J. Basic. Appl. Sci. Res., **2(11)11821-11828, 2012**
© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Numerical solutions of Differential Algebraic Equations by Differential Quadrature Method

Mehdi Ramezani[1], Mohesen Shahrezaee[2], Hassan Kharazi[3], Leily Heidarzadeh Kashany[4]

[1] Faculty of Mathematics, Tafresh University, Tafresh, Iran
[2,3] Department of Mathematics, Emam Hossein University, Tehran, Iran
[4]Faculty of Mathematics, Tafresh University, Tafresh, Iran

ABSTRACT

Differential Algebraic Equations (DAEs) appear in many fields of physics and have a wide range of applications in various branches of science and engineering. Finding reliable methods to solve DAEs has been the subject of many investigations in recent years. In this paper, numerical solution of DAEs considered that after reduction index using Differential Quadrature Method (DQM). The scheme is tested for some high-index DAEs and the results demonstrate that the method is very straightforward and can be considered as a powerful mathematical tool.

KEY WORDS: Differential algebraic equation, Reduction index, Schauder Bases, Tau method, Hessenberg forms, Differential quadrature method, DAE, DQM, Least Square method.

1 INTRODUCTION

Many physical problems are naturally described by a system of differential algebraic equations. These type of systems occur in the modelling of electrical networks, flow of incompressible fluids, optimal control, mechanical systems subject to constraints, power systems, chemical process simulation, computer-aided design and in many other applications. Finding new methods for solving DAEs has become an interesting task for mathematicians. To solve differential algebraic equations (DAEs), some numerical methods have been developed, using both BDF [5, 10, 11, 12] and implicit Runge Kutta methods [1, 4, 8]. These methods are only directly suitable for low index problems and often require that the problem to have special structure. Although many important applications can be solved by these methods there is a need for more general approaches.

Many researches also made important contributions to this method. In the DQM, derivatives of the unknown functions in the differential equations with respect to a coordinate direction are expressed as a linear weighted sums of all functional values at all grid points along that direction. The main idea of the method is to find the weighting coefficients using test functions whose functional values and derivative values at discrete points in the whole domain are known. Many authors have obtained weighting coefficients implicitly or explicitly using various test functions such as Legendre polynomials, Lagrange interpolation polynomials, spline functions, radial basis functions, harmonic functions with [11], etc. The DQM is an efficient discretization technique for obtaining accurate numerical solutions using small number of grid points. In this study, we apply cosine expansion based differential quadrature method (CDQ) defined in [6] for discretization to obtain fully discretized form of the DAEs, which are the system of equations.

2 DAEs and reduction of index

A system of DAEs is one that consists of ordinary differential equations (ODEs) coupled with purely algebraic equations, on the other hand, DAEs are everywhere singular implicit ODEs. The general form of DAEs is

$$F(x(t), x'(t), t) = 0, \quad F \in C^1(\mathbb{R}^{2m+1}, \mathbb{R}^m), \quad t \in [0, T], \tag{1}$$

where $\partial F/\partial x'$ is singular on $\mathbb{R}^{2m+1}$ [14]. Most DAEs arising in applications are in semi-explicit form and many are in the further restricted Hessenberg form [8]. The index-1 semi explicit DAEs is given by:

***Corresponding Author:** Mehdi Ramezani, Faculty of Mathematics, Tafresh University, Tafresh, Iran, ramezani@aut.ac.ir

$$\begin{cases} x'(t) = f(x(t), y(t), t), & f \in C(\mathbb{R}^{m+k+1}, \mathbb{R}^m), \\ 0 = g(x(t), y(t), t), & g \in C^1(\mathbb{R}^{m+k+1}, \mathbb{R}^k), \end{cases} \qquad t \in [0, T], \tag{2}$$

where $\partial g/\partial y$ is non-singular.

In such systems the Algebraic and differential variables are explicitly identified for higher-index DAEs as well, and the algebraic variables may all be eliminated using the same number of differentiations. These are called Hessenberg forms of the DAE and a re-given below.

