

## A Chebyshev Collocation Method for Stiff Initial Value Problems and Its Stability

SANGDONG KIM, JONGKYUM KWON, XIANGFAN PIAO AND PHILSU KIM<sup>\*†</sup>  
*Department of Mathematics, Kyungpook National University, Daegu, 702-701, Korea*  
*e-mail*: [skim@knu.ac.kr](mailto:skim@knu.ac.kr), [mathkjk26@hanmail.net](mailto:mathkjk26@hanmail.net), [piaoxf76@gmail.com](mailto:piaoxf76@gmail.com) and  
[kimps@knu.ac.kr](mailto:kimps@knu.ac.kr)

**ABSTRACT.** The Chebyshev collocation method in [21] to solve stiff initial-value problems is generalized by using arbitrary degrees of interpolation polynomials and arbitrary collocation points. The convergence of this generalized Chebyshev collocation method is shown to be independent of the chosen collocation points. It is observed how the stability region does depend on collocation points. In particular, A-stability is shown by taking the mid points of nodes as collocation points.

### 1. Introduction

Owing to their importance in many mathematical problems, systems of ordinary differential equations have received considerable attention. These systems exhibit a critical behavior known as stiffness. A system of the form  $\phi'(t) = f(t, \phi(t))$ ,  $\phi(t_0) = \phi_0$  is called a stiff problem if the Jacobian of  $f$  has at least one eigenvalue  $\lambda$  with  $|\lambda|$  large (see [17]). For solving stiff problems, whether a numerical method is A-stable is important because one can choose the step size based on accuracy considerations only, without worrying about stability constraints.

It is well-known that there is no A-stable explicit linear method with the convergence order larger than 2 (see [12]). Curtiss and Hirschfelder in [11] suggested the backward differentiation formulae (BDF) whose A-stability is obtained only if it has the convergence order less than 2. Moreover, as the convergence order increases, the BDF scheme loses its stability more and more on a part of the imaginary axis. In particular, the schemes with convergence order larger than 7 are generally unstable, even at the origin (see [17]). Since then, great efforts

---

\* Corresponding Author.

Received August 3, 2011; accepted September 27, 2011.

2010 Mathematics Subject Classification: 65N38, 65N06, 65R20.

Key words and phrases: Chebyshev Collocation method, BDF-type methods, Stiff initial-value problem, Absolute stability.

<sup>†</sup> This work was supported by basic science research program through the National Research Foundation of Korea(NRF) funded by the ministry of education, science and technology(grant number 2011-0009825).

have been done to develop numerical integration methods with strong stability properties desirable for solving various stiff systems. In this respect, there are lots of different methods based on Runge-Kutta or power series approaches (see [3, 6, 7, 8, 14, 15, 24, 28, 18]) and also the schemes based on the modification of the classical BDF (see [21, 17, 9, 10, 13, 19, 20, 27, 5]).

Recently, Ramos and Vigo-Aguiar in [21] presented a new BDF-type method with the fourth order convergence and  $A(\alpha)$  stability,  $\alpha = 89.7473^\circ$ . The procedure consists of two simple steps. The first step is the interval approximation of the true solution based on the Chebyshev interpolation polynomial of the fourth degree at the Chebyshev-Gauss-Lobatto (CGL) points. The second step is to collocate the approximated equation with the same node points used in the Chebyshev approximation. Hereafter, we would like to call the scheme a Chebyshev collocation method (CCM). There were several related and continued studies [22, 25, 26].

The goal of this paper is to generalize the Chebyshev collocation method in [21] with a concrete error analysis which employs the interpolation polynomials of arbitrary degrees and the arbitrary collocation points between the given interpolation nodes. It is proven that the convergence is independent of the choice of collocation points and that the scheme has a higher-order convergence (as one wants to have) with the usages of the interpolation polynomials of higher degrees. However, it turns out that the stability is quite sensitive to the choice of collocation points. It is a remarkable phenomenon that taking the mid points  $\eta_j$  of the CGL points  $s_j$  and  $s_{j+1}$  as collocation points makes the role of barrier for A-stability. If collocation points are chosen in the left interval  $[s_j, \eta_j]$  of  $\eta_j$ , then we get  $A(\alpha)$ -stability for a sufficiently large  $\alpha$ . This coincides with the result in [21]. Otherwise, the stability region becomes a finite domain. Overall, the stability region is shrinking more and more if collocation points are chosen continuously from the left interval  $[s_j, \eta_j]$  of  $\eta_j$  to its right interval  $[\eta_j, s_{j+1}]$ .

The rest of the manuscript is presented as follows. Section 2 is devoted to the derivation of the approximate scheme based on a general interpolation polynomial and the abstract convergence analysis is given in Section 3. In Section 4, a special case with the Chebyshev interpolation polynomial at the CGL points is analyzed. The stability for the scheme is discussed in Section 5. Several test problems are performed in Section 6 to guarantee the convergence analysis and the discussion of the stability. Finally, in section 7 we conclude the manuscript with some comments.

## 2. Derivation of an approximation method

Let us consider a scalar stiff initial-value problem of the form

$$(2.1) \quad \frac{d\phi}{dt} = f(t, \phi(t)), \quad t \in (t_0, T]; \quad \phi(t_0) = \phi_0.$$

We assume that the function  $f$  in (2.1) satisfies all the necessary requirements for the existence of a unique solution.

Now that an approximation  $y_m$  to the analytical solution  $\phi(t)$  for the problem (2.1) at the time  $t_m$  is known, we are interested in obtaining a numerical approximation  $y_{m+1}$  at the next point  $t_{m+1} = t_m + h$ , where  $h$  is a given time step size.

To develop an accurate way of approximating  $\phi(t_{m+1})$ , we transform the solution  $\phi(t)$  restricted to  $[t_m, t_{m+1}]$  into  $\bar{\phi}(s)$  on the reference interval  $[-1, 1]$  such that

$$(2.2) \quad \bar{\phi}(s) := \phi(t) = \phi(t(s)), \quad s \in [-1, 1],$$

with  $t = t(s) = t_m + \frac{h}{2}(1+s)$ . Hence, instead of (2.1) on the subinterval  $[t_m, t_{m+1}]$ , one may have

$$(2.3) \quad \frac{d\bar{\phi}(s)}{ds} = \frac{h}{2} f(t(s), \bar{\phi}(s)), \quad s \in [-1, 1].$$

Let  $\{s_j\}_{j=0}^n$  be the interpolation points in  $[-1, 1]$  such that  $-1 = s_0 < s_1 < \dots < s_n = 1$ . The interpolation  $p_n(s)$  of  $\bar{\phi}(s)$  is given by

$$(2.4) \quad p_n(s) = \sum_{k=0}^n \bar{\phi}(s_k) l_k(s),$$

where  $\{l_k(s)\}_{k=0}^n$  are the basis functions such that  $l_k(s_j) = \delta_{kj}$ . (Here  $\delta_{kj}$  denotes the Kronecker delta function.) From the property of interpolation, one can see

$$(2.5) \quad \sum_{k=0}^n l_k(s) = 1, \quad s \in [-1, 1].$$

Let  $r_n(s)$  be the truncation error for the interpolation  $p_n(s)$  defined by

$$r_n(s) = \bar{\phi}(s) - p_n(s).$$

Then, the differential equation (2.3) can be written as

$$(2.6) \quad \sum_{k=0}^n \bar{\phi}(s_k) \dot{l}_k(s) - \frac{h}{2} f\left(t(s), \sum_{k=0}^n \bar{\phi}(s_k) l_k(s)\right) \\ = -\dot{r}_n(s) + \frac{h}{2} \left( f(t(s), \bar{\phi}(s)) - f(t(s), p_n(s)) \right),$$

where  $\dot{l}_k(s) = \frac{d}{ds} l_k(s)$  and  $\dot{r}_n(s) = \frac{d}{ds} r_n(s)$ .

