BIT Numerical Mathematics (2025) 65:12
https://doi.org/10.1007/s10543-025-01056-x

BIT

RESEARCH PAPER

Block implicit methods with *L*-stability for parabolic problems

Shishun Li[1] · **Jing-Yuan Wang**[2] · **Xiao-Chuan Cai**[2]

Received: 15 July 2024 / Accepted: 7 February 2025 / Published online: 24 February 2025

Abstract
Block implicit methods (BIM) are a group of time integration schemes with desirable stability properties, high-order of accuracy and the capability to compute solutions across multiple time steps simultaneously without numerous initial values. In this paper, some BIM with *L*-stability are developed, similar to Runge–Kutta methods, a tableau including two matrices and two vectors defines a particular BIM with the required order of accuracy and stability properties. In particular, a type of *L*-stable BIM with a positive definite matrix and a positive diagonal matrix is introduced. In this study, we also extend the classical finite element theory, commonly for parabolic problems discretized using the Backward Euler or Crank–Nicolson schemes, to this type of BIM. Additionally, in order to solve the resulting large sparse linear system of equations, several tensor structure preserving domain decomposition preconditioners for Krylov subspace methods are also introduced. Finally, some numerical results are reported to demonstrate the effectiveness of the new methods.

Keywords Block implicit method · *L*-stability · Domain decomposition preconditioner · Parabolic problems

Mathematics Subject Classification 65L20 · 65M55 · 65M60

✉ Xiao-Chuan Cai
xccai@um.edu.mo

Shishun Li
lss6@sina.com

Jing-Yuan Wang
yc27479@um.edu.mo

[1] School of Mathematics and Statistics, Xinyang Normal University, Xinyang, People's Republic of China

[2] Department of Mathematics, University of Macau, Macau, People's Republic of China

1 Introduction

We consider the convection-diffusion equation

$$\begin{cases} u_t - \nabla \cdot (\alpha(x)\nabla u) + \beta(x) \cdot \nabla u = f(x,t), & \text{in } \Omega \times (0,T], \\ u(x,t) = 0, & \text{on } \partial\Omega \times (0,T], \\ u(x,0) = u_0(x), & \text{in } \Omega, \end{cases} \tag{1}$$

where $\Omega \subset R^d$ ($d = 2$ or 3) is a bounded, open polygonal (or polyhedral) domain, $f(x,t) \in L^2(\Omega \times [0,T])$. Assume all the coefficients are sufficiently smooth, and $0 < \alpha_0 \le \alpha(x) \le \alpha_1 < +\infty$. A typical low-order discretization of (1) by backward Euler in time and finite element (FEM) in space produces a system of equations with coefficient matrix

$$\boldsymbol{M} + \tau \boldsymbol{K}, \tag{2}$$

where τ is the time step size. $\boldsymbol{M}, \boldsymbol{K} \in \mathbb{R}^{N\times N}$ denote the mass and the stiffness matrix, respectively, N is the number of unknowns in space. Some higher order methods, such as the implicit Runge–Kutta (IRK) methods are attractive for solving (1) since their favorable stability properties, high-order of accuracy, and require no additional initial data. For example, the r-stage IRK can be expressed compactly using a Butcher tableau,

$$\begin{array}{c|c} \boldsymbol{c} & \widehat{\boldsymbol{A}} \\ \hline & \widehat{\boldsymbol{b}}^T \end{array}, \tag{3}$$

where $\widehat{\boldsymbol{A}} \in \mathbb{R}^{r\times r}$, $\boldsymbol{c} \in \mathbb{R}^r$ and $\widehat{\boldsymbol{b}} \in \mathbb{R}^r$. The discretization of (1) by IRK in time and FEM in space produces a system with coefficient matrix

$$\boldsymbol{I} \otimes \boldsymbol{M} + \tau \widehat{\boldsymbol{A}} \otimes \boldsymbol{K}, \tag{4}$$

where $\otimes$ denotes the Kronecker product. Among the implicit Runge–Kutta (IRK) schemes, the diagonally implicit Runge–Kutta (DIRK) method is commonly preferred due to its relatively straightforward implementation [14]. With the lower triangular structure of the Runge–Kutta matrix, the r-stage equations can be solved in r successive stages using a back substitution algorithm. However, this sequential stage by stage computation becomes inefficient when applied on large scale parallel computers. Furthermore, DIRK may suffer from the reduction of the order of accuracy and the stage-order is at most 2 [9]. The fully IRK (FIRK) methods enhance the stage-order but come with higher computational costs. Many effort has been made in order to construct effective and robust preconditioners for such schemes [11–13, 20, 22, 26, 27, 30]. Compared with (2), the matrix (4) lacks the positive definiteness of the matrix $\boldsymbol{K}$ since $\widehat{\boldsymbol{A}}$ is only positive stable. Moreover, the traditional finite element theory for parabolic problems discretized by the backward Euler or Crank–Nicolson schemes can not be extended for IRK until now.

Block implicit methods (BIM) was initially presented to provide initial values for predictor-corrector schemes [21] and share key attributes like strong stability and high-order accuracy, similar to IRK methods [5, 23, 24, 31, 32]. Recently, A-stable BIM

with up to 10th order of accuracy was proposed and studied in [18]. In this paper, we present a new group of BIM with L-stability for (1), which is better than A-stable schemes for stiff problems [9]. Similar to IRK, BIM with block size r can be described by a tableau [18]

$$\begin{array}{c|c} \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{a}^T & \boldsymbol{b}^T \end{array}, \tag{5}$$

where $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{r\times r}$ are *BIM matrices* and $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^r$ are *BIM vectors*. For (1), the coefficient matrix takes the form

$$\boldsymbol{A} \otimes \boldsymbol{M} + \tau \boldsymbol{B} \otimes \boldsymbol{K}. \tag{6}$$

In this paper, by carefully choosing the BIM matrices and vectors, we introduce several new BIM with special L-stabilities, and some corresponding preconditioning matrices without which the methods would be difficult to use for practical applications. More precisely, we construct a group of BIM with positive diagonal matrix $\boldsymbol{B}$ and positive definite matrix $\boldsymbol{A}$ such that the matrix (6) is sparse and has the same positive definiteness as (2), both properties is critically important if the systems are to be solved by a preconditioned Krylov subspace method. Similar to the implicit Euler and Crank–Nicolson methods for solving (1) [28], we establish *a prior* error estimates for finite element approximation based the Lax-Milgram theorem. Note that such analysis can not be obtained for FIRK since the FIRK matrix is only positive stable but not positive definite [20]. In the second part of the paper, we propose and study some domain decomposition (DD) based preconditioners for (6). DD has been widely used for solving the large system of equations arising from the discretization of partial differential equations [6–8, 15–17, 25, 29]. This work presents some DD preconditioners that preserve the tensor form from the BIM discretization.

The remaining part of the paper is organized as follows. In Sect. 2, we first introduce the general form of BIM with L-stability and present the matrix $\boldsymbol{B}^{-1}\boldsymbol{A}$ explicitly, then a series of BIM models with varying order of accuracy spanning from 2 to 8 are constructed. In Sect. 3, several convergence theorems are given for a model problem discretized by BIM in time and the finite element in space, and several tensor preserving domain decomposition based preconditioners are also presented for the resulting linear systems. Section 4 presents numerical experiments to illustrate the performance of BIM. Finally, some concluding remarks are given in Sect. 5.

2 Block implicit methods with *L*-stability

In this section, we first discuss the general form of BIM and several types of stabilities. Subsequently, we introduce some BIM with L-stability by employing the order conditions of linear multistep methods (LMM). Finally, some BIM with special matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are constructed explicitly.

2.1 Preliminaries of the block implicit methods

We consider an ordinary differential equation (ODE)

$$y' = f(t, y) \quad t \in (0, T], \quad y(t_0) = y_0 \in \mathbb{R}. \tag{7}$$

Given a uniform temporal partition $0 = t_0 < t_1 < \cdots < t_M = T$ with $\tau = t_i - t_{i-1}$, a BIM with block size r for (7) can be written in the following form

$$\begin{cases} a_{1r} y_{n+r} + \cdots + a_{11} y_{n+1} + a_{10} y_n = \tau(b_{1r} f_{n+r} + \cdots + b_{11} f_{n+1} + b_{10} f_n), \\ a_{2r} y_{n+r} + \cdots + a_{21} y_{n+1} + a_{20} y_n = \tau(b_{2r} f_{n+r} + \cdots + b_{21} f_{n+1} + b_{20} f_n), \\ \qquad\qquad\qquad\qquad \vdots \\ a_{rr} y_{n+r} + \cdots + a_{r1} y_{n+1} + a_{r0} y_n = \tau(b_{rr} f_{n+r} + \cdots + b_{r1} f_{n+1} + b_{r0} f_n), \end{cases} \tag{8}$$

where y_0 is a given initial value. Each equation of (8) is the classical LMM with order p, the coefficients a_{ij} and b_{ij} $(i = 1, 2, \cdots, r,\ j = 0, 1, \cdots, r)$ satisfy the order conditions

$$\begin{cases} C_{i,0} = \sum_{j=0}^{r} a_{ij} = 0, \\ C_{i,1} = \sum_{j=0}^{r} j a_{ij} - \sum_{j=0}^{r} b_{ij} = 0, \\ C_{i,2} = \frac{1}{2} \sum_{j=0}^{r} j^2 a_{ij} - \sum_{j=0}^{r} j b_{ij} = 0, \\ \qquad \cdots\cdots \\ C_{i,p} = \frac{1}{p!} \sum_{j=0}^{r} j^p a_{ij} - \frac{1}{(p-1)!} \sum_{j=0}^{r} j^{p-1} b_{ij} = 0, \qquad p = 3, 4, \cdots. \end{cases} \tag{9}$$

The error of (8) is $\boldsymbol{C}_{p+1} = (C_{1,p+1}\ C_{2,p+1}\ \cdots\ C_{r,p+1})^T$ and $C_{i,p+1}$ is the error constant of LMM. It is convenient to represent a BIM by a partitioned tableau of the form

$$\begin{array}{c|c} \boldsymbol{A} & \boldsymbol{B} \\ \hline \boldsymbol{a}^T & \boldsymbol{b}^T \end{array},$$

where $\boldsymbol{a} = (a_{10}\ a_{20}\ \cdots\ a_{r0})^T$, $\boldsymbol{b} = (b_{10}\ b_{20}\ \cdots\ b_{r0})^T$, and the BIM matrices

$$\boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1r} \\ a_{21} & a_{22} & \cdots & a_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r1} & a_{r2} & \cdots & a_{rr} \end{pmatrix} \quad \text{and} \quad \boldsymbol{B} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r1} & b_{r2} & \cdots & b_{rr} \end{pmatrix}.$$

In the following discussion, we assume $\boldsymbol{A}$ and $\boldsymbol{B}$ are nonsingular.

For BIM, the solutions at r time steps are computed simultaneously, so the corresponding stability functions may have different properties from the classical methods. To illustrate this, we present two special BIM applied to a simple ODE

$$y' = -2000(y - \cos(t)), \quad y(0) = 0, \quad t \in (0, 1.5]. \tag{10}$$

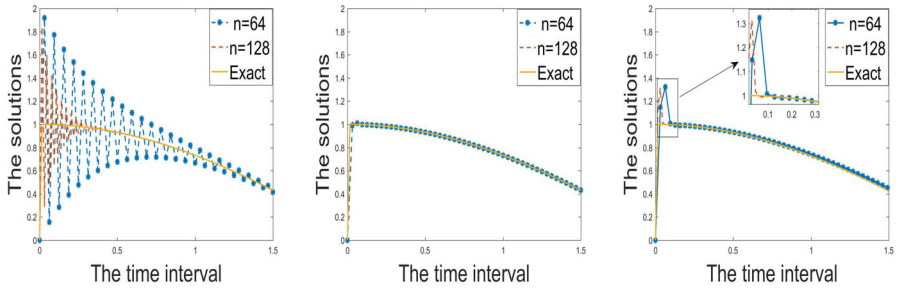

Fig. 1 Numerical solutions of (10) computed by the trapezoidal rule (left), Method A (middle) and Method B (right)

Method A (second-order):

$$\begin{cases} y_{n+2} - y_n = 2\tau f_{n+1}, \\ 3y_{n+2} - 4y_{n+1} + y_n = 2\tau f_{n+2}, \end{cases}$$

which is an one-step implicit method (with initial value $y_0 = y(0)$) obtained by coupling the mid-point rule and the BDF2 rule.

Method B (third-order):

$$\begin{cases} 12y_{n+3} - 12y_{n+2} = \tau(23f_{n+2} - 16f_{n+1} + 5f_n), \\ 11y_{n+3} - 18y_{n+2} + 9y_{n+1} - 2y_n = 6\tau f_{n+3}, \\ 3y_{n+3} - 3y_{n+1} = \tau(-2f_{n+3} + 13f_{n+2} - 8f_{n+1} + 3f_n), \end{cases}$$

which is an one-step implicit method (with initial value $y_0 = y(0)$) obtained by coupling three linear 3-step methods.

The numerical results of the trapezoidal rule, Methods A and B are presented in Fig. 1. It is well-known that the trapezoidal rule is second-order and A-stable, but not L-stable. The numerical comparisons show that Methods A and B damp out the transient phase much faster than the trapezoidal rule. The right subfigure also shows that the solutions at the first three time steps behave differently since the corresponding stability functions have different properties. These properties are typical for block methods, therefore below we introduce several new types of stability.

