

Research Article

L-Stable Method for Differential-Algebraic Equations of Multibody System Dynamics

Bowen Li,¹ Jieyu Ding ,^{1,2} and Yanan Li¹

¹School of Mathematics and Statistics, Qingdao University, Qingdao 266071, China

²Center for Computational Mechanics and Engineering Simulation, Qingdao University, Qingdao 266071, China

Correspondence should be addressed to Jieyu Ding; djy@qdu.edu.cn

Received 8 January 2019; Revised 22 March 2019; Accepted 28 April 2019; Published 29 May 2019

Academic Editor: Samuel N. Jator

Copyright © 2019 Bowen Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An L-stable method over time intervals for differential-algebraic equations (DAEs) of multibody system dynamics is presented in this paper. The solution format is established based on equidistant nodes and nonequidistant nodes such as Chebyshev nodes and Legendre nodes. Based on Ehle's theorem and conjecture, the unknown matrix and vector in the L-stable solution formula are obtained by comparison with Pade approximation. Newton iteration method is used during the solution process. Taking the planar two-link manipulator system as an example, the results of L-stable method presented are compared for different number of nodes in the time interval and the step size in the simulation, and also compared with the classic Runge-Kutta method, A-stable method, Radau IA, Radau IIA, and Lobatto IIIC methods. The results show that the method has the advantages of good stability and high precision and is suitable for multibody system dynamics simulation under long-term conditions.

1. Introduction

The stability of the numerical solution of differential equations is one of the important indicators to judge the quality of a numerical method. In the 1960s, after the innovation and development of Dahlquist, Widlund, Gear, and other scholars, the stability theory was deeply developed. Dahlquist [1] proposed a linear equation $y' = \mu y$ as a model equation and proposed A-stable, that is, $|y_r| \rightarrow 0$, $r \rightarrow \infty$, and all $h > 0$. Subsequently, Axellson [2] determines the a_{ik} of the $y_{i,k} = y_{n,r-1} + h \sum_{k=1}^n a_{ik} f(y_{k,r})$ by the quadrature coefficients at the zeros of $p_n - p_{n-1}$ or $p_n - p_{n-2}$, constructing a class of A-stable methods, where p_n is the Legendre polynomial. Widlund [3] proposed A(α)-stable, that is, if all solutions of $\alpha_k x_{n+k} + \dots + \alpha_0 x_n = h(\beta_k f_{n+k} + \dots + \beta_0 f_n)$ tend to zero as $n \rightarrow \infty$ when the method is applied, with a fixed h , to any differential equation of the form $y' = \mu y$, where μ is a complex constant which lies in the set $S_\alpha = \{z; |\arg(-z)| < \alpha, z \neq 0\}$. Gear [4] proposed stiffness stable, in order to generalize the A-stable concept to nonlinear problems. In 1975, Dahlquist [5] proposed the G-stable concept of single-method linear multistep method.

Butcher [6, 7] studied the nonlinear stability of Runge-Kutta method by using nonlinear equations $y' = f(y)$ as model equations, proposed B-stable and algebraic stability of the Runge-Kutta method, and established the algebraic stability criteria for general linear methods. Li Shoufu [8] established the (k,p,q) stability theory of the general linear method and the weak algebraic stability criterion of (k,p,q) . For a very rigid differential/algebraic equations, it is hoped that the rigid component can be attenuated as soon as possible, and the stable domain contains the left half complex plane; only A-stable is not enough. Ehle [9] proposed L-stable. Hairer [10] pointed out that the L-stable numerical method will introduce damping effect, but it is very effective for suppressing high-frequency oscillation, and some scholars have proposed implicit single-step and multistep methods based on L-stable for solving rigid equations, but these methods are based on ordinary differential equations. L-stable methods for DAEs are relatively rare.

The typical form of multibody system dynamic equations is DAEs, which consist of ordinary differential equations and algebraic constraint equations. The numerical solution is a matter of concern in the study of multibody system dynamics.

At present, most of the numerical solutions of the DAEs mainly include Direct integration method and Newmark method. Although the calculation process is relatively simple, the stability is poor. In the long-term simulation process, the error accumulation is more serious, and it needs to be corrected by combining constrained projection [11]. In this paper, the general form of L-stable method for multi-body system dynamics DAEs is established. Based on the equidistant nodes, Chebyshev nodes, and Legendre nodes, the concrete construction process is given and the planar two-link manipulator system verified and compared.

2. DAEs of Multibody System Dynamics

The DAEs of index-3 in multibody systems dynamics are

$$\begin{aligned} \mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \Phi_{\mathbf{q}}^T(\mathbf{q}, t) \boldsymbol{\lambda} - \mathbf{F}(\dot{\mathbf{q}}, \mathbf{q}, t) &= \mathbf{0} \\ \Phi(\mathbf{q}, t) &= \mathbf{0} \end{aligned} \quad (1)$$

where $\mathbf{q} = [q_1(t) \ q_2(t) \ \dots \ q_n(t)]^T$ are generalized coordinates of the system, $\dot{\mathbf{q}} = d\mathbf{q}/dt$ are generalized speed of the system, $\ddot{\mathbf{q}} = d^2\mathbf{q}/dt^2$ are generalized acceleration of the system, $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \dots, \lambda_m]^T$ are the Lagrange multiplier vectors, $\mathbf{M} \in R^{n \times n}$ is a symmetric positive definite mass matrix, $\mathbf{F} \in R^n$ are generalized force vectors, and $\Phi = [\Phi_1, \Phi_2, \dots, \Phi_m]^T$ are m complete displacement constraints.

For the constraint equation Φ , the first derivative and the second derivative are, respectively, obtained for the time t , and the DAEs form of the index-2 and the index-1 can be obtained by the combination of (1).

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \Phi_{\mathbf{q}}^T(\mathbf{q}, t) \boldsymbol{\lambda} - \mathbf{F}(\dot{\mathbf{q}}, \mathbf{q}, t) = \mathbf{0} \quad (2)$$

$$\Phi_{\mathbf{q}}(\mathbf{q}, t) \dot{\mathbf{q}} + \Phi_t(\mathbf{q}, t) = \mathbf{0}$$

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \Phi_{\mathbf{q}}^T(\mathbf{q}, t) \boldsymbol{\lambda} - \mathbf{F}(\dot{\mathbf{q}}, \mathbf{q}, t) = \mathbf{0}$$

$$\begin{aligned} \Phi_{\mathbf{q}}(\mathbf{q}, t) \ddot{\mathbf{q}} + [\Phi_{\mathbf{q}}(\mathbf{q}, t) \dot{\mathbf{q}}]_{\mathbf{q}} \dot{\mathbf{q}} + 2\Phi_{\mathbf{q}t}(\mathbf{q}, t) \dot{\mathbf{q}} + \Phi_{tt}(\mathbf{q}, t) \\ = \mathbf{0} \end{aligned} \quad (3)$$

When the initial values $\mathbf{q}_0 = \mathbf{q}(0)$, $\dot{\mathbf{q}}_0 = \dot{\mathbf{q}}(0)$ are known, the Runge-Kutta method can be used to directly solve the index-1 equation (3); however, if the selected step size is large, the accuracy will be reduced and the total energy will increase with time [12], smaller step will lead to lower computation efficiency and poor stability, and the index-3 equation (1) and the index-2 equation (2) cannot be solved directly using the Runge-Kutta method. The L-stable method is constructed below and can be used to solve the multibody system dynamics DAEs of index-1, -2, and -3.