Now it is required to calculate the coefficients  $\bar{\phi}(s_k)$ ,  $k = 1, \dots, n$  in (2.6) with a suitable approximation method. To do this, we will use a simple collocation method at collocation points  $\{\eta_k\}_{k=1}^n$  with  $-1 \leq \eta_1 < \dots < \eta_n \leq 1$  for the equation (2.6) among other numerical methods. Instead of solving (2.6) exactly, we consider the collocation method approximating (2.6) which becomes: find  $\alpha_k$  for  $k = 1, 2, \dots, n$  satisfying

$$(2.7) \quad \sum_{k=0}^n \alpha_k \dot{l}_k(\eta_j) - \frac{h}{2} f\left(t(\eta_j), \sum_{k=0}^n \alpha_k l_k(\eta_j)\right) = 0.$$

Here we will take the coefficient  $\alpha_0$  as  $y_m$  of the approximation to  $\bar{\phi}(s_0) = \phi(t_m)$ . Then one can get the system of equations for  $j = 1, \dots, n$ ,

$$(2.8) \quad \sum_{k=1}^n \alpha_k \dot{l}_k(\eta_j) - \frac{h}{2} f\left(t_{mj}, y_m l_0(\eta_j) + \sum_{k=1}^n \alpha_k l_k(\eta_j)\right) = -y_m \dot{l}_0(\eta_j),$$

where  $t_{mj} = t(\eta_j)$ . Note that the unknowns  $\alpha_k$  ( $k = 1, 2, \dots, n$ ) will be the approximate values to  $\bar{\phi}(s_k) = \phi(t(s_k))$ . Obviously, this algebraic system has to be solved by an iterative method. Solving the system (2.8), we obtain the required approximation  $\alpha_n$  to  $\phi(t(s_n)) = \phi(t_m + h)$ , which will be taken as  $y_{m+1}$  at the final point  $t_{m+1}$  on the interval  $[t_m, t_{m+1}]$ . Repeating this procedure along the integration interval  $[t_0, T]$ , a discrete solution for the problem in (2.1) is obtained. In the following section, we will provide the error analysis for  $\alpha_n - \bar{\phi}(s_n)$ .

**Remark 2.1.** Note that the developed scheme in this section is also applicable to a system of ordinary differential equations of the form

$$(2.9) \quad \Phi'(t) = \mathbf{f}(t, \Phi(t)), \quad t \in (t_0, T]; \quad \Phi(t_0) = \Phi_0,$$

where  $\Phi : \mathbb{R}^d \rightarrow \mathbb{R}^d$  and  $\mathbf{f} : [t_0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ .

### 3. Abstract error analysis

This section is devoting to give an abstract convergence analysis for the actual error

$$(3.1) \quad E_{m+1} = \phi(t_{m+1}) - y_{m+1}, \quad m \geq 0$$

between the exact solution  $\phi(t_{m+1})$  of (2.1) and the approximate solution  $y_{m+1} := \alpha_n$  of  $\phi(t_{m+1})$  obtained by (2.8).

We begin this section with the introduction of several notations. With the analytic solution  $\bar{\phi}(s)$  of the problem (2.3) and the approximate solution of the nonlinear system of algebraic equations (2.8), in each time interval  $[t_m, t_{m+1}]$ , let

$$\bar{\Phi} = [\bar{\phi}(s_0), \bar{\phi}(s_1), \dots, \bar{\phi}(s_n)]^T, \quad \Psi = [\alpha_0, \alpha_1, \dots, \alpha_n]^T,$$

where  $\alpha_0 = y_m$  and the super script  $T$  denotes the transpose.

For each  $j = 1, \dots, n$ , let us define  $F_j$  by

$$(3.2) \quad F_j(x_0, \dots, x_n) = \sum_{k=0}^n x_k \dot{l}_k(\eta_j) - \frac{h}{2} f\left(t_{mj}, \sum_{k=0}^n x_k l_k(\eta_j)\right).$$

The two equations (2.6) and (2.8) may be written as

$$(3.3) \quad F_j(\Psi) = 0, \quad F_j(\Phi) = -R_{nj}, \quad j = 1, \dots, n,$$

where

$$\begin{aligned}
 (3.4) \quad R_{nj} &= \dot{r}_n(\eta_j) - \frac{h}{2} \left( f(t_{mj}, \bar{\phi}(\eta_j)) - f(t_{mj}, p_n(\eta_j)) \right) \\
 &= \dot{r}_n(\eta_j) - \frac{h}{2} r_n(\eta_j) f_\phi(t_{mj}, \chi_j)
 \end{aligned}$$

for some  $\chi_j$  between  $\bar{\phi}(\eta_j)$  and  $p_n(\eta_j)$ . Using Taylor's theorem yields for  $j = 1, 2, \dots, n$ ,

$$(3.5) \quad F_j(\Phi) = F_j(\Psi) + (\bar{\phi}(s_0) - y_m) \frac{\partial F_j}{\partial x_0} + \sum_{k=1}^n (\bar{\phi}(s_k) - \alpha_k) \frac{\partial F_j}{\partial x_k},$$

where

$$(3.6) \quad \frac{\partial F_j}{\partial x_k} = \dot{l}_k(\eta_j) - \frac{h}{2} l_k(\eta_j) f_\phi \left( t_{mj}, \sum_{i=0}^n \xi_i l_i(\eta_j) \right)$$

for some  $\xi_i, i = 0, 1, \dots, n$  intermediate points between  $\bar{\phi}(s_i)$  and  $\alpha_i$ . Now, combining (3.3) and (3.5) leads to

$$(3.7) \quad \sum_{k=1}^n (\bar{\phi}(s_k) - \alpha_k) \frac{\partial F_j}{\partial x_k} = -R_{nj} - (\bar{\phi}(s_0) - y_m) \frac{\partial F_j}{\partial x_0},$$

which can be written in a simple matrix form as

$$(3.8) \quad \mathcal{J} \mathbf{e} = \mathbf{g}$$

where the matrix  $\mathcal{J}$ , vectors  $\mathbf{e}$  and  $\mathbf{g}$  are

$$(3.9) \quad \mathcal{J}(j, k) := \left( \frac{\partial F_j}{\partial x_k} \right), \quad j, k = 1, 2, \dots, n,$$

$$(3.10) \quad \mathbf{e}(j) := \bar{\phi}(s_j) - \alpha_j, \quad j = 1, 2, \dots, n,$$

$$(3.11) \quad \mathbf{g}(j) := -(E_m \frac{\partial F_j}{\partial x_0} + R_{nj}), \quad j = 1, 2, \dots, n,$$

respectively.

