Int. J. Nonlinear Anal. Appl. 11 (2020) No. 1, 199-206
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2018.14644.1769

A new approach for computing the exact solutions of DAEs in generalized Hessenberg forms

M. Golchian[a], M. Gachpazan[b], S. H. Tabasi[a,*]

[a]*School of Mathematics and Computer Sciences, Damghan University, P.O.Box 36715-364, Damghan, Iran.*
[b]*Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran.*

(Communicated by Madjid Eshaghi Gordji)

Abstract

In this paper, we propose a new method, with different approach and economical computing, that presents explicit formulas for the exact solutions of a large class DAEs in Hessenberg forms. First, we illustrate the method for linear time-varying DAEs in Hessenberg forms, in order to show the different approach and also the advantages of the method in computing, that make it economical. Then, we describe that the method is efficient for larger classes including special case of non-linear DAEs in Hessenberg forms. Some examples are given to illustrate the proposed method.

Keywords: Differential Algebraic Equations, The Hessenberg Forms, Linear Time-Varying, Linear Systems of Equations, Backward Substitution.
2010 MSC: Primary 65L80.

1. Introduction

Differential Algebraic Equations (DAEs) arise in several areas of sciences and engineering. Specially, in last two decades, lots of activities are done for improving the theory and computations of DAEs. In this paper, we are interested in a large class of DAEs in Hessenberg forms of arbitrary size r, that can be defined, both for linear time-varying and non-linear DAEs.

The Hessenberg forms arise in many applications for the higher index DAEs. Many of the mechanics and variational problems, are of size two and three [22], and also, some beam deflection problems are of size four [10]. Moreover, according to precise definition of solvability in [4], which is equivalent to existence

*Corresponding author
Email addresses: mmgolchian@gmail.com (M. Golchian), gachpazan@um.ac.ir (M. Gachpazan), tabasi@du.ac.ir (S. H. Tabasi)

Received: April 2018 *Revised:* November 2018

and uniqueness of the solutions, it has proved that DAEs in Hessenberg forms of size r, are solvable both for linear time-varying and non-linear DAEs.

Linear time-varying DAEs

$$A(t)x' + B(t)x = f(t), \tag{1.1}$$

with singular A, are better choices, than non-linear DAEs, to show our different approach, in comparison with the other approaches. First, we give an overview of existing approaches for solving linear time-varying DAEs. Linear DAEs with constant coefficients are completely studied. In fact, a comprehensive overview of solvability (or solution concepts) of these DAEs is described in [17], while this description for linear time-varying, according to their additional difficulties, such as non-constant rank, inconsistent initial value and etc, is briefly.

Several methods, both numerical and analytic, have been proposed for solving linear time-varying DAEs [6]-[9], [18], [20], [23]. Generally, the basic idea in all these methods, is based on transforming the DAEs to equivalent underlying Ordinary Differential Equations (ODEs), but with different approaches. For example, systems transferable to Standard Canonical Form (SCF) [5], derivative array approach [18], and differentiation index [19], are some of the most common approaches. However, all of these approaches, have been proved, in some sense, are equivalent (see [18] for more details).

All these transformations include changing coordinates $x = Qy$ and pre-multiplying by P where P and Q are invertible square matrices that both of them must be constructed. although, our method is also based on transformation that includes changing coordinates $x = Qy$ and pre-multiplying by P, that will be described in details in next section, but our proposed method is based on transforming the DAEs to equivalent underlying linear systems of algebraic (not differential) equations. Moreover, other approaches need to prove the existence of such matrices (P, Q) and during the proof, or after that, in order to compute x, they must "construct" P and Q. The construction of P and Q is their main problem in computing, while we just "determine" P and Q in our method and computing. The details, will be illustrated in next section.

2. PRELIMINARIES AND MAIN RESULTS

In this section, first, we define linear time-varying DAEs in Hessenberg forms of size r. Then, we illustrate the computations needed to transform it to linear system of equations in details. Presenting the exact solutions and computing them in practice, will be our next steps. In order to show that computations are as economical as possible, some of the advantages in computing, according to their necessity are mentioned. However, obvious ones, such as memory-consuming or time-consuming, computing simultaneously, shrink from unnecessary calculations and etc, are not mentioned. It should be noted that, the size of all vectors and matrices should be chosen such that all the product matrices be well-defined.