3. Construction of the DAEs L-Stable Format

3.1. General Form of L-Stable Method. In 1972, H. A. Watts and L. F. Shampine [13] proposed a block implicit one-step method for solving ordinary differential equations. The r discrete moments are made into a block vector, the output

values of r moments are obtained from the input values of one moment, and it is proved that the method is A-stable. In this method, the time step in each block vector is equidistant. This article improves on this method; the block vector is modified to correspond to the point in the single-step interval, and in the iterative solution process, a single-step interval $[t_k, t_{k+1}]$ and a time step $h = t_{k+1} - t_k$ are set, selecting $t_k < c_1 < c_2 < \dots < c_r = t_{k+1}$ and constructing DAEs solution format, where $c_i (i = 1, 2, \dots, r)$ can be an equidistant node or a nonequidistant node.

Taking the multibody system dynamics DAEs of index-1 equation (3) as an example, after the order reduction, the following first-order differential equations can be obtained.

$$\begin{aligned} \dot{\mathbf{z}} &= \mathbf{z} \\ \mathbf{M}(\mathbf{q}) \dot{\mathbf{z}} + \Phi_{\mathbf{q}}^T(\mathbf{q}, t) \boldsymbol{\lambda} - \mathbf{F}(\mathbf{z}, \mathbf{q}, t) &= \mathbf{0} \\ \Phi_{\mathbf{q}}(\mathbf{q}, t) \dot{\mathbf{z}} + [\Phi_{\mathbf{q}}(\mathbf{q}, t) \mathbf{z}]_{\mathbf{q}} \mathbf{z} + 2\Phi_{\mathbf{q}t}(\mathbf{q}, t) \mathbf{z} + \Phi_{tt}(\mathbf{q}, t) \\ &= \mathbf{0} \end{aligned} \quad (4)$$

Let $\mathbf{y} = [\mathbf{q}^T \ \mathbf{z}^T \ \boldsymbol{\lambda}^T]^T \in R^{2n+m}$, $\mathbf{y}_i = \mathbf{y}(c_i)$, $\dot{\mathbf{y}}_i = \dot{\mathbf{y}}(c_i)$, $i = 1, 2, \dots, r$, $\mathbf{Y} = [\mathbf{y}_1^T \ \mathbf{y}_2^T \ \dots \ \mathbf{y}_r^T]^T$, $\dot{\mathbf{Y}} = [\dot{\mathbf{y}}_1^T \ \dot{\mathbf{y}}_2^T \ \dots \ \dot{\mathbf{y}}_r^T]^T$; the construction solution formula is as follows

$$\mathbf{Y} = \mathbf{e} \otimes \mathbf{y}_k + h\mathbf{d} \otimes \dot{\mathbf{y}}_k + h\mathbf{B} \otimes \dot{\mathbf{Y}} \quad (5)$$

where $\mathbf{e} = [1 \ 1 \ \dots \ 1]^T \in R^r$, $\mathbf{d} \in R^r$, $\mathbf{B} \in R^{r \times r}$, $\mathbf{y}_k = \mathbf{y}(t_k)$, $\dot{\mathbf{y}}_k = \dot{\mathbf{y}}(t_k)$, and \otimes is Kronecker.

Bring (5) into the test equation

$$\begin{aligned} \mathbf{y}'(t) &= \alpha \mathbf{y}(t) \\ \mathbf{y}(t_0) &= \mathbf{y}_0, \end{aligned} \quad (6)$$

$$\text{Re}(\alpha) < 0$$

and get stability function

$$R(\lambda) = \mathbf{e}_r^T (\mathbf{I}_r - \lambda \mathbf{B})^{-1} (\mathbf{e} + \lambda \mathbf{d}) \quad (7)$$

where $\lambda = \alpha h$, \mathbf{I}_r is the r -order unit matrix, and $\mathbf{e}_r = [0 \ \dots \ 0 \ 1] \in R^r$. For the stability function, using the *Pade* approximation, we can construct L-stabilized solution formula.

Theorem 1 (*Ehle theorem and conjecture*). *For (k, j) - Pade approximation, if $k \leq j \leq k + 2$, then this Pade approximation is A-stable; if $k < j \leq k + 2$, then this Pade approximation is L-stable [14].*

Perform a Taylor expansion on \mathbf{y}_i , $i = 1, 2, \dots, r$ at t_k , let the coefficient of the $\mathbf{y}_k^{(r+1)} = \mathbf{y}^{(r+1)}(t_k)$ term be a pending constant, and omit the following high-order terms; one can get

$$\begin{aligned} \mathbf{y}_i \approx \mathbf{y}_k + h_i \dot{\mathbf{y}}_k + \frac{h_i^2}{2!} \ddot{\mathbf{y}}_k + \dots + \frac{h_i^r}{r!} \mathbf{y}_k^{(r)} + \frac{h_i}{(r+1)!} \mathbf{y}_k^{(r+1)}, \\ i = 1, 2, \dots, r \end{aligned} \quad (8)$$

where $\mu_i, i = 1, 2, \dots, r$, is pending constant and $h_i = c_i - t_k, i = 1, 2, \dots, r$. Bringing (8) into (5), one can get the following.

$$[\mathbf{B} \ \mathbf{d}] = \begin{bmatrix} h_1 & h_1^2 & \cdots & h_1^r & \mu_1 \\ h_2 & h_2^2 & \cdots & h_2^r & \mu_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h_r & h_r^2 & \cdots & h_r^r & \mu_r \end{bmatrix} \cdot \begin{bmatrix} 1 & 2h_1 & 3h_1^2 & \cdots & (r+1)h_1^r \\ 1 & 2h_2 & 3h_2^2 & \cdots & (r+1)h_2^r \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2h_r & 3h_r^2 & \cdots & (r+1)h_r^r \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}^{-1} \quad (9)$$

From (9), \mathbf{B}, \mathbf{d} , which are expressed using the undetermined constant, can be obtained bringing these into the stability function (7). The value of the undetermined constant $\mu_i, i = 1, 2, \dots, r$, can be obtained by comparison with the corresponding Pade approximation form, so that the value of \mathbf{B}, \mathbf{d} can be obtained bringing into formula (5) a specific solution formula.