Define the matrix  $\mathcal{J}_1$  as

$$(3.12) \quad \mathcal{J}_1(j, k) := \dot{l}_k(\eta_j), \quad j, k = 1, 2, \dots, n.$$

**Lemma 3.1.** *Assume that  $\mathcal{J}_1$  is nonsingular. Then the Jacobian matrix  $\mathcal{J}$  is nonsingular for a uniformly bounded function  $f_\phi$  and sufficiently small step size  $h$ .*

*Proof.* Recall that the  $(j, k)$  component of the matrix  $\mathcal{J}_2 = \mathcal{J} - \mathcal{J}_1$  is

$$-\frac{h}{2}l_k(\eta_j)f_\phi\left(t_{mj}, \sum_{i=0}^n \xi_i l_i(\eta_j)\right)$$

Thus, if  $f_\phi$  is uniformly bounded, then the matrix  $\mathcal{J}_2 = \mathcal{J} - \mathcal{J}_1$  has the asymptotic behavior with  $\|\mathcal{J}_2\| = O(h)$ , where  $\|\cdot\|$  denotes a matrix norm. Hence, for sufficiently small step size  $h$ , one may have  $\|\mathcal{J}_1^{-1}\mathcal{J}_2\| < 1$ . Also, by the theorem of Geometric series (see Theorem 7.10 in [4]),  $\mathcal{J} + \mathcal{J}_1^{-1}\mathcal{J}_2$  is nonsingular, where  $\mathcal{J}$  is the identity matrix. Therefore,  $\mathcal{J} = \mathcal{J}_1(\mathcal{J} + \mathcal{J}_1^{-1}\mathcal{J}_2)$  is nonsingular.  $\square$

**Lemma 3.2.** *Assume that  $\mathcal{J}_1$  is nonsingular. Let  $\mathbf{a} := [\dot{i}_0(\eta_1), \dots, \dot{i}_0(\eta_n)]^T$ . Then, all entries of  $\mathcal{J}_1^{-1}\mathbf{a}$  should be exactly  $-1$ .*

*Proof.* From the property of the partition of unity (2.5), it can be easily check that

$$-\dot{i}_0(\eta_j) = \sum_{k=1}^n \dot{i}_k(\eta_j), \quad j = 1, \dots, n.$$

By the definition of  $\mathcal{J}_1$  and  $\mathbf{a}$ , the above system can be written with the matrix form

$$\mathcal{J}_1\mathbf{x} = -\mathbf{a}, \quad \mathbf{x} = [1, \dots, 1]^T.$$

Since  $\mathcal{J}_1$  is non-singular, the above equation yields  $\mathcal{J}_1^{-1}\mathbf{a} = -\mathbf{x}$  which completes the proof.  $\square$

Let  $\mathbf{g}_1 := \left[\frac{\partial F_1}{\partial x_0}, \dots, \frac{\partial F_n}{\partial x_0}\right]^T$  be a  $n \times 1$  vector. For the vector  $\mathbf{a}$  in Lemma 3.2 and the vector  $\mathbf{g}$  in (3.11), let

$$\mathbf{g}_2 := \mathbf{g}_1 - \mathbf{a}, \quad \text{and} \quad \mathbf{r} := -E_m\mathbf{g}_1 - \mathbf{g}.$$

In virtue of the system equation (3.8) and Lemma 3.2, one can get

$$\begin{aligned} \mathbf{e} &= \mathcal{J}^{-1}\mathbf{g} \\ &= -(\mathcal{J}_1 + \mathcal{J}_2)^{-1}(E_m\mathbf{g}_1 + \mathbf{r}) \\ (3.13) \quad &= -E_m(\mathcal{J}_1 + \mathcal{J}_2)^{-1}(\mathbf{g}_2 + \mathbf{a}) - \mathcal{J}^{-1}\mathbf{r} \\ &= -E_m(\mathcal{J} + \mathcal{J}_1^{-1}\mathcal{J}_2)^{-1}\mathcal{J}_1^{-1}\mathbf{a} - E_m\mathcal{J}^{-1}\mathbf{g}_2 - \mathcal{J}^{-1}\mathbf{r} \\ &= E_m(\mathcal{J} + \mathcal{J}_1^{-1}\mathcal{J}_2)^{-1}\mathbf{I} - E_m\mathcal{J}^{-1}\mathbf{g}_2 - \mathcal{J}^{-1}\mathbf{r} \end{aligned}$$

where  $\mathcal{J}$  is the identity matrix,  $\mathbf{I} = [1, \dots, 1]^T$  and  $E_m$  is defined in (3.1).

From the definition of  $\mathcal{J}$ , it can be observed that both  $\|\mathcal{J}_1\|$  and  $\|\mathcal{J}_1^{-1}\|$  are uniformly bounded independent of the step size  $h$ . Thus, there is a constant  $C$  independent of  $h$  such that

$$\|\mathcal{J}^{-1}\| \leq \|\mathcal{J}_1^{-1}\| \|(I + \mathcal{J}_1^{-1}\mathcal{J}_2)^{-1}\| \leq \frac{\|\mathcal{J}_1^{-1}\|}{1 - \|\mathcal{J}_1^{-1}\|\|\mathcal{J}_2\|} \leq C.$$

Using the facts  $\mathbf{g}_2 = O(h)$  (see (3.6)),  $\|\mathcal{J}_2\| = O(h)$ , the theorem of Geometric series for  $(I + \mathcal{J}_1^{-1}\mathcal{J}_2)^{-1}$  and the equation (3.13), it is easy to check that

$$(3.14) \quad \begin{cases} |E_{m+1}| \leq (1 + Ch)|E_m| + D\delta_n, & m \geq 0, \\ E_0 = 0, \end{cases}$$

for some constants  $C$  and  $D$  independent of  $h$ , where

$$(3.15) \quad \delta_n = \max_{1 \leq j \leq n} |R_{nj}|.$$

Consequently, we show the following theorem.

**Theorem 3.1.** (Convergence) *For sufficiently small  $h$  with  $mh \leq T$ ,  $m \geq 0$ , the actual error  $E_m$  can be estimated by*

$$(3.16) \quad |E_m| \leq \frac{D}{C}(\exp(CT) - 1)\frac{\delta_n}{h}, \quad m \geq 0,$$

where  $C$  and  $D$  are the constants in (3.14).

*Proof.* By mathematical induction, it is easy to show that the difference equation (3.14) can be solved by

$$(3.17) \quad |E_m| \leq D\frac{(1 + Ch)^m - 1}{Ch}\delta_n, \quad m \geq 0.$$

If  $mh \leq T$ , then  $1 + Ch \leq \exp(Ch)$  and  $(1 + Ch)^m \leq \exp(Cmh) \leq \exp(CT)$  and so the inequality (3.17) provides the desired result.  $\square$

#### 4. Special cases

In the previous section, we discussed the convergence of the proposed scheme for a general interpolation polynomial  $p_n(s)$  interpolating  $\bar{\phi}(s)$  which is the exact solution of (2.3).

In this section, we take the Chebyshev interpolation polynomial  $p_n(s)$  such that

$$(4.1) \quad p_n(s) = \sum_{k=0}^n \bar{\phi}(s_k)l_k(s),$$

which interpolates the solution  $\bar{\phi}(s)$  of (2.3) at the CGL points

$$(4.2) \quad s_k = \cos \frac{(n - k)\pi}{n},$$

where

$$(4.3) \quad l_k(s) = \frac{a_k}{n} \sum_{j=0}^n{}'' T_j(s_k)T_j(s), \quad a_k = \begin{cases} 1, & k = 0, n, \\ 2, & \text{otherwise.} \end{cases}$$

Here  $T_k(s)$  is the Chebyshev polynomial of first kind of degree  $k$  and the double prime indicates that both the first and last terms in the summation are to be halved.

With the Chebyshev interpolation polynomial  $p_n(s)$ , we will show that the proposed scheme has the convergence of order  $n$ . That is, one can make a high-order convergence scheme as one wants to have with usages of the Chebyshev interpolation of degree  $n$  in (4.1).

It is well-known that the basis functions  $l_k(s)$  defined in (4.3) satisfy (2.5). Furthermore it easily seen that the following lemma [26].

