Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **9**, No. 2, pp. 501-514

DOI: 10.4208/aamm.2015.m1281
April 2017

A High-Efficient Algorithm for Parabolic Problems with Time-Dependent Coefficients

Chuanmiao Chen, Xiangqi Wang* and Hongling Hu

College of Mathematics and Computer Science, Key Laboratory of High Performance Computing and Stochastic Information Processing, Hunan Normal University, Changsha, Hunan 410081, China

Received 4 September 2015; Accepted (in revised version) 29 August 2016

Abstract. A high-efficient algorithm to solve Crank-Nicolson scheme for variable coefficient parabolic problems is studied in this paper, which consists of the Function Time-Extrapolation Algorithm (FTEA) and Matrix Time-Extrapolation Algorithm (MTEA). First, FTEA takes a linear combination of previous l level solutions ($U^{n,0}=\sum_{i=1}^{l}a_iU^{n-i}$) as good initial value of U^n (see Time-extrapolation algorithm (TEA) for linear parabolic problems, J. Comput. Math., 32(2) (2014), pp. 183–194), so that Conjugate Gradient (CG)-iteration counts decrease to $1/3\sim1/4$ of direct CG. Second, MTEA uses a linear combination of exact matrix values in level L, $L+s$, $L+2s$ to predict matrix values in the following $s-1$ levels, and the coefficients of the linear combination is deduced by the quadric interpolation formula, then fully recalculate the matrix values at time level $L+3s$, and continue like this iteratively. Therefore, the number of computing the full matrix decreases by a factor $1/s$. Last, the MTEA is analyzed in detail and the effectiveness of new method is verified by numerical experiments.

AMS subject classifications: 65M06, 65M12, 65B05, 65N22

Key words: Crank-Nicolson scheme, Time-Extrapolation, CG-iteration, variable coefficient parabolic.

1 Introduction

Consider a variable coefficient parabolic problem with initial boundary value conditions in a cylindric domain $Q=\Omega\times J$:

$$\begin{cases} u_t-\nabla(a(x,t)\nabla u)=f(x,t), & (x,t)\in Q=\Omega\times J, \quad J=(0,T), \\ u(x,t)=0, & x\in\partial\Omega, \quad t\in J=(0,T), \\ u(x,0)=\varphi(x), & x\in\Omega, \end{cases} \tag{1.1}$$

*Corresponding author.
Email: cmchen@hunnu.edu.cn (C. M. Chen), xiangqi.wang@foxmail.com (X. Q. Wang)

 ©2017 Global Science Press

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

where $a(x,t)$ depends on x and t, but not u. Finite Element Method (FEM) based on Crank-Nicolson (C-N) scheme is widely used to deal with parabolic problems.

However, to solve large scale problems, it needs a huge amount of calculation. After discretizing the equations by C-N scheme time-stepping and FEM, we have a large-scale linear equations in each time level. Therefore, there are two major challenges in numerical solution:

1. How to solve the large system of linear equations in each time level quickly;
2. Stiffness and load matrix (hereinafter referred to matrix) must be recalculated at each time level. The effective approaches to address these issues are discussed as follows.

First, in order to solve linear equations, many methods such as Successive Over Relaxation (SOR), Symmetric Successive Over-Relaxation (SSOR), Conjugate Gradient (CG)-iteration, Preconditioned CG (PCG)-iteration and so on have been proposed. The convergence speed for these methods is different. A simple and practical iterative method CG is used in this paper. Each iteration only requires the multiplication between matrix and vector. We use the compressed storage for sparse matrix, where the zero elements don't be saved and be computed. Therefore, the calculating time is linearly proportional to the number of unknowns. Suppose the initial U^0 is known, CG has good convergence

$$\|U^n-U\|_A \leq 2\rho^n \|U^0-U\|_A$$

after n-time iterations, where the convergence rate

$$\rho=\frac{\sqrt{cond(A)}-1}{\sqrt{cond(A)}+1}.$$

As the condition number of elliptic problems is $cond(A)=\mathcal{O}(h^{-2})$, and $\rho=1-\mathcal{O}(h)$, CG is almost the best in all iterative methods. There are two advantages of CG to solve parabolic problems [11]: 1). The solution U^{n-1} in the previous level is a good initial value for the current level, the error $\|U^n-U^{n-1}\|=\mathcal{O}(k)$; 2). The condition number of the matrix that needs to be inverted in each time-step when a parabolic problem is discretized by C-N scheme is not large. Therefore, CG is more effective in solving the parabolic problem than the elliptic problem. In fact, for the five-point scheme of parabolic equation, the condition number $cond(A)\approx 4r$, $r=ak/(2h^2)$, which is far less than the condition number of corresponding discrete elliptic problems where $Cond(A)=\mathcal{O}(h^{-2})$. Note that the convergence factor of CG-iteration is $\rho\approx e^{-\sqrt{1/r}}$ to solve parabolic equations. Therefore, CG-iteration has high efficiency. However, when the step-size becomes very small, e.g., $h=k\ll 10^{-3}$, the number of iterations for CG will increase greatly and overall computation would take a huge amount of time.