3.2. L-Stable Format Based on Equidistant Nodes. For the equidistant node $c_i (i = 1, 2, \dots, r)$, there are $c_i = t_k + (i/r)h, i = 1, 2, \dots, r$. For example, when $r = 3, h = 1$, the equidistant node is $c_i = t_k + i/3, i = 1, 2, 3$; at this time $h_i = i/3, i = 1, 2, 3$, and (9) becomes as follows.

$$[\mathbf{B} \ \mathbf{d}] = \begin{bmatrix} \frac{1}{3} & \frac{1}{9} & \frac{1}{27} & \mu_1 \\ \frac{2}{3} & \frac{4}{9} & \frac{8}{27} & \mu_2 \\ \frac{1}{3} & \frac{1}{9} & \frac{1}{27} & \mu_3 \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{2}{3} & \frac{1}{3} & \frac{4}{27} \\ 1 & \frac{4}{3} & \frac{4}{9} & \frac{32}{27} \\ 1 & 2 & 3 & 4 \\ 1 & 0 & 0 & 0 \end{bmatrix}^{-1} \quad (10)$$

Calculate \mathbf{B}, \mathbf{d} , substitute stability function (7), and compare it with (2, 3) – Pade approximation formula; one can solve undetermined constants $\mu_1 = 1/45, \mu_2 = 8/45, \mu_3 = 1$, and bringing them into the expression \mathbf{B}, \mathbf{d} , one can obtain

$$\mathbf{B} = \frac{1}{3} \begin{pmatrix} \frac{107}{120} & -\frac{37}{120} & \frac{3}{40} \\ \frac{15}{9} & \frac{15}{9} & -\frac{15}{8} \\ \frac{15}{8} & \frac{15}{8} & \frac{3}{8} \end{pmatrix}, \quad (11)$$

$$\mathbf{d} = \frac{1}{3} \begin{pmatrix} \frac{41}{120} \\ \frac{2}{5} \\ \frac{3}{8} \end{pmatrix}$$

Bringing them into (5) can be used to obtain a solution formula. The same method can be used to construct L-stable solution formats for other r values.

3.3. L-Stable Format Based on Nonequidistant Nodes. When the node $c_i (i = 1, 2, \dots, r)$ of the interval is the zero point of the Chebyshev polynomial, within the interval $[t_k, t_{k+1}]$, one gets the following.

$$c_i = t_k + \frac{h}{2} \left(1 + \cos \left(\frac{2i-1}{2(r-1)} \pi \right) \right), \quad (12)$$

$$i = 1, 2, \dots, r-1$$

Take $r = 3, h = 1$ as an example, get $c_1 = t_k + 1/2 - \sqrt{2}/4, c_2 = t_k + 1/2 + \sqrt{2}/4, c_3 = 1$, and bring it into (9); one can get

$$[\mathbf{B} \ \mathbf{d}] = \begin{bmatrix} h_1 & h_1^2 & h_1^3 & \mu_1 \\ h_2 & h_2^2 & h_2^3 & \mu_2 \\ 1 & 1 & 1 & \mu_3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2h_1 & 3h_1^2 & 4h_1^3 \\ 1 & 2h_2 & 3h_2^2 & 4h_2^3 \\ 1 & 2 & 3 & 4 \\ 1 & 0 & 0 & 0 \end{bmatrix}^{-1} \quad (13)$$

where $h_1 = 1/2 - \sqrt{2}/4, h_2 = 1/2 + \sqrt{2}/4$.

Bring \mathbf{B}, \mathbf{d} into the stability function (7) and make it equal to the (2, 3) – Pade approximation to get $\mu_1 = (10 - 5\sqrt{2})/32, \mu_2 = (10 + 5\sqrt{2})/32$, and $\mu_3 = 1$ and get the corresponding \mathbf{B}, \mathbf{d} :

$$\mathbf{B} = \begin{pmatrix} \frac{11-2\sqrt{2}}{24} & \frac{5-8\sqrt{2}}{24} & \frac{7\sqrt{2}-1}{48} \\ \frac{5+8\sqrt{2}}{24} & \frac{11+2\sqrt{2}}{24} & -\frac{7\sqrt{2}+1}{48} \\ \frac{2}{3} & \frac{2}{3} & -\frac{1}{6} \end{pmatrix}, \quad (14)$$

$$\mathbf{d} = \begin{pmatrix} \frac{\sqrt{2}-7}{48} & -\frac{\sqrt{2}+7}{48} & -\frac{1}{6} \end{pmatrix}^T,$$

and bringing these into (5), one can get the formula.

When the node $c_i (i = 1, 2, \dots, r)$ is the zero point of the Legendre polynomial, take $r = 3, h = 1$ as an example, get $c_1 = (1/2)(-\sqrt{3}/3 + 1), c_2 = (1/2)(\sqrt{3}/3 + 1), c_3 = 1$, get $\mu_1 = (3 - \sqrt{3})/12, \mu_2 = (3 + \sqrt{3})/12, \mu_3 = 1$, and get the corresponding \mathbf{B}, \mathbf{d} :

$$\mathbf{B} = \begin{pmatrix} \frac{27+\sqrt{3}}{72} & \frac{9-17\sqrt{3}}{72} & \frac{3+5\sqrt{3}}{72} \\ \frac{9+17\sqrt{3}}{72} & \frac{27-\sqrt{3}}{72} & \frac{3-5\sqrt{3}}{72} \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix}, \quad (15)$$

$$\mathbf{d} = \begin{pmatrix} -\frac{\sqrt{3}+3}{72} & \frac{\sqrt{3}-3}{72} & 0 \end{pmatrix}^T,$$

and bringing these into (5) can be used to obtain a solution formula. The L-stability solution format for nonequidistant nodes of other r values can be constructed using the same method.

4. Numerical Example

Planar two-link manipulator is shown in Figure 1; the length of the link is L_1, L_2 , where $L_1 = 1m, L_2 = \sqrt{3}m$, and the

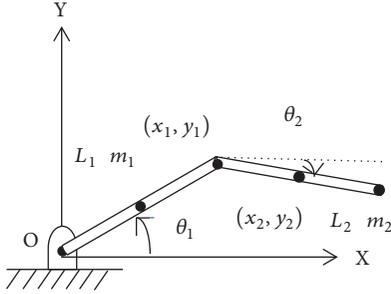


FIGURE 1: Planar two-link manipulator.

