J Sci Comput (2014) 58:499–516
DOI 10.1007/s10915-013-9755-3

# Spectral Collocation Methods for Differential-Algebraic Equations with Arbitrary Index

**Can Huang · Zhimin Zhang**

Received: 3 December 2012 / Revised: 27 May 2013 / Accepted: 10 July 2013 /
Published online: 20 July 2013


**Abstract** In this paper, a symmetric Jacobi–Gauss collocation scheme is explored for both linear and nonlinear differential-algebraic equations (DAEs) of arbitrary index. After standard index reduction techniques, a type of Jacobi–Gauss collocation scheme with $N$ knots is applied to differential part whereas another type of Jacobi–Gauss collocation scheme with $N + 1$ knots is applied to algebraic part of the equation. Convergence analysis for linear DAEs is performed based upon Lebesgue constant of Lagrange interpolation and orthogonal approximation. In particular, the scheme for nonlinear DAEs can be applied to Hamiltonian systems. Numerical results are performed to demonstrate the effectiveness of the proposed method.

**Keywords** Spectral collocation method · Differential-algebraic equation · Hamiltonian systems

**Mathematics Subject Classification (2000)** Primary: 65L10 · Secondary: 60N20

## 1 Introduction

In this paper, we consider the numerical solution of linear differential-algebraic equations (DAEs) with form

C. Huang (✉)
Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA
e-mail: canhuang2007@gmail.com

Z. Zhang
Department of Mathematics, Wayne State University, Detroit, MI 48202, USA
e-mail: ag7761@wayne.edu

Z. Zhang
Beijing Computational Science Research Center, Beijing 100084, China

$$\begin{cases} E(t)\dot{x}(t) = A(t)x(t) + f(t) \quad \text{fall all } t \in \mathbb{I}, \\ Bx(\underline{t}) + Dx(\bar{t}) = r, \end{cases} \tag{1}$$

where $\mathbb{I} = [\underline{t}, \bar{t}]$, $E, A : \mathbb{I} \to R^{n\times n}$ and $f : \mathbb{I} \to R^n$ are sufficiently smooth and $B, D \in R^{d\times n}$, $d \leq n$ is the number of differential equations. We also consider nonlinear DAEs of the form

$$\begin{cases} F(t, x, \dot{x}) = 0, \\ f(x(\underline{t}), x(\bar{t})) = 0, \end{cases} \tag{2}$$

where $F$ and $f$ are smooth functions. We assume that these equations possesses a unique solution $x^\star$.

Collocation methods for linear DAEs has been extensively studied, see Radau collocation [15], projected piecewise polynomial collocation [1], perturbed collocation [3] and symmetric collocation [9]. Collocation methods for general nonlinear DAE is explored in [10]. In [9–11], by index reduction technique, one can separate algebraic equations from differential ones. Furthermore, an $h$-version Legendre–Gauss scheme for differential part with $N$ knots and $h$-version Legendre–Gauss–Lobatto scheme for algebraic part with $N+1$ knots are applied to obtain a symmetric method which guarantees consistent approximations at the mesh points. In this work, however, we apply a $p$-version Jacobi–Gauss scheme (for differential part) and Jacobi–Gauss–Lobatto scheme (for algebraic part) to solve linear or nonlinear DAEs. Based upon the scheme, a convergence result is established for linear DAEs under standard assumptions and is confirmed by our numerical examples. Furthermore, the scheme for nonlinear DAEs can be applied for numerical solution of Hamiltonian systems. We compare phase plot of some Hamiltonian systems as DAEs with those in [7], in which a spectral collocation method is designed for Hamiltonian systems as ODEs.

This paper is organized as follows: In Sect. 2, some preliminary knowledge on the theory of DAEs and Jacobi polynomials are given. In Sect. 3, we formulate our algorithm for linear DAEs and provide a convergence analysis. Two numerical experiments are also included in this section. Sect. 4 is devoted to the algorithm and numerical experiments for nonlinear DAEs. We then give some conclusions in Sect. 5.

Throughout the paper, $C$ stands for a generic constant that is independent of the number of collocation points $N$ but may depend on the length of time interval $\bar{t} - \underline{t}$ and the dimension $n$.

## 2 Preliminaries

Application of index reduction techniques on the linear DAEs (1) and nonlinear DAEs (2), one obtain

$$\hat{E}(t)\dot{x}(t) = \hat{A}(t)x(t) + \hat{f}(t), \quad t \in [\underline{t}, \bar{t}] \tag{3}$$

with

$$\hat{E} = [\hat{E}_1, 0]^T, \ \hat{A} = [\hat{A}_1, \hat{A}_2]^T, \ \hat{f} = [\hat{f}_1, \hat{f}_2]^T. \tag{4}$$

and

$$\begin{cases} \hat{F}_1(t, x, \dot{x}) = 0, \\ \hat{F}_2(t, x) = 0, \quad t \in [\underline{t}, \bar{t}]. \end{cases} \tag{5}$$

Since solutions of (1) and (3), (2) and (5) are equivalent respectively, we assume that both the linear and nonlinear DAE that we work on is of the index reduced form and without causing any confusion, we keep the same notation as those in (1) and (2). We denote $a$ and $d$ the size of *algebraic and differential part* of (3) and (5). Interested readers are referred to [9,10] for more details. For the linear DAEs, we have

**Lemma 2.1** ([9]) *For $E, A \in C^k(\mathbb{I}, R^{n\times n})$, there exist point-wise nonsingular $P \in C^{k-1}(\mathbb{I}, R^{n\times n})$, $Q \in C^k(\mathbb{I}, R^{n\times n})$ such that*

$$PEQ = \begin{bmatrix} I_d & 0 \\ 0 & 0 \end{bmatrix}, \; PAQ - PE\dot{Q} = \begin{bmatrix} 0 & 0 \\ 0 & I_a \end{bmatrix}. \tag{6}$$

*In particular, $P$ has the special structure*

$$P = \begin{bmatrix} P_{11} & P_{12} \\ 0 & P_{22} \end{bmatrix}, \quad \text{with} \quad P_{11}(t) \in R^{d\times d}, \, P_{12} \in R^{d\times a}, \, P_{22}(t) \in R^{a\times a}. \tag{7}$$

*In addition, if $f \in C^{k-1}(\mathbb{I}, R^n)$, then $x \in C^{k-1}(\mathbb{I}, R^n)$ for every solution $x$ of (1).*

Next, we introduce Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$, which plays a center role throughout the paper. It is well-known that $P_n^{(\alpha,\beta)}(x)$ is a class of orthogonal polynomials associate with weight $\omega(x) = (1-x)^\alpha(1+x)^\beta$, $\alpha, \beta > -1$. Under the normalization $P_k^{(\alpha,\beta)}(1) = \binom{k+\alpha}{k}$, one has the expression,

$$P_k^{(\alpha,\beta)}(x) = \frac{1}{2^k}\sum_{l=0}^{k}\binom{k+\alpha}{k-l}\binom{k+\beta}{l}(x-1)^l(x+1)^{k-l}.$$

Jacobi polynomials satisfy the three-term recursive relations:

$$P_0^{(\alpha,\beta)}(x) = 1, \; P_1^{(\alpha,\beta)}(x) = \frac{1}{2}[(\alpha-\beta) + (\alpha+\beta+2)x],$$
$$a_{1,k}P_{k+1}^{(\alpha,\beta)}(x) = a_{2,k}P_k^{(\alpha,\beta)}(x) - a_{3,k}P_{k-1}^{(\alpha,\beta)}(x),$$

where

$$\begin{aligned} a_{1,k} &= 2(k+1)(k+\alpha+\beta+1)(2k+\alpha+\beta), \\ a_{2,k} &= (2k+\alpha+\beta+1)\left(\alpha^2-\beta^2\right) \\ &\quad + x\Gamma(2k+\alpha+\beta+3)/\Gamma(2k+\alpha+\beta), \\ a_{3,k} &= 2(k+\alpha)(k+\beta)(2k+\alpha+\beta+2). \end{aligned}$$