In this paper, the Time-Extrapolation Algorithm (TEA) [11] is used. It takes a linear combination of previous l level solutions,

$$U^{n,0}=\sum_{i=1}^{l} a_i U^{n-i}$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

as good initial value of U^n, which has high accuracy $\mathcal{O}(k^i)$, i=2,3. After applying a couple of CG iterations, we can get U^n with the desired accuracy. At present, the required CG-iteration count can be reduced to $1/3\sim1/4$.

Next, to calculate matrixes of parabolic problems with time-dependent coefficients $a(x,t)$ effectively, Matrix Time-Extrapolation Algorithm (MTEA) uses the exact matrix values in time levels L, $L+s$, $L+2s$ to construct quadratic extrapolation, and predict the matrix values at the following $s-1$ levels by extrapolating, then fully recalculate the matrix values at time level $L+3s$, and continue like this iteratively. Therefore, the number of computing the full matrix decreases by a factor $1/s$. For the linear problems, quadratic interpolation is simpler and more effective. However, for the nonlinear problems, because the coefficient depends on the solution u, only MTEA is feasible. This paper discusses the linear problem in order to analyze the accuracy of MTEA and the maximum admissible number s_m of extrapolation steps. Numerical experiments show that when $k=h=10^{-3}$, the time for matrix calculation and CG iteration is close each other. For three dimensional parabolic problems on the unit cube, it corresponds to 10^9 unknowns.

We recall that there are many classic methods to solve the large-scale problem of Partial Differential Equation (PDE). In 1977, Brandt proposed multi-grid method (MG) [1]. Later, this method was used for solving parabolic problems [9,10]. Cascadic multi-grid method (CMG) proposed by Borneman [2], Deuflhard [6] and Shaidurov [14] is a one-way multigrid method without the coarse mesh correction. Later, Shi and Xu [12], Du and Ming [8] applied this method to solve parabolic problems. In recent years, Chen and Hu etc. proposed an extrapolation cascadic multi-grid method(EXCMG) [4,5] which uses a linear combination of the solutions on previous 2 level coarse meshes to predict a better initial value of finite element solution on next-level fine mesh. This method has higher accuracy and converges more quickly. Therefore, it is more effective in solving the nonlinear problems. In this paper, however, we shall solve the large scale linear parabolic problem with time-dependent coefficients by using MTEA and CG-iteration, rather than multi-grid methods.

The rest of this paper is organized as follows: in Section 2, we give error estimates for a FEM with C-N time stepping. In Section 3, we introduce this fast algorithm in detail. In Section 4, the error estimate of MTEA is proven. In the last section, the numerical experiments are completed to support our theory.

2 Preliminaries

The weak formulation of (1.1) is

$$L(u,v)=(u_t,v)+A(u,v)=(f,v),\quad v\in H_0^1(\Omega), \tag{2.1}$$

where the bilinear form

$$A(u,v)=\int_\Omega a_{pq}(x,t)D_puD_qvdx$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

is assumed to be bounded and H_0^1-coercive, i.e., $A(v,v)\geq\gamma\|v\|_1^2$. In the cylindric domain $Q=\Omega\times J$, we introduce Sobolev space $H^m(J,H^n(\Omega))$ and norm

$$\|f\|_{L^2(\Omega)}=\Big(\int_\Omega|f|^2dx\Big)^{\frac{1}{2}},\qquad \|f\|_s=\Big(\int_\Omega\Sigma_{|\alpha|<s}|D^\alpha f|^2dx\Big)^{\frac{1}{2}},$$
$$\|f\|_{s,p}=\Big(\int_\Omega\Sigma_{|\alpha|<s}|D^\alpha f|^pdx\Big)^{\frac{1}{p}},\qquad |||f^{(m)}|||_n=\Big\{\sum_{j=0}^m\int_J\|D_t^jf\|_{n,\Omega}^2dt\Big\}^{1/2}.$$

Take the time nodes $t_j=jk$, $j=0,1,2,\cdots,N$, $T=Nk$, and the domain Ω is partitioned into quasi-uniform grids, $\Omega_h=\cup\tau$. Denote linear finite element space by

$$S^h=\{v\in C(\Omega),\ v|_\tau=\text{linear},\ \ v|_\Gamma=0\}.$$

Define the finite element solution $U^j(x)\in S^h$ satisfying C-N scheme

$$\Big(\frac{U^{j+1}-U^j}{k},v\Big)+A^{j+\frac{1}{2}}\Big(\frac{U^{j+1}+U^j}{2},v\Big)=(f^{j+1/2},v),\quad v\in S^h,\quad U^0=\varphi_h,\tag{2.2}$$

where $A^{j+\frac{1}{2}}$ denotes A with the coefficients $a_{pq}(t_{j+\frac{1}{2}},x)$, and φ_h is the elliptic projection of φ when $t=0$.