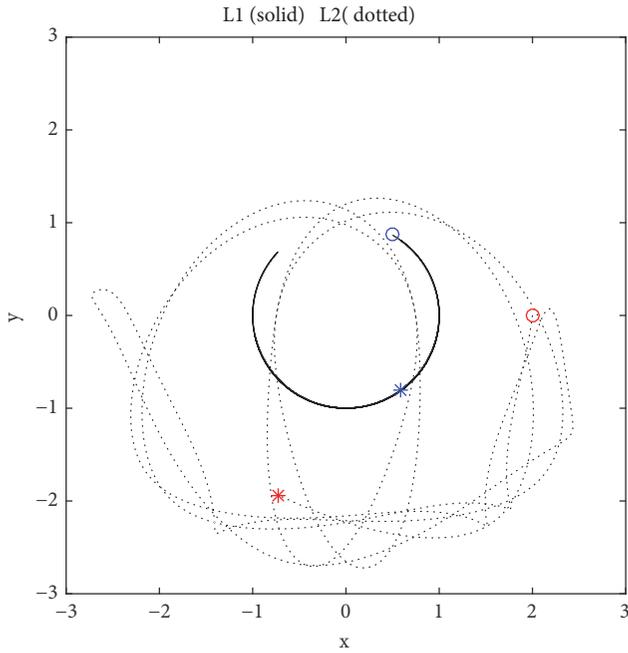


FIGURE 2: Displacement trajectory at the end of the connecting rod.

quality is m_1, m_2 , where $m_1 = 1\text{kg}, m_2 = 2\text{kg}$. The corners are denoted by θ_1 and θ_2 , and at the initial time, $\theta_1 = \pi/3, \theta_2 = -\pi/6$. $(x_1, y_1), (x_2, y_2)$ represent the coordinate position of the connecting rod centroid.

Based on the Lagrange function, planar two-link dynamics model is established, where $\mathbf{q} = (x_1, y_1, \theta_1, x_2, y_2, \theta_2)^T$, $\mathbf{M} = \text{diag}([m_1, m_1, I_1, m_2, m_2, I_2])$ is a generalized mass matrix, $I_1 = (1/12)m_1L_1^2$ and $I_2 = (1/12)m_2L_2^2$ are the moments of inertia of the connecting rod, and $\mathbf{F} = [0, -m_1g, 0, 0, -m_2g, 0]^T$ is a generalized force vector.

Here using the (2, 4) - *Pade*, we take $r=4$ as an example to divide the cells into four equal parts for the solution, and the step size $h=0.01$ and the time history 10s are selected and programmed using MATLAB. The Newton iteration method is used for calculation and the maximum allowable error is 10^{-3} . The displacement trajectory at the end of the connecting rod can be obtained as shown in Figure 2; the starting position of the connecting rod end is indicated by a circle symbol, and the ending position is represented by a star. The displacement time course of the connecting rod end is shown in Figure 3.

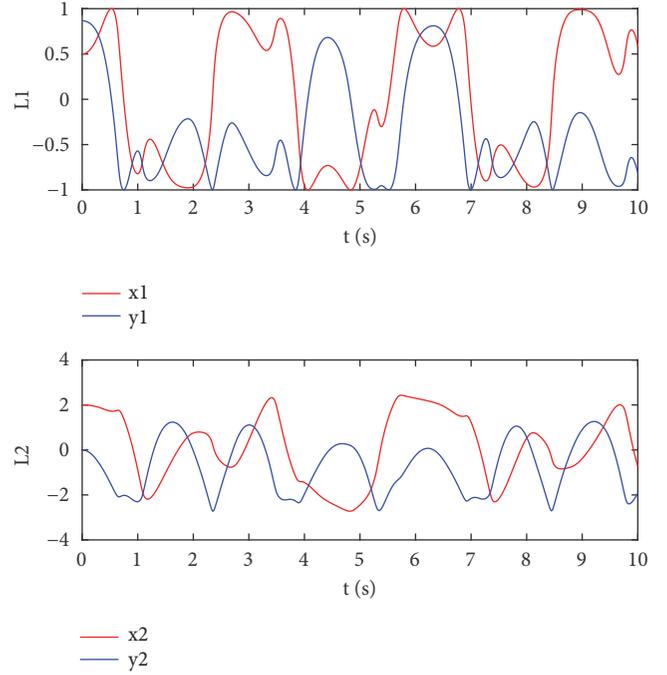


FIGURE 3: Displacement time history of the end of the rod.

The total energy, kinetic energy, and potential energy of the system are shown in Figure 4. Displacement constraints, speed-level constraints, and acceleration-level constraints are shown in Figure 5.

It can be seen from Figures 2–5 that the L-stable method given in this paper can maintain good stability during the simulation process, and the total energy error and the constraint error of each level are small, which is suitable for multibody system dynamics simulation. The numerical verification and comparison of the method are carried out in different situations.

(1) The simulation time is $t=10\text{s}$, the number of identical equidistant nodes ($r=4$) is taken in each time interval, and the results of different step sizes are compared in Table 1. When the step size is reduced, the time spent on program operation is increasing, but the maximum energy error ($\max|\varepsilon H|$) is decreasing; at the same time, displacement constraints, speed-level constraints, and acceleration-level constraints are better when the step size is smaller. Therefore, in the case of a short simulation time, choosing a smaller step size will improve the accuracy of the solution, and accuracy better maintains displacement constraints, speed-level constraints, and acceleration-level constraints.

(2) The simulation time is $t=10\text{s}$, and the step size is $h=0.01$, selecting the number of different equidistant nodes in each time interval for comparison ($r=3,4$) in Table 2. When $r=4$, although the time spent on program operation is increasing, the maximum energy error, displacement constraints, speed-level constraints, and acceleration-level constraints are less than $r=3$; therefore, an increase in the number of nodes can improve the accuracy.

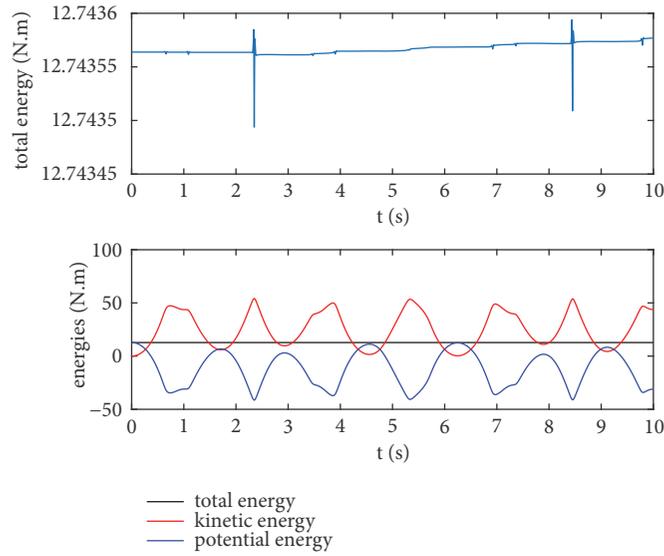


FIGURE 4: Total energy, kinetic energy, and potential energy time history of the system.