Especially, Legendre polynomial is exactly a class of Jacobi polynomial with $\alpha = \beta = 0$ and Chebyshev polynomials of the first kind is another class with $\alpha = \beta = -1/2$ up to a constant. It is clear that $P_n^{(\alpha,\beta)}(x)$ forms a complete orthogonal system associate with the norm

$$\|v\|_{\omega^{\alpha,\beta}} = \left(\int_{-1}^{1}|v(x)|^2\omega^{\alpha,\beta}(x)dx\right)^{1/2}.$$

Denote $\omega(x) = \omega^{-1/2,-1/2}(x)$ the Chebyshev weight function and introduce a semi-norm for the simplicity of analysis

$$|v|_{H_\omega^{m,N}(-1,1)} = \left( \sum_{k=\min(m,N+1)}^{m} |\partial_x^k v|^2_{L^2_\omega(-1,1)} \right)^{1/2}.$$

Let $I_N^c$ denote the interpolation operator on Chebyshev points, then we have [4]

$$\|u - I_N^c u\|_\infty \le C N^{1/2-m} |u|_{H_\omega^{m,N}(-1,1)}. \tag{8}$$

Let $h(t) \in C^k(\mathbb{I}, R^n)$ and write $u(x) = h\big[\frac{\bar{t}-\underline{t}}{2}(1+x) + \frac{\bar{t}+\underline{t}}{2}\big]$, $x \in [-1, 1]$. Let $P_N^{(\alpha,\beta)}$ be the Jacobi polynomial with degree $N$ and $I_N u \in P_N[-1, 1]$ interpolate $u$ at $N+1$ Jacobi–Gauss points, Jacobi–Gauss–Radau points or Jacobi–Gauss–Lobatto points, then we have the following lemma on the Legesgue constant, see [12].

**Lemma 2.2** *Let $\{l_j(x)\}_{j=0}^N$ be the Lagrange interpolation polynomials at Jacobi–Gauss points, then*

$$\begin{aligned} \|I_N^{\alpha,\beta}\|_\infty &:= \max_{x\in[-1,1]} \sum_{j=0}^{N} |l_j(x)| \\ &= \begin{cases} \mathcal{O}(\log N), & -1 < \alpha, \beta \le -\frac{1}{2}, \\ \mathcal{O}(N^{\gamma+1/2}), \ \gamma = \max(\alpha,\beta), & \text{otherwise.} \end{cases} \end{aligned} \tag{9}$$

The remainder of the interpolation is

$$u(x) - I_N u(x) = u[x_0, x_1, \ldots, x_N, x] v(x),$$

where $v(x) = (x-x_0)(x-x_1)\ldots(x-x_N)$ and $u[x_0, x_1, \ldots, x_N, x]$ is the divided difference of $u$. Hence,

$$u(x) - I_N u(x) = \frac{u[x_0, x_1, \ldots, x_N, x]}{L_{N+1}} P_{N+1}^{(\alpha,\beta)}(x),$$

where $L_{N+1} = \frac{\Gamma(2N+\alpha+\beta+3)}{2^{N+1}\Gamma(\alpha+\beta+N+2)(N+1)!} \approx \frac{2^{\alpha+\beta}}{\sqrt{\pi N}} 2^{N+1}$ is the leading coefficient of $P_{N+1}^{(\alpha,\beta)}(x)$. Moreover, if $u$ is sufficiently smooth, then the divided difference

$$u[x_0, x_1, \ldots, x_N, x] = \frac{u^{(N+1)}(\xi_x)}{(N+1)!}, \quad \xi_x \in (-1, 1),$$

which implies

$$\|u(x) - I_N u(x)\|_{L^\infty} \le C \frac{\sqrt{N+1}}{2^{N+1}(N+1)!} \|u^{(N+1)}(\xi_x)\|_{L^\infty}. \tag{10}$$

For the sake of analysis, we define the Lagrange interpolation at $(N+1)$ interpolation points, i.e.,

$$I_N u(x_i) = \sum_{i=0}^{N} u(x_i) l_i(x), \tag{11}$$

where $l_i(x)$ is the Lagrange interpolation basis function at these points.

It is obvious that for all $t \in [\underline{t}, \bar{t}]$ and $x \in [-1, 1]$

$$h(t) = u(x), h_N(t) = u_N(x),$$

Therefore,

$$(h - h_N)(t) = (u - u_N)(x) := e(x). \tag{12}$$

## 3 Linear Differential-Algebraic Equations

This section is devoted to the symmetric Jacobi collocation methods for linear DAEs and its convergence analysis.

Firstly, we transform the problem (3) onto an equivalent problem on interval $[-1, 1]$,

$$\begin{aligned} \frac{2}{b-a}E(\xi)\dot{x}(\xi) &= A_1(\xi)x(\xi) + f_1(\xi) \\ 0 &= A_2(\xi)x(\xi) + f_2(\xi),\ \xi \in [-1, 1]. \end{aligned} \tag{13}$$

Here, for simplicity of notation, the hats are omitted and subindex of $E$ is suppressed.

### 3.1 Formulation of Algorithm

On the interval $[-1, 1]$, let $\rho_j$'s be the zeros of Jacobi polynomial $P_N^{(\alpha,\beta)}(x)$, and $\sigma_i$'s be the associated Jacobi–Gauss–Lobatto points.

$$-1 \le \rho_0 < \cdots < \rho_{N-1} \le 1,\ -1 = \sigma_0 < \cdots < \sigma_N = 1. \tag{14}$$

Then, we have the interlacing property [6,13],

$$-1 = \sigma_0 < \rho_0 < \sigma_1 < \cdots < \rho_{N-1} < \sigma_N = 1. \tag{15}$$

In order to fix the idea, we use $T_k$, the Chebyshev polynomial of the first kind in algorithms. Nevertheless, our method is applicable to all Jacobi polynomials. Approximate the true solution of (13) by

$$x_N^m(\xi) = \sum_{k=0}^{N} C_k^m T_k(\xi),\ m = 1, \ldots, n. \tag{16}$$

The Jacobi coefficients $C_k^m \in R^{(N+1)\times 1}$ is determined by the following set of conditions:

$$\begin{aligned} r &= Bx_N(-1) + Dx_N(1), \\ \frac{2}{b-a}E(\rho_i)\dot{x}_N(\rho_i) &= A_1(\rho_i)x_N(\rho_i) + f_1(\rho_i), \quad i = 0, \ldots, N-1, \\ 0 &= A_2(\sigma_j)x_N(\sigma_j) + f_2(\sigma_j), \quad j = 0, \ldots, N. \end{aligned} \tag{17}$$

Note that the above system has exactly $(N+1)n$ unknowns and $d+Nd+(N+1)a = (N+1)n$ equations.