Applying BIM to a *Dahlquist* test equation

$$y' = \lambda y, \quad y_0 = 1, \quad z = \tau\lambda, \tag{11}$$

we obtain

$$\boldsymbol{C}(z)\boldsymbol{y} \equiv (\boldsymbol{A} - z\boldsymbol{B})\boldsymbol{y} = (z\boldsymbol{b} - \boldsymbol{a})y_n, \tag{12}$$

where $\boldsymbol{y} = (y_{n+1}\ y_{n+2}\ \cdots\ y_{n+r})^T$. Let $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ be the matrices obtained by replacing column i of $\boldsymbol{A}$ and $\boldsymbol{B}$ by $-\boldsymbol{a}$ and $-\boldsymbol{b}$, respectively. Denotes $|\boldsymbol{C}_i(z)| =$

$|\boldsymbol{A}_i - z\boldsymbol{B}_i|$ $(i = 1, 2, \ldots, r)$. Then the stability functions of BIM are defined as

$$R_{n+i}(z) = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}(z)|}, \quad \text{or} \quad R_{n+i}(z) = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}_{i-1}(z)|} \quad 1 \le i \le r, \tag{13}$$

where $\boldsymbol{C}_0(z) = \boldsymbol{C}(z)$. From (12), we have $y_{n+i} = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}(z)|} y_n$ and $y_{n+i} = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}_{i-1}(z)|} y_{n+i-1}$.

Definition 1 The sets

$$S_i = \{z \in \mathbb{C} : \quad |R_{n+i}(z)| \le 1\}, \ 1 \le i \le r$$

are called the stability domains of the BIM.

i) If $S_i \supset \mathbb{C}^-$ for all $1 \le i \le r$, i.e., each $R_{n+i}(z) = |\boldsymbol{C}_i(z)|/|\boldsymbol{C}_{i-1}(z)|$ is A-stable, and $\lim_{z\to\infty} R_{n+i}(z) = 0$, this method is called strong block r-step L-stable;
ii) If $S_i \supset \mathbb{C}^-$ for all $1 \le i \le r$, i.e., each $R_{n+i}(z) = |\boldsymbol{C}_i(z)|/|\boldsymbol{C}(z)|$ is A-stable, and $\lim_{z\to\infty} R_{n+i}(z) = 0$, this method is called block r-step L-stable;
iii) If $S_r \supset \mathbb{C}^-$, i.e., $R_{n+r}(z) = |\boldsymbol{C}_r(z)|/|\boldsymbol{C}(z)|$ is A-stable, and $\lim_{z\to\infty} R_{n+r}(z) = 0$, this method is called block rth-step L-stable.

2.2 The stability of BIM

Now we prove that the solution of (12) is uniquely determined by $\boldsymbol{B}^{-1}\boldsymbol{A}$, and is independent of the individual matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Let

$$\boldsymbol{W} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & 2 & \cdots & r \\ 0 & 1 & 2^2 & \cdots & r^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 2^r & \cdots & r^r \end{pmatrix} \quad \text{and} \quad \boldsymbol{H} = \begin{pmatrix} 0 & & & \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & r & 0 \end{pmatrix}$$

we have the following lemma.

Lemma 1 ([1]) *The elements of the matrix $\boldsymbol{W}^{-1}\boldsymbol{H}\boldsymbol{W}$ are*

$$(\boldsymbol{W}^{-1}\boldsymbol{H}\boldsymbol{W})_{ij} = \begin{cases} \displaystyle\sum_{p=1}^{i} \frac{1}{p} - \sum_{p=1}^{r-i} \frac{1}{p}, & i = j, \\ \displaystyle\frac{(-1)^{j-i}}{j-i} \frac{C_i^r}{C_j^r}, & i \ne j. \quad i, j = 0, 1, \ldots, r, \end{cases}$$

where $C_i^r = \binom{r}{i}$ denotes the binomial coefficient.

Define $\boldsymbol{B}_0, \boldsymbol{C}_0 \in \mathbb{R}^{r\times r}$ by

$$\boldsymbol{B}_0 = \begin{pmatrix} b_{10} & 0 & \cdots & 0 \\ b_{20} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ b_{r0} & 0 & \cdots & 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{C}_0 = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 2^{r-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & r & \cdots & r^{r-1} \end{pmatrix}. \tag{14}$$

For $i = 1, 2, \ldots, r$, let $\boldsymbol{E}_{i1}, \boldsymbol{E}_1 \in \mathbb{R}^{r\times r}$ be the matrices defined as follows:

$$\boldsymbol{E}_{i1} = [\boldsymbol{e}_i\ \boldsymbol{e}_0\ \cdots\ \boldsymbol{e}_0] \quad \text{and} \quad \boldsymbol{E}_1 = [\boldsymbol{e}\ \boldsymbol{e}_0\ \cdots\ \boldsymbol{e}_0], \tag{15}$$

where $\boldsymbol{e}_i$ is the column i of the identity matrix $\boldsymbol{I} \in \mathbb{R}^{r\times r}$, $\boldsymbol{e} = (1, 1, \cdots, 1)^T \in \mathbb{R}^r$ and $\boldsymbol{e}_0 = (0, 0, \cdots, 0)^T \in \mathbb{R}^r$. Then we have the following corollary.

Corollary 1 *Denote $\boldsymbol{D}_0 = diag(1, 2, \cdots, r)$ and $\boldsymbol{N}_0 = \boldsymbol{C}_0\boldsymbol{D}_0\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1}$, we have*

$$(\boldsymbol{E}_{i1}\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1})_{mj} = \begin{cases} \dfrac{(-1)^j}{-j}C_j^r, & m = i, \\ 0, & m \neq i, \end{cases} \quad \text{and } (\boldsymbol{N}_0)_{ij} = \begin{cases} \displaystyle\sum_{p=1}^{i}\frac{1}{p} - \sum_{p=1}^{r-i}\frac{1}{p}, & i = j, \\ \dfrac{(-1)^{i-j}}{i-j}\dfrac{C_j^r}{C_i^r}, & i \neq j. \end{cases}$$

Proof Set

$$\boldsymbol{F} = \begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{pmatrix}_{(r+1)\times(r+1)}.$$

A straightforward computation yields

$$\boldsymbol{W}^T\boldsymbol{F} = \begin{pmatrix} 0 & \boldsymbol{e}_1^T \\ \boldsymbol{e}_0 & bmC_0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{H}^T = \boldsymbol{F}\begin{pmatrix} 0 & \boldsymbol{e}_0^T \\ \boldsymbol{e}_0 & \boldsymbol{D}_0 \end{pmatrix}.$$

Therefore,

$$\begin{aligned} \boldsymbol{W}^T\boldsymbol{H}^T\boldsymbol{W}^{-T} &= \boldsymbol{W}^T\boldsymbol{F}\begin{pmatrix} 0 & \boldsymbol{e}_0^T \\ \boldsymbol{e}_0 & \boldsymbol{D}_0 \end{pmatrix}\boldsymbol{W}^{-T} = \begin{pmatrix} 1 & \boldsymbol{e}_0^T \\ \boldsymbol{e} & \boldsymbol{D}_0\boldsymbol{C}_0 \end{pmatrix}\boldsymbol{F}\begin{pmatrix} 0 & \boldsymbol{e}_0^T \\ \boldsymbol{e}_0 & \boldsymbol{D}_0 \end{pmatrix}\boldsymbol{W}^{-T} \\ &= \begin{pmatrix} 0 & \boldsymbol{e}_1^T \\ \boldsymbol{e}_0 & \boldsymbol{C}_0 \end{pmatrix}\begin{pmatrix} 0 & \boldsymbol{e}_0^T \\ \boldsymbol{e}_0 & \boldsymbol{D}_0 \end{pmatrix}\begin{pmatrix} 1 & \boldsymbol{e}_0^T \\ -\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1}\boldsymbol{e} & \boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1} \end{pmatrix} \\ &= \begin{pmatrix} -\boldsymbol{e}_1^T\boldsymbol{D}_0\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1}\boldsymbol{e} & \boldsymbol{e}_1^T\boldsymbol{D}_0\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1} \\ -\boldsymbol{C}_0\boldsymbol{D}_0\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1}\boldsymbol{e} & \boldsymbol{C}_0\boldsymbol{D}_0\boldsymbol{C}_0^{-1}\boldsymbol{D}_0^{-1} \end{pmatrix}. \end{aligned} \tag{16}$$

It follows from Lemma 1 and (16) that

$$(\boldsymbol{e}_1^T \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1})_j = \frac{(-1)^j}{-j} C_j^r \text{ and } (\boldsymbol{N}_0)_{ij} = \begin{cases} \sum_{p=1}^{i} \frac{1}{p} - \sum_{p=1}^{r-i} \frac{1}{p}, & i = j, \\ \frac{(-1)^{i-j}}{i-j} \frac{C_j^r}{C_i^r}, & i \neq j. \end{cases}$$

This completes the proof. □

By replacing $\boldsymbol{e}_i$ $(i = 1, 2, \ldots, r)$ in the identity matrix $\boldsymbol{I}$ by $\boldsymbol{e}$ and $\widetilde{\boldsymbol{b}} = \gamma \boldsymbol{e}_i$, define the matrices $\boldsymbol{Q}_i, \boldsymbol{I}_i \in \mathbb{R}^{r \times r}$ as follows:

$$\boldsymbol{Q}_i = [\boldsymbol{e}_1 \ \cdots \ \boldsymbol{e}_{i-1} \ \boldsymbol{e} \ \boldsymbol{e}_{i+1} \ \cdots \ \boldsymbol{e}_r], \quad \boldsymbol{I}_i = [\boldsymbol{e}_1 \ \cdots \ \boldsymbol{e}_{i-1} \ -\widetilde{\boldsymbol{b}} \ \boldsymbol{e}_{i+1} \ \cdots \ \boldsymbol{e}_r], \tag{17}$$

where γ is a constant. We obtain the following theorem.

Theorem 1 *For BIM, the solution of (12) is unique if $\boldsymbol{b}$ is chosen as a constant multiply of a column of $\boldsymbol{B}$, i.e., $\boldsymbol{b} = \gamma \boldsymbol{B} \boldsymbol{e}_i$, and the stability function is*

$$R_{n+i}(z) = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}(z)|} = \frac{|\boldsymbol{N}(\gamma)\boldsymbol{Q}_i - z\boldsymbol{I}_i|}{|\boldsymbol{N}(\gamma) - z\boldsymbol{I}|},$$

where $\boldsymbol{N}(\gamma) = \boldsymbol{N}_0 + \gamma \boldsymbol{E}_{i1} \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1}$. Let $\boldsymbol{v} = (1, 2^r, \cdots, r^r)^T$, the error is

$$\boldsymbol{C}_{r+1} = \frac{1}{(r+1)!} (\boldsymbol{D}_0 \boldsymbol{v} - (r+1) \boldsymbol{N}(\gamma)^{-1} \boldsymbol{v}).$$

Proof Following the order conditions (9) and (12), setting $C_{i1} = C_{i2} = \cdots = C_{ir} = 0$, we have

$$\boldsymbol{A} \boldsymbol{D}_0 \boldsymbol{C}_0 = \boldsymbol{B} \boldsymbol{C}_0 \boldsymbol{D}_0 + \boldsymbol{B}_0. \tag{18}$$

Since $\boldsymbol{b} = \gamma \boldsymbol{B} \boldsymbol{e}_i$ $(1 \leq i \leq r)$, we obtain

$$\boldsymbol{B}_0 = \gamma \boldsymbol{B} \boldsymbol{E}_{i1}. \tag{19}$$

From (18) and (19), we have

$$\boldsymbol{B} = \boldsymbol{A} \boldsymbol{N}(\gamma)^{-1}, \qquad \boldsymbol{N}(\gamma) = \boldsymbol{N}_0 + \gamma \boldsymbol{E}_{i1} \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1}. \tag{20}$$

Therefore,

$$\boldsymbol{C}(z) = (\boldsymbol{A} - z\boldsymbol{B}) = \boldsymbol{A} \boldsymbol{N}(\gamma)^{-1} (\boldsymbol{N}(\gamma) - z\boldsymbol{I}). \tag{21}$$

Setting $C_{i,0} = 0$ $(i = 1, 2, \cdots, r)$ in (9), it is clear that

$$\boldsymbol{a} = -\boldsymbol{A}\boldsymbol{e}. \tag{22}$$

Denote $\widetilde{\boldsymbol{b}} = \gamma \boldsymbol{e}_i$, it follows from (22) that

$$\begin{aligned}\boldsymbol{C}_i(z) &= \boldsymbol{A}_i - z\boldsymbol{B}_i \\ &= [\boldsymbol{a}_1 \ \cdots \ \boldsymbol{a}_{i-1} \ -\boldsymbol{a} \ \boldsymbol{a}_{i+1} \ \cdots \ \boldsymbol{a}_k] - z[\boldsymbol{b}_1 \ \cdots \ \boldsymbol{b}_{i-1} \ -\boldsymbol{b} \ \boldsymbol{b}_{i+1} \ \cdots \ \boldsymbol{b}_k] \\ &= \boldsymbol{A}[\boldsymbol{e}_1 \ \cdots \ \boldsymbol{e}_{i-1} \ \boldsymbol{e} \ \boldsymbol{e}_{i+1} \ \cdots \ \boldsymbol{e}_r] - z\boldsymbol{B}[\boldsymbol{e}_1 \ \cdots \ \boldsymbol{e}_{i-1} \ -\widetilde{\boldsymbol{b}} \ \boldsymbol{e}_{i+1} \ \cdots \ \boldsymbol{e}_r] \\ &= \boldsymbol{A}\boldsymbol{N}(\gamma)^{-1}(\boldsymbol{N}(\gamma)\boldsymbol{Q}_i - z\boldsymbol{I}_i). \end{aligned} \tag{23}$$

Combining (13), (21) and (23) implies

$$R_{n+i}(z) = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}(z)|} = \frac{|\boldsymbol{N}(\gamma)\boldsymbol{Q}_i - z\boldsymbol{I}_i|}{|\boldsymbol{N}(\gamma) - z\boldsymbol{I}|}.$$

Let $\boldsymbol{v} = (1, 2^r, \cdots, r^r)^T$, it follows from (9) that

$$\boldsymbol{C}_{r+1} = \frac{1}{(r+1)!}(\boldsymbol{A}\boldsymbol{D}_0\boldsymbol{v} - (r+1)\boldsymbol{B}\boldsymbol{v}).$$

Setting $\boldsymbol{A} = \boldsymbol{I}$, we have $\boldsymbol{B} = \boldsymbol{N}(\gamma)^{-1}$ and

$$\boldsymbol{C}_{r+1} = \frac{1}{(r+1)!}(\boldsymbol{D}_0\boldsymbol{v} - (r+1)\boldsymbol{N}(\gamma)^{-1}\boldsymbol{v}).$$

This completes the proof. □

From Theorem 1, it is evident that the stability function of BIM is uniquely determined by $\boldsymbol{N}(\gamma)$, and is independent of the individual matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Moreover, Corollary 1 shows that $\boldsymbol{N}(\gamma)$ can be presented explicitly by a given γ. The subsequent step involves constructing some BIM by choosing γ and studying the stability.