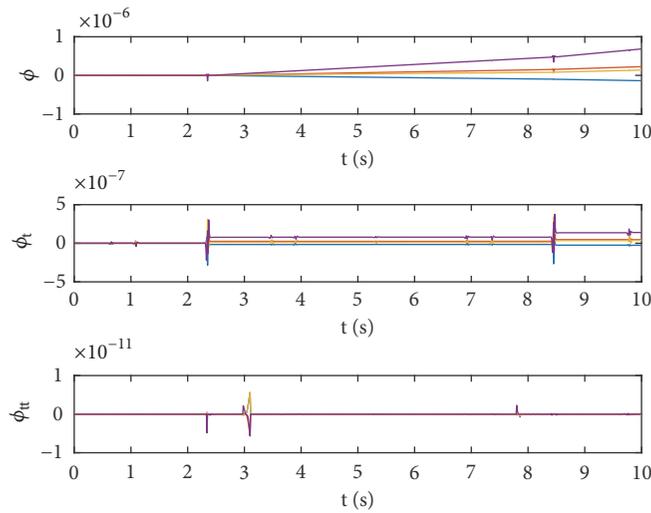


FIGURE 5: Time history of displacement and speed-level and acceleration-level constraints.

TABLE 1: Results of L-stable method under different steps.

	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
$h=0.01$	5.8906	$7.0007e-05$	$6.8459e-07$	$3.7480e-07$	$5.6595e-12$
$h=0.005$	10.8750	$8.8078e-07$	$1.9154e-09$	$6.6686e-09$	$1.9846e-12$
$h=0.001$	42.6406	$5.5074e-11$	$7.6766e-13$	$4.1800e-13$	$1.4211e-13$

TABLE 2: Results of the L-stable method under different nodes.

	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
$r=3$	1.6875	0.0021	$5.8212e-05$	$3.9267e-04$	$6.2629e-12$
$r=4$	5.8906	$7.0007e-05$	$6.8459e-07$	$3.7480e-07$	$5.6595e-12$

TABLE 3: Comparison of L-stable method and Explicit Runge-Kutta method at the same step size, $h=0.01$.

	Runtime/s	max $ e_H $	displacement constraints	speed-level constraints	acceleration-level constraints
L	5.8906	7.0007e-05	6.8459e-07	3.7480e-07	5.6595e-12
ERK4	0.3906	0.2628	0.0087	0.0021	9.9476e-14

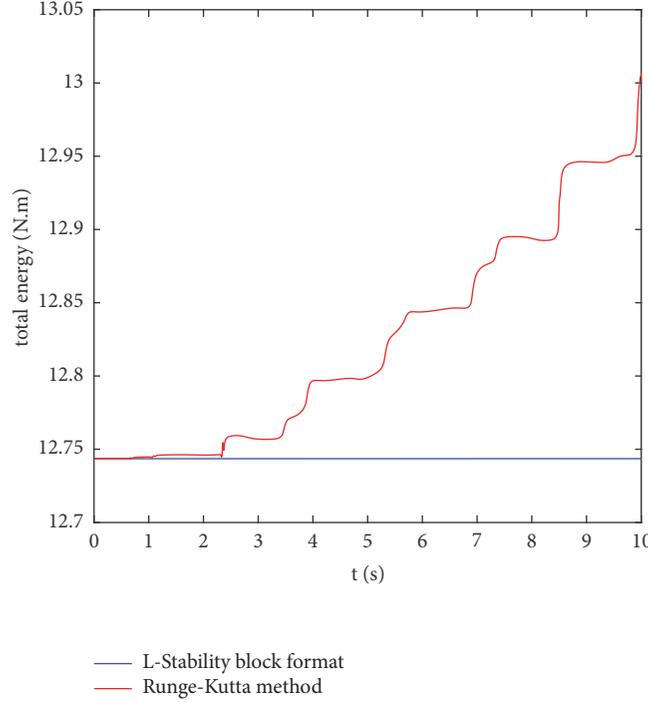


FIGURE 6: L-stable method and fourth-order Explicit Runge-Kutta method for total system time history (10s).

(3) The simulation time is $t=10s$, and the step size is $h=0.01$. We compare L-stable method when $r=4$ with fourth-order Explicit Runge-Kutta method (ERK4) in Table 3. In the L-stable method (L) for solving differential-algebraic equations, the maximum energy error, the displacement constraints, and the speed-level constraints are obviously smaller than those in the fourth-order Explicit Runge-Kutta method, although the acceleration-level constraints are not as good as those in the fourth-order Explicit Runge-Kutta method; the total energy of the L-stable method keeps stable with time, and the total energy of the fourth-order Explicit Runge-Kutta method has been increasing rapidly with time, as in Figures 6 and 7.

(4) The simulation time is $t=10s$, and the step size is $h=0.01$. Under the same number of nodes ($r=4$), the differential-algebraic equations of index-1, index-2, and index-3 are solved in L-stable method in Table 4. The time taken for the differential-algebraic equations of the L-stability block format to solve index-1, index-2, and index-3 is roughly the same, and the maximum energy errors for index-1 and index-2 are significantly smaller than those for index-3. The displacement constraints for index-3 are the smallest, the speed-level constraints for index-2 are the smallest, and the acceleration-level constraints for index-1 are the smallest. By comprehensive comparison, the overall accuracy of index-1 is higher.

(5) The step size is $h=0.01$; under the same step size, the L-stable method of index-3 selects equidistant nodes to be compared with nonequidistant nodes (Chebyshev nodes, Legendre nodes) in Table 5. By comparison, it is found that when the L-stable method selects nonequidistant nodes, the maximum energy error is smaller than equidistant nodes, and Chebyshev nodes have higher overall accuracy.