**Lemma 4.1.** *The matrix  $\mathcal{J}_1$  such that  $\mathcal{J}_1(j, k) := (\dot{l}_k(\eta_j))$  is non-singular.*

*Proof.* The proof can be found in [26]. □

This lemma guarantees that the convergence result in Theorem 3.1 for Chebyshev case holds and the bound of the actual error  $E_m = \phi(t_m) - y_m$  depends on the bound of  $\frac{\delta_n}{h}$ , where  $\delta_n$  is a bound of the quantity  $R_{nj}$  defined in (3.4). Furthermore,  $R_{nj}$  depends only on the truncation error for the interpolation polynomial  $p_n(s)$ .

Consequently to proof end, it remains to estimate  $\delta_n$  and then we can improve the inequality (3.16). Let us now estimate the error  $r_n(s)$  which is

$$(4.4) \quad r_n(s) := \bar{\phi}(\cos s) - p_n(\cos s).$$

Without confusion, for simplicity, let

$$\bar{\phi}(s) := \bar{\phi}(\cos s) \quad \text{and} \quad p_n(s) := p_n(\cos s).$$

Let us rewrite the interpolating polynomial  $p_n(s)$  in (4.1) as the following Lagrangian form

$$(4.5) \quad p_n(s) = \sum_{j=0}^n \frac{u_n(s)}{(\cos s - \cos s_j) \prod_{\substack{k=0 \\ k \neq j}}^n (\cos s_j - \cos s_k)} \bar{\phi}(s_j)$$

where

$$(4.6) \quad u_n(s) := \prod_{j=0}^n (\cos s - \cos s_j), \quad s \in [0, \pi].$$

Then, we get the error  $r_n(s)$  as

$$(4.7) \quad \begin{aligned} r_n(s) &= \bar{\phi}(s) - p_n(s) \\ &= \left(\frac{h}{2}\right)^{n+1} \frac{u_n(s)}{(n+1)!} \phi^{(n+1)}\left(t_m + \frac{h}{2}\xi(s)\right), \quad \xi(s) \in (-1, 1). \end{aligned}$$

We note that the mean-value point  $\xi(s)$  in (4.7) depends continuously on  $s$  and

$$(4.8) \quad \frac{d}{ds} \phi^{(n+1)}\left(t_m + \frac{h}{2}\xi(s)\right) = \frac{h}{2} \frac{\phi^{(n+2)}\left(t_m + \frac{h}{2}\chi(s)\right)}{n+2}$$



for some  $\chi(s) \in (-1, 1)$ . Applying the change of variables  $t = \cos s$  and  $t_j = \cos s_j$ , it may be shown that the polynomial  $u_n(s)$  in (4.6) can be simplified as

$$(4.9) \quad u_n(s) = u_n(\cos^{-1} t) = 2^{-n+1}(t^2 - 1)U_{n-1}(t) = -\frac{\sin s \sin(ns)}{2^{n-1}},$$

where  $U_{n-1}(t)$  is the Chebyshev polynomial of the second kind. In accordance with (4.9), it follows that

$$(4.10) \quad |u_n(s)| \leq \frac{1}{2^{n-1}},$$

$$(4.11) \quad \left| \dot{u}_n(s) \right| = \frac{1}{2^{n-1}} \left| (n-1) \cos(ns) \sin s + \sin((n+1)s) \right| \leq \frac{n}{2^{n-1}}.$$

Thus, applying (4.8), (4.10) and (4.11) to (4.7) leads to

$$\begin{aligned} \left| \dot{r}_n(s) \right| &\leq \left( \frac{h}{2} \right)^{n+1} \frac{1}{(n+1)!} \left[ \left| u_n(s) \right| \left| \frac{d}{ds} \phi^{(n+1)} \left( t_m + \frac{h}{2} \xi(s) \right) \right| \right. \\ &\quad \left. + \left| \phi^{(n+1)} \left( t_m + \frac{h}{2} \xi(s) \right) \right| \left| \frac{d}{ds} u_n(s) \right| \right] \\ &\leq \left( \frac{h}{2} \right)^{n+1} \frac{1}{2^{n-1}(n+1)!} \left[ \frac{h}{2(n+2)} \|\phi^{(n+2)}\|_\infty + n \|\phi^{(n+1)}\|_\infty \right]. \end{aligned}$$

Also, from (4.7) and (4.9), one can estimate  $r_n(s)$  with

$$\left| r_n(s) \right| \leq \left( \frac{h}{2} \right)^{n+1} \frac{1}{2^{n-1}(n+1)!} \|\phi^{(n+1)}\|_\infty.$$

We can bound  $R_{nj}$  with above estimates as follows:

$$(4.12) \quad \begin{aligned} |R_{nj}| &\leq \frac{h}{2} |f_\phi(t_{mj}, \xi)| |r_n(\eta_j)| + |\dot{r}_n(\eta_j)| \\ &\leq \left( \frac{h}{2} \right)^{n+1} \frac{1}{2^{n-1}(n+1)!} \left( \frac{h}{2(n+2)} \|\phi^{(n+2)}\|_\infty + (C \frac{h}{2} + n) \|\phi^{(n+1)}\|_\infty \right), \end{aligned}$$

where  $C$  is a positive constant arisen from the bound of  $|f_\phi(t_{mj}, \xi)|$ . Summarizing the above discussions, from Theorem 3.1, one can have the following convergence theorem.

**Theorem 4.1.** *For a sufficiently small  $h$  and all integers  $m \geq 0$  with  $mh \leq T$ , the actual error  $E_m$  satisfies*

$$(4.13) \quad |E_m| \leq C(\exp(CT) - 1)h^n, \quad m \geq 0,$$

for some constant  $C$  independent of  $h$ .

*Proof.* First, estimate  $\delta_n = \max_{1 \leq j \leq n} |R_{nj}|$  using (4.12). Then the proof will be

completed by using Theorem 3.1.  $\square$

**Remark 4.1.** We remark that the convergence analysis is independent of the choice of collocation points. However, we will show that the stability for the proposed scheme is dependent of the choice of collocation points  $\eta_j$ .

## 5. Stability analysis

For a calculation of the stability domain for the algorithm (2.8), we apply the method to the Dalquist's test problem,  $\phi' = \lambda\phi$ . Then, the system (2.8) arrives at

$$(5.1) \quad \sum_{k=1}^n \alpha_k \left( \dot{l}_k(\eta_j) - \frac{\lambda h}{2} l_k(\eta_j) \right) = \left( \frac{\lambda h}{2} l_0(\eta_j) - \dot{l}_0(\eta_j) \right) \bar{\phi}(s_0),$$

where  $j = 1, \dots, n$  and  $\alpha_k$  are the approximation to  $\bar{\phi}(s_k) = \phi(t(s_k))$ . Let  $A := (a_{jk})$  be a  $n \times n$  matrix such that

$$a_{jk} = \dot{l}_k(\eta_j) - \frac{\lambda h}{2} l_k(\eta_j), \quad j = 1, \dots, n, \quad k = 1, \dots, n,$$

and  $\mathbf{b} = [b_1, \dots, b_n]^T$  such that

$$b_j = \frac{\lambda h}{2} l_0(\eta_j) - \dot{l}_0(\eta_j), \quad j = 1, \dots, n.$$

Then, solving the system  $A\mathbf{x} = \mathbf{b}$  yields the difference equation

$$(5.2) \quad \phi_{m+1} = \mathcal{R}_n(\lambda h)\phi_m, \quad m \geq 0,$$

where  $\mathcal{R}_n(\lambda h)$  is the last component of the vector  $A^{-1}\mathbf{b}$ . For the function  $\mathcal{R}_n(z)$ , the stability domain of the method is defined as (see [16, p.16])

$$(5.3) \quad \mathcal{S}_n := \{z \in \mathbb{C} : |\mathcal{R}_n(z)| \leq 1\}.$$

In particular, recall that the method is called *A*-stable if the left-half complex plane is contained in  $\mathcal{S}_n$ .

