

Passage to the Differences for the BDF's Methods

Abdelouahab Salmi

Department of Mathematics, Badji Mokhtar University, Annaba, Algeria

Abstract: In this study, we give an improvement to the Backward Differentiation Formula (BDF). We use a formula for the differences that permits us to determine the new coefficients $\hat{\alpha}_i$ and $\hat{\gamma}_i$ in function of the classical coefficients α_i and γ_i of this method. Our proposed approach decreases the error and the cost of computation. Numerical results for concrete problems are given to support the theory.

Key words: Stiff systems, backward differentiation formula, differential-algebraic equations

INTRODUCTION

We are interested in initial value problems for the Differential-Algebraic Equation (DAE) of the form:

$$F(x, x', t) = 0$$

Where F , x and x' are s -dimensional vectors, F is assumed to be differentiable. We investigate some of the practical difficulties involved with implementing the BDF, for the DAE of the form,

$$0 = F_1(x, x', y, t), \quad 0 = F_2(x, y, t) \quad (1)$$

and the implicit differential equation of the form:

$$F(x, x', t) = 0 \quad (2)$$

Where the initial values of x and y are given, $\frac{\partial F}{\partial x'}$ is non-singular. The function F is assumed to be sufficiently smooth. The Jacobian matrix may be singular. As a special case, if $\frac{\partial F}{\partial x'}$ is non-singular,

equation (2) is locally a system of ODE's. However, if the Jacobian is singular, equation (2) is in fact a system of DAE's. In such a system, there are algebraic constraints on the variables. Systems of this form arise frequently in the modelling of engineering problems, for example the simulation of electrical networks, the solution of certain equations in fluid dynamics and the simulation of mechanical systems^[1-3]. Other areas of application are: control theory^[2], power systems^[3] and heat flow^[2].

For the stiff problems^[4,5] and differential-algebraic equations, our approach are based on BDF's^[5,6]:

$$\rho x_{n+1} = x'_{n+1} = -\frac{1}{h} \sum_{i=0}^k \alpha_i x_{n+1-i}, \quad k = \overline{1,6} \quad (3)$$

$$x_{n+1}^p = \sum_{i=1}^{k+1} \gamma_i x_{n+1-i}, \quad k = \overline{1,6} \quad (4)$$

Where $h = t_{n+1} - t_n$ and the coefficients α_i and γ_i depend on the stepsize h and the order k .

This type of methods was studied in^[1,7]. The contribution that we bring here to the BDF's permits the decrease of the global error as well as the explicit dependence of the coefficients α_i and γ_i on the stepsize h and the order k ^[5,6].

The basic idea of using BDF's for solving DAE systems was introduced by Gear^[2] and consists of replacing x' in (1) by a difference approximation and then solving the resulting equations for approximations to x and y . The simplest example of numerical ODE method for (1) is the backward Euler method. Using this approach, the derivative $x'(t_{n+1})$ at time t_{n+1} is approximated by a backward difference of $x(t)$ and the resulting system of nonlinear equations for x_{n+1} and y_{n+1} is,

$$0 = F_1\left(x_{n+1}, \frac{x_{n+1} - x_n}{t_{n+1} - t_n}, y_{n+1}, t_{n+1}\right),$$

$$0 = F_2(x_{n+1}, y_{n+1}, t_{n+1}) \quad (5)$$

The above system has a unique solution^[2] if the inverse of scaled Jacobian matrix:

$$hJ_n = \begin{pmatrix} \alpha_0 \frac{\partial F_1}{\partial x'} + h \frac{\partial F_1}{\partial x} & h \frac{\partial F_1}{\partial y} \\ h \frac{\partial F_2}{\partial x} & h \frac{\partial F_2}{\partial y} \end{pmatrix} \text{ exists.}$$

In an earlier paper^[3,7], it is showed that for the systems under considerations, assumptions must be placed on (1). These assumptions are satisfied in many applications. The k -step constant stepsize BDF converges to order of accuracy $O(h^k)$, where the

initial values are consistent and the functions are sufficiently smooth. Here we are concerned with the practical difficulties such as varying the stepsize and dealing with ill conditioned matrices which arise in implementing BDF methods for the solution of (1). An important characteristic for understanding both the properties of solutions to DAE's systems and the behaviour of numerical methods for solving these systems is the index of the system, for which a precise definition is given later in the next section. In order to better understand the index and its role in the structure and solution of DAE's systems, we write:

Assumption (1.1):

1. The index is less than or equal to one, or
2. The index is less than or equal to two and $\frac{\partial F_2}{\partial y} \equiv 0$, or
3. The index is less than or equal to three and the system has the form (1).