5. Accuracy Experiments and Analysis

The differential-algebraic equation of index-3 is used to solve the planar two-link model in time interval $t \in [0, 1]$, taking the result of fourth-order Explicit Runge-Kutta method $h = 0.000001$ as its approximate exact solution, and calculate the logarithmic relationship between the overall error of \mathbf{q} and the step size of the L-stable method at $t=1s$, $r=4$, as shown in Figure 8; the logarithm of the overall error and step size for each component of \mathbf{q} is shown in Figures 9 and 10, when $r=3$; the logarithmic relation curve of the overall error is shown in Figure 11; the logarithm of the overall error and step size for each component is shown in Figures 12 and 13. The formula $\log \|e\| / \log h$ is used to calculate the accuracy. When $r=4$, the L-stable method overall error can guarantee the fourth-order accuracy. When $r=3$, the L-stable method overall error can

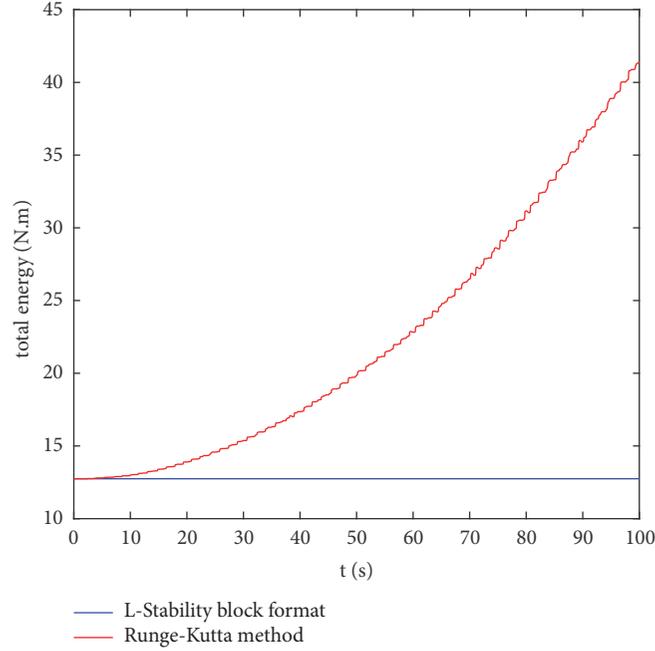


FIGURE 7: L-stable method and fourth-order Explicit Runge-Kutta method for total system time history (100s).

TABLE 4: Comparison of differential-algebraic equations for L-stable method solving indices-1,2,3.

	Runtime/s	max εH	displacement constraints	speed-level constraints	acceleration-level constraints
index -1	5.8906	7.0007e-05	6.8459e-07	3.7480e-07	5.6595e-12
index -2	5.4219	3.2163e-05	4.3949e-09	1.7764e-14	0.0704
index -3	5.6094	0.0011	2.7756e-15	5.9405e-04	0.9854

TABLE 5: Comparison of the results of the L-stable method taking uniform and nonuniform nodes.

	Runtime/s	max εH	displacement constraints	speed-level constraints	acceleration-level constraints
equidistant nodes	5.6094	0.0011	2.7756e-15	5.9405e-04	0.9854
Chebyshev nodes	5.2188	4.3245e-05	2.9976e-15	1.8094e-04	0.6880
Legendre nodes	5.1094	2.5327e-04	2.8866e-15	2.9047e-04	0.7560

guarantee the second-order accuracy; i.e., the accuracy of r is greater.

$$\mathbf{d} = \frac{1}{4} \begin{pmatrix} 1847 & 29 & 179 & 14 \\ 5040 & 90 & 560 & 45 \end{pmatrix}^T \tag{16}$$

6. Stability Analysis

According to Theorem 1, one can construct A-stable method and get matrix \mathbf{B} and \mathbf{d} .

$$\mathbf{B} = \frac{1}{4} \begin{pmatrix} \frac{2081}{62} & -\frac{109}{4} & \frac{191}{2} & -\frac{43}{5040} \\ \frac{45}{377} & \frac{15}{111} & \frac{45}{167} & -\frac{90}{31} \\ \frac{280}{64} & \frac{140}{8} & \frac{280}{64} & -\frac{560}{14} \\ \frac{45}{45} & \frac{15}{15} & \frac{45}{45} & \frac{45}{45} \end{pmatrix},$$

Taking the planar two-link manipulator as an example, with the simulation time being $t=10s$ and the step size being $h=0.01$, the differential-algebraic equations for the index-1, -2, and -3 are obtained in Table 6. It can be seen that the results of solving the index-1, -2 are basically the same as those obtained by the L-stable method, but when solving the index-3, there is a singularity that cannot be solved. It can be seen that the L-stable method is also helpful for solving high-index problems.

The L-stable method in this paper can be written into the corresponding Runge-Kutta method [15]. Taking the (1, 3) – *Pade* approximation as an example, the

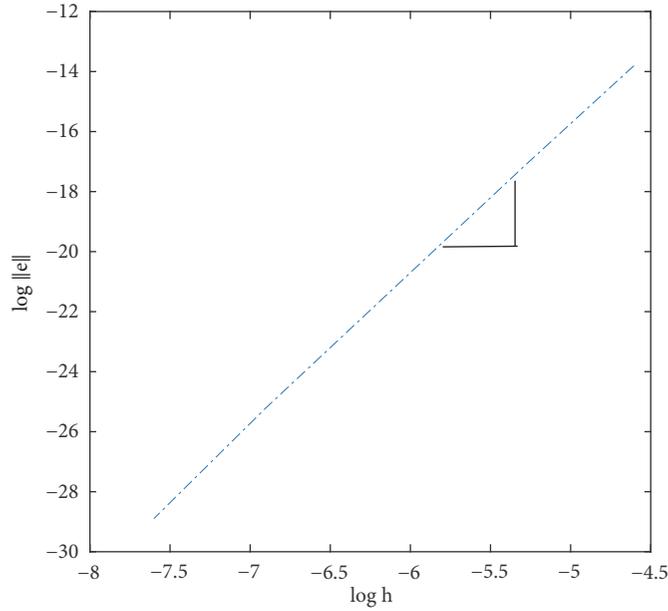


FIGURE 8: The overall error of \mathbf{q} and the logarithm of h .

TABLE 6: Comparison of DAEs for L-stable and A-stable method solving indices-1,2,3.

index-1	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
A-stable	6.2188	3.8045e-05	2.3863e-07	2.0615e-07	5.6648e-12
L-stable	5.8906	7.0007e-05	6.8459e-07	3.7480e-07	5.6595e-12
index-2	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
A-stable	5.4688	1.1145e-05	1.6059e-09	1.7729e-14	0.0214
L-stable	5.4219	3.2163e-05	4.3949e-09	1.7764e-14	0.0704
index-3	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
A-stable	Singular				
L-stable	5.6094	0.0011	2.7756e-15	5.9405e-04	0.9854

TABLE 7: Comparison of L-stable method ($r=3$) with Radau IA, Radau IIA, and Lobatto IIIC methods.