By the fact that $T_k'(\xi) = kU_{k-1}(\xi)$, where $U_k(\xi)$ is the Chebyshev polynomial of the second kind and denoting for all $i = 1, \ldots, n,\ j = 1, \ldots, a$ and $k = 1, \ldots, d$

$$\begin{aligned} &W_1 = \left[1, -1, \ldots, (-1)^N\right], \quad W_2 = [1, 1, \ldots, 1^N], \\ &f_{1,k} = \left[f_{1,k}(\rho_0), \ldots, f_{1,k}(\rho_{N-1})\right],\ f_{2,j} = \left[f_{2,j}(\sigma_0), \ldots, f_{2,j}(\sigma_N)\right], \\ &f_1 = \left[f_{1,1}, \ldots, f_{1,d}\right]^T, \quad f_2 = \left[f_{2,1}, \ldots, f_{2,a}\right]^T. \end{aligned}$$

$$U = \begin{bmatrix} 0 & 1 & \cdots & NU_{N-1}(\rho_0) \\ 0 & 1 & \cdots & NU_{N-1}(\rho_1) \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & NU_{N-1}(\rho_{N-1}) \end{bmatrix},$$

$$T = \begin{bmatrix} 1 & \rho_0 & \cdots & T_N(\rho_0) \\ 1 & \rho_1 & \cdots & T_N(\rho_1) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \rho_{N-1} & \cdots & T_N(\rho_{N-1}) \end{bmatrix}, \quad \tilde{T} = \begin{bmatrix} 1 & \sigma_0 & \cdots & T_N(\sigma_0) \\ 1 & \sigma_1 & \cdots & T_N(\sigma_1) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \sigma_N & \cdots & T_N(\sigma_N) \end{bmatrix}, \tag{18}$$

we have a system $AC = F$, where

$$C = \left[C_1^1, \ldots, C_N^1, \ldots, C_1^n, \ldots, C_N^n\right]^T,$$

$$A = \left[\begin{array}{ccc} B \bigotimes W_1 + D \bigotimes W_2 & & \\ \hline \frac{2}{b-a}E_{11}U - A_{1,11}T & \cdots & \frac{2}{b-a}E_{1n}U - A_{1,1n}T \\ \vdots & \ddots & \vdots \\ \frac{2}{b-a}E_{d1}U - A_{1,d1}T & \cdots & \frac{2}{b-a}E_{dn}U - A_{1,dn}T \\ \hline -A_{2,11}\tilde{T} & \cdots & -A_{2,1n}\tilde{T} \\ \vdots & \ddots & \vdots \\ -A_{2,a1}\tilde{T} & \cdots & -A_{2,an}\tilde{T} \end{array}\right], \quad F = \left[\begin{array}{c} r \\ \hline f_1 \\ \\ \hline f_2 \\ \\ \end{array}\right], \tag{19}$$

and $\bigotimes$ denotes the Kronecker product. Once $C$ is determined, we plug it back to (16) to obtain $x_N$.

### 3.2 Convergence Results

We try to derive an error estimate in the $L^\infty$-norm of the collocation scheme (17).

Since $N$-point Jacobi–Gauss–Lobatto points with parameters $(\alpha, \beta)$ are zeros of $P_{N-2}^{(\alpha+1,\beta+1)}(t)$ ($\pm 1$ excluded) [14], throughout the paper we only need to consider the case when both $\rho_i$'s and $\sigma_j$'s are zeros of certain type of Jacobi polynomials. Moreover, it is well-known that both $\rho_j$'s and $\sigma_j$'s are dense in $[-1, 1]$ as $N \to \infty$. Note that the distance between any two consecutive zeros of any type of Jacobi polynomials is bound by $\frac{\sqrt{1-t^2}}{N}$, $t \in [-1, 1]$ up to a constant independent of $N$ and $t$ [8]. We derive that $\rho_i - \sigma_i = \mathbb{O}(N^{-1})$, $i = 1, \ldots, N-1$.

To obtain the solvability of our algorithm, let us rewrite our algorithm in its equivalent form. For each component of $x^m$, its approximation

$$x_N^m = \sum_{k=0}^{N} C_k^m T_k(\xi) = \sum_{k=0}^{N} x_{k,m}^N l_k(\xi), \tag{20}$$

where $l_k(\xi)$ is the Lagrange interpolation polynomial on point set $\sigma_j$'s. Denote $v_{ki} = l_i'(\rho_k)$, $u_{ki} = l_i(\rho_k)$, $E_i = E(\rho_i)$, $A_{1i} = A_1(\rho_i)$, $A_{2j} = A_2(\sigma_j)$, and

$$x_N = \left[x_{0,1}^N, x_{0,2}^N, \ldots, x_{0,n}^N, \ldots, x_{N,1}^N, \ldots, x_{N,n}^N\right]^T,$$

for $k = 1, \ldots, N$, $i = 1, \ldots, N$. Plugging (20) into (13) yields a system $A' x_N = F'$, where

$$A' = \left[\begin{array}{ccc|c} \frac{2v_{00}}{b-a}E_0 - u_{00}A_{10} & \cdots & \cdots & \frac{2v_{0N}}{b-a}E_0 - u_{0N}A_{10} \\ -A_{20} & \cdots & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots \\ \frac{2v_{N-1,0}}{b-a}E_{N-1} & \cdots & \frac{2v_{N-1,N-1}}{b-a}E_{N-1} & \frac{2v_{N-1,N}}{b-a}E_{N-1} \\ -u_{N-1,0}A_{1,N-1} & \cdots & -u_{N-1,N-1}A_{1,N-1} & -u_{N-1,N}A_{1,N-1} \\ 0 & \cdots & -A_{2,N-1} & 0 \\ \hline C & & & D \\ & & & -A_2(1) \end{array}\right] \tag{21}$$

$F' = [f_1(\rho_0) \quad f_2(\sigma_0) \ldots f_1(\rho_{N-1}) \quad f_2(\sigma_{N-1}) \quad r \quad f_2(1)]^T$. Here, a missing condition $-A_2(1)x_N(1) = f_2(1)$ is used. Hence, if $A'$ is regular, our collocation algorithm is stable. To reorder rows and columns, we define

$$U_N = \begin{bmatrix} I_d & 0 & & & 0 & 0 & & & \\ 0 & 0 & & & I_a & 0 & & & \\ & & I_d & 0 & & & 0 & 0 & \\ & & 0 & 0 & & & I_a & 0 & \\ & & & \ddots & & & & \ddots & \\ & & & & I_d & & & & 0 \\ & & & & 0 & & & & I_a \\ & & & & & & & I_d & 0 \\ & & & & & & & 0 & I_a \end{bmatrix} \in R^{(N+1)n \times (N+1)n}. \tag{22}$$

*Remark 1* As is well-known, Lagrange interpolation is not stable for large $N$. Moreover, it requires a large computational cost to find the base functions. Hence, the equivalent algorithm is exhibited only for a theoretical purpose.

**Lemma 3.1** *For $N$ sufficiently large,*

$$l_j(\rho_i) = \begin{cases} \mathbb{O}(N^{-1}), & i \neq j \quad \text{and} \quad i \neq j-1, \\ \mathbb{O}(1), & \text{otherwise}. \end{cases}$$

*Proof* Let $\delta$ be a fixed number such that $\delta < 1 - |\rho_i|$. By the interlacing property, we can choose $\delta$ so small that $B(\sigma_j, \delta) \cap \rho = \{\rho_{j-1}, \rho_j\}$. From [15] (pp. 336), if $\sigma_j - \rho_i > \delta$,

$$\begin{aligned} l_j(\rho_i) &= \mathbb{O}\big(N^{-1/2}\big)|P_{N+1}^{(\alpha,\beta)\prime}(\sigma_j)|^{-1}, \\ &= \mathbb{O}\big(N^{-1/2}\big)\left(j^{\alpha+3/2}N^{-\alpha-2} + j^{\beta+3/2}N^{-\beta-2}\right), \\ &\leq \mathbb{O}\big(N^{-1}\big), \end{aligned} \tag{23}$$

otherwise,

$$l_j(\rho_i) = \mathbb{O}\big(N^{-1/2}\big)\frac{P_{N+1}^{(\alpha,\beta)}(\sigma_j) - P_{N+1}^{(\alpha,\beta)}(\rho_i)}{\sigma_j - \rho_i} = \mathbb{O}(1). \tag{24}$$

□

As an illustration, we plot an $l_{12}(x)$ for $N = 25$ and $(\alpha, \beta) = (1/2, 2/3)$, see Fig. 1. It is clear that $l_{12}(\rho_j)$ has the largest modulus at $\rho_{11}$ and $\rho_{12}$.