It is well known that the index of DAE's is a measure of the degree of singularity of the system and also widely regarded as an indication of certain difficulties for numerical methods. Thus DAE's can be difficult to solve when they have a higher index, i.e. an index greater than one^[1,2]. In this case an alternative treatment is the use of index reduction methods^[3,7].

Index and mathematical structure for DAE's: Since a DAE involves a mixture of differentiations and integrations, one may hope that applying analytical differentiations to a given system and eliminating as needed, repeatedly if necessary, will yield an explicit ODE system for all the unknowns. This section presents definitions of DAE's in various forms and introduces the notion of the index.

Definition 2.1: A system of initial value differential-algebraic equation can be written in its general nonlinear implicit or fully implicit form as in (6a), subject to initial conditions (6b),

$$F(x(t), x'(t), t) = 0 \tag{6a}$$

$$x(t_0) = x_0, x'(t_0) = x'_0 \tag{6b}$$

Where $F \in \mathbb{R}^n, x \in \mathbb{R}^n$ and $t \in \mathbb{R}$, $\frac{\partial F}{\partial x} = F_x$ is non-singular and $\frac{\partial F}{\partial x'} = F_{x'}$ is singular for all $x(t), x'(t)$ and t .

The existence of algebraic constraints on the variables is expressed by the singularity of the Jacobian $\frac{\partial F}{\partial x'}$ matrix. The initial conditions for x and x' defined in (6b) must be consistent in order for (6a) to admit solution. It is this last property that separates DAE's from implicit ODE's. It means that there are algebraic constraints on $x(t)$ in addition to the usual differential

equations, in the other words, the components of $x(t)$ are not all independent.

As in the case with both analytical and numerical solutions of ODE's, it is important to classify DAE's (and especially nonlinear DAE's) into forms with known properties and /or methods of solution. In particular, not all DAE's can be solved numerically by classical methods such as BDF^[2,3] or implicit Runge Kutta schemes^[3]. Thus it is desirable to have structural forms which predict when a DAE will be solvable.

Definition 2.2: For general DAE systems (6), the index along a solution $x(t)$ is the minimum number of differentiation of the system which would be required to solve for x' uniquely in terms of x and t . (i.e. to define an ODE for x). Thus, the index is defined in terms of the over determined system:

$$\begin{aligned} F(t, x, x') &= 0 \\ \frac{dF}{dt}(t, x, x', x'') &= 0 \\ \dots\dots\dots &\dots\dots\dots \\ \frac{d^p F}{dt^p}(t, x, x', \dots, x^{(p+1)}) &= 0 \end{aligned} \tag{7}$$

to be the smallest integer p such that x' in (7) can be solved in terms of x and t .

Special DAE forms: The general DAE system (6) can include problems which are not well defined in mathematical sense, as well as problems which result in failure by any direct discretisation method (i.e. a method based on discretisation of x and x' without first reformulating the equations). Fortunately, most of the higher- index problems encountered in practice can be expressed as a combination of more restrictive structures of ODEs coupled with constraints. In such systems, the algebraic and the differential variables are explicitly identified for higher-index DAE's as well and the algebraic variables may all be eliminated (in principle) using the same number of differentiations. These are called Hessenberg forms of the DAE and are given below.

Hessenberg index-1

$$\begin{aligned} x' &= f(t, x, z) \\ 0 &= g(t, x, z) \end{aligned}$$

Here, the Jacobian matrix g_x is assumed to be non-singular for all t . This is also often referred to as a semi-explicit index-1 system.

Hessenberg index-2

$$\begin{aligned} x' &= f(t, x, z) \\ 0 &= g(t, x) \end{aligned}$$

Here, the product of Jacobians $g_x f_z$ is non-singular for all t . This is a pure index-2 DAE.

Hessenberg index-3

$$x' = f(t, x, y, z)$$

$$z' = g(t, x, z)$$

$$0 = h(t, y)$$

Here the product of three matrix functions $h_y g_x f_z$ is non-singular for all t .

The index has proven to be a useful concept for classifying DAEs in order to construct and identify appropriate numerical methods.

BDF methods: Numerical approaches for the solution of differential-algebraic equations can be divided roughly into two classes:

- i. Direct discretisations of the given system and (ii) methods which involve a reformulation (e.g. index reduction), combined with discretisation. The desire for as direct a discretisation as possible arises because a reformulation may be costly. It may require more input from the user and it may involve more user intervention. The reason for the popularity of reformulation approach is that, as it turns out, direct discretisations are essentially limited in their utility to index-1 and semi-explicit index-2 DAE systems.