Definition 2.1. The DAE (1.1) is in *Hessenberg form of size r* if it can be written as

$$\begin{bmatrix} I & 0 & . & . & 0 \\ 0 & I & . & . & . \\ . & . & I & . & . \\ . & . & . & I & . \\ 0 & . & . & . & 0 \end{bmatrix} \begin{bmatrix} x_1' \\ . \\ . \\ . \\ x_r' \end{bmatrix} + \begin{bmatrix} B_{1,1} & * & * & B_{1,r-1} & B_{1,r} \\ B_{2,1} & * & * & B_{2,r-1} & 0 \\ 0 & * & * & * & . \\ . & . & * & * & . \\ 0 & . & 0 & B_{r,r-1} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ . \\ . \\ . \\ x_r \end{bmatrix} = \begin{bmatrix} f_1 \\ . \\ . \\ . \\ f_r \end{bmatrix}. \tag{2.1}$$

where $x = [x_1, x_2, ..., x_r]^T$, and x_i, for $i = 1, 2, ..., r$, are vectors. Also, $B_{i,j}$ are matrices, and the product matrix $C = B_{r,r-1}B_{r-1,r-2}...B_{2,1}B_{1,r}$ is non-singular for all t. It should be noted that the $B_{i+1,i}$ matrices do not need, in general, to be square or even invertible, but their sizes must let the product materix $C = B_{r,r-1}B_{r-1,r-2}...B_{2,1}B_{1,r}$ be well-defined and invertible matrix. In fact, it is only the product matrix C that needs to be square and non-singular [4]. Since, the Hessenberg forms of size two and three are the most common, and beside, the proposed algorithm for general Hessenberg form of arbitrary size r, may be needed to be more clarified, in the first example of last section will, we will apply the method for size 3 with all details, but here, we illustrate the proposed method for general Hessenberg form of arbitrary size r

First, we rewrite (2.1) by breaking it in to 3 parts as follows

$$B_{1,r}x_r = f_1 - x_1' - \sum_{j=1}^{r-1} B_{1,j}x_j = \tilde{f}_1, \tag{2.2}$$

$$B_{i+1,i}x_i = f_{i+1} - x_{i+1}' - \sum_{j=i}^{r-2} B_{i,j}x_j = \tilde{f}_{i+1}, \qquad i = 1,\dots,r-2, \tag{2.3}$$

$$B_{r,r-1}x_{r-1} = f_r = \tilde{f}_r. \tag{2.4}$$

For the remainder of this paper, we work with equations (2.2)-(2.4), instead of (2.1). As an obvious benefit, this kind of representation clearly shows that the $B_{i+1,i}$ matrices, as the coefficients of the unknowns, are exactly those matrices that appear in the product matrix $C = B_{r,r-1}B_{r-1,r-2}\dots B_{2,1}B_{1,r}$. In fact, since our only certain information is that, C is non-singular for all t, our approach is based on transforming all the coefficients $B_{i+1,i}$ to C. In order to achieve this aim, we are going to determine appropriate matrices like P_i and Q_i, independent of structure of the $B_{i+1,i}$ matrices, such that, changing coordinates $x_i = Q_iy_i$ and pre-multiplying by P_i, are both well-defined and also transform (2.2)-(2.4) to

$$Cy_r = g_r, \tag{2.5}$$

$$Cy_i = g_i, \qquad i = 2,\dots,r-2, \tag{2.6}$$

$$Cy_{r-1} = g_{r-1}, \tag{2.7}$$

where g_i are the updated versions of right hand sides of (2.2)-(2.4), after the transformation. Obviously, this work shows that, since the transformation is independent of the structure of the $B_{i+1,i}$ matrices, it is not involved with any of the existing and related difficulties, both in theory and computing, in comparison with other approaches.

Now, according to (2.2)-(2.4) and the structure of C, consider the following presentations of the product matrix C

$$C = (B_{r,r-1}B_{r-1,r-2}\dots B_{21})B_{1r}, \tag{2.8}$$

$$C = (B_{r,r-1}B_{r-1,r-2}\dots B_{i+2,i+1})B_{i+1,i}(B_{i,i-1}\dots B_{1r}), \tag{2.9}$$

$$C = B_{r,r-1}(B_{r-1,r-2}\dots B_{2,1}B_{1,r}). \tag{2.10}$$

Equations (2.8)-(2.10) show that the Q_i matrices should be constructed recursively as follows