	Runtime/s	max $ \epsilon H $	displacement constraints	speed-level constraints	acceleration-level constraints
L-stable	1.6875	0.0021	5.8212e-05	3.9267e-04	6.2629e-12
Radau IA	14.9063	1.4075	0.1803	0.1087	9.8100
Radau IIA	13.5625	0.1035	2.1396	8.2918e-04	9.8100
Lobatto IIIC	13.9375	0.1038	2.1396	0.0010	9.8100

corresponding fourth-order Runge-Kutta coefficient table is as follows.

$$\begin{array}{c|cccc}
 0 & 0 & 0 & 0 & 0 \\
 \frac{1}{3} & \frac{1}{72} & \frac{43}{72} & \frac{29}{72} & \frac{1}{8} \\
 \frac{2}{3} & \frac{1}{19} & \frac{1}{19} & \frac{1}{19} & \frac{1}{19} \\
 \frac{3}{3} & \frac{12}{1} & \frac{36}{3} & \frac{36}{3} & \frac{36}{3} \\
 1 & \frac{8}{1} & \frac{8}{3} & \frac{8}{3} & \frac{8}{1} \\
 \hline
 & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8}
 \end{array} \tag{17}$$

The Radau IA, Radau IIA, and Lobatto IIIC methods in the Implicit Runge-Kutta method are also L-stabilized [16]. The above method is used to calculate the planar two-link manipulator dynamics model. In order to compare the results with the results of this paper, the fourth-order method is selected for calculation. The step size is $h=0.01$, and the simulation time is $t=10s$. The comparison results are shown in Table 7.

By comparison, it is found that running time, maximum energy error, displacement constraints, speed-level constraints, and acceleration-level constraints of the L-stable method constructed in this paper for solving the multibody system dynamics model are better than those in the Radau IA,

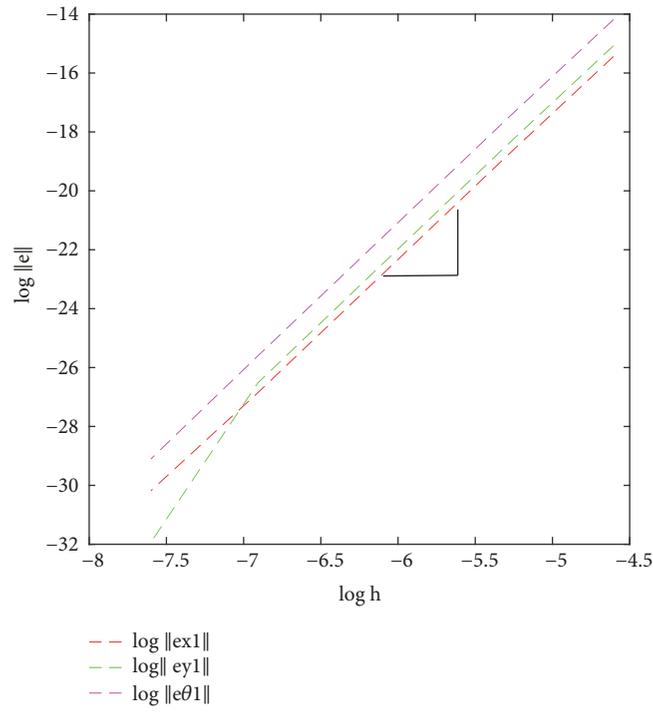


FIGURE 9: The overall error of x_1, y_1, θ_1 and the logarithm of h .

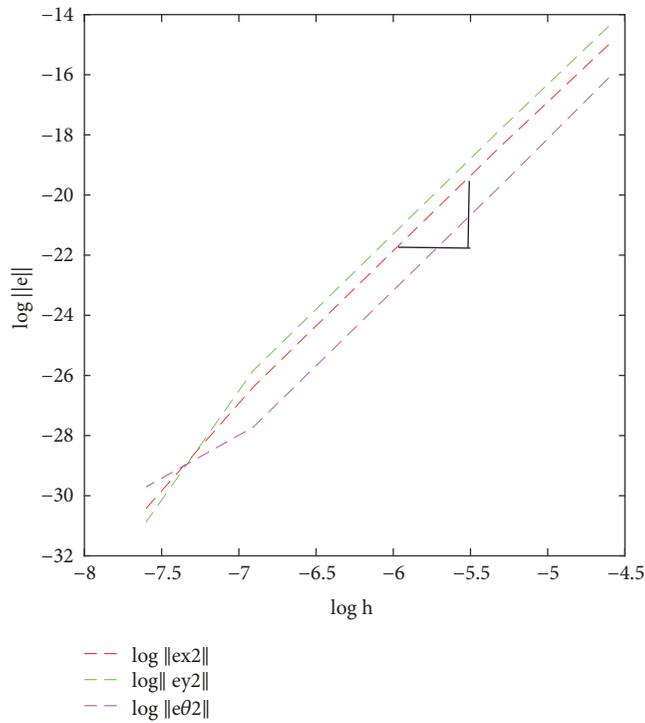


FIGURE 10: The overall error of x_2, y_2, θ_2 and the logarithm of h .

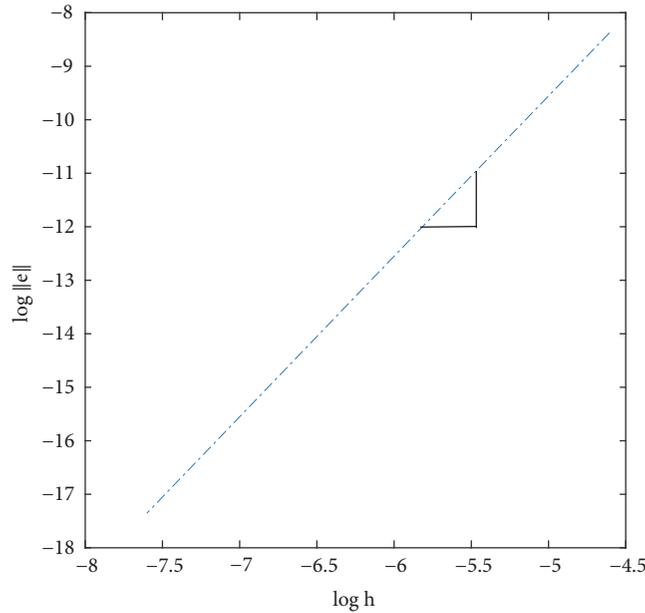


FIGURE 11: The overall error of q and the logarithm of h .