Theorem 2 *For $r = 2, 3, 4$ and 5, some BIM can be constructed by choosing $\boldsymbol{b}$ as $\boldsymbol{b} = \gamma\boldsymbol{B}\boldsymbol{e}_1$ and $\boldsymbol{b} = \gamma\boldsymbol{B}\boldsymbol{e}_r$, respectively. The details are presented in Table 1.*

Proof For brevity, we only present the proof for the case $\boldsymbol{b} = \gamma\boldsymbol{B}\boldsymbol{e}_1$. The coefficients of the stability functions are given in Table 6 in Appendix A, the other cases can be proved similarly.

If $r = 2$, it is clear that $r = 2r - 2$. The signs of the denominator $C(z)$ are alternating if $\gamma > -1/2$. Moreover, $\lim\limits_{z\to\infty} R_{n+2}(z) = \lim\limits_{z\to\infty} |\boldsymbol{C}_2(z)|/|\boldsymbol{C}(z)| = 0$ holds for any $\gamma \in \mathbb{R}$. From Theorem 4.9 in [9] and Definition 1, we observe that this method is block rth-step L-stable when $-1/2 < \gamma < \infty$. Additionally,

$$\lim_{z\to\infty} R_{n+1}(z) = \lim_{z\to\infty} \frac{|\boldsymbol{C}_1(z)|}{|\boldsymbol{C}(z)|} = \lim_{z\to\infty} \frac{|-2\gamma z^2 - z + 4\gamma + 2|}{|2z^2 + (-4\gamma - 3)z + 4\gamma + 2|} \leq 1$$

yields $-1 \leq \gamma \leq 1$. From Theorem 4.9 in [9], we see that the method is block r-step A-stable if $-1/2 < \gamma \leq 1$. Moreover, we obtain $\lim\limits_{z\to\infty} R_{n+1}(z) = 0$ if $\gamma = 0$, so this method is block r-step L-stable when $\gamma = 0$.

Table 1 Stability of BIM: ①-block r-step A-stable, ②-block rth-step A-stable, ③-block rth-step L-stable and ④-not block rth-step A-stable

	$r=2$		$r=3$		$r=4$		$r=5$	
	Stability	γ	Stability	γ	Stability	γ	Stability	γ
$\boldsymbol{b}=\gamma \boldsymbol{B}\boldsymbol{e}_1$	③	$\left(-\frac{1}{2},\infty\right)$	③	$\left[\frac{1}{3},+\infty\right)$	③	$\left[\frac{3+\sqrt{457}}{64},+\infty\right)$	④	$\emptyset$
	①	$\left(-\frac{1}{2},1\right]$	①	$\left[\frac{9}{19},1\right]$	①	$\left[\frac{75+\sqrt{102169}}{862},1\right]$		
$\boldsymbol{b}=\gamma \boldsymbol{B}\boldsymbol{e}_r$	①	$[-1,1)$	②	1	②	-1	②	1

For the case $r = 3$, it is easy to see that $r = 2r - 3$ and the signs of the denominator $C(z)$ are alternating if $\gamma > -1/3$. Moreover, for any $y \in \mathbb{R}$, we obtain

$$
\begin{aligned}
E_{n+1}(y) &= |\boldsymbol{C}(\mathbf{i}y)|^2 - |\boldsymbol{C}_1(\mathbf{i}y)|^2 = (37 - 36\gamma^2)y^6 + (171\gamma^2 - 24\gamma - 27)y^4, \\
E_{n+2}(y) &= |\boldsymbol{C}(\mathbf{i}y)|^2 - |\boldsymbol{C}_2(\mathbf{i}y)|^2 = 36y^6 + (288\gamma^2 + 24\gamma - 24)y^4, \\
E_{n+3}(y) &= |\boldsymbol{C}(\mathbf{i}y)|^2 - |\boldsymbol{C}_3(\mathbf{i}y)|^2 = 36y^6 + (243\gamma^2 - 27)y^4.
\end{aligned}
$$

A straightforward computation shows that $E_{n+1}(y) \geq 0$ for $\gamma \in [-1, -1/3] \cup [9/19, 1]$, $E_{n+2}(y) \geq 0$ for $\gamma \in (-\infty, -1/3] \cup [1/4, \infty)$ and $E_{n+3}(y) \geq 0$ for $\gamma \in (-\infty, -1/3] \cup [1/3, \infty)$. From Theorem 4.10 in [9], we see that $R_{n+1}(z)$ is A-stable for $\gamma \in [9/19, 1]$, $R_{n+2}(z)$ is A-stable for $\gamma \in [1/4, \infty)$ and $R_{n+3}(z)$ is A-stable for $\gamma \in [1/3, \infty)$. Therefore, this method is block r-step A-stable if $\gamma \in [9/19, 1]$. Since $\lim_{z\to\infty} R_{n+3}(z) = 0$, this method is block rth-step L-stable if $\gamma \in [1/3, \infty)$. Moreover, $\lim_{z\to\infty} R_{n+2}(z) = 0$ holds for $\gamma \in [1/3, \infty)$, but $\lim_{z\to\infty} R_{n+1}(z) \neq 0$ in this case, so this method is not block r-step L-stable

For the case $r = 4$, set $E_{n+i}(y) = |\boldsymbol{C}(\mathbf{i}y)|^2 - |\boldsymbol{C}_i(\mathbf{i}y)|^2$ $(i = 1, 2, 3, 4)$, for any $y \in \mathbb{R}$, we obtain

$$
\begin{aligned}
E_{n+1}(y) &= (144 - 144\gamma^2)y^8 + (1724\gamma^2 - 300\gamma - 224)y^6, \\
E_{n+2}(y) &= 144y^8 + (2240\gamma^2 - 112\gamma - 216)y^6, \\
E_{n+3}(y) &= 144y^8 + (2268\gamma^2 - 108\gamma - 216)y^6, \\
E_{n+4}(y) &= 144y^8 + (2048\gamma^2 - 192\gamma - 224)y^6.
\end{aligned}
$$

It is clear that $E_{n+1}(y) \geq 0$ for $\gamma \in \left[-1, \frac{75-\sqrt{102169}}{862}\right] \cup \left[\frac{75+\sqrt{102169}}{862}, 1\right]$, $E_{n+2}(y) \geq 0$ for $\gamma \in \left(-\infty, \frac{7-\sqrt{7609}}{280}\right] \cup \left[\frac{7+\sqrt{7609}}{280}, \infty\right)$, $E_{n+3}(y) \geq 0$ for $\gamma \in (-\infty, -2/7] \cup [1/3, \infty)$ and $E_{n+4}(y) \geq 0$ for $\gamma \in \left(-\infty, \frac{3-\sqrt{457}}{64}\right] \cup \left[\frac{3+\sqrt{457}}{64}, \infty\right)$. So $R_{n+i}(z)$ are I-stable for some γ. Let $z = \alpha + \mathbf{i}\beta$ $(\alpha \leq 0)$, we have

$$
\begin{aligned}
&|\boldsymbol{C}(z)|^2 - |\boldsymbol{C}_4(z)|^2 \\
&= 144\beta^8 + \left[576\alpha^2 + (-1152\gamma - 600)\alpha + 2048\gamma^2 - 192\gamma - 224\right]\beta^6 \\
&+ \left[864\alpha^4 + (-3456\gamma - 1800)\alpha^3 + (6144\gamma^2 + 4416\gamma + 1008)\alpha^2 + (-11776\gamma^2 - 1472\gamma + 344)\alpha\right]\beta^4 \\
&+ \left[576\alpha^6 + (-3456\gamma - 1800)\alpha^5 + (6144\gamma^2 + 9408\gamma + 2688)\alpha^4 + (-23552\gamma^2 - 13312\gamma - 2192)\alpha^3\right. \\
&\left. + (15360\gamma^2 + 5184\gamma + 480)\alpha^2 + (-13440\gamma^2 - 4992\gamma - 480)\alpha\right]\beta^2 \\
&+ 144\alpha^8 + (-1152\gamma - 600)\alpha^7 + (2048\gamma^2 + 4800\gamma + 1456)\alpha^6 + (-11776\gamma^2 - 11840\gamma - 2536)\alpha^5 \\
&+ (15360\gamma^2 + 14400\gamma + 2784)\alpha^4 + (-38016\gamma^2 - 21888\gamma - 3168)\alpha^3 + (9216\gamma^2 \\
&+ 6912\gamma + 1152)\alpha^2 + (-18432\gamma^2 - 9216\gamma - 1152)\alpha.
\end{aligned}
$$

We observe that the coefficients of the powers of α are either constants, or polynomials of degree 1 or 2 of γ, A tedious and straightforward computation shows that all the coefficients of the odd powers of α are less than or equal to 0 and all the coefficients

of the even powers of α are greater than or equal to 0 if $\gamma \in \left[\frac{3+\sqrt{457}}{64}, \infty\right)$. Then, $|\boldsymbol{C}(z)|^2 \geq |\boldsymbol{C}_4(z)|^2$ holds for all $\alpha \leq 0$. Further, it is clear that $\lim_{z\to\infty} R_{n+4}(z) = 0$. Therefore, this method is block rth-step L-stable if $\gamma \in \left[\frac{3+\sqrt{457}}{64}, \infty\right)$. Analogously, we have $\lim_{z\to\infty} R_{n+3}(z) = 0$ and $\lim_{z\to\infty} R_{n+2}(z) = 0$ if $\gamma \in \left[\frac{3+\sqrt{457}}{64}, \infty\right)$, but $\lim_{z\to\infty} R_{n+1}(z) \neq 0$ in this case, so this method is not block r-step L-stable. Moreover, we are able to prove this method is also block r-step A-stable if $\gamma \in \left[\frac{75+\sqrt{102169}}{862}, 1\right]$.

For the case $r = 5$, we obtain

$$\begin{aligned} E_{n+5}(y) &= |\boldsymbol{C}(\mathbf{i}y)|^2 - |\boldsymbol{C}_5(\mathbf{i}y)|^2 \\ &= 3600y^{10} + (84375\gamma^2 - 12000\gamma - 8375)y^8 \\ &\quad + (-46875\gamma^2 + 2500\gamma + 2375)y^6. \end{aligned}$$

It is easy to see that there exists no $\gamma \in \mathbb{R}$ (the possible γ is an empty set $\emptyset$) such that $E_{n+5}(y) \geq 0$ holds for all $y \in \mathbb{R}$. So $R_{n+5}(z)$ is not I-stable and this method is not block rth-step A-stable. We denote it by the notation ④ in Table 1. □

Remark 1 Note that the methods mentioned in Theorem 2 are block r-step L-stable for $r = 2$ and $\gamma = 0$, and we can also prove that there exists no block r-step L-stable methods if $k \geq 3$.

From Table 1, we see that there exists no fifth-order BIM which satisfies the stability ①, ② and ③ except the case $\boldsymbol{b} = \gamma \boldsymbol{B}\boldsymbol{e}_5$. In this case, we have

$$|\boldsymbol{C}(z)| = |\boldsymbol{C}_5(-z)| = -60z^5 + 149z^4 - 275z^3 + 360z^2 - 300z + 120$$

if $\gamma = 1$. So $E_{n+5}(y) = 0$ holds for all $y \in \mathbb{R}$, and $R_{n+5}(z)$ is I-stable. Unfortunately, we are unable to prove mathematically that $R_{n+5}(z)$ is analytic. Instead, we obtain the following five roots of $|\boldsymbol{C}(z)|$ numerically:

$$0.0219 + 1.4164\boldsymbol{i},\ 0.0219 - 1.4164\boldsymbol{i},\ 0.7499 + 0.7058\boldsymbol{i},\ 0.7499 - 0.7058\boldsymbol{i},\ 0.9398.$$

This implies that $R_{n+5}(z)$ is analytic, and this method is block rth-step A-stable.

Table 1 shows that the BIM is block rth-step L-stable only when $\boldsymbol{b} = \gamma \boldsymbol{B}\boldsymbol{e}_1$, and the order is no more than 4. Now we will construct more higher order BIM with L-stability by choosing different $\boldsymbol{b}$.