2.1 Hessenberg index- ν

The index-2 Hessenberg DAEs is given by:

$$\begin{cases} x'(t) = f(x(t), y(t), t), & f \in C^1(\mathbb{R}^{m+k+1}, \mathbb{R}^m), \\ 0 = g(x(t), t), & g \in C^2(\mathbb{R}^{m+1}, \mathbb{R}^k), \end{cases} \qquad t \in [0, T], \tag{3}$$

where $(\partial g/\partial x)(\partial f/\partial y)$ is non-singular [14]. DAEs in Hessenberg form of index ν have the form

$$\begin{cases} x_1' = f_1(x_1, \ldots, x_{\nu-1}, x_\nu) \\ \quad \vdots \\ x_{\nu-1}' = f_{\nu-1}(x_{\nu-2}, x_{\nu-1}) \\ \end{cases} \tag{4}$$

with

$$\frac{\partial f_\nu}{\partial x_{\nu-1}} \frac{\partial f_{\nu-1}}{\partial x_{\nu-2}} \cdots \frac{\partial f_2}{\partial x_1} \frac{\partial f_1}{\partial x_\nu} \ \textit{nonsingular} \tag{5}$$

for all relevant points $(x_1, \cdots, x_\nu) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_\nu}$.

2.2 Reducing index

We consider a linear (or linearized) model problem,

$$\begin{cases} X^{(m)} = \sum_{j=0}^{m} A_j X^{(j-1)} + By + q, \\ 0 = CX + r \end{cases} \tag{6}$$

where A_j, B and C are smooth functions of t, $0 \le t \le t_f$, $A_j(t) \in \mathbb{R}^{n \times n}$, $j = 1, \cdots, m$, $B(t) \in \mathbb{R}^n$, $C(t) \in \mathbb{R}^{n \times n}$, $n \ge 2$, and CB is nonsingular for each t (hence the DAE has index $m+1$). The inhomogeneities are $q(t) \in \mathbb{R}$ and $r(t) \in \mathbb{R}$. The DAE 6 will be transformed into an implicit DAE form by representing a simple formulation. For this reason, we put

$$y = (CB)^{-1} C \left(X^{(m)} - \sum_{j=1}^{m} A_j X^{(j-1)} - q \right) \tag{7}$$

and by substituting 7 in 6, we obtain an implicit DAE which has index m, as follows,

$$\sum_{j=0}^{m} E_j X^{(j)} = \hat{q} \tag{8}$$

where $E_j(t) \in \mathrm{R}^{n\times n}$, $j = 1,\ldots,m$, and except $E_0(t)$, others are singular matrices.

Theorem 1: Consider DAEs (Equation 6), when it has index-$2, n = 2$ and $k = 1$. This problem is equivalent to the following index-1 DAE system:

$$E_1 X' + E_0 X = \hat{q}, \tag{9}$$

such that

$$E_0 = \begin{pmatrix} b_1a_{21} - b_2a_{11} & b_1a_{22} - b_2a_{12} \\ c_1 & c_2 \end{pmatrix}, \qquad E_1 = \begin{pmatrix} b_2 & -b_1 \\ 0 & 0 \end{pmatrix},$$
$$\hat{q} = \begin{pmatrix} b_2q_1 - b_1q_2 \\ 0 \end{pmatrix},$$

and

$$y = (CB)^{-1} C[X' - AX - q]. \tag{10}$$

The proof of this theorem is presented in Ref.[5].

Theorem 2: Consider DAEs (Equation 6) with index-$2, n = 3$ and $k = 2$. This problem is equivalent to the following index-1 DAE system:

$$\begin{pmatrix} M \\ 0 \end{pmatrix} X' + \begin{pmatrix} -MA \\ C \end{pmatrix} X = \begin{pmatrix} Mq \\ -r \end{pmatrix} \tag{11}$$

such that

$$M = \left(b_{21}b_{32} - b_{22}b_{31} \quad b_{12}b_{31} - b_{11}b_{32} \quad b_{11}b_{22} - b_{12}b_{21}\right)_{1\times 3}$$

and

$$y = (CB)^{-1} C[X' - AX - q]. \tag{12}$$

The proof is presented in Ref.[15].

