Table 1: Comparison of solutions using STWS method (Example 1)

	z_1 in the first-order formulation (Sivaramakrishnan and Srisailam, 1985)		y_1 in the second-order formulation (Thanushkodi *et al*., 1988)		Value of x (Present method)	
Time (Sec)	Block pulse value	Discrete value	Block pulse value	Discrete value	Block pulse value	Discrete value
0.000		5.00	5.00		5.00	
0.125	4.416161	3.832323	4.416162	3.832323	4.416162	3.832323
0.375	3.501937	3.171550	3.501937	3.171550	3.501937	3.171550
0.625	2.975986	2.780422	2.975986	2.780422	2.975986	2.780422
0.875	2.659330	2.538239	2.659330	2.538239	2.659330	2.538239

Table 2: Comparison of solutions using STWS method (Example 1)

	$z_2 = z_1^{(1)}$ in the first-order formulation (Sivaramakrishnan and Srisailam, 1985)		$y_1^{(1)}$ in the second-order formulation (Thanushkodi *et al*., 1988)		Value of $x^{(1)}$ (Present method)	
Time (Sec)	Block pulse value	Discrete value	Block pulse value	Discrete value	Block pulse value	Discrete value
0.000		-6.0		-6.0		-6.0
0.125	-4.670707	-3.341414	-4.670707	-3.341414	-4.670707	-3.341414
0.375	-2.643094	-1.944773	-2.643094	-1.944773	-2.643094	-1.944773
0.625	-1.564512	-1.184251	-1.564512	-1.184251	-1.564512	-1.184251
0.875	-0.968773	-0.753215	-0.968773	-0.753215	-0.968773	-0.753215

Generalizing the above analysis, it is demonstrated that for a differential equation of (2n-1)th order with r unknown dependent variables, the originally given matrix to be inverted is of dimensions (r×r) in the present analysis whereas the size of matrix to be inverted becomes (nr×nr) in the second-order state space formulation and in the case of the first-order state space formulation, its size becomes (2nr×2nr).

Example 2

Let us consider the fourth order linear ordinary differential equation as

$$15\, x^{(4)}(t) - 2\, x^{(2)}(t) - 80\, x^{(1)}(t) = 0 \tag{18}$$

with the initial conditions

$$x(0) = 1.8354,\ x^{(1)}(0) = 0.5,\ x^{(2)}(0) = 1,\ x^{(3)}(0) = 0.5$$

Let us define the state variables as

$$y_1 = x,\ y_1^{(1)} = x^{(1)},\ y_2 = y_1^{(2)} = x^{(2)},\ y_2^{(1)} = y_1^{(3)} = x^{(3)},\ y_2^{(2)} = x^{(4)} \tag{19}$$

With the help of Eq. 19, the second order state space system of Eq. 18 becomes

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_1^{(2)}(t) \\ y_2^{(2)}(t) \end{bmatrix} = \begin{bmatrix} -40 & 3 \\ 0 & 0.4 \end{bmatrix} \begin{bmatrix} y_1^{(1)}(t) \\ y_2^{(1)}(t) \end{bmatrix} \tag{20}$$

with the following initial conditions:

$$\begin{bmatrix} y_1(0) \\ y_2(0) \end{bmatrix} = \begin{bmatrix} 1.8354 \\ 1.0000 \end{bmatrix} \text{and} \begin{bmatrix} y_1^{(1)}(0) \\ y_2^{(1)}(0) \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

It is of interest to mention that Eq. 20 represents the stiff system.

Taking $z_1 = y_1$, $z_2 = y_1^{(1)}$, $z_3 = y_2$ and $z_4 = y_2^{(1)}$, the first-order state space form of Eq. 20 may be obtained as

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_1^{(1)}(t) \\ z_2^{(1)}(t) \\ z_3^{(1)}(t) \\ z_4^{(1)}(t) \end{bmatrix} = \begin{bmatrix} 0 & -40 & 0 & 3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0.4 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{bmatrix} \tag{21}$$

with the following initial conditions:

$$\begin{bmatrix} z_1(0) \\ z_2(0) \\ z_3(0) \\ z_4(0) \end{bmatrix} = \begin{bmatrix} 1.8354 \\ 0.5 \\ 1.0 \\ 0.5 \end{bmatrix}$$

The present algorithm performs the numerical computation of Eq. 18 as it is where as the Eq. 20 is numerically solved by the second-order state space STWS (Sekar *et al*., 2004) and the Eq. 21 has been solved using the first-order state space STWS adopted by Sivaramakrishnan and Srisailam (1985) and Ponalagusamy *et al*. (2001). The computed results are shown in Table 3 and 4.