Corollary 2 *For BIM, the solution of (12) is unique if $\boldsymbol{b}$ is chosen as a linear combination of the columns of $\boldsymbol{B}$, and the stability function is*

$$R_{n+i}(z) = \frac{|\boldsymbol{C}_i(z)|}{|\boldsymbol{C}(z)|} = \frac{|\widetilde{\boldsymbol{N}}\boldsymbol{Q}_i - z\boldsymbol{I}_i|}{|\widetilde{\boldsymbol{N}} - z\boldsymbol{I}|}, \tag{24}$$

where $\boldsymbol{Q}_i$ *and* $\boldsymbol{I}_i$ *are defined in (17),* $\widetilde{\boldsymbol{N}} \equiv \widetilde{\boldsymbol{N}}(\gamma_1, \cdots, \gamma_r) = \boldsymbol{N}_0 + \sum_{i=1}^{r} \gamma_i \boldsymbol{E}_{i1} \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1}$, γ_i *are constants, and* $\widetilde{\boldsymbol{b}} = \sum_{i=1}^{r} \gamma_i \boldsymbol{e}_i$. *The error*

$$\boldsymbol{C}_{r+1} = \frac{1}{(r+1)!}(\boldsymbol{D}_0 \boldsymbol{v} - (r+1)\widetilde{\boldsymbol{N}}^{-1}\boldsymbol{v}).$$

Here, $\boldsymbol{N}_0$ *and* $\boldsymbol{v}$ *are the same as that in Theorem 1.*

We present some L-stable BIM defined by matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ satisfying

$$\begin{cases} \boldsymbol{A} = \boldsymbol{B}\widetilde{\boldsymbol{N}}, \quad \widetilde{\boldsymbol{N}} = \boldsymbol{C}_0 \boldsymbol{D}_0 \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1} + \sum_{i=1}^{r} \gamma_i \boldsymbol{E}_{i1} \boldsymbol{C}_0^{-1} \boldsymbol{D}_0^{-1}, \\ \boldsymbol{a} = -\boldsymbol{A}\boldsymbol{e} = -\boldsymbol{B}\widetilde{\boldsymbol{N}}\boldsymbol{e}, \\ \boldsymbol{b} = \boldsymbol{A}\widetilde{\boldsymbol{x}} - \boldsymbol{B}\boldsymbol{e} = \boldsymbol{B}\widetilde{\boldsymbol{N}}\widetilde{\boldsymbol{x}} - \boldsymbol{B}\boldsymbol{e}, \end{cases} \tag{25}$$

where $\widetilde{\boldsymbol{x}} = (1, 2, \ldots, r)^T$. Note that if $\widetilde{\boldsymbol{b}} = \gamma_1 \boldsymbol{e}_1$, $\boldsymbol{N}(\gamma)$ is a special case of $\widetilde{\boldsymbol{N}}$, such as $\boldsymbol{N}(\gamma)$ is exactly $\widetilde{\boldsymbol{N}}(\gamma, 0)$, $\widetilde{\boldsymbol{N}}(\gamma, 0, 0)$ and $\widetilde{\boldsymbol{N}}(\gamma, 0, 0, 0)$ for the cases $r = 2, 3, 4$, respectively. From Theorem 2 and Corollary 2, we come up some matrices $\widetilde{\boldsymbol{N}}$ for $2 \le r \le 8$ as follow:

$$\widetilde{\boldsymbol{N}}_2 \equiv \widetilde{\boldsymbol{N}}(1, 0) = \begin{pmatrix} 2 & 0 \\ -2 & 3/2 \end{pmatrix}, \quad \widetilde{\boldsymbol{N}}_3 \equiv \widetilde{\boldsymbol{N}}(1, 0, 0) = \begin{pmatrix} 5/2 & -1/2 & 1/6 \\ -1 & 1/2 & 1/3 \\ 3/2 & -3 & 11/6 \end{pmatrix},$$

$$\widetilde{\boldsymbol{N}}_4 \equiv \widetilde{\boldsymbol{N}}(1, -1, 0, 0) = \begin{pmatrix} 19/6 & -3/2 & 5/6 & -1/6 \\ -14/3 & 3 & -2/3 & 1/6 \\ 1/2 & -3/2 & 5/6 & 1/4 \\ -4/3 & 3 & -4 & 25/12 \end{pmatrix},$$

$$\widetilde{\boldsymbol{N}}_5 \equiv \widetilde{\boldsymbol{N}}(0.25, -0.55, 0.05, 0, 0) = \begin{pmatrix} 1/6 & 3/4 & -1/6 & 1/48 & 0 \\ -13/4 & 29/12 & -5/6 & 7/16 & -23/300 \\ 1/2 & -5/4 & 1/2 & 7/16 & -1/25 \\ -1/3 & 1 & -2 & 13/12 & 1/5 \\ 5/4 & -10/3 & 5 & -5 & 137/60 \end{pmatrix},$$

$$\widetilde{\boldsymbol{N}}_6 \equiv \widetilde{\boldsymbol{N}}(0, -1, 0, 0, 0, 0) = \begin{pmatrix} -77/60 & 5/2 & -5/3 & 5/6 & -1/4 & 1/30 \\ -32/5 & 83/12 & -16/3 & 13/4 & -16/15 & 3/20 \\ 3/20 & -3/4 & 0 & 3/4 & -3/20 & 1/60 \\ -2/15 & 1/2 & -4/3 & 7/12 & 2/5 & -1/30 \\ 1/4 & -5/6 & 5/3 & -5/2 & 77/60 & 1/6 \\ -6/5 & 15/4 & -20/3 & 15/2 & -6 & 49/20 \end{pmatrix},$$

$$\widetilde{\boldsymbol{N}}_7 \equiv \widetilde{\boldsymbol{N}}(0, -0.2, 0.6, 0.5, 0, 0, 0) = \begin{pmatrix} -29/20 & 3 & -5/2 & 5/3 & -3/4 & 1/5 & -1/42 \\ -26/15 & 79/60 & -2/3 & 11/12 & -38/75 & 3/20 & -2/105 \\ 43/10 & -69/10 & 27/4 & -17/4 & 111/50 & -19/30 & 11/140 \\ 103/30 & -99/20 & 29/6 & -33/8 & 27/10 & -41/60 & 17/210 \\ 1/12 & -1/3 & 5/6 & -5/3 & 47/60 & 1/3 & -1/42 \\ -1/5 & 3/4 & -5/3 & 5/2 & -3 & 29/20 & 1/7 \\ 7/6 & -21/5 & 35/4 & -35/3 & 21/2 & -7 & 363/140 \end{pmatrix},$$

$$\widetilde{N}_8 \equiv \widetilde{N}(-0.2, -0.1, 0.8, 0.5, 0, 0, 0, 0) =$$
$$\begin{pmatrix} -447/140 & 63/10 & -217/30 & 77/12 & -399/100 & 49/30 & -83/210 & 3/70 \\ -38/35 & 9/20 & 2/15 & 1/2 & -34/75 & 13/60 & -2/35 & 11/1680 \\ 453/70 & -117/10 & 869/60 & -51/4 & 423/50 & -107/30 & 123/140 & -27/280 \\ 416/105 & -34/5 & 128/15 & -35/4 & 32/5 & -38/15 & 64/105 & -37/560 \\ 1/28 & -1/6 & 1/2 & -5/4 & 9/20 & 1/2 & -1/14 & 1/168 \\ -2/35 & 1/4 & -2/3 & 5/4 & -2 & 19/20 & 2/7 & -1/56 \\ 1/6 & -7/10 & 7/4 & -35/12 & 7/2 & -7/2 & 223/140 & 1/8 \\ -8/7 & 14/3 & -56/5 & 35/2 & -56/3 & 14 & -8 & 761/280 \end{pmatrix}.$$

2.3 Construction of BIM with positive definite BIM matrices

In the previous section, some BIM with up to 8th order of accuracy were introduced. For the given $\widetilde{N}$, the matrices $\boldsymbol{B}$ and $\boldsymbol{A}$ can be chosen flexibly and it does not affect the stability and the error. However, there exists no strategy to chose $\boldsymbol{A}$ and $\boldsymbol{B}$ such that the corresponding BIM has desirable properties. Since $\widetilde{N}$ is positive stable but not positive definite, the rest of the section focuses on the selection of $\boldsymbol{A}$ and $\boldsymbol{B}$ with the desired positive definiteness and sparsity.

Now we introduce some lemmas about the positive stable matrix.

Lemma 2 ([19]) *$\boldsymbol{N}$ is positive stable if and only if there exists a symmetric positive definite (SPD) matrix $\boldsymbol{B}$ such that*

$$\boldsymbol{B}\boldsymbol{N} + \boldsymbol{N}^T\boldsymbol{B} \tag{26}$$

is positive definite.

Especially, if (26) is positive definite with a positive diagonal matrix $\boldsymbol{B}$, $\boldsymbol{N}$ is also called diagonally stable or "Lyapunov diagonally stable"[3, 10]. However, it is difficult to find a positive diagonal matrix $\boldsymbol{B}$ for a given diagonally stable $\boldsymbol{N}$. Note that a positive stable matrix may not be diagonally stable. The following lemma provides a necessary condition for a diagonally stable $\boldsymbol{N}$, i.e., all the diagonal elements of $\boldsymbol{N}$ are positive.

Lemma 3 ([3]) *A matrix $\boldsymbol{N}$ is diagonally stable if and only if for every nonzero positive semi-definite matrix $\boldsymbol{B}$, $\boldsymbol{B}\boldsymbol{N}$ has a positive diagonal element.*

Below, we present three pairs of $\boldsymbol{B}$ and $\boldsymbol{A}$ according to the given $\widetilde{N}$:

(1) $\boldsymbol{B}$ is chosen as a SPD matrix such that $\boldsymbol{A} = \boldsymbol{B}\widetilde{N}$ is positive definite.

From Lemma 2, we see that there exists a SPD matrix $\boldsymbol{B}$ such that $\boldsymbol{B}\widetilde{N}$ is positive definite, but the matrix $\boldsymbol{B}$ is difficult to find for a given $\widetilde{N}$.

(2) $\boldsymbol{B}$ is chosen as a positive diagonal matrix such that $\boldsymbol{A} = \boldsymbol{B}\widetilde{N}$ is positive definite.

By using the techniques presented in [4], we present some computed $\boldsymbol{B}$ for $\widetilde{N}(1, 0)$, $\widetilde{N}(1, 0, 0)$, $\widetilde{N}(1, -1, 0, 0)$ and $\widetilde{N}(0.25, -0.55, 0.05, 0, 0)$ as follows:

$$\boldsymbol{B}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \boldsymbol{B}_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/10 \end{pmatrix}, \quad \boldsymbol{B}_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 4/5 & 0 & 0 \\ 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/20 \end{pmatrix},$$

$$\boldsymbol{B}_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1/5 & 0 & 0 & 0 \\ 0 & 0 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 1/10 & 0 \\ 0 & 0 & 0 & 0 & 1/100 \end{pmatrix}.$$

For other cases, the matrices $\widetilde{\boldsymbol{N}}$ are only positive stable but not diagonally stable since some diagonal elements are not positive.

(3) $\boldsymbol{B}$ is chosen as an identity matrix such that $\boldsymbol{A} = \widetilde{\boldsymbol{N}}$.

It is trivial to choose such matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ since $\widetilde{\boldsymbol{N}}$ can be given explicitly, however the matrix $\boldsymbol{A}$ is only positive stable but not positive definite.

Now, we present some BIM with L-stability.

Scheme 1 *(BIM with L-stability) Let $\widetilde{\boldsymbol{N}}$ be defined by $\widetilde{\boldsymbol{N}}_r$ for $2 \le r \le 8$, the methods are described by the BIM matrices and vectors*

$$\begin{cases} \boldsymbol{B} = \boldsymbol{B}_r, \ \boldsymbol{A} = \boldsymbol{B}\widetilde{\boldsymbol{N}}_r, \ \boldsymbol{a} = -\boldsymbol{A}\boldsymbol{e}, \ \boldsymbol{b} = \boldsymbol{A}\widetilde{\boldsymbol{x}} - \boldsymbol{B}\boldsymbol{e}, & \text{for} \quad r = 2, 3, 4, 5, \\ \boldsymbol{B} = \boldsymbol{I}, \quad \boldsymbol{A} = \boldsymbol{B}\widetilde{\boldsymbol{N}}_r, \ \boldsymbol{a} = -\boldsymbol{A}\boldsymbol{e}, \ \boldsymbol{b} = \boldsymbol{A}\widetilde{\boldsymbol{x}} - \boldsymbol{B}\boldsymbol{e}, & \text{for} \quad r = 6, 7, 8. \end{cases} \tag{27}$$

3 Preconditioners for BIM for parabolic equations

In this section, we extend the traditional finite element theory for parabolic problems discretized by the backward Euler or Crank–Nicolson schemes to BIM. Then we introduce some parallel DD preconditioners for the model problem (1) discretized using BIM in time and regular finite element in space.

Let u_i be the solution at time t_i, by using the traditional time-stepping method, such as backward Euler scheme, for the time discretization, the variational form of (1) at time t_i is to find $u_i \in H_0^1(\Omega)$, $i \ge 1$, such that

$$(u_i, v) + \tau B(u_i, v) - (u_{i-1}, v) = \tau(f, v) \quad \forall v \in H_0^1(\Omega), \tag{28}$$

where the bilinear form is

$$B(u, v) = \int_\Omega (\alpha \nabla u \cdot \nabla v + \beta \cdot \nabla u v) dx \quad \forall u, \ v \in H_0^1(\Omega).$$

We assume there exists a constant $c_0 > 0$ such that

$$B(v, v) \ge c_0 \|v\|_1^2 \ \ \forall v \in H_0^1(\Omega). \tag{29}$$

Different from the traditional time-stepping methods, solutions at r time steps are computed simultaneously with BIM. Denote $\boldsymbol{u} = (u_1, u_2, \cdots, u_r)^T \in \boldsymbol{H}_0^1(\Omega) \equiv (H_0^1(\Omega))^r$, $\boldsymbol{f} = (f_1, f_2, \cdots, f_r)^T \in \boldsymbol{L}^2(\Omega \times [0, T]) \equiv (L^2(\Omega \times [0, T]))^r$. Set $\nabla \boldsymbol{u} =$

$(\nabla u_1, \nabla u_2, \cdots, \nabla u_r)^T$. By using BIM for the time discretization, the variational form of (1) is

$$\mathcal{A}_\tau(\boldsymbol{u}, \boldsymbol{v}) \equiv (\boldsymbol{A}\boldsymbol{u}, \boldsymbol{v}) + \tau \mathcal{B}(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{g}, \boldsymbol{v}), \tag{30}$$

where

$$\mathcal{B}(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega (\alpha \boldsymbol{B}\nabla \boldsymbol{u} \cdot \nabla \boldsymbol{v} + \boldsymbol{B}(\beta \cdot \nabla \boldsymbol{u})\boldsymbol{v}) dx \quad \forall \boldsymbol{u},\ \boldsymbol{v} \in \boldsymbol{H}_0^1(\Omega)$$

and

$$(\boldsymbol{g}, \boldsymbol{v}) = \tau(\boldsymbol{B}\boldsymbol{f}, \boldsymbol{v}) - (\boldsymbol{a} \otimes u_0, \boldsymbol{v}) + \tau(\boldsymbol{b} \otimes f_0, \boldsymbol{v}) - \tau(\boldsymbol{b} \otimes \alpha \nabla u_0, \nabla \boldsymbol{v}) - \tau\,(\boldsymbol{b} \otimes (\beta \cdot \nabla u_0), \boldsymbol{v})\,.$$

Define the $\|\cdot\|_\tau$ norm $\|\boldsymbol{v}\|_\tau^2 = \|\boldsymbol{v}\|^2 + \tau|\boldsymbol{v}|_1^2$, where $\|\cdot\|$ and $|\cdot|_1$ are the $\boldsymbol{L}^2(\Omega)$ norm and $\boldsymbol{H}_0^1(\Omega)$ seminorm in the Sobolev space, respectively. We obtain some basic estimates for the bilinear form $\mathcal{A}_\tau(\cdot, \cdot)$.