$$Q_1 = B_{1,r}, \tag{2.11}$$

$$Q_{i+1} = B_{i+1,i}Q_i, \qquad i = 1,\dots,r-2, \tag{2.12}$$

$$Q_r = I. \tag{2.13}$$

After this step, (2.2)-(2.4) become

$$B_{1,r}Q_ry_r = f_1 - x_1' - \sum_{j=1}^{r-1} B_{1,j}x_j, \tag{2.14}$$

$$B_{i+1,i}Q_iy_i = f_{i+1} - x_{i+1}' - \sum_{j=i}^{r-2} B_{i,j}x_j, \qquad i = 1,\dots,r-2, \tag{2.15}$$

$$B_{r,r-1}Q_{r-1}y_{r-1} = f_r. \tag{2.16}$$

It should be noted that, since we will use backward substitution, there is no need to substitute x_i with $Q_i y_i$ in the right hand side, in this step.

The process for P_i is almost the same, but in different order. Again, according to (2.8)-(2.10), the P_i matrices should be constructed as follows

$$P_1 = I, \tag{2.17}$$
$$P_i = P_{i+1} B_{i+2,i+1}, \qquad i = r-2, \dots, 1, \tag{2.18}$$
$$P_r = P_1 B_{1,r}. \tag{2.19}$$

The most important point in this step, as an advantage in memory and time consuming, is to note that, in practice, the P_i matrices DO NOT need to be saved, or even pre-multiplied in the left hand side of (2.2)-(2.4), since, according to (2.8)-(2.10) we already know that the answer will be Cy_i. In fact, the P_i matrices are constructed just to update the right hand sides of (2.2)-(2.4) as follows

$$g_r = f_1 - x_1' - \sum_{j=1}^{r-1} B_{1j} x_j, \tag{2.20}$$

$$g_i = P_i(f_{i+1} - x_{i+1}' - \sum_{j=i}^{r-2} B_{ij} x_j), \qquad i = r-2, \dots, 1, \tag{2.21}$$

$$g_{r-1} = P_{r-1} f_r. \tag{2.22}$$

Moreover, even there is no need to define the new vector g_i and we just use it to make the description easier. After this step, (2.2)-(2.4) will be transformed to simple following form

$$Cy_i = g_i, \qquad i = r-1, r-2, \dots, 1, r. \tag{2.23}$$

The method for solving $Cy_i = g_i$ is, in some sense, arbitrary and depends on our facilities and requirements. Now, recalling that $x_i = Q_i y_i$, according to (2.11)-(2.13), complete the computing of the x_i for $i = r-1, r-2, \dots, 1, r$. The computations will continue by using backward substitution. The computed $x_i = Q_i y_i$, by considering the order of computing, will be substituted in the right hand sides of previous equations and so on. It is clear that, having the same coefficient for all the equations is another advantage of our computing.

After the transformation, by solving (2.5)-(2.7) and using backward substitution, we can compute y_i and according to $x_i = Q_i y_i$, the explicit formulas for the exact solutions can be presented clearly as follows

$$x_i = Q_i C^{-1} P_i g_i, \qquad i = r-1, r-2, \dots, 1, r. \tag{2.24}$$

Of course, this presentation of x_i is, in some sense, symbolic. As a matter of fact, as we illustrated, computing x_i in practice, is different. From numerical angle of view, computing C^{-1} is expensive. So, instead of computing C^{-1} directly, we can solve $Cy_i = g_i$ for computing. In fact, it depends on size of the product matrix C.

3. generalization

In this section, we want to generalize our method for lager classes of DAEs in Hessenberg form, including special case of non-linear DAEs in Hessenberg forms. In order to achieve this goal, consider (2.2)-(2.4), but this time, as follows

$$B_{1r} x_r = \tilde{f}_1(t, x_1, x_1', x_2, \dots, x_{r-1}), \tag{3.1}$$
$$B_{i+1,i} x_i = \tilde{f}_{i+1}(t, x_{i+1}, x_{i+1}', x_{i+2}, \dots, x_{r-1}), \qquad i = 1, \dots, r-2, \tag{3.2}$$
$$B_{r,r-1} x_{r-1} = \tilde{f}_r(t). \tag{3.3}$$

where $\tilde{f}_i, i = 1, ..., r$, are non-linear and sufficiently smooth functions of their variables. As we illustrated in previous section, the process of transforming the DAE to $Cy_i = g_i$, is completely independent of the structure of $\tilde{f}_i$, as a function of t, x_i and their derivatives, and according to (2.20)-(2.22), for g_i, as well. Since we have used backward substitution, without loss of generality and only under the assumption of definition 2.1, the method allows us to find the exact solutions of following DAEs that we call them *DAEs in generalized Hessenberg form of size r*