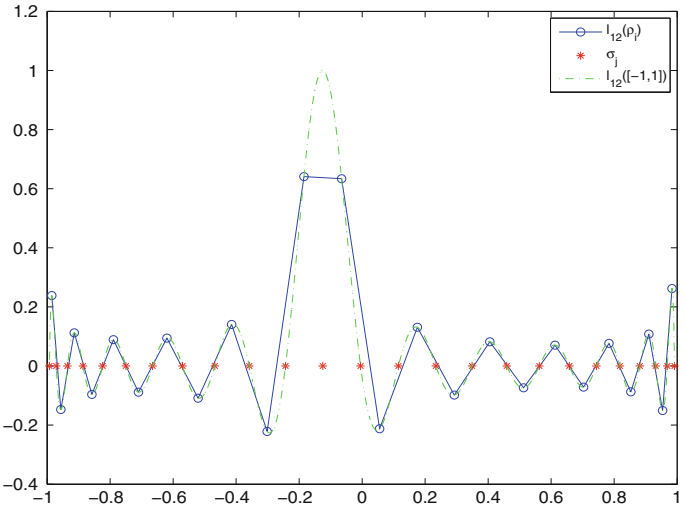


**Fig. 1** A plot of $l_{12}(x)$ for $N = 25$ and $(\alpha, \beta) = (1/2, 2/3)$

Next, we prove the invertibility of $A'$ by multiplying it from the left and from the right, respectively, by

$$T_P = \operatorname{diag}\left(\operatorname{diag}\begin{pmatrix} P_{11}(\rho_i) & P_{12}(\sigma_i) \\ 0 & P_{22}(\sigma_i) \end{pmatrix}_{i=0,\ldots,N-1}, \begin{bmatrix} I & 0 \\ 0 & P_{22}(1) \end{bmatrix}\right),$$
$$T_Q = \operatorname{diag}\Big(Q(\sigma_0), \ldots, Q(\sigma_{N-1}), Q(1)\Big). \tag{25}$$

**Lemma 3.2** *Assume $A_2$, $Q \in C^2$. Then $A'$ is invertible for sufficiently large $N$.*

*Proof* Since $l_i(x) = \frac{P_{N+1}(x)}{P'_{N+1}(\sigma_i)(x-\sigma_i)}$, after a simple calculation, we have

$$v_{ij} = l_i'(\rho_j) = -\frac{l_i(\rho_j)}{|\rho_j - \sigma_i|}. \tag{26}$$

Clearly, from Lemma (3.1),

$$v_{ij} = \begin{cases} \mathbb{O}(N), & j = i, \text{ or } i-1; \\ \mathbb{O}(1), & j \neq i, \text{ or } i-1 \text{ and } |\sigma_j - \rho_i| = \mathbb{O}(N^{-1}); \\ o(1), & \text{otherwise.} \end{cases} \tag{27}$$

Hence, for $j \neq i$ or $i-1$,

$$\begin{aligned} I_1 &= [P_{11}(\rho_i) \quad P_{12}(\sigma_i)] \begin{bmatrix} \frac{2v_{ij}}{b-a} E_i - u_{ij} A_{1i} \\ 0 \end{bmatrix} Q(\sigma_j) \\ &= \frac{2v_{ij}}{b-a}(P_{11}E)(\rho_i)Q(\sigma_j) - u_{ij}(P_{11}A_1)(\rho_i)Q(\sigma_j) \\ &= \frac{2v_{ij}}{b-a}(P_{11}EQ)(\rho_i) - u_{ij}(P_{11}A_1)(\rho_i)Q(\sigma_j) + e_1, \end{aligned} \tag{28}$$

where $e_1 = \frac{2v_{ij}}{b-a}(P_{11}E)(\rho_i)(Q(\sigma_j) - Q(\rho_i))$. Apply the smoothness assumption of $P$, $E$ and $Q$,

$$\begin{aligned} e_1 &= \frac{2l_i'(\rho_j)}{b-a}(P_{11}E)(\rho_i)\dot{Q}(\xi)(\rho_j - \sigma_i), \\ &= -l_i(\rho_j)(P_{11}E)(\rho_i)\dot{Q}(\xi), \\ &= \mathbb{O}(1)l_i(\rho_j). \end{aligned} \tag{29}$$

However, in this case, $l_i(\rho_j)/l_i'(\rho_j) \approx 0$ by the L'Hopital's rule. Therefore, (29) and the definition of $u_{ij}$ yield

$$I_1 = \frac{2v_{ij}}{b-a}[I\ 0] + o(v_{ij}). \tag{30}$$

Similarly, if $j = i-1$, $I_1 = \frac{2v_{ij}}{b-a}[I\ 0] + \mathbb{O}(1)$.

If $j = i$,

$$\begin{aligned} I_2 &= [P_{11}(\rho_i) \quad P_{12}(\sigma_i)] \begin{bmatrix} \frac{2v_{ij}}{b-a}E_i - u_{ij}A_{1i} \\ -A_{2i} \end{bmatrix} Q(\sigma_i) \\ &= \frac{2v_{ii}}{b-a}(P_{11}E)(\rho_i)Q(\sigma_i) - u_{ii}(P_{11}A_1)(\rho_i)Q(\sigma_i) - (P_{12}A_2Q)(\sigma_i) \\ &= \frac{2v_{ii}}{b-a}(P_{11}EQ)(\rho_i) - u_{ii}(P_{11}A_1)(\rho_i)Q(\sigma_i) - (P_{12}A_2Q)(\sigma_i) + e_2, \\ &= \frac{2v_{ii}}{b-a}[I\ 0] + \mathbb{O}(1). \end{aligned} \tag{31}$$

At last,

$$\begin{bmatrix} I & 0 \\ 0 & P_{22}(1) \end{bmatrix} \begin{bmatrix} B \\ 0 \end{bmatrix} Q(1) = \begin{bmatrix} B_{11} & B_{12} \\ 0 & 0 \end{bmatrix},$$
$$\begin{bmatrix} I & 0 \\ 0 & P_{22}(1) \end{bmatrix} \begin{bmatrix} D \\ -A_2(1) \end{bmatrix} Q(1) = \begin{bmatrix} D_{11} & D_{12} \\ 0 & -I \end{bmatrix}.$$

Hence, we have

$$M = T_P A' T_Q =$$
$$\left[\begin{array}{ccc|c} \frac{2v_{00}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(1) & \cdots & \frac{2v_{0,N-1}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(\frac{1}{N}) & \frac{2v_{0N}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(\frac{1}{N}) \\ -I_{a\times a} & \cdots & 0 & 0 \\ \ddots & & \ddots & \ddots \\ \frac{2v_{N-1,0}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(\frac{1}{N}) & \cdots & \frac{2v_{N-1,N-1}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(1) & \frac{2v_{N-1,N}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(1) \\ 0 & \cdots & -I_{a\times a} & 0 \\ \hline B_{11} \quad B_{12} & & & D_{11} \quad D_{12} \\ 0 \quad\ 0 & & & 0 \quad -I_{a\times a} \end{array}\right] \tag{32}$$

Rewrite the above matrix as

$$M = \left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right].$$

Clearly, if $D_{11}^{-1}$ exists,

$$\tilde{D}^{-1} = \begin{bmatrix} D_{11}^{-1} & D_{11}^{-1}D_{12} \\ 0 & -I \end{bmatrix}.$$

Note that

$$\begin{bmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & \tilde{D} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\tilde{D}^{-1}\tilde{C} & I \end{bmatrix} = \begin{bmatrix} \tilde{A} - \tilde{B}\tilde{D}^{-1}\tilde{C} & \tilde{B} \\ 0 & \tilde{D} \end{bmatrix}.$$

Therefore, the invertibility of $\tilde{A} - \tilde{B}\tilde{D}^{-1}\tilde{C}$ implies the invertibility of $A'$. By a simple matrix manipulation,

$$\tilde{A} - \tilde{B}\tilde{D}^{-1}\tilde{C} = \begin{bmatrix} \frac{2v_{00}}{b-a}[I_{d\times d}\ 0] - \frac{2v_{0N}}{b-a}D_{11}^{-1}B_{11} + \mathbb{O}(1) & \cdots & \frac{2v_{0,N-1}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(\frac{1}{N}) \\ -I_{a\times a} & \cdots & 0 \\ \ddots & \ddots & \\ \frac{2v_{N-1,0}}{b-a}[I_{d\times d}\ 0] - \frac{2v_{N-1,N}}{b-a}D_{11}^{-1}B_{11} + \mathbb{O}(1) & \cdots & \frac{2v_{N-1,N-1}}{b-a}[I_{d\times d}\ 0] + \mathbb{O}(1) \\ 0 & \cdots & -I_{a\times a} \end{bmatrix}$$

It is clear that the matrix is diagonal dominant, hence invertible. □

This result indicates that our algorithm is well-posed. In the following, we find the error estimates for the algorithm.