By theorems 1 and 2 , the above system can be transformed to the following full rank DAE system, with n equations and n unknowns,

$$\begin{cases} \overline{M}[X^{(m)} - \sum_{j=1}^{m} A_j X^{(j-1)} - q] = 0, \\ CX + r = 0, \end{cases}$$

i.e.,

$$\begin{pmatrix} \overline{M} \\ 0 \end{pmatrix} X^{(m)} + \begin{pmatrix} -\overline{M}A_m \\ 0 \end{pmatrix} X^{(m-1)} + \cdots + \begin{pmatrix} -\overline{M}A_2 \\ 0 \end{pmatrix} X' + \begin{pmatrix} -\overline{M}A_1 \\ C \end{pmatrix} X = \begin{pmatrix} \overline{M}q \\ -r \end{pmatrix},$$

or

$$E_m X^{(m)} + E_{m-1} X^{(m-1)} + \cdots + E_1 X' + E_0 X = \hat{q}. \tag{13}$$

Index of 13 is equal to m .

3 Differential quadrature method

Differential quadrature method (DQM) is a numerical method for evaluating derivatives of a sufficiently smooth function, proposed by Bellman and Casti in 1971. In other words, the derivatives of a smooth function are approximated with weighted sums of the function values at a group of so-called nodes. Suppose function $x(t)$ is sufficiently smooth on the interval $[a,b]$. On the interval, N distinct nodes are defined:

$$a = t_1 < t_2 < \cdots < t_N = b \tag{14}$$

The function values on these nodes are assumed to be

$$x(1), x(2), \cdots, x(N) \tag{15}$$

Based on DQM, the first and second order derivatives on each of these nodes are given by

$$\frac{dx(t_i)}{dt} \approx \sum_{j=1}^{N} a_{ij} x(j) = a_i^T . x_N, \quad i = 1,2,\cdots,N \tag{16}$$

$$\frac{d^2 x(t_i)}{dt^2} \approx \sum_{j=1}^{N} b_{ij} x(j) = b_i^T . x_N, \quad i = 1,2,\cdots,N \tag{17}$$

The coefficients a_{ij} and b_{ij} are the weighting coefficients of the first and second order derivatives with respect to t, respectively. Using the Lagrange interpolating functions, Shu and Richards [6] gave a convenient and recurrent formula for determining the derivative weighting coefficients as follows:

$$\begin{aligned} a_{ij} &= \frac{M(t_i)}{(t_i - t_j).M(t_j)} \qquad && i \neq j, i, j = 1,2,\cdots,N \\ a_{ii} &= -\sum_{j=1,\, i\neq j}^{N} a_{ij} && i = 1,2,\cdots,N \end{aligned} \tag{18}$$

also for second derivative weighting coefficients we have

$$\begin{aligned} b_{ij} &= 2[a_{ii}.a_{ij} - \frac{a_{ij}}{(t_j - t_i)}] \qquad && i \neq j, i, j = 1,2,\cdots,N \\ b_{ii} &= -\sum_{j=1,\, i\neq j}^{N} b_{ij} && i = 1,2,\cdots,N \end{aligned} \tag{19}$$

Where

$$M(t_i) = \Pi_{j=1,\, i\neq j}^{N} (t_i - t_j).$$

When above equations is used, there is no restriction on the choice of the nodes [6].

Similarly we may obtain formulas for higher order derivatives by using the higher order weighting coefficients, which are expressed as $e_{ij}^{(m)}$ to avoid confusion. They are characterized by recurrence [7]

$$e_{ij}^{(m)} = m[a_{ij}.e_{ii}^{(m-1)} - \frac{e_{ij}^{(m-1)}}{t_j - t_i}], \quad i, j = 1,2,\cdots,N, i \neq j, m = 2,3,\cdots,N-1$$
$$e_{ii}^{(m)} = -\sum_{j=1,\ j\neq i}^{N} e_{ij}^{(m)}, \qquad i = 1,2,\cdots,N$$

Where $a_{ij} = e_{ij}^{(1)}$ and $b_{ij} = e_{ij}^{(2)}$.

4 Formulating the problem

Consider 12 and 13, now by theorem 1 , 2 and employing DQM we have

$$E_m e_i^{(m)T} X_N + E_{m-1} e_i^{(m-1)T} X_N + \cdots + E_1 e_i^{(1)T} X_N + E_0 X_N = \hat{q}_N. \tag{20}$$

and

$$y = (CB)^{-1} C\left(e_i^{(m)T} X_N - \sum_{j=1}^{m} A_j e_i^{(j-1)T} X_N - q \right) \tag{21}$$

That with $m = 1$ we will have

$$E_1 a_i^T X_N + E_0 X_N = \hat{q}_N \tag{22}$$

and

$$y = (CB)^{-1} C\left(a_i^T X_N - AX_N - q_N\right) \tag{23}$$

After implimentation initial conditions 22 can be expressed in the following vector matrix form

$$Ax = b$$

Where in this equation A is an $(2N)\times(2N-2)$ matrix of known constants, x is an $(2N-2)\times 1$ vector of unknown values and b is an $(2N)\times 1$ vector of known values that this equation can be solve by Least Square method.