$$B_{1r}x_r = \tilde{F}_1(t, x_1, ..., x_1^{(n_1)}, ..., x_{r-1}, ..., x_{r-1}^{(n_{r-1})}), \tag{3.4}$$

$$B_{i+1,i}x_i = \tilde{F}_{i+1}(t, x_{i+1}, ..., x_{i+1}^{(n_{i+1})}, ..., x_{r-1}, ..., x_{r-1}^{(n_{r-1})}), \qquad i = 1, ..., r-2, \tag{3.5}$$

$$B_{r,r-1}x_{r-1} = \tilde{F}_r(t), \tag{3.6}$$

where $n_i \in \mathbb{N}$ and $\tilde{F}_i, i = 1, ..., r$, are also like $\tilde{f}_i$, non-linear and sufficiently smooth functions of their variables. The second and third examples of last section will ***

4. applications

Applications of DAEs overlap to some extent, but the most famous groups of these applications that are based on how the equations are derived rather than on the type of equations that result, are: *Constrained Variational Problems*, *Network Modeling*, *Model Reduction and Singular Perturbations*, *Chemical proces* and *Discretization of PDEs* [4].

Although, the efficiency of proposed method can be shown for any of these five groups with several examples, we have tried to select some special examples that show this efficiency better.

Example 4.1. *Consider the linear DAE in Hessenberg form of size 3*

$$x_1' + B_{1,1}x_1 + B_{1,2}x_2 + B_{1,3}x_3 = f_1, \tag{4.1}$$

$$x_2' + B_{2,1}x_1 + B_{2,2}x_2 = f_2, \tag{4.2}$$

$$B_{3,2}x_3 = f_3. \tag{4.3}$$

where $C = B_{3,2}B_{2,1}B_{1,3}$ is non-singular.
If we rewrite (*) as follows

$$B_{1,3}x_3 = f_1 - x_1' - B_{1,1}x_1 - B_{1,2}x_2, \tag{4.4}$$

$$B_{2,1}x_1 = f_2 - x_2' - B_{2,2}x_2, \tag{4.5}$$

$$B_{3,2}x_3 = f_3. \tag{4.6}$$

then, in order to determine the P_i and Q_i matrices that transform the coefficients of the unknowns on the left hand side to C, it can be easily seen that we should have

Example 4.2. *Consider the non-linear following DAE*

$$A(t)x' + B(t)x = f(t), \qquad t \neq -1 \tag{4.7}$$

Although, it may be difficult to verify that the given system is in generalized Hessenberg form, but if we rewrite it as *and by using this fact, that the product matrix*

$$C = B_{3,2}B_{2,1}B_{1,3} = 3(t+1)^3$$

*is non-singular for all $t \neq 1$, then it can be easily seen that (*) is in generalized Hessenberg form of size 3. Now, the same as Example 1, by applying the propsed algorithm, after computing $Q-i$ and P_i matrices, for $i = 1,2,3$ from *-* and *-* respectively, we have*

$$A(t) = \begin{bmatrix} I_{2\times 2} & 0 & 0 \\ 0 & I_{3\times 3} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$B(t) = \begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} t+1 \\ 0 \end{bmatrix} \\ \begin{bmatrix} t+1 & t+2 \\ 2t+2 & 2t+4 \\ t+1 & 1 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \end{bmatrix} & \begin{bmatrix} t+1 & t+1 & 0 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \end{bmatrix},$$

$$f(t) = \begin{bmatrix} \begin{bmatrix} t^6+t^5+t+1 \\ 1 \end{bmatrix} \\ \begin{bmatrix} t^2+6t+3 \\ 5t^2+8t+6 \\ 4t^3+2t+2 \end{bmatrix} \\ \begin{bmatrix} t^4+2t^3+t^2+2t+2 \end{bmatrix} \end{bmatrix}.$$

Example 4.3. *Consider a linear (or linearized) semi-explicit DAEs*

$$\begin{cases} x^{(m)} = \sum_{j=1}^{m} A_j(t)x^{(j-1)} + B(t)y + q(t) \\ 0 = C(t)x + r(t), \end{cases} \tag{4.8}$$

where $A_j(t) \in \mathbb{R}^{n\times n}$ for $j = 1,2,...,m$, $B_j(t) \in \mathbb{R}^{n\times k}$, $C_j(t) \in \mathbb{R}^{k\times n}$, $q_j(t) \in \mathbb{R}^n$, $r_j(t) \in \mathbb{R}^k$, for $n \geq 2$ and $1 \leq k \leq n$, are smooth real valued functions of t, for $t_0 \leq t \leq t_f$. Also, $E(t) = C(t)B(t)$ is non-singular for all t in interested interval.