To reduce the requirement of regularity, we integrate the Eq. (2.1) directly in the time element $[t_j,t_{j+1}]$ [11], which is different from the classical analysis in [16]

$$(u^{j+1}-u^j,v)+\int_{t_j}^{t_{j+1}}A(u,v)dt=\int_{t_j}^{t_{j+1}}(f,v)dt,$$

where $u^j(t,x)=u(t_j,x)$. Then using the midpoint and trapezoidal formulas, we have

$$(u^{j+1}-u^j,v)+\frac{k}{2}A^{j+\frac{1}{2}}(u^{j+1}+u^j,v)=k(f^{j+\frac{1}{2}},v)+r_{1h}(v),\tag{2.3}$$

where the error

$$r_{1h}(v)=\mathcal{O}(k^2)\int_{t_j}^{t_{j+1}}(\|u_{tt}\|_1\|v\|_1+\|f_{tt}\|\|v\|)dt,$$

which only includes u_{tt} and f_{tt}, but not u_{ttt} directly. We shall point out that $u_{ttt}\in L^2(Q)$ is stronger requirement than $f_{tt}\in L^2(Q)$. Actually by the theory of PDE, we know that the solution u_{tt}, $D_x^4u\in L^2(Q)$, if and only if the initial value $\varphi(x)$ and f are suitably smooth, for example,

$$\varphi(x)\in H_0^1(\Omega)\cap H^3(\Omega),\quad f,f_t,D_x^2f\in L^2(Q),$$

besides, the data φ and f satisfy the first order consistence on $\Gamma_T=t=0\cap\partial\Omega$, i.e.,

$$u_t=0\quad\text{on}\ \ \Gamma_T\ \ \text{leads to}\ \ G=f-Au=0\quad\text{on}\ \ \Gamma_T.$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

To guarantee the regularity $u_{ttt}\in L^2(Q)$, the second order consistence $G_t=0$ on Γ_T is also required. Otherwise, as to be pointed out by Thomee [16], only the regularities in positive time can be obtained.

Subtracting (2.2) with (2.3), and following M. Wheeler [17], we take the elliptic projection R_hu of u as a comparison function, denote $\rho(t)=u-R_hu$, notice $A(\rho,v)=0$, $v\in S^h$. Decomposing the error of the discrete solution U into $e=u-U=\rho-\theta$, then $\theta=U-R_hu$ satisfies

$$(\theta^{j+1}-\theta^j,v)+\frac{k}{2}A^{j+1}(\theta^{j+1}+\theta^j,v)=r_h(v),$$

where $\theta^j=\theta(t_j)$, $\rho^j=\rho(t_j)$, $r_h(v)=r_{1h}(v)+r_{2h}(v)$ and

$$r_{2h}(v)=|(\rho^j-\rho^{j-1},v)|\le Ch^2\int_{t_{j-1}}^{t_j}\|u_t\|_2dt\|v\|.$$

Lemma 2.1 (see [11]). *For $\theta=U-R_hu$, the following error estimate holds:*

$$\|\theta^n\|+\Big(\int_0^{t_n}\|\theta\|_1^2dt\Big)^{\frac{1}{2}}\le Ch^2\int_0^{t_n}\|u_t\|_2dt+Ck^2\Big((\|u_{tt}\|_1^2)^{\frac{1}{2}}+\int_0^{t_n}\|f_{tt}\|dt\Big). \tag{2.4}$$

The time-difference of error $e=u-U$ has high order

$$\|e^j-e^{j-1}\|\le C(u)k(h^2+k^2). \tag{2.5}$$

(This conclusion is the basis of TEA).

Lemma 2.2. *Suppose $d\le 3$ and u is suitably smooth, we can deduce that the gradient of C-N finite element solution U is uniformly bounded in h ($k=\mathcal{O}(h)$).*

$$|\nabla U|\le|\nabla u|+|\nabla(u-R_hu)|+|\nabla\theta|\le C_1+|\nabla\theta|\le C. \tag{2.6}$$

Proof. The first term is clearly bounded. Next, we will prove that the second term is also bounded.

It is well known that finite element function v has inverse estimate $\|v\|_{1,\infty}\le Ch^{-d/2}\|v\|_1$, and interpolation function u_I has error estimates

$$\|u-u_I\|_{1,p}\le C_1,\quad 2\le p\le\infty,\quad \text{and}\quad \|u-u_I\|_{1,p}\le Ch,\quad 2\le p<\infty.$$

If the grid is uniform, for $d=2,3$, it has been proven superconvergence of gradient [3]

$$\|u_h-u_I\|_1\le Ch^2\|u\|_3,$$

then using the inverse estimate, we directly have

$$\|u-u_h\|_{1,\infty}\le C_1+\|u_h-u_I\|_{1,\infty}\le C_1+Ch^{2-d/2}\|u\|_3\le C.$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

If the grid is quasi-uniform, for $d=2$, using $\|u-u_h\|_1 \le Ch$, we still have

$$\|u-u_h\|_{1,\infty} \le C_1 + Ch^{-1}\|u_h-u_I\|_1 \le C + Ch^{-1}(\|u-u_h\|_1 + \|u-u_I\|_1) \le C.$$