Some numerical methods have been developed, using both BDF^[2,7] and implicit Runge kutta methods^[3]. These methods are only directly suitable for low index problems and often require that the problem has a special structure. Although many important applications can be solved by these methods, there is a need for more general approach as proposed in^[2,3].

In this paper, our approach is based on the methods of backward differentiation formulas (BDFs):

$$\rho x_{n+1} = x'_{n+1} = -\frac{1}{h} \sum_{i=0}^k \alpha_i x_{n+1-i}, k = \overline{1,6} \tag{8}$$

$$x_{n+1}^p = \sum_{i=1}^{k+1} \gamma_i x_{n+1-i}, k = \overline{1,6} \tag{9}$$

This type of methods has been studied by^[1,7]; the contribution that we bring to the BDF's, permits the decrease of the global error as well as the explicit dependence of the coefficients α_i and γ_i on the step h and the order k ^[5,6].

Computation of the coefficients α_i and γ_i :

The coefficients α_i depend on the values $t_{n-1}, t_n, \dots, t_{n+1-k}$ and are determined so that the formula (1) gives an exact result for $x(t) = p_i(t) = \left(\frac{t_{n+1}-t}{h}\right)^i$ ^[4,6].

While substituting the polynomials $p_i(t)$ in (8) and (9), one gets the following linear system:

$$V_1 \alpha = e_2 \text{ and } V_2 \gamma = e_1 \tag{10}$$

With $e_1 = (1, 0, \dots, 0)^T, e_2 = (0, 1, \dots, 0)^T$,

$\alpha = (\alpha_0, \alpha_1, \dots, \alpha_k)^T$ and $\gamma = (\gamma_1, \gamma_1, \dots, \gamma_{k+1})^T$.

The matrices V_1 and V_2 are of Vendermonde type:

$$V_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & x_2 & \dots & x_{k+1} \\ \dots & \dots & \dots & \dots \\ 0 & x_2^k & \dots & x_{k+1}^k \end{bmatrix} \text{ with}$$

$$x_i = \frac{t_{n+1} - t_{n+2-i}}{h}, i = \overline{1, k+1}$$

And

$$V_2 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_{k+1} \\ x_1^2 & x_2^2 & \dots & x_{k+1}^2 \\ \dots & \dots & \dots & \dots \\ x_1^k & x_2^k & \dots & x_{k+1}^k \end{bmatrix} \text{ with}$$

$$x_i = \frac{t_{n+1} - t_{n+1-i}}{h}, i = \overline{1, k+1}.$$

While taking into account the specific pace of these systems, we can find the coefficients with a reduced computation cost^[5,6].

Proposition 3.1^[6]: The coefficients γ_i and α_i are given by,

$$\gamma_i = \prod_{i \neq v=1}^{k+1} \frac{t_{n+1} - t_{n+1-v}}{t_{n+1-i} - t_{n+1-v}}, i = \overline{1, k+1} \tag{11}$$

$$\alpha_0 = -\sum_{v=1}^k \frac{t_{n+1} - t_n}{t_{n+1-i} - t_{n+1-v}} \tag{12}$$

and

$$\alpha_i = \frac{t_{n+1} - t_n}{t_{n+1} - t_{n+1-i}} \prod_{i \neq v=1}^k \frac{t_{n+1} - t_{n+1-v}}{t_{n+1-i} - t_{n+1-v}}, i = \overline{1, k}$$

The formulas (11) and (12) can be used for the computation of the coefficients for different steps and an order $1 \leq k \leq 6$ ^[5].

Passage to the differences: To decrease the values of rounded errors when using the formulas (8) and (9), we

replace the values x_{n+1-i} in (8) and (9) by the differences: $\Delta x_{n-i} = x_{n+1-i} - x_{n-i}$. In this case the formulas (8) and (9) take the following form:

$$x'_{n+1} = -\frac{1}{h} \sum_{i=0}^{k-1} \hat{\alpha}_i \Delta x_{n-i}, k = \overline{1,6} \tag{13}$$

$$x_{n+1}^p = x_n + \sum_{i=1}^k \hat{\gamma}_i \Delta x_{n-i}, k = \overline{1,6} \tag{14}$$

Table 1: Errors in x and y for example 4.1 using BDF with a difference.