Lemma 4 *Suppose that $\boldsymbol{A}$ is positive definite and $\boldsymbol{B}$ is positive diagonal, we have*

$$\mathcal{A}_\tau(\boldsymbol{u}, \boldsymbol{v}) \le C\|\boldsymbol{u}\|_\tau \|\boldsymbol{v}\|_\tau \quad \forall \boldsymbol{u}, \boldsymbol{v} \in (H_0^1(\Omega))^k \tag{31}$$

and

$$\mathcal{A}_\tau(\boldsymbol{v}, \boldsymbol{v}) \ge c_1\|\boldsymbol{u}\|_\tau^2 \quad \forall \boldsymbol{v} \in (H_0^1(\Omega))^k, \tag{32}$$

where $C > 0$ and $c_1 > 0$ are independent of τ.

Proof Since $\boldsymbol{B}$ is positive diagonal, it follows from the Cauchy-Schwarz inequality that

$$\begin{aligned}
\mathcal{A}_\tau(\boldsymbol{u}, \boldsymbol{v}) &= \sum_{i,j=1}^r a_{ij}(u_i, v_j) + \tau \sum_{i=1}^r b_{ii}(\alpha \nabla u_i, \nabla v_i) + \tau \sum_{i}^r b_{ii}(\beta \cdot \nabla u_i, v_i) \\
&\le a_{max} \sum_{i,j=1}^r \|u_i\|\|v_j\| + C\tau b_{max} \sum_{i=1}^r (\|\nabla u_i\|\|\nabla v_i\| + \|\nabla u_i\|\|v_i\|) \\
&\le C(\|\boldsymbol{u}\|\|\boldsymbol{v}\| + \tau\|\nabla \boldsymbol{u}\|\|\nabla \boldsymbol{v}\|) \\
&\le C\|\boldsymbol{u}\|_\tau \|\boldsymbol{v}\|_\tau,
\end{aligned}$$

where $a_{max} = \max_{i,j=1}^r |a_{ij}|$ and $b_{max} = \max_{i=1}^r |b_{ii}|$. Since $\boldsymbol{A}$ is positive definite and $\boldsymbol{B}$ is positive diagonal, it follows from (29) that

$$\begin{aligned}
\mathcal{A}_\tau(\boldsymbol{v}, \boldsymbol{v}) &= \sum_{i,j=1}^r a_{ij}(v_i, v_j) + \tau \sum_{i}^r b_{ii} B(v_i, v_i) \\
&\ge \lambda_{min}(\boldsymbol{A}_H)\|\boldsymbol{v}\|^2 + c_0 b_{min}\tau \sum_{i=1}^r \|v_i\|_1^2 \\
&\ge c_1\|\boldsymbol{v}\|_\tau^2,
\end{aligned}$$

where $c_1 = \min\{\lambda_{min}(\boldsymbol{A}_H), c_0 b_{min}\}$, $\lambda_{min}(\boldsymbol{A}_H)$ denotes the minimum eigenvalue of $\boldsymbol{A}_H$ and $b_{min} = \min_{i=1}^{r} |b_{ii}|$. This completes the proof. □

We remark that $\mathcal{A}_\tau(\cdot,\cdot)$ is nonsymmetric but positive since $\boldsymbol{A}$ is positive definite and $\boldsymbol{B}$ is positive diagonal. Hence, the solution of (30) is unique by the theorems of Riesz or Lax and Milgram. Let $\boldsymbol{V}_h \in (H_0^1(\Omega))^k$ be a piecewise linear continuous finite element space. The finite element solution of (30) is to find $\boldsymbol{u}_h \in \boldsymbol{V}_h$ such that

$$\mathcal{A}_\tau(\boldsymbol{u}_h, \boldsymbol{v}_h) = (\boldsymbol{g}, \boldsymbol{v}_h) \quad \forall \boldsymbol{v}_h \in \boldsymbol{V}_h. \tag{33}$$

Theorem 3 *Suppose that $\boldsymbol{A}$ is positive definite and $\boldsymbol{B}$ is positive diagonal, there exists a constant $C > 0$ such that*

$$\|\boldsymbol{u} - \boldsymbol{u}_h\| \le \frac{Ch}{\sqrt{\tau}} \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau \quad \boldsymbol{u}_h \in \boldsymbol{V}_h$$

and

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau \le C \|\boldsymbol{u}\|_\tau.$$

Proof By the definition of the τ-norm, we have

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|^2 \le Ch^2 |\boldsymbol{u} - \boldsymbol{u}_h|_1^2 \le \frac{Ch^2}{\tau} \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau^2 - \frac{Ch^2}{\tau} \|\boldsymbol{u} - \boldsymbol{u}_h\|^2,$$

it is clear that

$$\|\boldsymbol{u} - \boldsymbol{u}_h\| \le \frac{Ch}{\sqrt{Ch^2 + \tau}} \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau \le \frac{Ch}{\sqrt{\tau}} \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau. \tag{34}$$

Combining (31) and (32) implies

$$\mathcal{A}_\tau(\boldsymbol{u} - \boldsymbol{u}_h, \boldsymbol{u}) \le C \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau \|\boldsymbol{u}\|_\tau$$

and

$$\mathcal{A}_\tau(\boldsymbol{u} - \boldsymbol{u}_h, \boldsymbol{u}) = \mathcal{A}_\tau(\boldsymbol{u} - \boldsymbol{u}_h, \boldsymbol{u} - \boldsymbol{u}_h) \ge c_1 \|\boldsymbol{u} - \boldsymbol{u}_h\|_\tau^2.$$

Then we obtain $\|\boldsymbol{u} - \boldsymbol{u}\|_\tau \le C \|\boldsymbol{u}_h\|_\tau$. □

Discretized by the finite element basis functions, the variational form (33) is equivalent to the following linear system of equations

$$\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{U}} \equiv (\boldsymbol{A} \otimes \boldsymbol{M} + \tau \boldsymbol{B} \otimes \boldsymbol{K})\, \boldsymbol{\mathcal{U}} = \boldsymbol{\mathcal{F}}, \quad \boldsymbol{\mathcal{A}} \in \mathbb{R}^{rN \times rN},\ \boldsymbol{\mathcal{F}} \in \mathbb{R}^{rN}, \tag{35}$$

where $\boldsymbol{A} \in \mathbb{R}^{r\times r}$ and $\boldsymbol{B} \in \mathbb{R}^{r\times r}$ are the BIM matrices, $\boldsymbol{M} \in \mathbb{R}^{N\times N}$ and $\boldsymbol{K} \in \mathbb{R}^{N\times N}$ denote the mass matrix and the stiffness matrix, respectively. $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{F}}$ are vectors corresponding to the nodal values of $\boldsymbol{u}_h$ and $\boldsymbol{g}$.

Next we introduce several overlapping Schwarz preconditioners for solving (35) with a Krylov subspace method. First we define a sequence of nested conforming

meshes $\mathcal{T}_l = \{K_i^l\}_{i=1}^{N^l}$ $(l = 1, 2, \ldots, L)$ obtained by uniform refinement of an initial mesh $\mathcal{T}_0$. Set $h_l = \max_i \operatorname{diam}(K_i^l)$, and denote by H and h as the mesh sizes of $\mathcal{T}_0$ and $\mathcal{T}_L$, respectively. Then, we denote $V_l = V_{h_l}$ $l = 0, 1, \ldots, L$ as the space of continuous piecewise linear functions associated with the mesh $\mathcal{T}_l$. Let $\Omega = \bigcup_{i=1}^{N_l} \Omega_{li}'$ be an overlapping decomposition on each level l $(l \geq 1)$, where Ω_{li}' is extended from the nonoverlapping subdomains Ω_{li} by adding several layers of fine mesh elements. For $l = 1, 2, \ldots, L$ and $i = 1, 2, \ldots, N_l$, the finite element space on Ω_{li}' is defined as $V_l^i = V_l \cap H_0^1(\Omega_{li}')$. Let N_0, N_{li} and N be the dimension of V_0, V_l^i and V_h, we define the restriction matrices $\boldsymbol{R}_0 \in \mathbb{R}^{N_0 \times N}$: $V_h \to V_0$ and $\boldsymbol{R}_l^i \in \mathbb{R}^{N_{li} \times N}$: $V_h \to V_l^i$. Extend the single time step spaces to r time steps,

$$\boldsymbol{V}_h = (V_h)^r, \quad \boldsymbol{V}_0 = (V_0)^r \quad \text{and} \quad \boldsymbol{V}_l^i = (V_l^i)^r, \ (i = 1, 2, \ldots, N_l),$$

we have the following decomposition in space and time,

$$\boldsymbol{V}_h = \boldsymbol{\mathcal{R}}_0^T \boldsymbol{V}_0 + \sum_{l=1}^{L} \sum_{i=1}^{N_l} {\boldsymbol{\mathcal{R}}_l^i}^T \boldsymbol{V}_l^i, \tag{36}$$

where the block restriction matrices $\boldsymbol{\mathcal{R}}_0 \in \mathbb{R}^{rN_0 \times rN}$: $\boldsymbol{V}_h \to \boldsymbol{V}_0$ and $\boldsymbol{\mathcal{R}}_l^i \in \mathbb{R}^{rN_{li} \times rN}$: $\boldsymbol{V}_h \to \boldsymbol{V}_l^i$ are defined as

$$\boldsymbol{\mathcal{R}}_0 = diag\{\boldsymbol{R}_0, \boldsymbol{R}_0, \cdots, \boldsymbol{R}_0\} \quad \text{and} \quad \boldsymbol{\mathcal{R}}_l^i = diag\{\boldsymbol{R}_l^i, \boldsymbol{R}_l^i, \cdots, \boldsymbol{R}_l^i\}.$$

Denote $\boldsymbol{\mathcal{B}}_0 = \boldsymbol{\mathcal{R}}_0^T \boldsymbol{\mathcal{A}}_0^{-1} \boldsymbol{\mathcal{R}}_0$ and $\boldsymbol{\mathcal{B}}_{li} = {\boldsymbol{\mathcal{R}}_l^i}^T \boldsymbol{\mathcal{A}}_{li}^{-1} \boldsymbol{\mathcal{R}}_l^i$, we define a tensor preserving space-time multilevel multiplicative Schwarz preconditioner

$$\boldsymbol{\mathcal{B}}_{MS} = \left(\boldsymbol{\mathcal{I}} - (\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{B}}_0 \boldsymbol{\mathcal{A}}) \prod_{l=1}^{L} (\boldsymbol{\mathcal{I}} - \sum_{i=1}^{N_l} \boldsymbol{\mathcal{B}}_{li} \boldsymbol{\mathcal{A}}) \right) \boldsymbol{\mathcal{A}}^{-1}, \tag{37}$$

where

$$\boldsymbol{\mathcal{A}}_0 = \boldsymbol{A} \otimes \boldsymbol{M}_0 + \tau \boldsymbol{B} \otimes \boldsymbol{K}_0 \quad \text{and} \quad \boldsymbol{\mathcal{A}}_{li} = \boldsymbol{A} \otimes \boldsymbol{M}_{li} + \tau \boldsymbol{B} \otimes \boldsymbol{K}_{li} \tag{38}$$

are the restriction of $\boldsymbol{\mathcal{A}}$ to the coarse space $\boldsymbol{V}_0$ and the subspaces $\boldsymbol{V}_l^i$ $(i \geq 1)$, respectively. Here, $\boldsymbol{M}_0 = \boldsymbol{R}_0 \boldsymbol{M} {\boldsymbol{R}_0}^T$, $\boldsymbol{K}_0 = \boldsymbol{R}_0 \boldsymbol{K} {\boldsymbol{R}_0}^T$, $\boldsymbol{M}_{li} = \boldsymbol{R}_l^i \boldsymbol{M} {\boldsymbol{R}_l^i}^T$ and $\boldsymbol{K}_{li} = \boldsymbol{R}_l^i \boldsymbol{K} {\boldsymbol{R}_l^i}^T$

In practical applications, $\boldsymbol{\mathcal{A}}_0^{-1}$ and $\boldsymbol{\mathcal{A}}_{li}^{-1}$ are often approximated to reduce the computational cost. Below, we present several modified Schwarz preconditioners by changing the dense matrix $\boldsymbol{A}$.