For $d=3$, we have to use L^p-estimate of finite element [3]

$$\|u-u_h\|_{1,p} \le Ch\|u\|_{2,p}, \quad 2 \le p < \infty.$$

When $p \ge d$, we still have

$$\begin{aligned}\|u-u_h\|_{1,\infty} &\le C_1 + Ch^{-d/p}\|u_h-u_I\|_{1,p} \\ &\le C_1 + Ch^{-d/p}(\|u-u_h\|_{1,p} + \|u-u_I\|_{1,p}) \le C.\end{aligned}$$

Last, we discuss the third term. By (2.6) and time inverse estimate, we obtain

$$\|\theta^j\|_1 \le Ck^{-1/2}\Big(\int_{t_j}^{t_{j+1}} \|\theta\|_1^2 dt\Big)^{\frac{1}{2}} \le Ck^{-1/2}\Big(\int_0^{t_n} \|\theta\|_1^2 dt\Big)^{\frac{1}{2}} \le Ck^{-1/2}(h^2+k^2).$$

Hence, applying the inverse estimate when $d \le 3$, we obtain

$$\|\nabla\theta^j\|_{L^\infty} \le Ch^{-d/2}\|\theta^j\|_1 \le Ch^{-d/2}k^{-1/2}(h^2+k^2) \le C, \quad \text{if } k = \mathcal{O}(h).$$

The Lemma follows by these estimates. □

3 New fast algorithm

After discretizing the Eq. (1.1) by C-N scheme time-stepping and FEM, we have a large-scale linear equations in each time level t_j

$$\Big(M + \frac{k}{2}\hat{K}^j\Big)U^j = \Big(M - \frac{k}{2}\hat{K}^j\Big)U^{j-1} + kF^{j-1/2} = B^j, \quad \hat{K}^j = K^{j-1/2}, \tag{3.1}$$

where $A^j = M + \frac{k}{2}\hat{K}^j$, M and $\hat{K}^j$ are mass and stiffness matrices, and $F^{j-1/2}$ is the load matrix.

In this section we shall discuss two important algorithms in detail.

3.1 Function time extrapolation algorithm (FTEA or TEA) [11]

Assume that the solutions U^j of the previous p levels are known. We use the linear combination formula 3DpL to get initial value $U^{n,0}$ in level n $(n > p)$

$$U^{n,0} = \sum_{i=1}^{p} \omega_i U^{n-i}, \quad W(p) = [\omega_1, \omega_2, \cdots, \omega_p], \quad p = 3,4,5,6,7, \tag{3.2}$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

with the third order accuracy for smooth function $u(t)$. Obviously, by difference formula $u(t+3k)-3u(t+2k)+3u(t+k)-u(t)=k^3u'''(\xi)$, we have a predict formula $u(t+3k)=3u(t+2k)-3u(t+k)+u(t)+\mathcal{O}(k^3)$. However, numerical experiments show that this formula is not good in CG-iteration, though it has third order accuracy. We found an important fact that the CG-iteration solutions U^j at previous p levels obtained by terminate condition $Res=\|AU^{n,q}-B^n\|\le\epsilon$ ($U^{n,q}$ is the q-th iteration in CG which uses $U^{n,0}$ as initial value), have the rounding error $\pm 0.5\epsilon$. However, their linear combination (3.1) will bring about the maximum deviation $\pm 0.5\epsilon*7$. But these rounding errors are stochastic, we should select these coefficient $a_j\approx\pm 1$, such that these stochastic errors in linear combination can partially cancel each other. We have searched the best groups of coefficients

$$W(7)=[1,1,-0.95,0.9,-1,-0.9,0.95]$$

called 3D7L. Using FTEA, the number of CG iteration decreases by a factor $1/3\sim 1/4$.

3.2 Matrix time-extrapolation algorithm (MTEA)

Denote the matrix values in levels $j=L,L+s,L+2s$ by K^j, and let s be the number of extrapolation steps. Based on these three K^j, MTEA is used to get the subsequent $s-1$ matrix values K^{L+i} (i=$2s+1,2s+2,\cdots,3s-1$) by extrapolation, then calculate exact value K in level $L+3s$, and continue like this iteratively. We can take $s=5\sim 10$, which is discussed in Section 5. MTIA is to pridict matrix values K^{L+i} with $i=1,2,\cdots,s-1,s+1,\cdots,2s-1$, (MTIA is effective only for linear problems, so we mainly study MTEA in this paper).

Bellow, we will analyze the deviation of matrix caused by quadratic extrapolation and interpolation.