5 Test problems

The above result allows us to calculate some numerical solutions of differential equations. we present some numerical results to demonstrate the efficiency of DQM for solving the DAEs, also DQM comparisons with Schauder Bases [9] and Pseudospctral methods [3]. These examples are chosen such that their exact solutions are known.The numerical computations have been done by the software Matlab edition 2011 .

Example 1: Consider the linear index-2 semi-explicit DAEs problem:

$$\begin{aligned} X' &= AX + By + q, \\ 0 &= CX + r, \end{aligned} \tag{24}$$

where

$$A = \begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 \\ 1+2t \end{pmatrix}, \qquad q = \begin{pmatrix} -\sin(t) \\ 0 \end{pmatrix}, \qquad C^T = \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

and $r(t) = -(e^{-t} + sin(t))$ with $x_1(0) = 1$ and $x_2(0) = 0$ .The exact solutions of this problem are:

$$x_1(t) = e^{-t}, \qquad x_2(t) = \sin(t), \qquad y(t) = \frac{\cos(t)}{1+2t}.$$

From Theorem 1, the index-2 DAEs (Equation (24)) can be converted to the index-1 DAEs:

$$\begin{cases} x_2 = -x_1 + e^{-t} + \sin(t), \\ x_1' = x_2 - x_1 - \sin(t), \end{cases}$$

with $x_1(0) = 1$ and $x_2(0) = 0$.

Employing DQM 16 for this system equations, obtain the following system algebraic equations

$$\begin{cases} x_2(i) + x_1(i) = e^{t_i} + \sin t_i \\ \sum_{j=1}^{N} a_{ij} x_1(j) - x_2(i) + x_1(i) = -\sin t_i \end{cases} \qquad i = 1, \cdots, N$$

Where $a_{ij}, (i, j = 1, \cdots, N)$ are weighting coefficents and $x_1(i), x_2(i)(i = 1, \cdots, N)$ are values x_1, x_2 in nodes $t_1 = 0 \le t_2 \le \cdots \le 1 = t_N$.

After implimentation initial conditions last system can be expressed in the following vector matrix form

$$Ax = b$$

Where in this equation A is an $(2N) \times (2N-2)$ matrix of known constants, x is an $(2N-2) \times 1$ vector of unknown values and b is an $(2N) \times 1$ vector of known values.

Table 1: Numerical solution of $x_1(t)$ and $x_2(t)$ and $y(t)$ by Schauder Bases Tau method and DQM

	Schauder Bases Tau method		DQM	
N	e_x	e_y	e_x	e_y
5	$2.0E-1$	$2.1E-1$	$2.0E-5$	$2.4E-4$
10	$2.4E-2$	$1.0E-3$	$4.1E-12$	$1.1E-10$
20	$1.7E-4$	$2.2E-4$	$6.6E-14$	$3.1E-15$

The obtained approxtimate values by DQM and Schauder Bases Tau method, relative errors are reported in Table 1 .
In this example, we use "e_x" and "e_y" to denote the maximum over all components of the errors in x and y.

Example 2: Consider the DAE is in pure index-2, for $0 \le t \le 1$,

$$\begin{cases} x_1'(t) = 10(t-2)y - 9e^t \\ x_2'(t) = 9y + \left(\dfrac{11-t}{2-t}\right)e^t \\ (t+2)x_1 + (t^2-4)x_2 - (t^2+t-2)e^t = 0 \end{cases} \tag{25}$$