- Simplified Schwarz preconditioner. By removing the part $\boldsymbol{A} \otimes \boldsymbol{M}_{li}$ of $\boldsymbol{\mathcal{A}}_{li}$ in (38), we have

$$\boldsymbol{\mathcal{B}}_{MS}^{(1)} = \left(\boldsymbol{\mathcal{I}} - (\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{B}}_0 \boldsymbol{\mathcal{A}}) \prod_{l=1}^{L} (\boldsymbol{\mathcal{I}} - \sum_{i=1}^{N_l} \boldsymbol{\mathcal{B}}_{li}^{(1)} \boldsymbol{\mathcal{A}}) \right) \boldsymbol{\mathcal{A}}^{-1}, \tag{39}$$

  where $\boldsymbol{\mathcal{B}}_{li}^{(1)} = \boldsymbol{\mathcal{R}}_l^{i\,T} \boldsymbol{\mathcal{A}}_{li}^{(1)\,-1} \boldsymbol{\mathcal{R}}_l^i$ and $\boldsymbol{\mathcal{A}}_{li}^{(1)} = \tau \boldsymbol{B} \otimes \boldsymbol{K}_{li}$.
- Diagonal Schwarz preconditioner. By replacing $\boldsymbol{A}$ by its diagonal part $\boldsymbol{A_D}$ in (38), we have

$$\boldsymbol{\mathcal{B}}_{MS}^{(2)} = \left(\boldsymbol{\mathcal{I}} - (\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{B}}_0 \boldsymbol{\mathcal{A}}) \prod_{l=1}^{L} (\boldsymbol{\mathcal{I}} - \sum_{i=1}^{N_l} \boldsymbol{\mathcal{B}}_{li}^{(2)} \boldsymbol{\mathcal{A}}) \right) \boldsymbol{\mathcal{A}}^{-1}, \tag{40}$$

  where $\boldsymbol{\mathcal{B}}_{li}^{(2)} = \boldsymbol{\mathcal{R}}_l^{i\,T} \boldsymbol{\mathcal{A}}_{li}^{(2)\,-1} \boldsymbol{\mathcal{R}}_l^i$ and $\boldsymbol{\mathcal{A}}_{li}^{(2)} = \boldsymbol{A_D} \otimes \boldsymbol{M}_{li} + \tau \boldsymbol{B} \otimes \boldsymbol{K}_{li}$.
- Lower-triangular Schwarz preconditioner. By replacing $\boldsymbol{A}$ by its lower triangular, including the diagonal, part $\boldsymbol{A_L}$ in (38), we have

$$\boldsymbol{\mathcal{B}}_{MS}^{(3)} = \left(\boldsymbol{\mathcal{I}} - (\boldsymbol{\mathcal{I}} - \boldsymbol{\mathcal{B}}_0 \boldsymbol{\mathcal{A}}) \prod_{l=1}^{L} (\boldsymbol{\mathcal{I}} - \sum_{i=1}^{N_l} \boldsymbol{\mathcal{B}}_{li}^{(3)} \boldsymbol{\mathcal{A}}) \right) \boldsymbol{\mathcal{A}}^{-1}, \tag{41}$$

  where $\boldsymbol{\mathcal{B}}_{li}^{(3)} = \boldsymbol{\mathcal{R}}_l^{i\,T} \boldsymbol{\mathcal{A}}_{li}^{(3)\,-1} \boldsymbol{\mathcal{R}}_l^i$ and $\boldsymbol{\mathcal{A}}_{li}^{(3)} = \boldsymbol{A_L} \otimes \boldsymbol{M}_{li} + \tau \boldsymbol{B} \otimes \boldsymbol{K}_{li}$.

4 Numerical experiments

In this section, we focus on the proposed Schwarz preconditioners for the L-stable BIM described in Sect. 2 by looking at the condition number and the eigenvalue distribution. To understand the parallel performance of (37) for solving linear parabolic equations we present the number of GMRES iterations and the compute time when time step size is reduced and the number of processors is increased. Finally, we provide some numerical results for a nonlinear problem.

Example 1 We consider an advection–diffusion equation

$$\begin{cases} u_t - \Delta u + u_x + u_y = f, & \text{in } \Omega \times (0, 1], \\ u(x, y, t) = 0, & \text{on } \partial\Omega \times (0, 1], \\ u(x, y, 0) = \sin(\pi x) \sin(\pi y) e^{xy}, & \text{in } \Omega, \end{cases}$$

where $\Omega = [0, 1] \times [0, 1]$, f is chosen such that the exact solution is $u(x, y, t) = \sin(\pi x) \sin(\pi y) \cos(pt) e^{xy}$.

Note that the large parameter p leads to high oscillations in time and generates a certain degree of complexity. For simplicity, the notations "δ", "N_p", "NIT", "IT" and "Time(s)" denote the overlapping size, the number of subdomains, the number of nonlinear iterations, the number of GMRES iterations and the compute time (seconds), respectively.

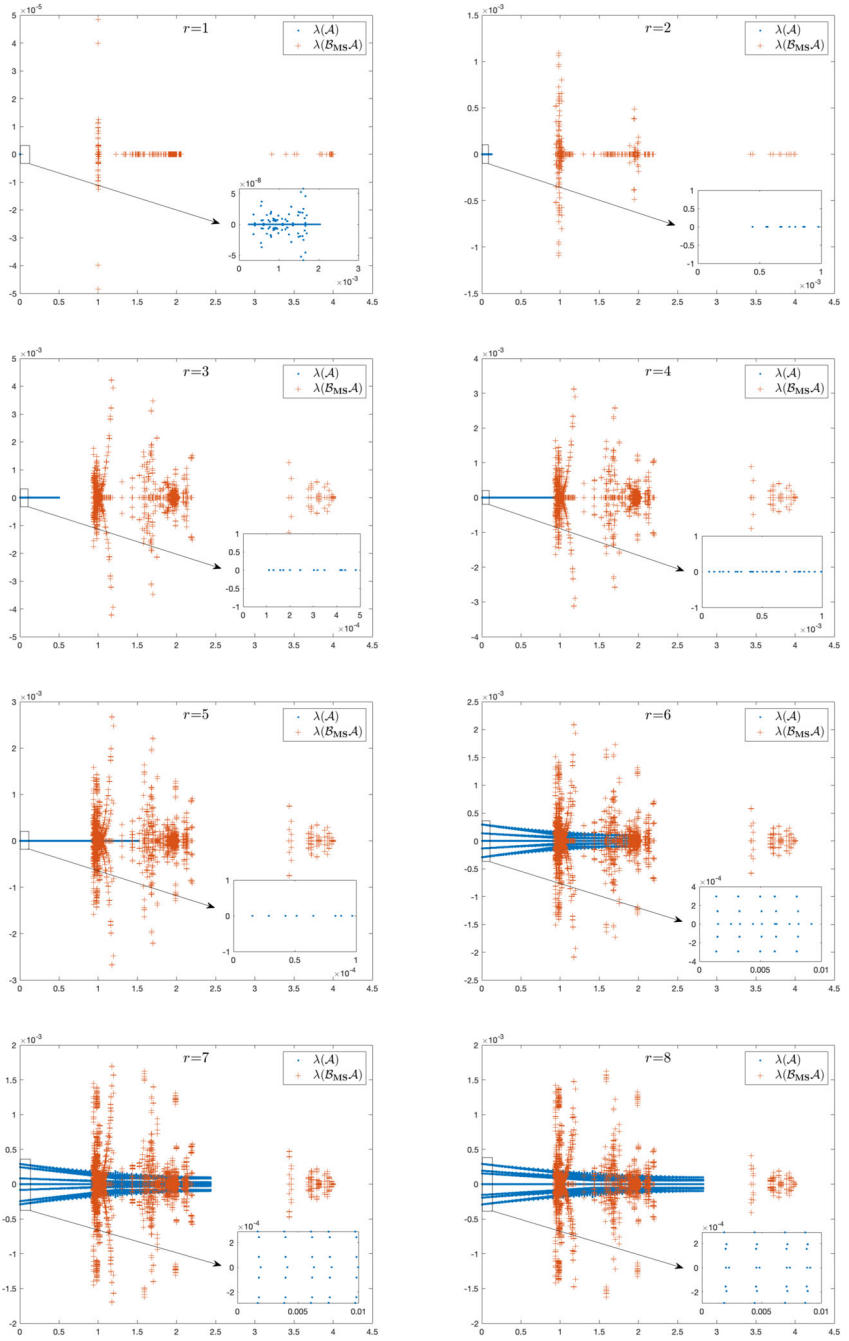

Fig. 2 All eigenvalues of the matrices $\mathcal{A}$ (blue) and $\mathcal{B}_{MS}\mathcal{A}$ (red) for Backward Euler ($r = 1$) and Scheme 1 with various block size r, zoom-in is for the eigenvalues of $\mathcal{A}$ (near origin)

Table 2 The condition numbers of the matrices with different two-level preconditioners for Backward Euler ($r = 1$) and Scheme 1

r	$\kappa(\mathcal{A})$	$\kappa(\mathcal{B}_{MS}^{(1)}\mathcal{A})$	$\kappa(\mathcal{B}_{MS}^{(2)}\mathcal{A})$	$\kappa(\mathcal{B}_{MS}^{(3)}\mathcal{A})$	$\kappa(\mathcal{B}_{MS}\mathcal{A})$
1	9.45	4.8	4.8	4.8	4.8
2	407.6	33.8	36.0	34.2	34.2
3	5968.9	120.0	120.2	119.4	119.2
4	17134.8	132.8	133.2	132.0	131.9
5	99637.5	148.4	148.6	147.7	147.5
6	2438.2	195.4	194.9	194.4	193.7
7	4019.2	321.9	320.9	319.4	319.4
8	6077.2	487.9	486.1	485.4	486.2

Here, $h_1 = 1/16$, $h_2 = 1/64$, $\tau = h^{2/r}$, $\delta = 2$ and $N_p = 16$

4.1 The condition number and the eigenvalue distribution

In the experiments, we test the two-level preconditioners (37), (39), (40) and (41) for Backward Euler ($r = 1$) as well as Scheme 1. The coarse mesh size and the fine mesh size are set as $H = 1/16$ and $h = 1/64$. We use the piecewise linear continuous finite element in space, and the time step size τ is chosen as $\tau = h^{2/r}$ so that the spatial and temporal error terms are comparable. The overlapping size and the number of subdomains are set as $\delta = 2$ and $N_p = 16$. All the preconditioners are constructed exactly; i.e., the inverse of the submatrix and coarse matrix are computed exactly.

Table 2 provides insights into the condition number in the 2-norm for the unpreconditioned matrix $\mathcal{A}$ and the matrix $\mathcal{A}$ preconditioned by $\mathcal{B}_{MS}^{(1)}$, $\mathcal{B}_{MS}^{(2)}$, $\mathcal{B}_{MS}^{(3)}$ and $\mathcal{B}_{MS}$, respectively. It shows that all the preconditioners reduce the condition number greatly, and the preconditioner $\mathcal{B}_{MS}$ performs slightly better than the others. It is interesting that the condition number of $\mathcal{A}$ grows firstly with r from 2 to 5, after a drop from $r = 5$ to $r = 6$, it then increases again with r ranging from 6 to 8. The reason is that, by construction, $\boldsymbol{A}$ is positive definite for $2 \le r \le 5$ and positive stable for $6 \le r \le 8$. On the other hand, the condition number of the preconditioned matrices always increase with r.

In practical applications, the linear system is often solved by the Schwarz preconditioners accelerated by GMRES [25, 29], so the eigenvalue distribution of the preconditioned matrix is an important indicator for the performance of GMRES. In Fig. 2, the eigenvalues of the matrices $\mathcal{A}$ and $\mathcal{B}_{MS}\mathcal{A}$ are plotted in the complex plane. We observe that some eigenvalues of $\mathcal{A}$ are close to the origin and the eigenvalues of $\mathcal{B}_{MS}\mathcal{A}$ are notably distant from the origin.

4.2 Numerical studies of the convergence rate

In the experiments, we solve the linear systems using a restarted GMRES(30) preconditioned by (37). The algorithms are implemented on top of PETSc[2]. The domain Ω on each level is divided into N_p subdomains and each subdomain is assigned

Table 3 The number of GMRES iterations and compute time for solving Example 1 by using preconditioners (37) for backward Euler ($r = 1$) and Scheme 1 with various block size r

N_p		64			128			256		
r		$\delta = 0$	$\delta = 1$	$\delta = 2$	$\delta = 0$	$\delta = 1$	$\delta = 2$	$\delta = 0$	$\delta = 1$	$\delta = 2$
1	IT	2	1	1	2	1	1	2	2	1
	Time(s)	1.19	1.11	1.12	0.86	0.64	0.65	0.45	0.42	0.42
2	IT	3	2	2	3	2	2	3	3	2
	Time(s)	2.97	2.79	2.84	1.67	1.57	1.58	0.96	0.96	0.90
3	IT	3	3	3	3	3	3	3	3	3
	Time(s)	5.60	5.66	5.68	3.11	3.14	3.17	1.70	1.73	1.75
4	IT	4	4	4	4	4	4	4	4	4
	Time(s)	9.82	9.82	9.94	5.37	5.39	5.44	2.91	2.97	2.98
5	IT	6	4	4	8	4	4	8	5	4
	Time(s)	16.42	15.50	15.56	9.37	8.32	8.40	5.17	4.66	4.52
6	IT	11	8	6	13	9	7	14	10	7
	Time(s)	27.03	25.08	23.72	15.46	14.01	13.41	8.78	7.88	7.25
7	IT	15	9	7	16	10	9	18	11	9
	Time(s)	41.42	36.03	34.65	22.98	19.97	19.72	13.35	11.25	10.74
8	IT	14	10	8	17	11	11	19	11	11
	Time(s)	53.12	48.88	47.92	30.79	26.85	27.39	17.96	14.49	14.86

The other parameters are chosen as $h_1 = 1/128$, $h_2 = 1/512$, $h_3 = 1/2048$, $\tau = h_3^{2/r}/4$ and $p = 3.5\pi$

to one processor. For solving the subdomain systems, we use ILU(1) (incomplete LU with a single level of fill-in) with the exception of employing LU on the coarsest mesh. We evaluate the preconditioned GMRES algorithms for Scheme 1 with r ranging from 2 to 8. For different BIM, the stopping conditions for GMRES are set to be $rtol = \max\{10^{-r-7}, 10^{-13}\}$ (the relative convergence tolerance) since the size of the linear systems changes with r.