**Theorem 3.3** *Let $x(\xi)$ be the true solution of transformed DAE (13) and $x_N(\xi)$ be the approximation given by (17) based upon our collocation method. If the given data $E(\xi), A(\xi) \in C^{m+1}([-1,1], R^{n\times n})$, and $f \in C^m([-1,1], R^n)$, then for sufficiently large $N$,*

$$\|x - x_N\|_\infty \le C \begin{cases} (\log N)N^{3/2-m}, & -1 < \alpha, \beta < -1/2, \\ N^{\gamma+2-m}, & \gamma = \max(\alpha,\beta), \quad \textit{otherwise}. \end{cases} \tag{33}$$

*Proof* Let $I_N$ be the interpolation operator on $\sigma_0, \ldots, \sigma_N$ and $\tilde{I}_{N-1}$ be the one on $\rho_0, \ldots, \rho_{N-1}$. Clearly, the true solution satisfies

$$\begin{aligned} r &= Bx(-1) + Dx(1), \\ \frac{2}{b-a}E(\rho_i)\dot{x}(\rho_i) &= A_1(\rho_i)x(\rho_i) + f_1(\rho_i), \quad i = 0, \ldots, N-1, \\ 0 &= A_2(\sigma_j)x(\sigma_j) + f_2(\sigma_j), \quad j = 0, \ldots, N. \end{aligned} \tag{34}$$

Multiplying both sides of the first and the third equation of (34) by $l_j(\xi)$, the second equation by $\tilde{l}_i(\xi)$, respectively, and summing over $i$, $j$ yield

$$\begin{aligned} r &= BI_Nx(-1) + DI_Nx(1), \\ \frac{2}{b-a}\tilde{I}_{N-1}(E\dot{x}) &= \tilde{I}_{N-1}(A_1x) + \tilde{I}_{N-1}f_1, \\ 0 &= I_N(A_2x) + I_Nf_2. \end{aligned} \tag{35}$$

We follow the same fashion for (17) to obtain a system for $x_N$, then subtract the system obtained from (35),

$$\begin{aligned} 0 &= BI_N(x(-1) - x_N(-1)) + DI_N(x(1) - x_N(1)), \\ 0 &= \frac{2}{b-a}\tilde{I}_{N-1}(E_1\dot{x} - E\dot{x}_N) - \tilde{I}_{N-1}(A_1x - A_1x_N), \\ 0 &= I_N(A_2x - A_2x_N). \end{aligned} \tag{36}$$

Recall the definition of $I_N^c$. Since $I_N A_2$ and $I_N x$ are polynomials, We have the following estimate from (5.5.28) of [4],

$$\begin{aligned}\|I_N(A_2x) - (I_NA_2)(I_Nx)\|_\infty &= \|I_N^c(A_2x) - (I_N^cA_2)(I_N^cx)\|_\infty \\ &= \|I_N^c((I_N^cA_2)(I_N^cx)) - (I_N^cA_2)(I_N^cx)\|_\infty \\ &\le CN^{1/2-2m}|I_N^cA_2\cdot I_N^cx|_{H_\omega^{m,N}(-1,1)}. \end{aligned} \tag{37}$$

Similar estimation can be obtained for $\|I_N(A_2x_N) - (I_NA_2)(I_Nx_N)\|_\infty$, $\|\tilde{I}_{N-1}(E\dot{x}) - (\tilde{I}_{N-1}E)(\tilde{I}_{N-1}\dot{x})\|_\infty$, $\|\tilde{I}_{N-1}(E\dot{x_N}) - (\tilde{I}_{N-1}E)(\tilde{I}_{N-1}\dot{x_N})\|_\infty$, $\|\tilde{I}_{N-1}(A_1x) - (\tilde{I}_{N-1}A_1)(\tilde{I}_{N-1}x)\|_\infty$ and $\|\tilde{I}_{N-1}(A_1x_N) - (\tilde{I}_{N-1}A_1)(\tilde{I}_{N-1}x_N)\|_\infty$. However, all these terms are high-order terms. Combine all these estimates, (36) can be written as

$$\begin{cases} 0 = BI_N(x(-1) - x_N(-1)) + DI_N(x(1) - x_N(1)), \\ \mathcal{O}(N^{1/2-2m}) = \frac{2}{b-a}(I_{N-1}E_1)[\tilde{I}_{N-1}\dot{x} - \tilde{I}_{N-1}\dot{x}_N] - (\tilde{I}_{N-1}A_1)[\tilde{I}_{N-1}x - \tilde{I}_{N-1}x_N], \\ \mathcal{O}(N^{1/2-2m}) = (I_NA_2)(I_Nx - I_Nx_N). \end{cases} \tag{38}$$

Also, by simple calculation,

$$\begin{aligned}\|\tilde{I}_{N-1}\dot{x} - \tilde{I}_{N-1}\dot{x}_N - (\dot{I_N}x) + (I_N\dot{x}_N)\|_\infty &= \|\tilde{I}_{N-1}\dot{x} - (\dot{I_N}x)\|_\infty \\ &\le CN^{3/2-m}. \end{aligned} \tag{39}$$

We obtain

$$\begin{cases} 0 = BI_N(x(-1) - x_N(-1)) + DI_N(x(1) - x_N(1)), \\ \mathcal{O}(N^{3/2-m}\|I_N\|_\infty) = \frac{2}{b-a}(I_NE_1)[(\dot{I_N}x) - \dot{x}_N] - (I_NA_1)[I_Nx - x_N], \\ \mathcal{O}(N^{1/2-2m}) = (I_NA_2)(I_Nx - x_N). \end{cases}$$

Writing the right hand side of the last two groups of equations as $w$ and applying the transformation in Lemma 2.1 yields that

$$\begin{bmatrix} I_d & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & I_a \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \tag{40}$$

$$[B_1\ B_2]\begin{bmatrix} y_1(-1) \\ y_2(-1) \end{bmatrix} + [D_1\ D_2]\begin{bmatrix} y_1(1) \\ y_2(1) \end{bmatrix} = 0, \tag{41}$$

where

$$I_Nx - x_N = Q\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad Pw = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \quad BQ(-1) = [B_1\ B_2], \quad DQ(1) = [D_1,\ D_2].$$

Matrix functions $P$ and $Q$ are given in Lemma 2.1. The differences $P(E - I_NE)Q = \mathcal{O}(N^{-m-1})$ and $P(A - I_NA)Q = \mathcal{O}(N^{-m-1})$ are absorbed into functions $g$. Clearly,

$$y_1(\xi) = y_1(-1) + \int_{-1}^{\xi} g_1(s)ds,\ \ y_2(\xi) = -g_2(\xi). \tag{42}$$

The equation is equivalent to [11]

$$\begin{cases} \dot{y}_1 = g_1(\xi), \\ y_1(-1) = -(B_1 + D_1)^{-1}\int_{-1}^{1} g_1(s)ds. \end{cases}$$
$$y_2(\xi) = -g_2(\xi). \tag{43}$$

Hence, it is enough for us to consider the corresponding initial value condition for our problem. Obviously, $\|y\| \leq C\|g\|$, which implies

$$\|I_N x - x_N\|_\infty \leq C\|I_N\|_\infty N^{3/2-m}. \tag{44}$$

Therefore,

$$\begin{aligned}\|x - x_N\|_\infty &\leq \|x - I_N x\|_\infty + \|I_N x - x_N\|_\infty \\ &\leq C \begin{cases} (\log N) N^{3/2-m}, & -1 < \alpha, \beta < -1/2, \\ N^{\gamma+2-m}, & \gamma = \max(\alpha, \beta), \quad \text{otherwise.} \end{cases}\end{aligned} \tag{45}$$

□

### 3.3 Numerical Experiments

We present two examples here to illustrate the effectiveness of our algorithm. Both examples are from [9]. Unconfined to the type of points in our theory, we use three different types of collocation points, which is presented in Table 1.