We now present the explicit formula on  $\mathcal{R}_n(z)$  for the presented method in this paper and draw the corresponding stability domain  $\mathcal{S}_n$  using the symbolic calculation with Mathematica for two specific cases about the effects of the chosen collocation points. The first case is to choose collocation points as node points. The second case is to choose collocation points as the mid points of nodes.

Case 1. (Collocation points = node points)

Suppose that we take collocation points as the node points, that is,  $\eta_j = s_j$ ,

$j = 1, \dots, n$ . Then, the definition of  $l_k(s)$  and  $\mathcal{R}_n(z)$  in (4.3) and (5.2) give

$$\begin{aligned}
 \mathcal{R}_1(z) &= \frac{(1)}{(1, -1)}, \\
 \mathcal{R}_2(z) &= \frac{(4, 1)}{(4, -3, 1)}, \\
 \mathcal{R}_3(z) &= \frac{(96, 32, 3)}{(96, -64, 19, -3)}, \\
 \mathcal{R}_4(z) &= \frac{(384, 144, 20, 1)}{(384, -240, 68, -11, 1)}, \\
 \mathcal{R}_5(z) &= \frac{(30720, 12288, 2016, 160, 5)}{(30720, -18432, 5088, -832, 85, -5)}, \\
 \mathcal{R}_6(z) &= \frac{(368640, 153600, 27648, 2688, 140, 3)}{(368640, -215040, 58368, 9600, 1036, -73, 3)}, \\
 \mathcal{R}_7(z) &= \frac{(20643840, 8847360, 1689600, 184320, 12096, 448, 7)}{(20643840, -11796480, 3164160, -522240, 58176, -4480, 231, -7)}, \\
 \mathcal{R}_8(z) &= \frac{(82575360, 36126720, 7188480, 844800, 63360, 3024, 84, 1)}{(82575360, -46448640, 12349440, -2042880, 232320, -18864, 1092, -43, 1)},
 \end{aligned}
 \tag{5.4}$$

where the notation  $(a_0, \dots, a_k)$  means the polynomial  $\sum_{j=0}^k a_j z^j$  with respect to  $z$ . The Fig. 1 shows the stability domains for  $\mathcal{S}_n$ ,  $n = 2, \dots, 5$  and it can be observed that A-stability holds only for  $n = 2$ . The stability regions widen and violates the left-half complex plane when the degrees of the Chebyshev interpolation polynomial are increasing.

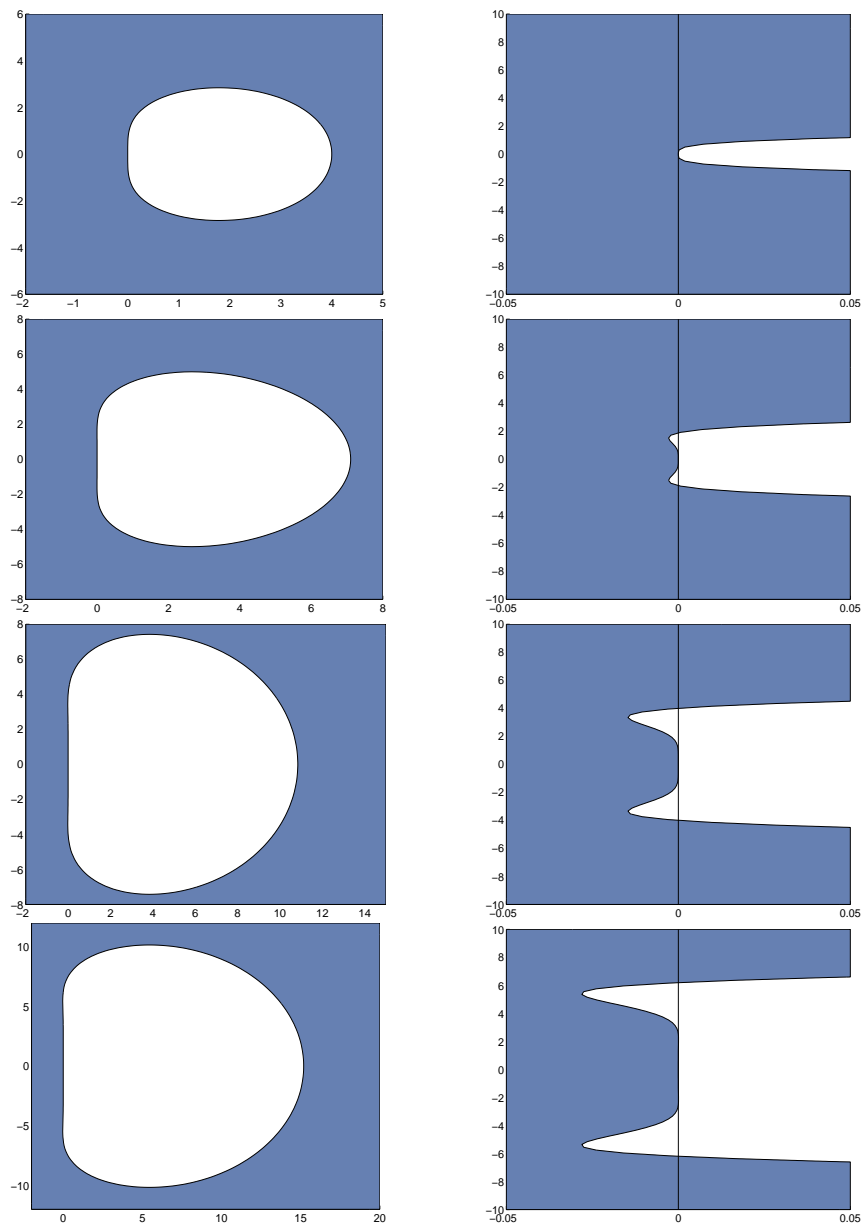
This first case is now compared with the case for the chosen collocation points as the mid points of the nodes by calculating the stability domain.

Case 2. (Collocation points = mid points of node points)

Let collocation points  $\eta_j$  be

$$\eta_j = \cos \xi_j, \quad \xi_j = \frac{2n - 2j + 1}{2n} \pi \quad j = 1, \dots, n.$$

Then it is not same with the mid points of the node points. However, in the sense of  $\xi_j = \frac{1}{2} \left( \frac{(n-j)\pi}{n} + \frac{(n-j+1)\pi}{n} \right)$ , we call  $\eta_j$  the mid points of the node points hereafter.



**Fig. 1.** (Left) Stability domain. (Right) Detail near the origin. From the first row to the final row, the convergence order  $N$  is increasing from 2 to 5.