1. Take $a(x,t)$ at three nodes $t_j=Lk$, $(L+s)k$, $(L+2s)k$ to construct quadratic interpolation I_2a to obtain $\bar{a}^j(x,t)$ ($t=jk$) in levels $j=L+i$, $i\in(1,s-1)\cup(s+1,2s-1)\cup(2s+1,3s-1)$. The perturbation $\alpha=a(x,t)-\bar{a}(x,t)$ depends on k and s. From the expression of remainder of quadratic interpolation

$$a-I_2a=\frac{1}{3!}(t-t_1)(t-t_2)(t-t_3)D_t^3a(x,\xi),\quad t_1<\xi<t,$$

we know that

$$\alpha^j=a^j-\bar{a}^j=\frac{1}{6}i(i-s)(i-2s)k^3D_t^3a(x,\xi),\quad \xi\in(Lk,(L+3s)k),$$

where $a^j=a(t_j,x)$, $\bar{a}^j=\bar{a}(t_j,x)$. Denoting

$$C_a=\max\left|\frac{\partial^3a(x,\xi)}{6\partial t^3}\right|,\quad \alpha^j=a^j-\bar{a}^j,$$

we have

$$|\alpha^j|\le C_ai(s+i)(2s+i)k^3,\quad j=L+2s+i,\quad i=1,\cdots,s-1. \tag{3.3}$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

For quadratic extrapolation, $\max|\alpha^j| \le C_a \cdot 6(sk)^3$, $(i \in (2s+1, 3s-1))$.

For quadratic interpolation,

$$\max|\bar{\alpha}^j| \le C_a \cdot \frac{2}{3\sqrt{3}}(s'k)^3, \quad i \in (1, s-1) \cup i \in (s+1, 2s-1),$$

is only 1/16 of $\max|\alpha^j|$ of quadratic extrapolation. So a larger s can be taken to calculate. The comparison between quadratic extrapolation and interpolation will be given in Section 5.

The estimate for $a \in C^3$ is not suitable to solve the nonlinear parabolic equation, whose coefficient $a(x,t,u)$ depends on u. We need the following result.

2. If $a \in H^3(J)$, $\|a\|_{3,J} \le C_a$, by Bramble-Hilbert lemma we have the variation estimate with the weak norm

$$\int_{(L+2s)k}^{(L+3s)k} \|\alpha\|^2 dt \le C(sk)^6 \int_{Lk}^{(L+2s)k} \|\partial_t^3 a(x,t)\|^2 dt, \tag{3.4}$$

which will be used in Section 4.

4 Error analysis

We know that the small change in the coefficients of original PDE brings about the resulting perturbation of solution, see Strang-Fix [15].

Lemma 4.1 (see [15]). *For an elliptic problem*

$$-\nabla(a(x)\nabla u) + b(x)u = f(x), \quad u|_\Gamma = 0.$$

Suppose that a, b and f are changed to $\bar{a}$, $\bar{b}$ and $\bar{f}$, respectively, so that $\max(|a-\bar{a}|, |b-\bar{b}|, |f-\bar{f}|) < \varepsilon$. *Then the solution is also perturbed $\|u_h - \bar{u}_h\|_1 < C\varepsilon$. Especially, when $\varepsilon = h^2$, we have $\|u_h - \bar{u}_h\|_1 = \mathcal{O}(h^2)$ which satisfies the error estimates of bilinear finite element.*

In what follows, we will study MTEA to solve parabolic problems (2.1).

Theorem 4.1. *For variable coefficient parabolic equations* (1.1), *define the C-N finite element solution U, and extrapolation step-number s. When quadratic extrapolation on time level j is used, coefficient $a(x,t)$, $f(x,t)$ are changed to be $\bar{a}(x,t)$, $\bar{f}(x,t)$. Assume that the corresponding elliptic operator $\bar{A}$ is bounded and H_0^1-coercive, $\bar{A}(v,v) \ge \bar{\gamma}\|v\|_1^2$. Then the finite element solution $\bar{U}$ has the error estimation*

$$\|\bar{U} - U\| + |||\bar{U} - U|||_1^* \le CM(sk)^3, \quad M = |||D_t^3 a||| + |||D_t^3 f|||, \tag{4.1}$$

where

$$|||v|||_1^* = \Big(k \sum_{j=2s}^{n} \|v^j\|_1^2\Big)^{1/2}$$

is the discrete L^2-norm.

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

Proof. Let the U^j in previous $2s$ levels be calculated exactly, the perturbed solution $\bar{U}^j \in S^h$ ($j \geq 2s$) satisfies C-N scheme

$$(\bar{U}^j - \bar{U}^{j-1}, v) + k\bar{A}^{j-\frac{1}{2}}\left(\frac{\bar{U}^j + \bar{U}^{j-1}}{2}, v\right) = k(\bar{f}^{j-1/2}, v), \quad v \in H_0^1(\Omega), \quad U^0 = \varphi_h. \tag{4.2}$$