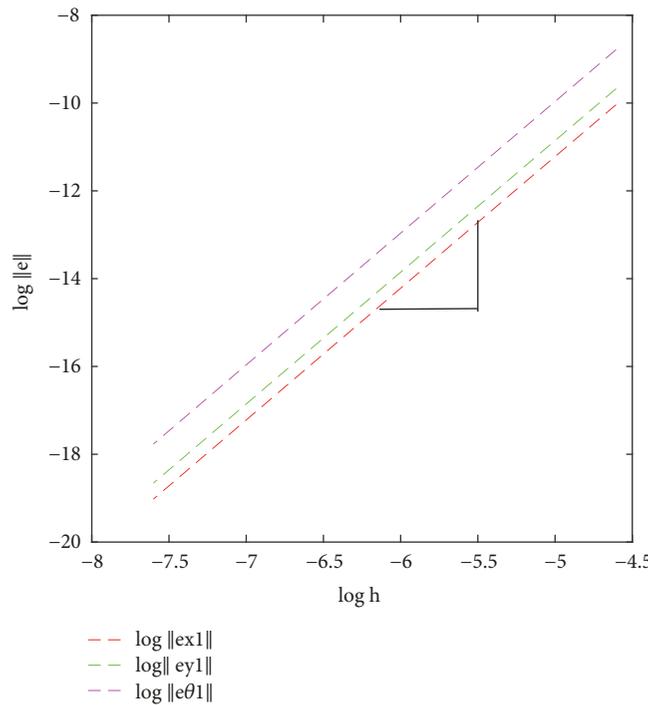


FIGURE 12: The overall error of x_1, y_1, θ_1 and the logarithm of h .

Radau IIA, and Lobatto IIIC methods of the Implicit Runge-Kutta, although these methods are L-stable.

7. Conclusion

This paper constructs L-stable method for solving differential-algebraic equations of multibody system dynamics. The numerical results show that, under the equidistant

node, when the same number of nodes is selected, the smaller the step size, the higher the accuracy, and when the step size is fixed, the more the nodes, the higher the accuracy. When compared with the Explicit Runge-Kutta method, the L-stable method better maintains the displacement constraints and speed-level level constraints, the maximum energy error is significantly smaller than the Explicit Runge-Kutta method, and the total energy retention effect is better under long-term

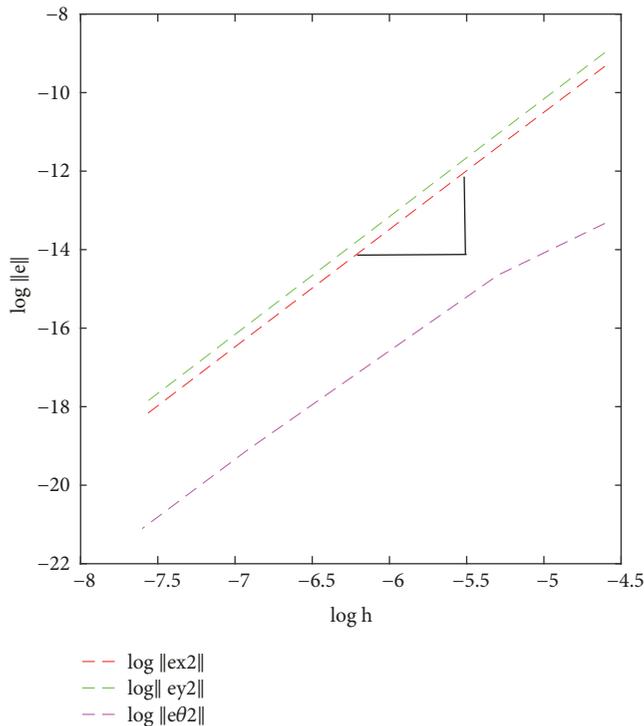


FIGURE 13: The overall error of x_2, y_2, θ_2 and the logarithm of h .

simulation. When compared with A-stable method, L-stable method has no singularity in calculating high-index systems of equations, and each index keeps good results. When compared with Radau IA, Radau IIA, and Lobatto IIIC method, L-stable method is more efficient and has better performance than the above methods. The numerical results show that the nonequidistant nodes (Chebyshev nodes and Legendre nodes) result accuracy is higher than equidistant node. The method is simple in mathematics, easy to use, and easy to program. It is easy to implement in computers and worthy of research and promotion. How to improve the calculation efficiency while ensuring the numerical accuracy is the content that needs further study in the future.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study has been supported by the National Natural Science Foundations of China under Grant no. 11472143, 11772166.

References

- [1] G. G. Dahlquist, "A special stability problem for linear multistep methods," *BIT Numerical Mathematics*, vol. 3, pp. 27–43, 1963.
- [2] O. Axelsson, "A class of A-stable methods," *BIT Journal*, vol. 9, no. 3, pp. 185–199, 1969.
- [3] O. B. Widlund, "A note on unconditionally stable linear multistep methods," *BIT Numerical Mathematics*, vol. 7, pp. 65–70, 1967.
- [4] C. W. Gear, *The Automatic Integration of Stiff Ordinary Differential Equations*, North Holland publishing Company, Amsterdam, Netherlands, 1963.
- [5] G. Dahlquist, *Error Analysis for a Class of Methods for Stiff Nonlinear Initial Value Problems*, Springer-Verlag, Berlin, Germany, 1975.
- [6] J. C. Butcher, "A stability property of implicit Runge-Kutta methods," *BIT Journal*, vol. 15, no. 4, pp. 358–361, 1975.
- [7] K. Burrage and J. C. Butcher, "Stability criteria for implicit Runge-Kutta methods," *SIAM Journal on Numerical Analysis*, vol. 16, no. 1, pp. 46–57, 1979.
- [8] S. F. Li, "Nonlinear stability of general linear methods," *Journal of Computational Mathematics*, vol. 9, no. 2, pp. 97–104, 1991.
- [9] B. L. Ehle, "A-stable methods and Pade Approximations to the exponential," *SIAM Journal on Mathematical Analysis*, vol. 4, no. 5, pp. 671–680, 1973.
- [10] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II stiff and differential-Algebraic problems*, Science Press, Beijing, China, 2nd edition, 2006.
- [11] J. Y. Ding and Z. K. Pan, "Generalized- α projection method for differential-algebraic equations of multibody dynamics," *Engineering Mechanics*, vol. 30, no. 4, pp. 380–384, 2013.
- [12] L. P. Wen, C. H. Yang, and H. Y. Wen, "Stability and asymptotic stability of multistep runge-kutta methods for nonlinear functional-integro-differential equations," *Natural Science Journal of Xiangtan University*, vol. 40, no. 1, pp. 1–5, 2018.
- [13] H. A. Watts and L. F. Shampine, "A-stable block implicit one-step methods," *Bit Numerical Mathematics*, vol. 12, no. 2, pp. 252–266, 1972.
- [14] Z. D. Yuan, G. Fei, and D. G. Liu, *Numerical Solution of Initial Value Problems for Stiff Ordinary Differential Equations*, Science Press, Beijing, China, 2016.
- [15] Z. Q. Wu, "L-Stable methods for numerical solution of structural dynamics equations and multibody dynamics equations," in *Changsha, Hunan: National University of Defense Technology*, pp. 1–132, 2009.
- [16] F. S. Zhu, "A Class of Runge-Kutta methods with L-stability and B-stability," *Journal of Mathematics*, vol. 21, no. 2, pp. 183–188, 2001.