From the initial conditions $x_1(0) = 1$, $x_2(0) = 1$ we have the exact solution

$$x_1(t) = e^t, \quad x_2(t) = e^t, \quad y(t) = -\frac{e^t}{(2-t)}$$

By using theorem 2 we have:

$$\begin{cases} 9x_1'(t) + 10(2-t)x_2'(t) = (29-10t)e^t \\ (t+2)x_1 + (t^2-4)x_2 = (t^2+t-2)e^t \end{cases} \tag{26}$$

and

$$y(t) = \frac{1}{19(t-2)} x_1^{'}(t) + \frac{1}{19} x_2^{'}(t) + \frac{20-t}{19(t-2)} e^t \tag{27}$$

Employing DQM 16 for system equations 26 obtain the system algebraic equations

$$\begin{cases} 9\sum_{j=1}^{N} a_{ij} x_1(j) + 10(2-t_i)\sum_{j=1}^{N} a_{ij} x_2(j) = (29-10t_i)e^{t_i} \\ (t_i+2)x_1(i) + (t_i^2-4)x_2(i) = (t_i^2+t_i-2)e^{t_i} \qquad i = 1,\cdots,N \end{cases}$$

After implimentation initial conditions last system can be expressed in the $Ax = b$. This problem is solved using DQM and is Compared with Pseudospectral method in Table 2.

Table 2: Numerical solution of $x_1(t)$ and $x_2(t)$ and $y(t)$ by Pseudospectral and DQM

	Pseudospectral		**DQM**	
N	e_x	e_y	e_x	e_y
5	$7.0E-6$	$7.3E-6$	$2.1E-5$	$1.2E-5$
10	$1.3E-11$	$4.0E-10$	$2.1E-10$	$1.2E-10$
15	$2.0E-13$	$2.2E-12$	$1.3E-12$	$1.1E-11$

6 Conclusion

We have presented a numerical method that allows some DAEs to be solved with a low computational cost. The use of the DQM provided for the high accuracy to the exact solution. Through example which has exact solution, it was found that in order to obtain accurate numerical results demonstrate DQM is efficient.

REFERENCES

[1] L. Jay: Specialized RungeKutta methods for index 2 differential-algebraic equations, Math. Comput. 75 (2005), pp. 641- 654.

[2] J. Huang, J. Jia, and M. Minion: Arbitrary order Krylov deffered correction methods for differential algebraic equations, J. Comput. Phys. 221 (2007), pp. 739 -760.

[3] F. Soltanian, S.M. Karbassi, M.M. Hosseini: Application of Hes variational iteration method for solution of differential-algebraic equations, Chaos, Solitons Fractals 41(1) (2009), pp. 436-445.

[4] U.M. Ascher and L.R. Petzold: Projected implicit RungeKutta methods for differential-algebraic equations, SIAM J. Numerical Analysis 28 (1991), pp. 1097-1120.

[5] U.M. Ascher,L.R. Petzold: Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations, Philadelphia, PA, 1998.

[6] C. Shu: Differential Quadrature Method and its Application in Engineering, Springer, Brlin, 2000.

[7] Zhi Zong, Yingyan Zhang: Advanced Differential Quadrature Methods, CRC Press, Boca Raton ,2009.

[8] K. E. Brenan, S. L. Campbell, L R. Petzold: Numerical solution of Initial Value Problems in Differential

Algebraic, Siam, New York, 1989.

[9] M.Shahrezaee, M. Ramezani, L.H.Kashani, H.Kharazi: Numerical Tau Method for Solving DAEs in Banach Spaces with Schauder Bases. 2 (2012) 99-114.

[10] U.M. Ascher: On symmetric schemes and differential-algebraic equations, SIAM J. Sci. Stat. Comput. 10 (1989) 937-949.

[11] C.W. Gear, L.R. Petzold: ODE systems for the solution of differential-algebraic systems, SIAM J. Numer. Anal. 21 (1984) 716-728.

[12] U.M. Ascher, L.R. Petzold: Stabilityof computational methods for constrained dynamics systems, SIAM J. Sci. Comput. 14 (1993) 95-120.

[13] E. Celik,E. Karaduman,M. Bayram: A numerical method to solve chemical differential algebraic equations,Int. J. Quant. Chem. 89 (2002) 447-451.

[14] H. Wang and Y. Song: Regularization methods for solving differential-algebraic equations, Appl. Math. Comput. 119 (2001), pp. 283-296.

[15] E. Babolian and M.M. Hosseini : Reducing index, and pseudospectral methods for differential-algebraic equations, Appl. Math. Comput. 140 (2003), pp. 77-90.