$\bar{A}$ is second-order elliptic operator with extrapolation coefficient $\bar{a}$

$$\bar{A}(u,v) = \int_\Omega \bar{a}_{pq}(x,t) D_p u D_q v dx,$$

which is bounded and H_0^1-coercive if sk suitably small. Subtracting (2.2) with (4.2), denote the errors $w^j = U^j - \bar{U}^j$, $w^{j-1} = U^{j-1} - \bar{U}^{j-1}$, and $w^j \equiv 0$ ($j \leq 2s$), which satisfy the equality

$$\begin{aligned}&(w^j - w^{j-1}, v) + \frac{k}{2}\bar{A}^{j-\frac{1}{2}}(w^j + w^{j-1}, v)\\ =&\frac{k}{2}(\bar{A}^{j-\frac{1}{2}} - A^{j-\frac{1}{2}})(U^j + U^{j-1}, v) + k(f^{j-1/2} - \bar{f}^{j-1/2}, v).\end{aligned}$$

Taking $v^j = w^j + w^{j-1}$ and using Lemma 2.2, we have

$$\begin{aligned}\|w^j\|^2 - \|w^{j-1}\|^2 + \frac{k}{2}\bar{\gamma}\|v^j\|_1^2 &\leq \frac{k}{2}\int_\Omega (\alpha^j D_p(U^j + U^{j-1})) D_q v^j dx + k(\beta^j, v^j)\\ &\leq \frac{k}{2}(\|\alpha^j \nabla(U^j + U^{j-1})\| + 2\|\beta^j\|)\|v^j\|_1.\end{aligned}$$

By using Young inequality and eliminating the right term $\|v^j\|_1$, we have

$$\|w^j\|^2 - \|w^{j-1}\|^2 + \frac{k\bar{\gamma}}{4}\|v^j\|_1^2 \leq Ck(C_0^2\|\alpha^j\|^2 + \|\beta^j\|^2).$$

Then sum over $j \in (L+2s+1, L+3s-1)$ and apply (3.4). Finally sum over all time intervals till $j = n$, and note that $w^j = \alpha^j = \beta^j = 0$ ($j \leq 2s$), we get

$$\|w^n\|^2 + \frac{\bar{\gamma}}{4}|v|_1^{*2} \leq Ck\sum_{j=2s}^{n}(\|\alpha^j\|^2 + \|\beta^j\|^2) \leq C(sk)^6 M^2.$$

This completes the proof of (4.1) and theorem. □

5 Numerical examples

In the cylindric domain $Q = (0,1)^2 \times (0,T)$, we consider

$$\begin{cases} u_t - \nabla((1+x+y+e^t)\nabla u) = f(x,y,t), & (x,y) \in \Omega, \quad t \in (0,T),\\ u(x,y,t) = 0, & (x,y) \in \partial\Omega, \quad t \in (0,T),\\ u(x,y,0) = \varphi(x,y), & (x,y) \in \Omega, \end{cases} \tag{5.1}$$

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

Table 1: Time to calculate K, F and CG for different $k=h$.

$m=500$	K-time	F-time	CG-time	CG-num.	CGe-time
$k=h=0.01$	0.3653s	0.1827s	0.0022s	46.00	0.1012s
$k=h=0.001$	36.42s	18.16s	0.22s	146.7	32.27s

Table 2: Comparison of computational times for three methods when $k=h=0.001$, $m=500$.

$k=h=0.001$	K,F-num.	Ka-time	Fa-time	K+F-time	CG-num./level	CGa-time
FEM	500	5.058h	2.522h	7.581h	146.7	4.482h
MTEA+3D7L	62	0.627h	0.313h	*0.94h	35.19	*1.075h
MTIA+3D7L	22	0.223h	0.111h	0.334h	36.71	1.122h

with the exact solution

$$u(x,y,t)=\sin(\pi x)\sin(\pi y)\sqrt{1+t}.$$

Taking the time levels $m=500$, step-size $k=h=1/N$, $N=100,1000$. We discuss C-N scheme and bilinear finite element, and then compare the computational time needed to calculate the stiffness matrix (K), the load matrix (F) and the CG-iteration (when $k=h=0.001$, CG iteration is terminated by $Res=\|b-Au\|\leq 10^{-9}$).

The symbols in Table 1 stand for:

$K-time$:	the time to calculate one stiffness matrix
$F-time$:	the time to calculate one load matrix
$CG-time$:	the time to calculate one CG iteration
$CG-num.$:	the average number of CG iterations per time step
$CGe-time$:	the average time of CG iterations per time step

From Table 1, we can conclude that

1. Since compression storage and CG are used, the computational time is proportional to the number of unknowns $N=h^{-d}$ $(d=2)$. When $k=h$ is decreased to $1/10$, the calculation time of K, F and one CG-irritation are increased to $10^d=100$;
2. The number of CG iterations increases with

$$\sqrt{\frac{ak}{h^2}}=\sqrt{\frac{a}{h}},$$

see [11], and increases by a factor of $\sqrt{10}\approx 3.2$ in this experiment.

We use MTEA + 3D7L ($s=19$) and MTIA + 3D7L ($s=24$) to solve the parabolic equation, respectively, and compare the two methods with FEM.