Then, the definition of  $l_k(s)$  and  $\mathcal{R}_n(z)$  in (4.3) and (5.2) give

$$\begin{aligned}
 \mathcal{R}_1(z) &= \frac{(2, 1)}{(2, -1)}, \\
 \mathcal{R}_2(z) &= \frac{(16, 8, 1)}{(16, -8, 1)}, \\
 \mathcal{R}_3(z) &= \frac{(192, 96, 18, 1)}{(192, -96, 18, -1)}, \\
 (5.5) \quad \mathcal{R}_4(z) &= \frac{(3072, 1536, 320, 32, 1)}{(3072, -1536, 320, -32, 1)}, \\
 \mathcal{R}_5(z) &= \frac{(61440, 30720, 6720, 800, 50, 1)}{(61440, -30720, 6720, -800, 50, -1)}, \\
 \mathcal{R}_6(z) &= \frac{(1474560, 737280, 165888, 21504, 1680, 72, 1)}{(1474560, 737280, 165888, 21504, 1680, 72, 1)}, \\
 \mathcal{R}_7(z) &= \frac{(41287680, 20643840, 4730880, 641520, 56448, 3136, 98, 1)}{(41287680, -20643840, 4730880, -641520, 56448, -3136, 98, -1)}.
 \end{aligned}$$

These stability functions satisfy the A-stability as follows:

**Lemma 5.1.** *For the stability functions  $R_n(z)$  in (5.5), the domain  $S_n$  defined in (5.3) is the left-half complex plane.*

*Proof.* Let  $N_n(z)$  and  $D_n(z)$  be the numerator and denominator of  $\mathcal{R}_n(z)$ , respectively,  $n = 1, 2, \dots, 7$  in (5.5). Define  $G_n(z)$  by

$$G_n(z) = |N_n(z)|^2 - |D_n(z)|^2.$$

Then, it can be shown that  $G_n(z) = c_n x p_n(x, y)$ , where  $z = x + iy$ ,  $c_n$  are positive constants and  $p_n(x, y)$  are polynomials containing only even degree terms with respective to  $x$  and  $y$ . Here  $i = \sqrt{-1}$ . In fact, some manipulations give

$$\begin{aligned}
 G_1(z) &= 8x, \\
 G_2(z) &= 32x(16 + x^2 + y^2), \\
 G_3(z) &= 24x(3x^4 + 3(32 + y^2)^2 + x^2(320 + 6y^2)), \\
 G_4(z) &= 128x(x^6 + 6144(24 + y^2) + y^4(176 + y^2) + x^4(368 + 3y^2) \\
 &\quad + x^2(18432 + 544y^2 + 3y^4)), \\
 G_5(z) &= 40x(5x^8 + 5898240(32 + y^2) + 4x^6(1168 + 5y^2) \\
 &\quad + 6x^4(116224 + 1888y^2 + 5y^4) \\
 &\quad + y^4(107520 + 1984y^2 + 5y^4) \\
 &\quad + 20x^2(1277952 + 35328y^2 + 432y^4 + y^6)),
 \end{aligned}$$

$$\begin{aligned}
G_6(z) &= 96x(3x^{10} + 1132462080(40 + y^2) + x^8(5936 + 15y^2)) \\
&\quad + 3y^4(5308416 + 55296y^2 + 784y^4 + y^6) \\
&\quad + 6x^4(34111488 + 623616y^2 + 4144y^4 + 5y^6) \\
&\quad + x^2(6417285120 + 149815296y^2 + 1873920y^4 + 12922y^6 + 15y^8) \\
&\quad + 6x^6(338944 + 5y^2(672 + y^2)), \\
G_7(z) &= 56x(7x^{12} + 14x^{10}(1856 + 3y^2) + 707788800(86016 + 1837y^2)) \\
&\quad + x^8(17472912 + 113792y^2 + 105y^4) \\
&\quad + 4x^6(948445440 + 11968512y^2 + 48832y^4 + 35y^6) \\
&\quad + y^4(13316751360 + 109347840y^2 + 862032y^4 + 9856y^6 + 7y^8) \\
&\quad + 2x^2(4433942937600 + 87304273920y^2 \\
&\quad + 920962560y^4 + 7326144y^6 + 32704y^8 + 21y^{10}) \\
&\quad + x^4(309266841600 + 7y^2(762516480 + 6313056y^2 + 23296y^4 + 15y^6)).
\end{aligned}$$

Since  $p_n(x, y) > 0$ , the above equations show that  $|N_n(z)| \leq |D_n(z)|$  if and only if  $x \leq 0$ , and also  $|N_n(z)| > |D_n(z)|$  if and only if  $x > 0$  for  $n = 1, 2, \dots, 7$ . Hence the domain  $\mathcal{S}_n$  is exactly the left-half complex plane.  $\square$

## 6. Numerical test

For solving the linear system occurred in the Newton's iteration for the non-linear system (2.8), we use LU decomposition for a matrix. We test both a scalar stiff problem and a stiff system of problems. The  $n$ -th order CCM with taking collocation points as the node points is denoted by CBDF $n$ , whereas taking collocation points as the mid points of the node points is denoted by MBDF $n$ . In order to confirm the convergence of CBDF $n$  and MBDF $n$ , we are choosing the investigated error as the maximum distance between the approximated solution and the analytic solution at the node points. The rates of convergence are computed by  $\log(\text{Err}(h)/\text{Err}(h/2))/\log(2)$ , where  $\text{Err}(h)$  is the absolute maximum error when the mesh size is chosen by  $h$ .

**Example 6.1.** As a first example, we test the problem

$$\frac{dy}{dx} = \lambda(y - \sin(x)) + \cos(x), \quad x \in [0, 1]; \quad y(0) = 1,$$

where  $\lambda$  is a parameter. The exact solution is

$$y(x) = e^{\lambda x} + \sin(x).$$

The errors of the numerical results obtained from both CBDF $n$  and MBDF $n$ ,  $n = 4, 8$ , are reported in Table 1 and 2, respectively. The errors of CBDF with MBDF are plotted using the logarithm scale in Figure 2 with the fixed step size

**Table 1**

Results for Example 6.1, using the fourth-order CCM with different collocation points.

$\lambda$	$h$	$\lambda h$	Err(CBDF4)	Err(MBDF4)
4	$2^{-1}$	2	1.1960e+00	1.6189e-01
	$2^{-2}$	1	3.6960e-02	5.9739e-03
	$2^{-3}$	0.5	1.6394e-03	3.1515e-04
	$2^{-4}$	0.25	8.6873e-05	1.8811e-05
	$2^{-5}$	0.125	5.0097e-06	1.1619e-06
1	$2^{-1}$	0.5	3.1568e-05	6.1319e-06
	$2^{-2}$	0.25	1.6794e-06	3.6461e-07
	$2^{-3}$	0.125	9.6935e-08	2.2498e-08
	$2^{-4}$	0.0625	5.8241e-09	1.4015e-09
	$2^{-5}$	0.03125	3.5694e-10	8.7438e-11
-1	$2^{-1}$	-0.5	5.7477e-07	2.0864e-07
	$2^{-2}$	-0.25	4.0743e-08	1.2093e-08
	$2^{-3}$	-0.125	2.7326e-09	7.4152e-10
	$2^{-4}$	-0.0625	1.7726e-10	4.6115e-11
	$2^{-5}$	-0.03125	1.1296e-11	2.8610e-12
-10	$2^{-1}$	-5	6.4274e-04	6.0100e-03
	$2^{-2}$	-2.5	2.4842e-04	5.0024e-04
	$2^{-3}$	-1.25	3.8822e-05	2.6806e-05
	$2^{-4}$	-0.625	3.2886e-06	1.3043e-06
	$2^{-5}$	-0.3125	2.5142e-07	7.7862e-08

$h = 2^{-4}$  and the parameter  $\lambda$  from  $-32$  to  $32$ . Table 1, 2 and Fig. 2 show that the errors for both CBDF and MBDF depend on the values  $h\lambda$  and convergence order.

**Example 6.2.** Consider a traditional problem in [11]

$$\frac{dy}{dx} = 5(y - x^2), \quad x \in [0, 2]; \quad y(0) = \frac{3}{25}.$$

The exact solution is

$$y(x) = \frac{1}{25}(e^{5x} + 2 + 10x + 25x^2).$$

With the step size  $h = 2^{-n}$  for various  $n = 2, 3, \dots, 6$  over  $2^n$  steps, the computed errors for CBDF and MBDF are reported in Table 3 and 4. The numerical results show that both CBDF and MBDF have good qualitative behaviors and support the convergence analysis developed in previous section.