Firstly, we consider the cases with the spatial mesh sizes $h_1 = 1/128$, $h_2 = 1/512$, $h_3 = 1/2048$, the time step size $\tau = h_3^{2/r}/4$, and vary the number of subdomains N_p and the overlapping size δ. From Table 3, we observe that the number of GMRES iterations and compute time do not change much with N_p and δ for each case of $r = 1, 2, 3, 4$, and they decrease with the increase of δ for each case of $r = 5, 6, 7, 8$. Although the number of GMRES iterations increases slightly with N_p, the compute time reduces nearly by half when doubling the number of processors N_p. Table 3 also shows that the number of iterations increases slightly from $r = 1$ to $r = 5$, but increase drastically from $r = 5$ to $r = 6$, afterward, it increases slowly again with r goes from 6 to 8. Moreover, there is an increase in the compute time with increasing r. This implies that the linear systems from higher order methods are more difficult to solve.

Next, we present some numerical comparisons of the preconditioners (37) for Scheme 1 with various r. We choose a larger value of p and solve the problem one block at a time till the final time $T = 1.0$. Let $\|u - u_h\|$ be the error in $\boldsymbol{L}^2$ norm

Table 4 The averaged number of GMRES iterations, the total compute time and the error in L^2-norm at the final time by using preconditioner (37) for Scheme 1 with various block size r and different time step size τ

		$p = 11.5\pi$			$p = 28.2\pi$		
r	τ	IT	Time(s)	$\|u - u_h\|$	IT	Time(s)	$\|u - u_h\|$
2	1/128	2.25	13.60	3.76e−3	2.50	14.43	3.64e−2
	1/256	2.20	27.01	9.35e−4	2.48	28.11	8.59e−3
	1/512	2.18	53.32	2.34e−4	2.55	56.31	2.11e−3
	1/1024	2.05	102.57	6.01e−5	2.42	108.23	5.28e−4
	1/2048	1.78	190.89	1.41e−5	2.04	196.92	1.27e−4
	1/4096	1.60	359.96	6.92e−6	1.84	371.78	3.18e−5
3	1/64	3.13	10.50	2.16e−3	3.00	10.22	2.46e−3
	1/128	2.95	18.58	6.03e−4	3.09	19.07	2.20e−2
	1/256	2.94	35.79	7.84e−5	3.05	36.43	1.25e−3
	1/512	2.96	69.81	1.05e−5	3.06	71.36	2.02e−4
	1/1024	2.92	137.65	1.17e−6	3.03	140.33	1.98e−5
	1/2048	2.71	261.76	5.41e−7	2.90	272.12	2.48e−6
4	1/32	5.25	9.35	7.81e−4	5.00	9.13	3.27e−1
	1/64	4.00	13.97	3.39e−4	3.94	13.87	5.31e−2
	1/128	3.91	25.52	2.00e−5	4.03	25.93	1.27e−3
	1/256	3.89	48.41	1.36e−6	4.06	49.86	7.43e−5
	1/512	3.80	93.63	1.35e−7	4.00	96.37	4.80e−6
	1/1024	3.42	171.36	5.29e−8	3.70	181.15	3.81e−7
5	1/32	5.86	12.84	1.79e−2	6.29	13.37	3.16e−1
	1/64	5.23	19.47	1.04e−4	4.92	18.75	4.50e−3
	1/128	4.04	29.99	9.47e−6	4.04	30.00	1.23e−3
	1/256	3.60	52.49	4.07e−7	4.00	56.19	4.31e−6
	1/512	3.40	95.94	6.21e−8	3.79	104.09	5.27e−7
	1/1024	3.18	182.27	5.25e−8	3.41	195.79	6.43e−8
6	1/32	20.00	37.17	1.18e−2	23.50	42.39	7.06e−1
	1/64	18.45	59.87	2.52e−4	26.73	82.89	6.11e−2
	1/128	7.86	56.21	1.80e−6	17.82	111.09	9.79e−4
	1/256	5.02	75.40	7.73e−8	7.42	100.76	1.82e−6
	1/512	4.94	143.85	5.16e−8	5.07	146.21	1.05e−7
	1/1024	4.38	258.26	5.14e−8	4.68	271.23	8.59e−8
7	1/32	29.80	57.95	3.39e−2	14.80	33.58	5.84e+0
	1/64	18.60	72.21	1.15e−4	27.30	100.72	5.50e−2
	1/128	12.21	91.29	1.12e−6	16.05	115.37	1.07e−3
	1/256	5.16	87.23	5.68e−8	10.22	150.08	5.87e−6
	1/512	4.88	160.19	5.15e−8	5.04	163.89	5.51e−8
8	1/32	29.00	58.10	1.02e−2	13.75	21.09	4.26e+0

Table 4 continued

r	τ	$p = 11.5\pi$			$p = 28.2\pi$		
		IT	Time(s)	$\|u - u_h\|$	IT	Time(s)	$\|u - u_h\|$
	1/64	18.13	71.90	3.52e−5	32.25	114.01	1.24e−1
	1/128	16.25	122.10	1.66e−7	15.43	116.96	1.07e−3
	1/256	6.25	108.08	5.44e−8	11.38	173.53	1.55e−6
	1/512	4.97	174.00	5.14e−8	5.20	181.10	8.97e−8

The other parameters are chosen as $N_p = 256$, $\delta = 1$, $h_1 = 1/128$, $h_2 = 1/512$ and $h_3 = 1/2048$

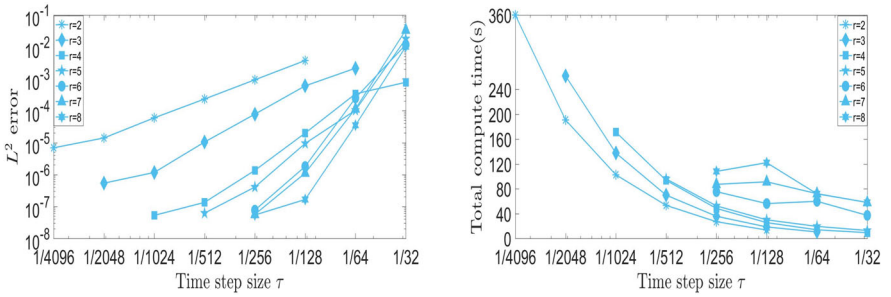

Fig. 3 The error in L^2-norm (top) and the total compute time ($p = 11.5\pi$) by using preconditioner (37) for Scheme 1

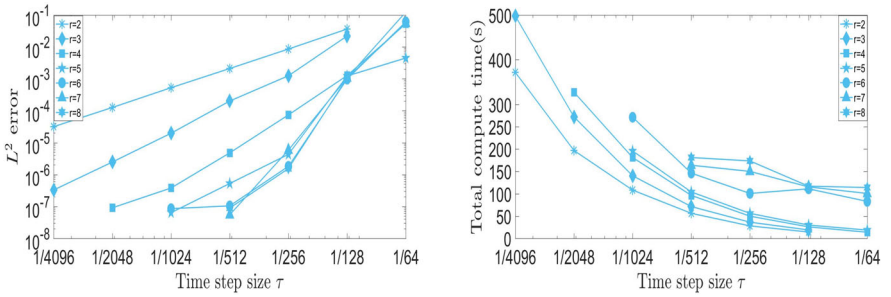

Fig. 4 The error in L^2-norm (top) and the total compute time ($p = 28.2\pi$) by using preconditioner (37) for Scheme 1

between the numerical solution and the exact solution at the final time. For each r, Figs. 3 and 4 show that smaller error can be obtained when we reduce the time step size, but more compute time is required. From Table 4, we observe that the averaged number of GMRES iterations and the total compute time increase when we increase p from 11.5π to 28.2π. To achieve the same accuracy 10^{-6}, the preconditioners (37) for Scheme 1 with $r = 4, 5, 6$ require less compute time than the other cases. And the case $r = 7$ performs much better than the other cases when the desired accuracy is 10^{-8}. Although fewer averaged number GMRES iterations is required for smaller r, it takes more compute time to achieve the same error at the final time since smaller time step size and larger number of block are used. In short, the preconditioners for

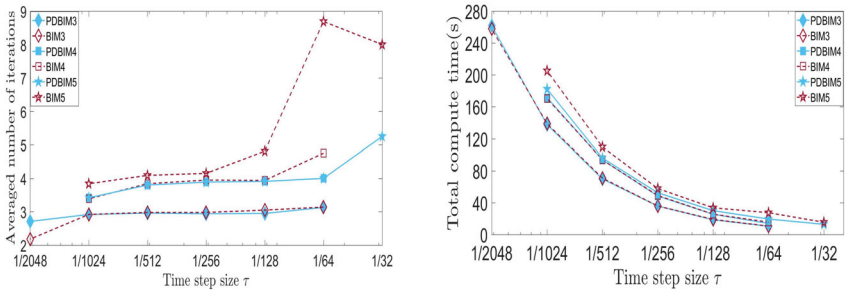

Fig. 5 The averaged number of GMRES iterations (top) and the total compute time ($p = 11.5\pi$) by using preconditioner (37) for BIM with $\boldsymbol{B} = \boldsymbol{I}$, $\boldsymbol{A} = \widetilde{\boldsymbol{N}}_r$ (denoted by BIM3, BIM4 and BIM5) and preconditioner (37) for Scheme 1 (denoted by PDBIM3, PDBIM4 and PDBIM5)

higher-order BIM show better performance compared to those for lower-order BIM in this particular test problem.

It is worth noting that the averaged GMRES iterations and the compute time grow drastically from $r = 5$ to $r = 8$ in the cases $\tau = 1/32$ and $\tau = 1/64$, then they increase a little from $r = 6$ to $r = 8$. We observe that the BIM matrix $\boldsymbol{A}$ is positive definite for $r = 2, 3, 4, 5$ but is only positive stable for $r = 6, 7, 8$. Therefore, we also test the preconditioners (37) for BIM with $\boldsymbol{B} = \boldsymbol{I}_r$ and $\boldsymbol{A} = \widetilde{\boldsymbol{N}}_r$ ($r = 3, 4, 5$), which are denoted by "BIM3", "BIM4"and "BIM5", respectively, and the preconditioner (37) for Scheme 1 with $r = 3, 4, 5$ are denoted by "PDBIM3", "PDBIM4"and "PDBIM5", respectively. From Fig. 5, we see that BIM3 performs as well as PDBIM3 in term of the compute time, but PDBIM5 requires fewer GMRES iterations and less compute time than BIM5. Unfortunately, we have not found any BIM with positive definite $\boldsymbol{A}$ and positive $\boldsymbol{B}$ when $r > 5$.

4.3 Numerical results for a nonlinear problem

In the previous sections we studied the proposed method for linear parabolic equations, and in this section we show that the method can be straightforwardly extended to nonlinear problems by linearizing the nonlinear term via a fixed-point iteration.

Example 2 We consider a nonlinear convection-diffusion–reaction equation

$$\begin{cases} u_t - \Delta u + uu_x + uu_y + u^3 = f, & \text{in } \Omega \times (0, 1], \\ u(x, y, t) = 0, & \text{on } \partial\Omega \times (0, 1], \\ u(x, y, 0) = \sin(\pi x)\sin(\pi y)e^{xy}, & \text{in } \Omega, \end{cases}$$

where $\Omega = [0, 1] \times [0, 1]$, f is chosen such that the exact solution is $u(x, y, t) = \sin(\pi x)\sin(\pi y)\cos(pt)e^{xy}$, where p is parameter to be chosen in the numerical experiment.