BDF		BDF with difference		
E_{\max}^x	E_{\max}^y	E_{\max}^x	E_{\max}^y	E_{\max}^y
5.0000 E-07	7.1000 E-07	4.8990 E-08	7.0000 E-07	7.0000 E-07
3.9000 E-11	4.2000 E-11	3.8700 E-12	4.1100 E-12	4.1100 E-12
1.8000 E-15	1.8000 E-16	1.8200 E-16	1.8200 E-16	1.8200 E-16
4.6100 E-15	6.4000 E-17	4.5900 E-17	6.3700 E-17	6.3700 E-17

Table 2: Convergence of the BDF scheme with a difference for Stiff problem.

Tolerance		$\epsilon=10^{-3}$			
Method	BDF5	BDF6	BDF6	BDF6	BDF6
Values/Errors	E_1	E_2	E_1	E_2	E_2
t=2.0	7.5951 E-03	2.0934 E-03	5.3060 E-03	2.4746 E-05	2.4746 E-05
t=3.0	6.4092 E-03	7.1329 E-04	4.2336 E-03	7.8208 E-04	7.8208 E-04
t=6.0	6.6100 E-03	2.7391 E-04	5.0564 E-03	2.1270 E-03	2.1270 E-03
Tolerance		$\epsilon=10^{-4}$			
Method	BDF5	BDF6	BDF6	BDF6	BDF6
Values/Errors	E_1	E_2	E_1	E_2	E_2
t=2.0	1.4042 E-03	3.5832 E-04	3.5691 E-04	2.0970 E-04	2.0970 E-04
t=3.0	1.1559 E-03	3.1592 E-04	1.7552 E-04	6.7362 E-04	6.7362 E-04
t=6.0	3.4594 E-03	1.5085 E-05	2.9635 E-03	2.8997 E-04	2.8997 E-04

with $x_1(0) = 1$ and $q(t) = \begin{pmatrix} (v+1)e^t \\ (1+(v-1)/(2-t))e^t \end{pmatrix}$,

$$r(t) = -(t^2 + t - 2)e^t.$$

The exact solution is: $x_1 = x_2 = e^t$, $y = -e^t/(2-t)$; the numerical results with $v=1000$ is given in Table 1.

The comparison between the general results given in table 1 and the published results in^[7] and^[6] shows the advantage of using the proposed methods of this paper, for this example.

Example 4.2 (Stiff problem):

$$\begin{cases} x'_1 = -30x_1 + 29x_2 + 3 \\ x'_2 = 70x_1 - 70x_2, t \in [0,10] \\ \text{avec } x_1(0) = x_2(0) = 1 \end{cases}$$

In Table 2, we give the maximum over all components of the errors (E_1) and the truncature error (E_2) evaluated at $t=2.0$, $t=3.0$ and $t=6.0$. This example is solved, using BDF with the formulas (15) and (16).

For a tolerance $\epsilon \leq 10^{-3}$, BDF5 and BDF6 converge, but BDF6 has a faster convergence rate than BDF5. The comparison between the general results mentioned in Table 2 and the published results in^[6] shows the advantage of using the proposed methods of this paper, for this example.

CONCLUSION

The implementation of the numerical methods analyzed in this article shows that the results found by our approach of BDF's, as using the formula to the differences, is better than those of^[4] and^[6]; and can be improved more in maintaining the same order for the following steps and while using a good approximation

of the Jacobian matrix^[7]. This improvement can overcome the problem of instability that can be met for small tolerances.

ACKNOWLEDGEMENT

This work was supported in part by the accord program CMEP number 04 MDU 606.

REFERENCES

1. Brenan, K.E. and B.E. Enquist, 1988. Backward differentiation approximation of nonlinear differential algebraic equations and supplement S7-S17. *Math. Comp.*, 51: 659-676.
2. Brenan, K.E., S.L. Campbell and L.R. Petzold, 1995. *Numerical Solution Of Initial-Value Problems in Differential-Algebraic Equations*. 2nd Edn., SIAM, Philadelphia, PA.
3. Hairer, E. and G. Wanner, 1991. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer-Verlag.
4. Enright, W.H., T.E. Hull and B. Lendberg, 1975. Comparing numerical methods for stiff systems of Ode's. *BIT* 15: 10-48.
5. Salmi, A., 1995. Study of some numerical methods for the stiff problems. Magister Thesis. University of Annaba.
6. Nafaa K. and A. Salmi, 1996. Contributions aux méthodes de différentiation en arrière pour les problèmes rigides. *Maghreb Math. Rev.*, 5: 129-149.
7. Lotstedt, P. and L.R. Petzold, 1986. Numerical solution of nonlinear differential equations with algebraic constraints I: Convergence results for backward differential formulas. *Math. Comp.*, 46: 491-516.