*Example 1* Consider the DAE

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & -t & 0 \\ -1 & t & 1 \end{bmatrix} \dot{x} = \begin{bmatrix} -1 & t & 0 \\ 0 & 0 & 0 \\ 0 & t^2 & 1 \end{bmatrix} x + \begin{bmatrix} e^{t/2} \\ 0 \\ 0 \end{bmatrix}, \quad t \in [-5, 0]$$
$$[1\ 7\ 0]\, x(-5) + [0\ 4\ 1]\, x(0) = 6.$$

This is an index-two problem with solution

$$x(t) = e^{t/2}\left(1 - \frac{t}{2}, -\frac{1}{2}, t^2 + 4t + 8\right).$$

Since the solution is smooth, from Table 2 and Fig. 2, we actually observe a super geometric rate of convergence ($\mathcal{O}\left(\frac{\gamma}{N}\right)^{\sigma N}$), where $\gamma, \sigma$ are some constants [16].

**Table 1** Three sets of collocation points

| | $\rho$ | $\sigma$ |
|---|---|---|
| Set 1 | Gauss–Legendre points | Gauss–Legendre–Lobatto points |
| Set 2 | Gauss–Legendre–Lobatto points | Gauss–Legendre–Radau points |
| Set 3 | Gauss–Legendre points | Gauss–Legendre points |

**Table 2** Errors for Example 1

| Number of points | Set 1 | Set 2 | Set 3 |
|---|---|---|---|
| 5 | 5.7025e−2 | 8.2836e−2 | 5.9604e−2 |
| 10 | 9.7657e−6 | 2.2007e−5 | 9.9000e−6 |
| 15 | 1.8526e−10 | 5.3087e−10 | 1.8701e−10 |
| 20 | 6.9944e−15 | 7.9936e−15 | 1.3323e−14 |

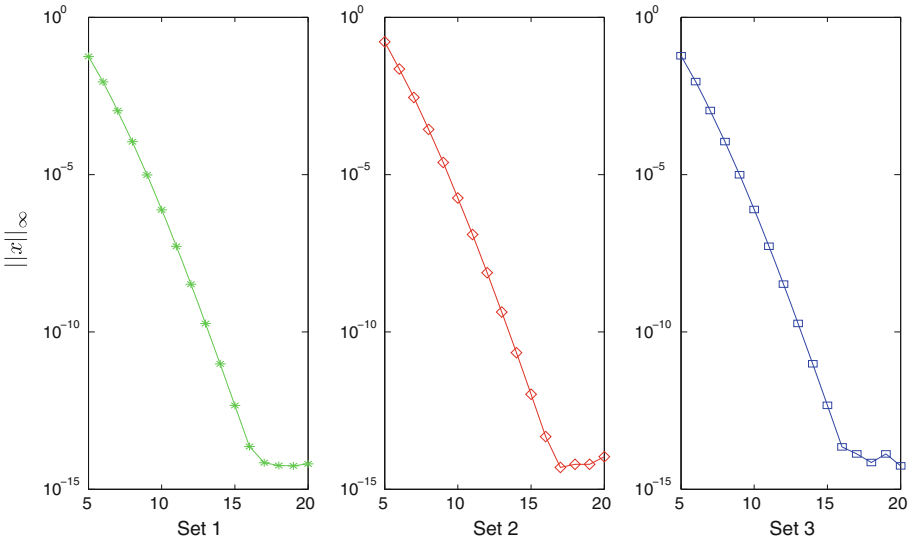


**Fig. 2** Numerical errors for Example 1

*Example 2* Consider the problem of form (1) with

$$
\begin{aligned}
E(t) &= \begin{bmatrix} 1 & t & 0 \\ t & 1 & -t \\ p(t)-2 & -t(p(t)-2) & 0 \end{bmatrix}, \\
A(t) &= \begin{bmatrix} \kappa - \frac{1}{2-t} & \frac{2}{2-t} - \kappa t & (2-t)\kappa \\ \frac{\kappa-1}{2-t} - t - 1 & -t\frac{\kappa-1}{2-t} - 1 & t + \kappa - \frac{\kappa p(t)}{2+t} \\ \kappa t(t^2-3) - \frac{p(t)-2}{2-t} & 2\frac{p(t)-2}{2-t} - 4\kappa & \kappa(p(t)(2-t) - t^3 + 6t - 4) \end{bmatrix}, \\
f(t) &= \begin{bmatrix} \frac{3-t}{2-t} \\ 2 + \frac{(\kappa+2)p(t)+\dot{p}(t)}{t^2-4} - 2\frac{tp(t)}{(t^2-4)^2} \\ (p(t)-2)\frac{3-t}{2-t} - \kappa(t^2+t-2) \end{bmatrix} e^t,
\end{aligned} \tag{46}
$$

where $t \in [0, 1]$, parameter $\kappa \in R$ and a smooth function $p$ is a $C^1$ smooth function. The boundary condition is $x_1(0) = 1$. We choose $\kappa = 20$ and

$$
p(t) = -\left(1 + \operatorname{erf}\left(\frac{t - 1/3}{\sqrt{2\varepsilon}}\right)\right), \quad \varepsilon = 10^{-2}
$$

There exists an inner layer at $t = 1/3$ in $p$ and also in the solution

$$
x(t) = \frac{e^t}{t^2+1} \begin{bmatrix} 1 + t - \frac{t^2}{2-t} + \frac{tp(t)}{t^2-4} \\ 1 - t - \frac{t}{2-t} + \frac{p(t)}{t^2-4} \\ -\frac{t^2+1}{2-t} \end{bmatrix}.
$$

If $\varepsilon$ is too small, the solution is not smooth. Hence, it is not advantageous to use the $p$-version method. For the choice of $\varepsilon = 10^{-2}$, Table 3 records the numerical errors and Fig. 3 shows the rate of convergence as desired.

**Table 3** Errors for Example 2

| Number of points | Set 1 | Set 2 | Set 3 |
|---|---|---|---|
| 10 | 1.0926e−2 | 1.0902e−2 | 1.9384e−2 |
| 20 | 1.0531e−5 | 4.5635e−5 | 9.6400e−5 |
| 30 | 6.3180e−9 | 1.1820e−8 | 6.5315e−8 |
| 40 | 7.0655e−13 | 5.3695e−12 | 7.9923e−12 |