We discretize the problem using a rth order BIM in time and finite element in space on a 1024×1024 mesh, and solve the linear systems in each fixed-point iteration using a restarted GMRES(30) preconditioned by a three-level Schwarz (37) whose coarse

Table 5 The averaged number of nonlinear iterations (NIT) and GMRES iterations (IT), the total compute time and the error in L^2-norm at the final time ($T = 1.0$) by using preconditioner (37) for Scheme 1 with various block size r and different time step size τ

		$p = 11.5\pi$				$p = 28.2\pi$			
r	τ	NIT	IT	Time(s)	$\|u - u_h\|$	NIT	IT	Time(s)	$\|u - u_h\|$
2	1/64	7.91	4.00	98.56	1.57e−2	7.38	4.00	97.22	1.78e−1
	1/128	6.84	4.00	168.49	3.94e−3	6.67	4.00	167.49	3.52e−2
	1/256	5.87	4.00	284.15	9.79e−4	5.83	4.00	280.32	8.24e−3
	1/512	5.02	4.00	489.70	2.43e−4	5.09	4.00	471.66	2.02e−3
	1/1024	4.38	4.00	782.37	6.05e−5	4.40	4.00	785.68	5.01e−4
	1/2048	3.76	4.00	1321.79	1.49e−5	3.78	4.00	1350.94	1.24e−4
3	1/64	9.18	5.00	148.36	2.04e−3	8.36	5.00	131.57	6.17e−2
	1/128	7.77	5.00	238.49	5.89e−4	8.16	5.00	253.24	8.23e−3
	1/256	6.34	5.00	404.57	7.71e−5	6.66	5.00	424.32	1.23e−3
	1/512	5.40	5.00	648.78	1.05e−5	5.47	5.00	667.85	1.99e−4
	1/1024	4.63	5.00	1090.21	1.54e−6	4.68	5.00	1117.04	1.88e−5
	1/2048	4.01	5.00	1847.92	4.00e−7	4.12	5.00	1923.38	1.83e−6
4	1/32	10.62	6.00	110.62	2.57e−3	11.75	5.97	116.13	2.97e−1
	1/64	10.31	5.06	182.48	3.57e−4	9.06	5.06	163.37	5.37e−2
	1/128	8.44	5.00	300.08	2.06e−5	9.12	5.00	324.37	1.28e−3
	1/256	7.08	5.00	505.55	1.60e−6	7.56	5.00	553.70	7.21e−5
	1/512	5.99	5.00	848.83	3.30e−7	6.20	5.00	881.37	5.28e−6
	1/1024	5.26	5.00	1450.86	2.48e−7	5.37	5.00	1482.17	1.05e−6
	1/2048	4.65	5.00	2519.72	2.43e−7	4.70	5.00	2559.65	7.85e−7
5	1/32	10.86	6.99	164.95	1.67e−2	18.43	6.62	254.37	2.91e−1
	1/64	10.31	6.00	249.90	1.11e−4	9.23	6.00	221.04	4.53e−3
	1/128	9.04	5.92	423.75	9.56e−6	9.04	5.94	438.95	1.26e−3
	1/256	7.54	5.86	733.76	5.74e−7	8.02	5.86	765.03	3.57e−6
	1/512	6.50	5.86	1217.90	2.51e−7	6.63	5.86	1240.49	1.66e−7
	1/1024	5.61	5.85	2045.11	2.43e−7	5.72	5.87	2121.43	7.46e−7
	1/2048	4.86	6.00	3613.84	2.43e−7	4.87	6.00	3577.12	7.66e−7

The other parameters are chosen as $N_p = 256$, $\delta = 1$, $h_1 = 1/64$, $h_2 = 1/256$ and $h_3 = 1/1024$

meshes are obtained by coarsening the fine mesh twice with a coarsening factor of 4. In the experiment, the fixed-point and GMRES iterations are terminated with a relative tolerance of 10^{-r-6}. Notably, we use an order-dependent stopping criterion to ensure higher accuracy, preventing early stop of the iterative solver. The numerical results for Example 2 are listed in Table 5. For each r, it is clear that smaller error can be obtained when reducing the time step size, and the number of nonlinear iterations goes down with the decrease of the time step size. To achieve the same level of accuracy, a higher order method takes less compute time than a lower order method because the time step size is larger. It is interesting to see that for higher order methods, such as $r = 4$ or 5, the limitation of the spatial mesh is easily reached. For example, in the bottom few rows

corresponding to $r = 4$ and 5 in Table 5, the errors stop decreasing even if we decrease the time step. For example. for the more difficult case when $p = 28.2\pi$, it makes sense to use a small time step size $1/2048$ for $r = 4$, but not for $r = 5$. Similarly to the linear cases, the number of GMRES iterations for higher-order PDBIM is slightly higher than that for the lower-order methods due to larger condition numbers. When comparing the total compute time, higher order methods necessitate less time to achieve the same accuracy at the final time, because of the reduced number of time steps. As p increases from 11.5π to 28.2π, the averaged number of nonlinear iterations and the total compute time increase as expected, reflecting the increased difficulty of the problem. In summary, the method behaves similarly for solving nonlinear problems when the problem is linearized by the fixed-point method.

5 Conclusions

We introduced a general form of BIM with L-stability based on the order conditions of linear multistep methods. The relation between the BIM matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is obtained, and the matrix $\boldsymbol{B}^{-1}\boldsymbol{A}$ can be presented explicitly for schemes up to order 8. We constructed a type of BIM with order of accuracy ranging from 2 to 5 by choosing positive definite matrix $\boldsymbol{A}$ and positive diagonal matrix $\boldsymbol{B}$. The corresponding matrix $\boldsymbol{\mathcal{A}}$ from the BIM discretization in time and FEM in space is sparse, and it has the same positive definiteness as the matrix $\boldsymbol{K}$. Moreover, the traditional finite element theory for backward Euler and Crank–Nicolson schemes also holds for BIM. Some tensor preserving overlapping Schwarz preconditioners were also introduced for BIM applied to the parabolic equations. Numerical experiments demonstrated that the preconditioners perform well in terms of the condition number, the eigenvalue distribution, the parallel scalability, and the compute time for both linear and nonlinear problems. Generally speaking, the higher-order BIM are beneficial since larger time steps can be used to reduce the computational cost.

Appendix

See Table 6.

Table 6 Coefficients of the stability functions for BIM with $\boldsymbol{b} = \gamma \boldsymbol{b}_1$

		z^6	z^5	z^4	z^3	z^2	z	1
$r=2$	$\lvert C_1(z)\rvert$					-2γ	-1	4γ+2
	$\lvert C_2(z)\rvert$						4γ+1	4γ+2
	$\lvert C(z)\rvert$					2	$-4\gamma-3$	4γ+2
$r=3$	$\lvert C_1(z)\rvert$				6γ	$2-3\gamma$	$-12\gamma-6$	18γ+6
	$\lvert C_2(z)\rvert$					$-6\gamma-1$	6γ	18γ+6
	$\lvert C_3(z)\rvert$					9γ+2	24γ+6	18γ+6
	$\lvert C(z)\rvert$				-6	18γ+11	$-30\gamma-12$	18γ+6
$r=4$	$\lvert C_1(z)\rvert$			-12γ	$10\gamma-3$	20γ+11	$-60\gamma-18$	48γ+12
	$\lvert C_2(z)\rvert$				8γ+1	$-16\gamma-1$	$-12\gamma-6$	48γ+12
	$\lvert C_3(z)\rvert$				$-6\gamma-1$	$-4\gamma-1$	36γ+6	48γ+12
	$\lvert C_4(z)\rvert$				16γ+3	56γ+11	84γ+18	48γ+12
	$\lvert C(z)\rvert$			12	$-48\gamma-25$	104γ+35	$-108\gamma-30$	48γ+12
$r=5$	$\lvert C_1(z)\rvert$		60γ	$12-65\gamma$	$-75\gamma-50$	375γ+105	$-540\gamma-120$	300γ+60
	$\lvert C_2(z)\rvert$			$-30\gamma-3$	80γ+5	$15-15\gamma$	$-240\gamma-60$	60
	$\lvert C_3(z)\rvert$			15γ+2	-5γ	$-105\gamma-15$	60γ	300γ+60
	$\lvert C_4(z)\rvert$			$-20\gamma-3$	$-30\gamma-5$	105γ+15	360γ+60	300γ+60
	$\lvert C_5(z)\rvert$			75γ+12	305γ+50	615γ+105	660γ+120	300γ+60
	$\lvert C(z)\rvert$		-60	300γ+137	-770γ-225	1065γ+255	$-840\gamma-180$	300γ+60
$r=6$	$\lvert C_1(z)\rvert$	-360γ	$462\gamma-60$	294γ+274	$-2520\gamma-675$	5040γ+1020	$-5040\gamma-900$	2160γ+360
	$\lvert C_2(z)\rvert$		144γ+12	$-456\gamma-26$	$360\gamma-45$	1080γ+300	$-2880\gamma-540$	2160γ+360
	$\lvert C_3(z)\rvert$		$-54\gamma-6$	54γ+4	360γ+45	$-720\gamma-60$	$-720\gamma-180$	2160γ+360
	$\lvert C_4(z)\rvert$		48γ+6	24γ+4	$-360\gamma-45$	$-360\gamma-60$	1440γ+180	2160γ+360
	$\lvert C_5(z)\rvert$		$-90\gamma-12$	$-186\gamma-26$	360γ+45	2160γ+300	3600γ+540	2160γ+360
	$\lvert C_6(z)\rvert$		432γ+60	1944γ+274	4680γ+675	6840γ+1020	5760γ+900	2160γ+360
	$\lvert C(z)\rvert$	360	-2160γ-882	6264γ+1624	$-10440\gamma-2205$	11160γ+2100	$-7200\gamma-1260$	2160γ+360

Declarations

Conflict of interest We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with this work.

References

1. Aceto, L., Trigiante, D.: On the *A*-stable methods in the GBDF class. Nonlinear Anal. Real World Appl. **3**, 9–23 (2002)
2. Balay, S., Abhyankar, S., Adams, M.F., Benson, S., Brown, J., Brune, P., Buschelman, K., Constantinescu, E., Dalcin, L., Dener, A., Eijkhout, V., Faibussowitsch, J., Gropp, W.D., Hapla, V., Isaac, T., Jolivet, P., Karpeev, D., Kaushik, D., Knepley, M.G., Kong, F., Kruger, S., May, D.A., McInnes, L.C., Mills, R.T., Mitchell, L., Munson, T., Roman, J.E., Rupp, K., Sanan, P., Sarich, J., Smith, B.F., Zampini, S., Zhang, H., Zhang, H., Zhang, J.: PETSc Users Manual. Argonne National Laboratory, Argonne, IL (2023)
3. Barker, G.P., Berman, A., J Plemmons, R.: Positive diagonal solutions to the Lyapunov equations. Linear Multilinear Algebra **5**, 249–256 (1978)
4. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. SIAM, Philadelphia (1994)
5. Brugnano, L., Trigiante, D.: Block implicit methods for ODEs. In: Recent Trends in Numerical Analysis, vol. 3, pp. 81–105. Nova Science Publishers, Huntington, NY (2001)
6. Cai, X.-C.: Additive Schwarz algorithms for parabolic convection-diffusion equations. Numer. Math. **60**, 41–61 (1991)
7. Cai, X.-C.: Multiplicative Schwarz methods for parabolic problems. SIAM J. Sci. Comput. **15**, 587–603 (1994)
8. Cong, C., Cai, X.-C., Gustafson, K.: Implicit space-time domain decomposition methods for stochastic parabolic partial differential equations. SIAM J. Sci. Comput. **36**, C1–C24 (2014)
9. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations. II: Stiff and Differential-Algebraic Problems. Springer, Berlin (1996)
10. Hershkowitz, D., Schneider, H.: Scalings of vector spaces and the uniqueness of Lyapunov scaling factors. Linear Multilinear Algebra **17**, 203–226 (1985)
11. Jay, L.O.: Inexact simplified Newton iterations for implicit Runge–Kutta methods. SIAM J. Numer. Anal. **38**, 1369–1388 (2000)
12. Jay, L.O., Braconnier, T.: A parallelizable preconditioner for the iterative solution of implicit Runge–Kutta type methods. J. Comput. Appl. Math. **111**, 63–76 (1999)
13. Jiao, X., Wang, X., Chen, Q.: Optimal and low-memory near-optimal preconditioning of fully implicit Runge–Kutta schemes for parabolic PDEs. SIAM J. Sci. Comput. **43**, A3527–A3551 (2021)
14. Kennedy, C.A., Carpenter, M.H.: Diagonally Implicit Runge–Kutta Methods for Ordinary Differential Equations: A Review. Langley Research Center, NASA Technical Memorandum, Hampton, VA (2016)
15. Li, S., Cai, X.-C.: Convergence analysis of two-level space-time additive Schwarz method for parabolic equations. SIAM J. Numer. Anal. **53**, 2727–2751 (2015)
16. Li, S., Shao, X., Cai, X.-C.: Multilevel space-time additive Schwarz methods for parabolic equation. SIAM J. Sci. Comput. **40**, A3012–A3037 (2018)
17. Li, S., Shao, X., Chen, R.: Multilevel space-time multiplicative Schwarz preconditioner for parabolic equations. Numer. Linear Algebra Appl. **28**, e2390 (2021)
18. Li, S., Wang, J.-Y., Cai, X.-C.: A-stable high order block implicit methods for parabolic equations. SIAM J. Numer. Anal. **61**, 1858–1884 (2023)
19. Lyapunov, A.: Problème Générale de la Stabilité: du Mouvement. Communicutiuns of the Mathematical Society of Khurkow, Khurkow (1892)
20. Mardal, K.-A., Nilssen, T.K., Staff, G.A.: Order-optimal preconditioners for implicit Runge–Kutta schemes applied to parabolic PDEs. SIAM J. Sci. Comput. **29**, 361–375 (2007)
21. Milne, W.E.: Numerical Solution of Differential Equations. Wiley, New York (1953)
22. Rana, Md.M., Howle, V.E., Long, K., Meek, A., Milestone, W.: A new block preconditioner for implicit Runge–Kutta methods for parabolic PDE problems. SIAM J. Sci. Comput. **43**, S475–S495 (2021)
23. Rosser, J.B.: A Runge–Kutta for all seasons. SIAM Rev. **9**, 417–452 (1967)

24. Shampine, L.F., Watts, H.A.: Block implicit one-step methods. Math. Comput. **23**, 731–740 (1969)
25. Smith, B., Bjørstad, P., Gropp, W.: Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations. Cambridge University Press, Cambridge (1996)
26. Southworth, B.S., Krzysik, O., Pazner, W., Sterck, H.D.: Fast solution of fully implicit Runge–Kutta and discontinuous Galerkin in time for numerical PDEs, part I: the linear setting. SIAM J. Sci. Comput. **44**, A416–A443 (2022)
27. Staff, G.A., Mardal, K.-A., Nilssen, T.K.: Preconditioning of fully implicit Runge–Kutta schemes for parabolic PDEs. Model. Identif. Control **27**, 109–123 (2006)
28. Thomée, V.: Galerkin Finite Element Methods for Parabolic Problems, 2nd edn. Springer, Berlin (2006)
29. Toselli, A., Widlund, O.B.: Domain Decomposition Methods-Algorithms and Theory. Springer, Berlin (2005)
30. Van Lent, J., Vandewalle, S.: Multigrid methods for implicit Runge–Kutta and boundary value method discretizations of PDEs. SIAM J. Sci. Comput. **27**, 67–92 (2005)
31. Watts, H.A.: *A*-stable Block Implicit One-Step Methods. The University of New Mexico, Albuquerque (1971)
32. Watts, H.A., Shampine, L.F.: *A*-stable block one-step methods. BIT **12**, 252–266 (1972)