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

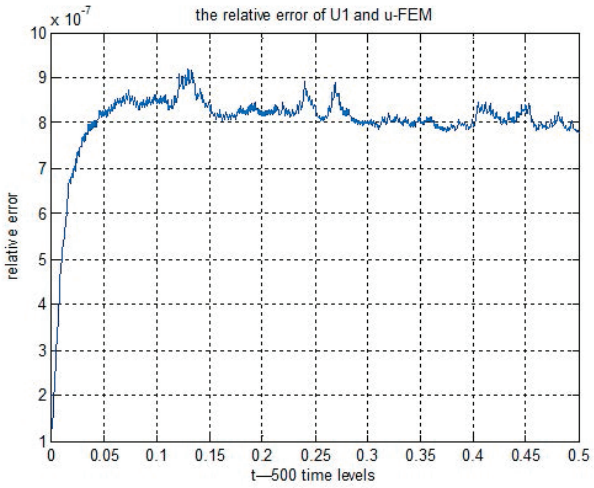


Figure 1: The relative error of FEM and exact solution.

Symbol description in Table 2:

$K,F-num.$:	the total number of stiffness or load matrixes
$Ka-time$:	the time to calculate all stiffness matrices
$Fa-time$:	the time to calculate all load matrices
$K+F-time$:	the time to calculate all stiffness and load matrices
$CG-num./level$:	the average number of CG iterations per time step
$CGa-time$:	the total time of CG iterations for all time steps

As shown in Table 2, we can conclude:

1. The efficiency of computing matrix is increased by 8 times when quadratic extrapolation is used (because META is used only after the level $2s$, the efficiency is decreased, less than 19 times). If quadratic interpolation is used, the efficiency will be improved by 23 times.
2. The extrapolation formula 3D7L can reduce the number of CG iteration to a factor $1/3\sim1/4$ of direct CG-iteration. This effect is seen clearly in the results.
3. By using MTEA+3D7L, the time for computing K and F are decreased from $7.8h$ to $T_1=0.94h$. The calculating time for CG are decreased from $4.482h$ to $T_2=1.075h$. T_2 and $T_1=\mathcal{O}(h^{-d})$ are close each other. If $k=h\ll0.0001$, then $T_2=\mathcal{O}(h^{-d}h^{-1/2})$ increases faster than $T_1=\mathcal{O}(h^{-d})$.

Next, the error curves $\|\bar{U}^n-u^n\|$ of three methods are shown in Figs. 1, 2 and 3. Here, we take $k=h=0.001$, $a(x,y,t)=1+x+y+e^t$, U is the finite element solution, u is the true solution. We see that the error in Fig. 3 is equal to that in Fig. 1, which means that

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

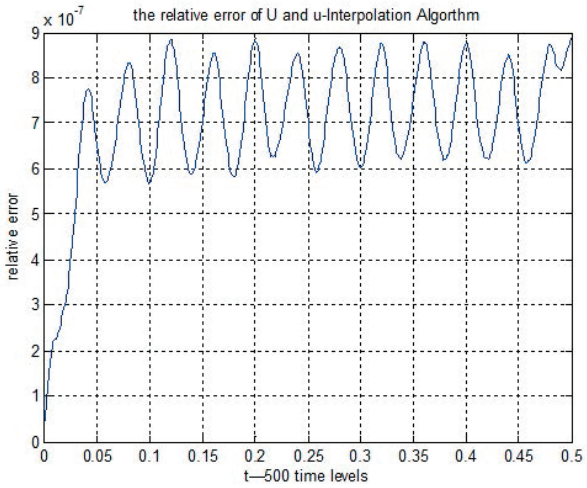


Figure 2: The relative error of quadratic interpolation and exact solution.

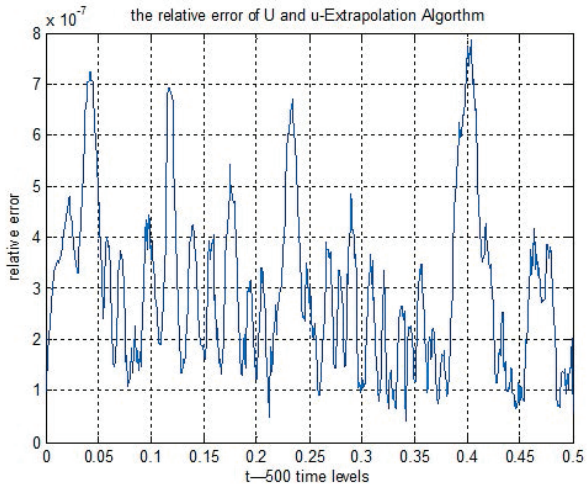


Figure 3: The relative error of quadratic extrapolation and exact solution.

MTEA+3D7L can increase the computational efficiency without affecting the calculation accuracy.

Last, we discuss the maximum admissible step-number s_m for MTEA+3D7L in $J=(0,T)$, $T=500k$. For the fixed k and h, increasing s (so the deviation $|a-\bar{a}|+|f-\bar{f}|$ increases), there is a maximum value s_m so that CG-iteration is still convergent when $s>s_m$ (even if the number of iterations increases slightly), the error $\|\bar{U}-u\|$ exceeds the tolerance δ. Table 3 shows the various values s_m corresponding to different k $(=h)$ and a, $10a$, $100a$. From (4.1), we can predict that if the deviation $(sk)^3$ equals to the optimal order

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

Table 3: s with $a(x,y,t)$, k, and h varying.