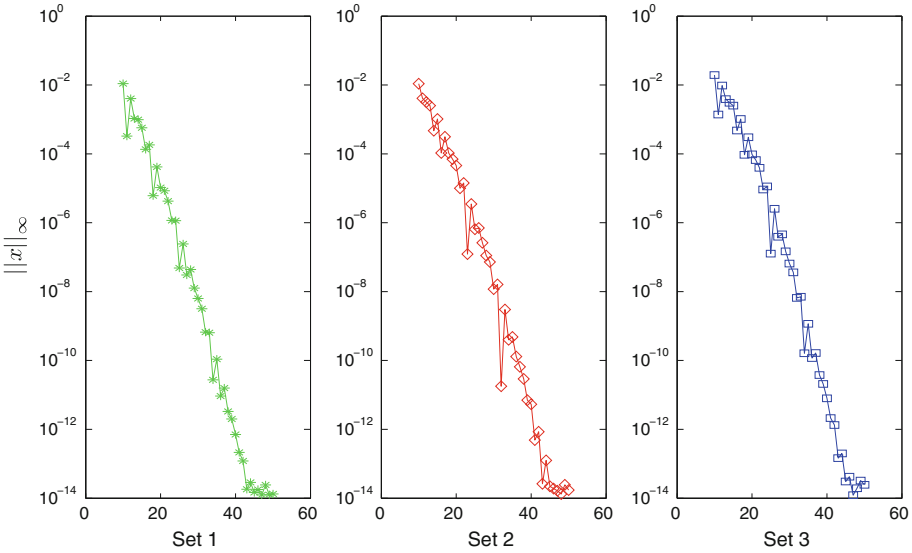


**Fig. 3** Numerical errors for Example 2

## 4 Nonlinear Differential-Algebraic Equations

Similar as linear DAEs, we first transform (5) into its canonical form,

$$\begin{cases} F_1(\xi, x, \frac{2}{b-a}\dot{x}) = 0, \\ F_2(\xi, x) = 0, \quad \xi \in [-1, 1], \\ f(x(-1), x(1)) = 0. \end{cases} \tag{47}$$

### 4.1 Formulation of Algorithm

The collocation points that we use are exactly the same as those in (14), where $\rho$ is for differential part and $\sigma$ is for algebraic part.

Denote $x_N^m(\xi) = \sum_{k=0}^{N} C_k^m T_k(\xi),\ m = 1, \ldots, n$ the approximation of $x$ in (47). We have the following set of conditions nonlinear equations:

$$\begin{aligned} f(x_N(-1), x_N(1)) &= 0, \\ F_1(\rho_i, x_N(\rho_i), \frac{2}{b-a}\dot{x}_N(\rho_i)) &= 0, \quad i = 0, \ldots, N-1, \\ F_2(\sigma_j, x_N(\sigma_j)) &= 0, \quad j = 0, \ldots, N. \end{aligned} \tag{48}$$

Now we convert it into a nonlinear equation of $C = (C^1, \ldots, C^n)^T$.

$$\begin{cases} f\Big([W_1C^1, \ldots, W_1C^n]^T, [W_2C^1, \ldots, W_2C^n]^T\Big) = 0, \\ F_1\Big(\rho, [TC^1, \ldots, TC^n]^T, \frac{2}{b-a}[UC^1, \ldots, UC^n]^T\Big) = 0, \\ F_2\Big(\sigma, [\tilde{T}C^1, \ldots, \tilde{T}C^n]^T\Big) = 0, \end{cases} \tag{49}$$

where $T$, $U$ and $\tilde{T}$ are defined in (18). We use Newton-iteration to solve the above system.

### 4.2 Numerical Experiments for Nonlinear DAEs

We present four examples in this section to illustrate the application of spectral collocation method for nonlinear DAEs. One important field of application is to solve Hamiltonian systems [5].

Given a Hamiltonian system:

$$\begin{cases} \frac{dp(t)}{dt} = -\frac{\partial H}{\partial q}, & p(0) = p_0, \\ \frac{dq(t)}{dt} = \frac{\partial H}{\partial p}, & q(0) = q_0, \end{cases} \tag{50}$$

where $p(t), q(t) \in R^n$. Setting $(x_1, \ldots, x_{2n}) = (p, q)^T$, $x_{2n+1} = H$, we obtain a nonlinear DAE on the interval $[0, T]$

$$\begin{cases} \frac{dx_i(t)}{dt} = -\frac{\partial H}{\partial q_i}, \quad i = 1, \ldots, n \ (Free) \\ \frac{dx_j(t)}{dt} = \frac{\partial H}{\partial p}, \quad j = n+1, \ldots 2n \ (Free) \\ \frac{dx_{2n+1}(t)}{dt} = 0, \\ H(x_1, \ldots, x_n, x_{n+1}, \ldots, x_{2n}) - x_{2n+1} = 0, \\ x_i(0) = x_0, \quad i = 1, \ldots, 2n, \\ x_{2n+1}(0) = H_0. \end{cases} \tag{51}$$

It should be pointed out that the system above is overdetermined and in practice, one need to abandon one equation among all free equations. Throughout this subsection, we choose $\rho$ as Legendre–Gauss points and $\sigma$ as Legendre–Gauss–Lobatto points.

*Remark 2* On the interval $[nT, (n+1)T]$, taking the transform $t = \frac{2nT+T}{2} + \frac{T}{2}\xi$, $\xi \in [-1, 1]$, we can obtain the same system as (51) except that $x_i(-1) = \hat{x}_0$, $x_{2n+1}(-1) = \hat{H}_0$, where $\hat{x}_0$ and $\hat{H}_0$ are the value of $x_i(1)$ and $x_{2n+1}(1)$ obtained from the preceding interval, respectively. To better approximate energy, we may specify initial condition as $x_i(0) = x_0$, $i = 1, \ldots, 2n$ and take a poststablization process [2] at the end of each step:

$$\begin{aligned} \tilde{x}_i(-1) &= \hat{x}_0, \\ x_i(-1) &= \tilde{x}_i - F(\tilde{x})h(\tilde{x}), \end{aligned}$$

where $h = H_x$ and $F = H^T(HH^T)^{-1}$.

*Example 3* We first consider an example from [10].

$$\begin{cases} \dot{x}_1 = (\epsilon + x_2 - p_2(t))x_4 + \dot{p}_1(t), \\ \dot{x}_2 = \dot{p}_2(t), \\ \dot{x}_3 = x_4, \\ 0 = (x_1 - p_1(t))(x_4 - e^t). \end{cases}$$

**Table 4** Errors for Example 3

| Number of points | 5 | 10 | 20 | 30 |
|---|---|---|---|---|
| $\max_{1\le i\le 4} \|x(\sigma_i) - x_i\|$ | 1.9566e0 | 2.6436e−3 | 1.1258e−10 | 4.2188e−15 |

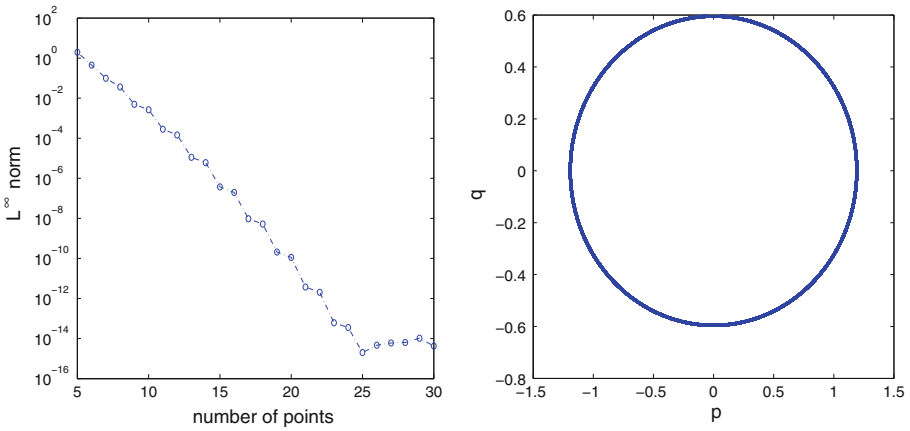


**Fig. 4** (*Left*) numerical errors for example 3; (*Right*) a phase plot of $(p, q)$ on $[0, 10^4]$—for Example 4

Choosing the boundary condition

$$x_1(0) = P_1(0) + \epsilon, \quad x_3(0) = 1, \quad x_2(1) = p_2(1),$$

the exact solution of the problem is

$$x^*(t) = \left(\epsilon e^t + p_1(t), p_2(t), e^t, e^t\right).$$

We choose $p_1(t) = \sin(4\pi t)$, $p_2(t) = \sin(t)$, and $\epsilon = 1/2$ as parameters. Using our algorithm, we obtain Table 4 and Fig. 4(Left). Clearly, the convergence rate is geometric.