Table 3: Comparison of numerical values of x using STWS (Example 2)

Time (sec)	Numerical value of x Present algorithm	Sekar *et al.* (2004)	Sivaramakrishnan and Srisailam (1985)	Ponalagusamy *et al.* (2001)
0.0	1.8353960	1.8353960	1.8353960	1.8353960
0.6	2.0987460	2.0987460	2.0987460	2.0987460
1.2	2.4188192	2.4188192	2.4188192	2.4188192
1.8	2.8257115	2.8257115	2.8257115	2.8257115
2.4	3.3429737	3.3429737	3.3429737	3.3429737

Table 4: Comparison of numerical values of x using STWS (Example 2)

Time (sec)	Numerical value of x Present algorithm	Sekar *et al.* (2004)	Sivaramakrishnan and Srisailam (1985)	Ponalagusamy *et al.* (2001)
0.0	1.8353960	1.8353960	1.8353960	1.8353960
0.125	1.8944808	1.8944808	1.8944808	1.8944808
0.375	1.9971855	1.9971855	1.9971855	1.9971855
0.625	2.1106055	2.1106055	2.1106055	2.1106055
0.875	2.2359540	2.2359540	2.2359540	2.2359540
1.125	2.3744856	2.3744856	2.3744856	2.3744856

It is observed that the results computed by the present algorithm tally exactly with that of Sivaramakrishnan and Srisailam (1985), Ponalagusamy *et al*. (2001) and Sekar *et al*. (2004) with the advantage of considerable reduction in the computational efforts. The matrix to be inverted in the present method has just reduced itself to a scalar (1×1), compared to a (2×2) matrix involved with the second-order state space STWS and an (4×4) matrix in the first-order state space STWS formulation.

Generalizing the above analysis, it is verified that for a differential equation of 2nth order with r unknown dependent variables, the originally given matrix to be inverted is of dimensions ($r\times r$) in the present analysis whereas the size of matrix to be inverted becomes ($nr\times nr$) in the second-order state space formulation and in the case of the first-order state space formulation, its size becomes ($2nr\times 2nr$).

DISCUSSION

The present numerical algorithm given by Eq. 12 has been applied to several practical problems (second and third order state space systems representing mechanical vibrating systems, electrical power systems, dynamic systems, etc.). A comparative study reveals that the results computed by the present numerical algorithm are exactly the same as obtained by Sivaramakrishnan and Srisailam (1985), Thanushkodi *et al*. (1988), Palanisamy and Arunachalam (1987) and Sekar *et al*. (2004) in their respective illustrative examples that are solved using the first/second order state space formulation via STWS (Table 1-4), with the significant feature of considerable reduction in the system matrix structure and the size of the matrix to be inverted resulting in lesser computing time and saving in storage space. (Table 5). It is of importance to mention here that the present method does not require the Kronecker product of matrices and there is no need for the operational matrices of integration. Unlike the approaches adapted by Sivaramakrishnan and Srisailam (1985), Thanushkodi *et al*. (1988), Palanisamy and Arunachalam (1987) and Sekar *et al*. (2004), the new version of computational algorithm of Single-Term-Walsh-Series(STWS) solves the given state space system with any order without converting into its lower order which in turn, implies that the original size of system matrix and the size of matrix to be inverted are not increased and remained the same. So, the proposed new numerical algorithm is computationally very effective in lesser computing time as well as storage space.

Table 5: Comparative Analysis of Present and Existing Numerical Techniques

	General case				Particular case		
Specification	Sivarama-krishnan and Srisailam (1985)	Thanuskodi *et al.* (1990)	Murugesan *et al.* (2000)	Present study	Sivarama-krishnan and Srisailam (1985)	Thanushkodi *et al.* (1990)	Present study
Order of linear Differential equation	2n or 2n-1	2n or 2n-1	2n or 2n-1	2n or 2n-1	3	3	3
Actual number of dependent variables to be determined	r	r	1	r	1	1	1
Actual number of size of the matrix	(r×r)	(r×r)	(1×1)	(r×r)	(1×1)	(1×1)	(1×1)
Resultant size of the matrix obtained by respective method	(2n×2n)	(nr×nr)	(n×n)	(r×r)	(4×4)	(2×2)	(1×1)
Size of the matrix to be inverted	(2n×2n)	(nr×nr)	(n×n)	(r×r)	(4×4)	(2×2)	(1×1)
Resultant number of dependent variables to be numerically computed	2nr	nr	n	r	4	2	1