**Table 2**

Results for Example 6.1, using the eighth-order CCM with different collocation points.

$\lambda$	$h$	$\lambda h$	Err(CBDF8)	Err(MBDF8)
8	$2^{-1}$	4	1.6764e+00	9.1385e-02
	$2^{-2}$	2	1.8897e-03	1.2237e-04
	$2^{-3}$	1	4.0710e-06	3.3011e-07
1	$2^{-1}$	0.5	2.1014e-12	2.0206e-13
	$2^{-2}$	0.25	6.2172e-15	4.8850e-15
	$2^{-3}$	0.125	3.5527e-15	8.8818e-15
-1	$2^{-1}$	-0.5	4.0634e-14	6.6613e-15
	$2^{-2}$	-0.25	6.6613e-16	1.1102e-15
	$2^{-3}$	-0.125	6.6613e-16	2.4425e-15
-10	$2^{-1}$	-5	2.4448e-07	1.4115e-06
	$2^{-2}$	-2.5	8.4352e-09	7.9739e-09
	$2^{-3}$	-1.25	1.0413e-10	3.1853e-11

**Example 6.3.** Consider the nonlinear initial value problem in [2]

$$\frac{dy}{dx} = \frac{\lambda y(x)(1-y(x))}{2y(x)-1}, \quad x \in [0, 10]; \quad y(0) = \frac{5}{6}.$$

The exact solution is given by

$$y(x) = \frac{1}{2} + \sqrt{\frac{1}{4} - \frac{5}{36}e^{-\lambda x}}$$

with a parameter  $\lambda$ .

The numerical results and the required computational costs are compared in Table 5,6. For  $\lambda = 1$ , the result of MBDF is a little bit superior to that of CBDF with the step size  $h$  small enough (see Table 5). For a large parameter  $\lambda = 50$ , two results of CBDF and MBDF are quite similar (see Table 6).

**Example 6.4.** As a scalar stiff problem, we test the problem taken from [1, 23]

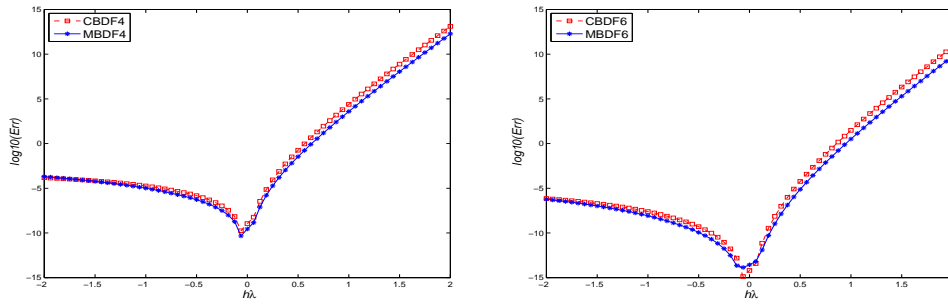
$$y'(x) = -100y(x) + 99e^{2x}, \quad x > 0; \quad y(0) = 0$$

with the exact solution given by

$$y(x) = \frac{33}{34} \left( e^{2x} - e^{-100x} \right).$$

The solution involves rapidly decaying transient term and hence the problem is a





**Fig. 2.** Errors of CBDF and MBDF along  $h\lambda$  with a fixed mesh size  $h = \frac{1}{16}$ .

**Table 3**

Results for Example 6.2, using the fourth-order CCM with different collocation points.

$h$	Err(CBDF4)	rate	Err(MBDF4)	rate
$2^{-2}$	4.33282331e+0	-	6.59761458e-1	-
$2^{-3}$	1.75624855e-1	4.6247	3.20902844e-2	4.3617
$2^{-4}$	8.91046477e-3	4.3009	1.86861636e-3	4.1021
$2^{-5}$	5.03263451e-4	4.1461	1.14669533e-4	4.0264
$2^{-6}$	2.99267156e-5	4.0718	7.13367580e-6	4.0067

**Table 4**

Results for Example 6.2, using the sixth-order CCM with different collocation points.

$h$	Err(CBDF6)	rate	Err(MBDF6)	rate
$2^{-2}$	8.33393245e-3	-	8.30451604e-4	-
$2^{-3}$	8.85779355e-5	6.5559	1.08430277e-5	6.2591
$2^{-4}$	1.14698377e-6	6.2710	1.61616981e-7	6.0680
$2^{-5}$	1.63827280e-8	6.1295	2.60195065e-9	5.9568

**Table 5**

Results for Example 6.3, using the fourth-order CCM with different collocation points when  $\lambda = 1$ .

$h$	Err(CBDF4)	cpu	Err(MBDF4)	cpu
$2^{-1}$	5.63e-06	5.38e-03	1.00e+00	5.58e-03
$2^{-2}$	4.77e-07	8.63e-03	1.00e+00	8.85e-03
$2^{-3}$	3.58e-08	1.69e-02	1.01e-08	2.52e-02
$2^{-4}$	2.39e-09	2.73e-02	6.33e-10	4.29e-02
$2^{-5}$	1.54e-10	5.38e-02	3.97e-11	8.31e-02

**Table 6**

Results for Example 6.3, using the fourth-order CCM with different collocation points when  $\lambda = 50$ .

$h$	Err(CBDF4)	cpu	Err(MBDF4)	cpu
$2^{-7}$	2.48e-06	2.59e-01	8.95e-07	4.57e-01
$2^{-8}$	1.97e-07	5.15e-01	5.90e-08	8.89e-01
$2^{-9}$	1.37e-08	8.31e-01	3.75e-09	1.44e+00
$2^{-10}$	9.04e-10	1.67e+00	2.37e-10	2.86e+00
$2^{-11}$	5.79e-11	3.30e+00	1.48e-11	5.61e+00

**Table 7**

Results for Example 6.4, using the fourth-order CCM with different collocation points.

$h$	Err(CBDF4)	cpu	Err(MBDF4)	cpu
$2^{-4}$	4.67e-04	1.01e-03	1.07e-02	1.06e-03
$2^{-5}$	3.72e-04	1.55e-03	1.16e-03	1.71e-03
$2^{-6}$	7.44e-05	2.70e-03	6.83e-05	2.86e-03
$2^{-7}$	7.17e-06	5.08e-03	3.25e-06	5.10e-03
$2^{-8}$	5.63e-07	9.05e-03	1.85e-07	9.75e-03
$2^{-9}$	4.00e-08	1.98e-02	1.14e-08	2.01e-02

**Table 8**

Results for Example 6.5 at final time  $t = 1$ , using the third-order CBDF and MBDF.

$\Delta x$	$h$	Err(CBDF3)	cpu	Err(MBDF3)	cpu
1.0e-1	1.0e-1	9.44e-6	0.008	9.26e-6	0.009
	5.0e-2	8.86e-6	0.009	8.75e-6	0.010
	2.5e-2	8.74e-6	0.020	8.72e-6	0.020
5.0e-2	1.0e-1	2.81e-6	0.010	2.65e-6	0.010
	5.0e-2	2.26e-6	0.015	2.15e-6	0.015
	2.5e-2	2.14e-6	0.025	2.12e-6	0.026
2.5e-2	1.0e-1	1.21e-6	0.014	1.05e-6	0.017
	5.0e-2	6.64e-7	0.026	5.58e-7	0.032
	2.5e-2	5.54e-7	0.046	5.28e-7	0.048
1.25e-2	1.0e-1	8.16e-7	0.043	6.56e-7	0.055
	5.0e-2	2.69e-7	0.068	1.63e-7	0.084
	2.5e-2	1.53e-7	0.126	1.33e-7	0.155
6.25e-3	1.0e-1	7.17e-7	0.220	5.57e-7	0.284
	5.0e-2	1.70e-7	0.438	6.46e-8	0.521
	2.5e-2	5.42e-8	0.744	3.48e-8	0.896

stiff differential equation. The problem has been integrated on the interval  $[0, 0.5]$  and the results are listed in Table 7. The numerical results show that both CBDF4 and MBDF4 have good qualitative similar behaviors.