After Example 1 and 2, it is obvious that the problem is in generalized Hessenberg form. Numerical solutions of this problem, according to its importance, that we will discuss it later in this example, have been considered in several papers. For example, Direct Method by Using the Operational Matrices of Chebyshev Cardinal Functions [13], Homotopy Perturbation Method [27], Adomian Decomposition Method [14], Sinc-Collocation Method [28], Reducing Index Method [16], Differential Quadrature Method[26], Numerical Tau Method with Schauder Bases [25], Predicted Sequential Regularization Method [21], Projected Collocation Method [2], Pseudo-Spectral Method [24] and etc, but they are all numerical.

In order to solve the problem with our proposed method, let $x = B(t)u$ in second equation, so $E(t)u = -r(t)$ that implies $u = -E^{-1}(t)r(t)$, and consequently

$$x = -B(t)E^{-1}(t)r(t). \tag{4.9}$$

Now, by substituting the calculeted x in first equation, pre-multiplying in $C(t)$ and then in $E^{-1}(t)$, we have

$$y = E^{-1}(t)C(t)(x^{(m)} - \sum_{j=1}^{m} A_j(t)x^{(j-1)} - q(t)). \tag{4.10}$$

Now, consider the special case of (32) for $m = 1$, that is

$$\begin{cases} x' = A(t)x + B(t)y + q(t) \\ 0 = C(t)x + r(t), \end{cases} \tag{4.11}$$

The correctness of our explicit representation of the solutions for this special case, only for y, has been proved, for $n = 2$ and $k = 1$, in [3] and, for $n = 3$ and $k = 2$, in [15]. As we mentioned in the begining of this section, one of the most importatnt applications of DAEs is in solving discretized (or semi-discretized) PDEs that achieved by using method of line (MOL). For example, the incompressible Navier-Stokes equations can be formulated as (4.11) by semi-discretization in space [29].

5. Conclusions

In this paper, we have proposed a new method, with different approach and economical computing, that presents explicit formulas for the exact solutions of a large class of DAEs in Hessenberg forms. Frist, we have illustrated the method for linear time-varying DAEs in Hessenberg forms, in order to show the different approach and also the advantages of the method in computing, that make it economical. Then, we have described that the method is efficient for larger classes including special case of non-linear DAEs in Hessenberg forms.

References

[1] P. Amodio and F. Mazzia, ”Boundary value methods for solution of DAEs”, *Numer. Math.*, vol. 66, pp. 411-421, 1994.

[2] U. M. Ascher, L. R. Petzold, ”Projected collocation for higher-index differential-algebraic equations, *Journal of Computational and Applied Mathematics*, vol. 43, pp. 243-259, 1992.

[3] E. Babolian and M. M. Hosseini, ”Reducing index, and pseudospectral methods for differential-algebraic equations”, *Appl. Math. Comput.*, vol. 140, pp. 77-90, 2003.

[4] K. E. Brenan, S. L. Campbell and L. R. Petzold, ”Numerical solution of initial-value problems in differential-algebraic equations”, *SIAM Classics in Appl. Math., SIAM, Philadelphia*, 2nd ed, 1996.

[5] S. L. Campbell, ”One canonical form for higher-index linear time-varying singular systems”, *Circuits Systems Signal Process*, vol. 2, no. 3, pp. 311-326, 1983.

[6] S. L. Campbell, ”A general form for solvable linear time varying singular systems of differential equations”, *SIAM J. Math. Anal.*, vol. 18, pp. 1101-1115, 1987.

[7] S. L. Campbell, ”Singular Systems of Differential Equations”, *Pitman, Boston*, 1980.

[8] S. L. Campbell, ”Singular Systems of Differential Equations II”, *Pitman, Boston*, 1982.

[9] S. L. Campbell and R.L. Petzold, ”Canonical forms and solvable singular systems of differential equations”, *SIAM J. Algebr. Discrete Methods*, vol. 4, pp. 517521, 1983.