CONCLUSIONS

A new STWS method in 2nth order state-space formulation for the analysis of the generalized linear, non-singular or singular, time varying system governing 2nth order state-space equation has been introduced. The pivotal characteristic of this method is that the sizes of the system matrices and the matrix to be inverted are smaller (the original size of the given system matrices and the size of matrices to be inverted are unaltered) in the present numerical algorithm than those in the existing methods (first and second-order state-space formulations via STWS) as the value of 2n (the order of state space system) increases. It is observed from the research works done by Sivaramakrishnan and Srisailam (1985), Thanushkodi *et al.* (1988), Palanisamy and Arunachalam (1987) and Sekar *et al.* (2004) that their approach does inevitably make the given system of equations more complicated in its structure and requires the inversion of very large matrices. Hence, the algorithm presently developed reduces appreciably computational burden and this is applicable to handle large-scale problems. This is simple and recursive in nature and it is easy to be implemented for a digital computer. Further, the recursive relations provide block pulse values, discrete values and continuous approximation for any length of time and there is no restriction on k as in the case with Walsh series approach. It can also be parallelized by following any of the strategies such as ones presented in Chu *et al.* (1993) and Li *et al.* (1999) in the case of computing matrix inverse which, in turn, reduces the computational burden.

REFERENCES

Boukas, E.K., 2006. Discrete-Time Systems with time-varying time delay: Stability and Stabilizability, Mathematical Problems Eng., pp: 1-10.

Campbell, S.L. and N.J. Rose, 1982. A second order singular linear system arising in electric power systems analysis. Int. J. Systems Sci., 13: 101-108.

Chen, W.L. and Y.P. Shih, 1978. Analysis and optimization of time varying linear systems via Walsh functions. Int. J. Control, 27: 917-982.

Chu, E., A. George and D. Quesnel, 1993. Parallel matrix inversion on a subcube-grid. Parallel Computing, 19: 243-256.

Guo, G., Z. Ma and J. Qiao, 2006. State-PID feedback control with application to a robot vibration absorber. Int. J. Modeling, Identification Control,1: 38-43.

Li, K., Y. Pan and S.Q. Zheng, 1999. Parallel matrix computations using a reconfigurable pipelined optical bus. J. Parallel Distributed Computing, 50: 13-30.

Momani, S. and Z. Odibat, 2007. Numerical approach to differential equations of frational order. J. Computational Applied Mathematics, 207: 96-110.

Murugesan, K., D. Paul Dhayabaran and E.C. Henry Amirtharaj, 2000. A study of second order state-space systems of time-invariant and time-varying transistor circuits using the STWS Technique. Int. J. Electronics, 89: 305-315.

Palanisamy, K.R. and V.P. Arunachalam, 1987. Analysis of vibrating systems via single-term Walsh series approach. Int. J. Sys. Sci., 18: 1457-1466.

Ponalagusamy, R., K. Murugesan, D. Paul Dhayabaran and E.C. Henry Amirtharaj, 2001. Numerical solution of heat flow problem by a combined method of Rayleigh-Ritz with STWS and RKHM. AMSE J., 38: 29-48.

Rao, G.P., K.R. Palanisamy and T. Srinivasan, 1980. Extension of computation beyond the limit of normal interval in Walsh Series analysis of dynamical systems. IEEE Trans. Auto. Control, 25: 517-519.

Rao, G.P., 1983. Piecewise Constant Orthogonal Functions and their Applications to Systems and Control. Springer-Verlag, Berlin.

Sekar, S., V. Murugesh and K. Murugesan, 2004. Numerical strategies for the system of second order IVPs using the RK-Butcher algorithms. Int. J. Comp. Sci. Applic., 1: 96-117.

Sepehrian, B. and M. Razzaghi, 2005. Solutionof non-linear Volterra-Hammerstein integral equations via single term walsh-series method, pp: 547-554.

Sivaramakrishnan, A.Y. and M.C. Srisailam, 1985. Walsh series solution of second and third order state-space systems. Int. J. Sys. Sci., 16: 633-639.

Sheih, L.S., C.K. Yeung and B.C. Mcinnis, 1978. Solution of state space equations via block-pulse functions. Int. J. Control, 28: 383-390.

Thanushkodi, K., S. Palanichamy, K.R. Palanisamy and V.P. Arunachalam, 1988. Single-term Waish series solution of second order state-space systems. AMSE Rev., 9: 15-22.

Thanashkodi, K., S. Palanichamy and S.K. Sivanandam, 1990. Analysis of hidden oscillations in sampled-data systems via single term Walsh series approach. AMSE J., 19: 37-41.