**Example 6.5.** As a stiff system of differential equations, we test the initial value problem (IVP)

$$\begin{cases} \frac{d\mathbf{u}}{dt} = \mathbf{A}\mathbf{u}(t), & t > 0, \\ \mathbf{u}(0) = [u_1(0), \dots, u_N(0)]^T, \end{cases}$$

where  $A$  is the  $N \times N$  tridiagonal matrix given by

$$A = \frac{1}{(\Delta x)^2} \text{tridiag}([1, -2, 1]), \quad \Delta x = \frac{1}{N+1},$$

and

$$u_i(0) = 2 \sin(\pi x_i), \quad x_i = i\Delta x, \quad i = 1, \dots, N.$$

The IVP can be derived by the method of line approach for the linear parabolic

equation

$$\begin{cases} \frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, & x \in (0, 1), \quad t > 0 \\ u(0, t) = u(1, t) = 0, & t > 0, \\ u(x, 0) = 2 \sin(\pi x), & x \in (0, 1), \end{cases}$$

with the exact solution

$$u(x, t) = 2e^{-\pi^2 t} \sin(\pi x).$$

As stated in [22], the system becomes very stiff for large values of  $N$ . The numerical results and the required cpu times of the third-order CBDF and MBDF are presented in Table 8. Both methods have comparable numerical results for the same spatial mesh size  $\Delta x$  and time step length  $h$ . That is, the error bound does not depend on the choice of collocation points as shown in the mathematical analysis of the previous sections. It can be observed that both methods MBDF and CBDF have a similar error bound and require a similar cpu time.

## 7. Conclusion

A generalization of the CCM developed by [21] is discussed using interpolation polynomials of higher degrees and the arbitrary collocation points between interpolation nodes. Also, a concrete abstract convergence analysis is performed and it turns out that the convergence is independent of the choice of collocation points. As a special case, we consider the Chebyshev interpolation polynomials of higher degrees at CGL points and show how the stability region changes depending on the choice of collocation points. It is a remarkable phenomenon that the mid points of the CGL points as collocation points, which makes the role of barrier for  $A$ -stability, do only create  $A$ -stability.

## References

- [1] R. R. Ahmad, N. Yaacob, A. H. Mohd Murid, *Explicit methods in solving stiff ordinary differential equations*, Int. J. Comput. Math., **81**(2004), 1407-1415.
- [2] J. Álvarez, J. Rojo, *An improved class of generalized Runge-Kutta methods for stiff problems. Part I: The scalar case*, Appl. Math. Comput., **130**(2002), 537-560.
- [3] J. Álvarez, J. Rojo, *An improved class of generalized Runge-Kutta methods for stiff problems. Part II: the separated system case*, Appl. Math. Comput., **159**(2004), 717-758.
- [4] K. E. Atkinson, *An Introduction to Numerical Analysis*, 2nd ed., John Wiley & Sons, New York, 1989.
- [5] P. N. Brown, G. D. Byrne, A. C. Hindmarsh, *VODE: A variable coefficient ODE solver*, SIAM J. Sci. Statist. Comput., **10**(1989), 1038-1051.
- [6] J. C. Butcher, *Implicit Runge-Kutta processes*, Math. Comput., **18**(1964), 50-64.

- [7] J. C. Butcher, *Integration processes based on Radau quadrature formulas*, Math. Comput., **18**(1964), 233-244.
- [8] J. C. Butcher, G. Wanner, *Runge-Kutta methods: some historical notes*, Appl. Numer. Math., **22**(1996), 113-151.
- [9] J. R. Cash, *On the integration of stiff systems of ODE's using extended backward differential formulae*, Numer. Math., **34**(1980), 235-246.
- [10] J. R. Cash, S. Considine, *An MEBDF code for stiff initial value problems*, ACM Trans. Math. Software, **18**(2)(1992), 142-155.
- [11] C. F. Curtiss, J. O. Hirschfelder, *Integration of stiff equations*, Proc. Natl. Acad. Sci., **38**(1952), 235-243.
- [12] G. Dahlquist, *A special stability problem for linear multistep methods*, BIT, **3**(1963), 27-43.
- [13] C. Fredebeul, *A-BDF: a generalization of the backward differentiation formulae*, SIAM J. Numer. Anal., **35**(5)(1998), 1917-1938.
- [14] N. Guzel, M. Bayram, *On the numerical solution of stiff system*, Appl. Math. Comput., **170**(2005), 230-236.
- [15] A. Hadi Alim and A. Khader *Error control policy for initial value problems with discontinuities and delays*, Kyungpook Math. J., **48**(2008), 665-684.
- [16] E. Hairer, S. P. Norsett, G. Wanner, *Solving ordinary differential equations I*, Nonstiff problems, Springer Series in Computational Mathematics, Springer 1996.
- [17] E. Hairer, G. Wanner, *Solving ordinary differential equations. II. Stiff and differential-algebraic problems*, Springer Series in Computational Mathematics, Springer 1996.
- [18] E. Hairer, G. Wanner, *Stiff differential equations solved by Radau methods*, J. Comput. Appl. Math., **111**(1999), 93-111.
- [19] A. C. Hindmarsh, *ODEPACK, A systematized collection of ODE solvers*, in: R.S. Stepleman et al., eds, Scientific Computing, North-Holland, Amsterdam, (1983), 55-64.
- [20] L. G. Ixaru, G. V. Berghe, H. D. Meyer, *Exponentially fitted variable two-step BDF algorithm for first order ODEs*, Comput. Phys. Comm., **150**(2003), 116-128.
- [21] H. Ramos, J. Vigo-Aguiar, *A fourth-order Runge-Kutta method based on BDF-type Chebyshev approximations*, J. Comput. Appl. Math., **204**(2007), 124-136.
- [22] H. Ramos, J. Vigo-Aguiar, *An almost L-stable BDF-type method for the numerical solution of stiff ODEs arising from the method of lines*, Numer. Methods Partial Differential equations, **23**(2007), 1110-1121.
- [23] H. Ramos, *A non-standard explicit integration scheme for initial value problems*, Appl. Math. Comput., **189**(2007), 710-718.
- [24] J. Vigo-Aguiar, H. Ramos, *A family of A-stable Runge-Kutta collocation methods of higher order for initial-value problems*, IMA J. Numer. Anal., **27**(2007), 798-817.
- [25] J. Vigo-Aguiar, J. Martín-Vaquero, H. Ramos, *Exponential fitting BDF-Runge-Kutta algorithms*, Comput. Phys. Comm., **178**(2008), 15-34.

- [26] H. Ramos, R. Carcia-Rubio, *Analysis of a Chebyshev-based backward differentiation formulae and reallion with Runge-Kutta collocation methods*, Inter. J. Comp. Math., to appear.
- [27] M. M. Stabrowski, *An efficient algorithm for solving stiff ordinary differential equations*, Simul. Pract. Theory, **5**(1997), 333-344.
- [28] X. Y. Wu, J. L. Xia, *Two low accuracy methods for stiff systems*, Appl. Math. Comput., **123**(2001), 141-153.