$a(x,y,t)$ \ k,h	$\frac{1}{200}$	$\frac{1}{400}$	$\frac{1}{800}$	$\frac{1}{1600}$
$1+x+y+e^t$	8	13	17	24
$10(1+x+y+e^t)$	4	8	12	16
$100(1+x+y+e^t)$	3	5	7	11

$\mathcal{O}(k^2)$, then $s=k^{-1/3}$, but actually, s_m also depends on a.

We can conclude from Table 3:

1. s_m increases with $k^{-1/2}$ roughly, and more quickly than $k^{-1/3}$;
2. When the coefficient a is replaced by $100a$, the s_m decreases to $\frac{1}{2.2}s_m$, which means that the decrease of s is insensitive to a. Even if a is very large, the value of s still can be the range of $5\sim10$ (if $k=h\leq0.001$), which gives us confidence that the method can be used to study the nonlinear problem.

6 Conclusions

If using extrapolation formula 3D7L, the number of CG iterations is about $1/3\sim1/4$ of direct CG-iteration. Our MTEA just needs to calculate one matrix per s time-levels. When the coefficient a increases to $100a$ and even $1000a$, we can still expect taking step-number $s=5\sim10$, as long as $k,h\leq0.001$. This gives us confidence to study nonlinear parabolic problems.

Acknowledgments

This work is supported by the Ph.D. Candidates' Innovative Research Project of Hunan Province of China under Grant No. CX2015B123 and National Natural Science Fund of China under Grant No. 11301176. We would also like to thank the reviewers for their valuable suggestions and comments.

References

[1] A. BRANDT, *Multi-level adaptive solutions to boundary value problems*, Math. Comput., 31 (1977), pp. 333–390.
[2] F. BORNEMANN AND P. DEUFLHARD, *The cascadic multigrid method for elliptic problems*, Numer. Math., 75(2) (1996), pp. 135–152.
[3] C. M. CHEN AND Y. Q. HUANG, High Accuracy Theorey of Finite Elements, Changsha: Hunan Science and Technique Press, 1995.
[4] C. M. CHEN, H. L. HU, Z. Q. XIE AND C. L. LI, *Analysis of extrapolation cascadic multigrid method*, Science China, Series A, 51(8) (2008), pp. 1349–1360.

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press

[5] C. M. CHEN, Z. C. SHI AND H. L. HU, *On extrapolation cascadic multigrid method,* J. Comput. Math., 29(6) (2011), pp. 684–697.
[6] P. DEUFLHARD, P. LEINEN AND H. YSERENTANT, *Concepts of an adaptive hierarchical finite element code,* IMPACT Comput. Sci. Eng., 1 (1989), pp. 3–35.
[7] J. DOUGLAS JR. AND T. DOUPONT, *Galerkin methods for parabolic equations,* SIAM J. Numer. Anal., 7 (1970), pp. 575–626.
[8] Q. DU AND P. B. MING, *Cascadic multigrid methods for parabolic problems,* Science in China Series A, 51(8) (2008), pp. 1415–1439.
[9] W. HACKBUSH, *Fast numerical solution of time-periodic parabolic problem by a multigrid method,* SIAM J. Sci. Comput., 2 (1981), pp. 198–206.
[10] W. HACKBUSCH, *Parabolic multigrid methods,* in Computing Methods in Applied Sciences and Engineering, VI, R. Glowinski and J. L. Lions, eds., North C Holland, Amsterdam: 1984, pp. 189–197.
[11] H. L. HU, C. M. CHEN AND K. J. PAN, *Time-extrapolation algorithm (TEA) for linear parabolic problems,* J. Comput. Math., 32(2) (2014), pp. 183–194.
[12] Z. C. SHI AND X. J. XU, *Cascadic multigrid for parabolic problems,* J. Comput. Math., 18(5) (2000), pp. 551–560.
[13] Y. SAAD, Iterative Method for Sparse Linear Systems, Boston: PWS Publishing Company, 1996.
[14] V. SHAIDUROV, *Some estimates of the rate of convergence for the cascadic conjugate gradient method,* Comput. Math. Appl., 31(4-5) (1996), pp. 161–171.
[15] G. STRANG AND G. FIX, An Analysis of the Finite Element Method, Wellesley-Cambridge Press, 1973.
[16] V. THOMEE, Galerkin Finite Element Methods for Parabolic Problems, Second edition, New York: Springer-Verlag, 2006.
[17] M. WHEELER, *A priori L_2 error estimates for Galerkin approximations to parabolic partial differential equations,* SIAM J. Numer. Anal., 10(4) (1973), pp. 723–759.

https://doi.org/10.4208/aamm.2015.m1281 Published online by Cambridge University Press