*Example 4* Consider a simple linear Hamiltonian system [7]:

$$\begin{cases} p'(t) = -4q(t), \\ q'(t) = p(t). \end{cases}$$

with initial condition $p(0) = 0, q(0) = 0.6$ and Hamiltonian $H(p, q) = 1/2p^2 + 2q^2 = 0.72$. Although the Hamiltonian system itself is linear, if we consider the Hamiltonian as a variable and apply our algorithm, we obtain a nonlinear DAE. In [7], the system is treated as ODEs and numerically solved by spectral collocation method. On $[0, 10^5]$ , their algorithm has an energy error at the terminal point $2.2949 \times 10^{-10}$ whereas our algorithm has an error $3.2968 \times 10^{-11}$. Furthermore, applying the poststablization approach mentioned above, we can reduce the error to machine epsilon. Hence, our algorithm is much more accurate from this point of view. Figure 4(Right) shows that our algorithm keeps the symplectic structure of the system also. However, our algorithm takes longer CPU time to obtain the result, which is 1, 142 s while theirs takes only 621 s for 20 collocation points.

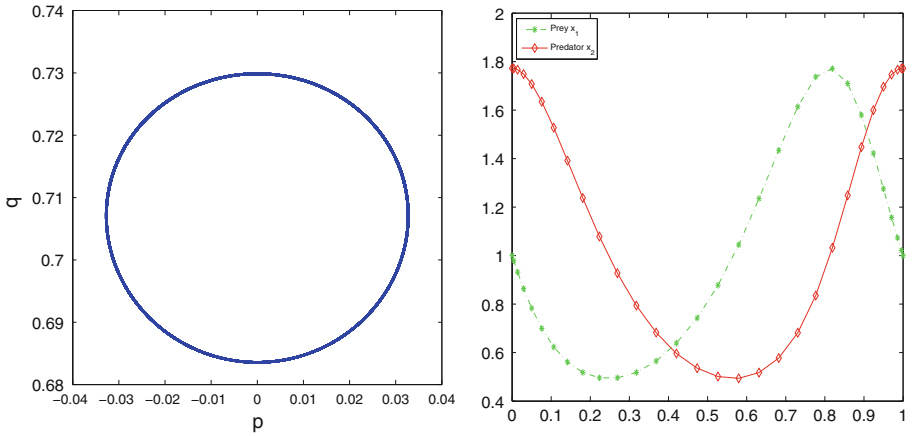


**Fig. 5** (*Left*) a phase plot of $(p, q)$ on $[0, 5000]$—for Example 6; (*Right*) prey/predator interaction in one cycle—for Example 5

*Example 5* Consider the Lotka–Volterra system for a predator/prey interaction, which consists of two equations [10].

$$\begin{cases} \dot{x}_1 = x_1(1 - x_2), \\ \dot{x}_2 = -cx_2(1 - x_1), \end{cases} \tag{52}$$

where $c > 0$. In the system, the quantity,

$$H = c(x_1 - \log x_1) + (x_2 - \log x_2)$$

remains constant along time, which forces solutions $x_1$ and $x_2$ to be periodic. But the period is unknown. As in [10], we introduce new variables $x_3 = H$ and $x_4 = T$ and transform the equation into the interval $[0, 1]$.

$$\begin{cases} \dot{x}_1 = x_1(1 - x_2)x_4, \\ \dot{x}_3 = 0, \\ \dot{x}_4 = 0, \\ c(x_1 - \log x_1) + (x_2 - \log x_2) - x_3 = 0, \ t \in [0, 1] \end{cases} \tag{53}$$

with initial conditions $x_1(0) = 1, x_1(0) = x_1(1)$, and $x_3(0) = 2.2$. For the choice of $c = 1$, we obtain $T$ converges to 6.4943297198 as the number of collocation points increases. Figure 5(Right) represents the computed interaction between prey and predator in one cycle for 30 collocation points.

*Example 6* Consider a system with Hamiltonian $H(p, q) = p^2 - q^2 + q^4$ [7]. The corresponding system of nonlinear ODEs is

$$\begin{cases} \dot{p} = 2q - 4q^3, \\ \dot{q} = 2p \end{cases}$$

with initial condition $p(0) = 0, q(0) = 0.73$. Hence, the Hamiltonian is $-0.24891759$. Our DAE algorithm takes only 17s to obtain a result on $[0, 5000]$, with an energy error $4.72 \times 10^{-13}$. It can be further reduced to machine epsilon by applying the poststablization process. A phase plot can be found in Fig. 5(Left).

## 5 Conclusion

In this work, a Jacobi spectral collocation method is applied to solve both linear and nonlinear DAE with arbitrary index. We apply a type of Jacobi–Gauss collocation scheme with $N$ knots to differential part of DAEs whereas another type of Jacobi–Gauss collocation scheme with $N + 1$ knots to algebraic part of the equation. Convergence analysis for linear DAEs shows that the spectral collocation method works efficiently if the solution itself is smooth enough. In particular, the scheme for nonlinear DAEs can be applied to solve Hamiltonian systems and it keeps the symplectic structure of the system. However, it may take more CPU seconds to obtain a result compared with the method in [7] since we enhance the dimension of the problems.

## References

1. Ascher, U., Petzold, L.: Projected collocation for higher-order higher-index differential-algebraic equations. J. Comput. Appl. Math. **43**, 243–259 (1992)
2. Ascher, U., Petzold, L.: Computer methods for ordinary differential equations and differential-algebraic equations. SIAM, Philadelphia (1998)
3. Bai, Y.: A perturbed collocation method for boundary value problems in differential-algebraic equations. Appl. Math. Comput. **45**, 269–291 (1991)
4. Canuto, C., Hussaini, M., Quarteroni, A., Zang, T.: Spectral Methods: Fundamentals in Single Domains. Springer, Berlin (2006)
5. Cherry, T.: On periodic solutions of Hamiltonian systems of differential equations. Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character **227**, 137–221 (1928)
6. Cimwanga, N.M.: Interlacing zeros of linear combinations of classical orthogonal polynomials. Thesis, University of Pretoria (2009)
7. Kanyamee, N., Zhang, Z.: Comparison of a spectral collocation method and symplectic methods for Hamiltonian systems. Int. J. Numer. Anal. Mod. **8**, 86–104 (2011)
8. Krasovsky, I.V.: Asymptotic distribution of zeros of polynomials satisfying difference equations. J. Comput. Appl. Math. **150**, 57–70 (2003)
9. Kunkel, P., Stöver, R.: Symmetric collocation methods for linear differential-algebraic boundary problems. Numer. Math. **91**, 475–501 (2002)
10. Kunkel, P., Mehrmann, V., Stöver, R.: Symmetric collocation for unstructured nonlinear differential-algebraic equations of arbitrary index. Numer. Math. **98**, 277–304 (2004)
11. Kunkel, P., Mehrmann, V.: Differential-Algebraic Equations: Analysis and Numerical Solution. European Mathematical Society, Zürich (2006)
12. Mastroianni, G., Occorsio, D.: Optimal systems of nodes for Lagrange interpolation on bounded intervals: a survey. J. Comput. Appl. Math. **134**, 325–341 (2001)
13. Shen, J., Tao, T., Wang, L.L.: Spectral Methods: Algorithms, Analysis and Applications. Springer, New York (2011)
14. Stancu, D.D., Coman, G., Blaga, P.: Numerical Analysis and Approximation theory (in Romanian), vol. II. Cluj University Press, Cluj-Napoca (2002)
15. Szegö, G.: Orthogonal Polynomials, vol. XXIII. American Mathematical Society, Providence (1939)
16. Zhang, Z.: Superconvergece of spectral collocation and $p$-version methods in one dimensional problems. Math. Comput. **74**, 1621–1636 (2005)