[10] K. D. Clark, ”A structural form for higher index semistate equations I: Theory and applications to circuit and control”, *Linear Algebra Appl.*, vol. 98, pp. 169-197, 1988.

[11] C. W. Gear and L. R. Petzold, ”ODE methods for the solution of differential-algebraic systems, *SIAM J. Numer. Anal.*, vol. 21 , pp. 716-728, 1984.

[12] E. Hairer, C. Lubich and M. Roche, ”The Numerical Solution of DA Systems by Runge-Kutta”, *Springer-Verlag, Berlin*, 1989.

[13] M. Heydari, G. Barid Loghmani, S. M. Hosseini and S. M. Karbassi, ”Direct Method to Solve Differential-Algebraic Equations by Using the Operational Matrices of Chebyshev Cardinal Functions”, *Journal of Mathematical Extension*, vol. 7, No. 2, pp. 25-47, 2013.

[14] M. M. Hosseini, ”Adomian decomposition method for solution of differential algebraic equations”, *J. Comput. Appl. Math.*, vol. 197 , pp. 495-501, 2006.

[15] M. M. Hosseini, ”An index reduction method for linear Hessenberg systems”, *J. Appl. Math. Comput.*, vol. 171, pp. 596-603, 2005.

[16] M. M. Hosseini, ”Reducing Index Method for Differential Algebraic Equations”, Proceedings of the 7th WSEAS International Conference on Applied Mathematics, 27,

[17] A. Ilchman and T. Reis, ” Survey in Diferential-Algebraic Equations I”, *Springer*, 2014.

[18] P. Kunkel and V. Mehrmann, ”Diferential-Algebraic Equations:Analysis and Numerical Solution”, *EMSPublishing House, Zurich, Switzerland*, 2006.
[19] P. Kunkel and V. Mehrmann, ”Stability properties of diferential-algebraic equations and spin-stabilized discretizations”, *Electr. Trans. on Numerical Analysis*, vol. 26, pp. 385-420, 2007.
[20] R. Lamour, R. Mrz and C. Tischendorf, ”Differential Algebraic Equations: A Projector Based Analysis”, *Differential-Algebraic Equations Forum*, vol. 1, Springer, Heidelberg, 2013.
[21] P. Lin and R. J. Spiteri, ”A PREDICTED SEQUENTIAL REGULARIZATION METHOD FOR INDEX-2 HESSENBERG DAES”, *SIAM J. NUMER. ANAL.*, vol. 39, No. 6, pp. 18891913, 2002.
[22] N. H. McClamroch, ”Singular systems of differential equations as dynamic models for constrained robot systems”, *Technical Report RSD- TR-2-86, Univ. of Michigan Robot Systems Division*, 1986.
[23] P. J. Rabier and W. C. Rheinboldt, ”Theoretical and numerical analysis of differential-algebraic equations”, *Handbook of Numerical Analysis, Elsevier, Amsterdam*, vol. VIII, pp. 183-537, 2002.
[24] M. Saravi, E. Babolian, R. England and M. Bromilow, ”System of linear ordinary differential and differential-algebraic equations and pseudo-spectral method”, *Comput. Math. Appl.*, vol.59, pp. 1524-1531, 2010.
[25] M. Shahrezaee, M. Ramezani, L. H. Kashany and H. Kharazi, ”Numerical Tau method for solving DAEs in Banach spaces with Schauder bases, *J. Appl. Math. Bio.*, vol. 2 , pp.99-114, 2012.
[26] M. Ramezani, M. Shahrezaee, H. and L. H. Kashany, ”Numerical solutions of Differential Algebraic Equations by Differential Quadrature Method”, *J. Basic. Appl. Sci. Res.*, vol. 2, No. 11, pp. 11821-11828, 2012.
[27] F. Soltanian, M. Dehghan and S. M. Karbassi, ”Solution of the differential algebraic equations via homotopy perturbation method and their engineering applications”, *International J. Comput. Math.*, vol. 1, pp. 1-25, 2009.
[28] S. Yeganeh, A. Saadatmandi, F. Soltanian and M. Dehghan,”The numerical solution of differential-algebraic equations by sinc-collocation method”, *Comp. Appl. Math.*, 2013.
[29] Z. Zheng, L. Petzold, ”A Framework for the Analysis of Second Order Projection Methods”, doi=10.1.1.81.7260