Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **9**, No. 1, pp. 157-172

DOI: 10.4208/aamm.2015.m1045
February 2017

Efficient and Stable Exponential Runge-Kutta Methods for Parabolic Equations

Liyong Zhu*

School of Mathematics and Systems Science, Beihang University, Xueyuan Road No. 37, Haidian District, Beijing 100191, China

Received 7 April 2015; Accepted (in revised version) 31 December 2015

Abstract. In this paper we develop explicit fast exponential Runge-Kutta methods for the numerical solutions of a class of parabolic equations. By incorporating the linear splitting technique into the explicit exponential Runge-Kutta schemes, we are able to greatly improve the numerical stability. The proposed numerical methods could be fast implemented through use of decompositions of compact spatial difference operators on a regular mesh together with discrete fast Fourier transform techniques. The exponential Runge-Kutta schemes are easy to be adopted in adaptive temporal approximations with variable time step sizes, as well as applied to stiff nonlinearity and boundary conditions of different types. Linear stabilities of the proposed schemes and their comparison with other schemes are presented. We also numerically demonstrate accuracy, stability and robustness of the proposed method through some typical model problems.

AMS subject classifications: 65M06, 65M22, 65Y20

Key words: Exponential Runge-Kutta method, explicit scheme, linear splitting, discrete fast Fourier transforms, Allen-Cahn equation.

1 Introduction

The exponential integrator method is one of the many methods for solving stiff differential equations, see [7] for a recent review. Due to their stable and high-order accuracy for time integration, the exponential integrator based schemes have attracted many researcher's interests [2–5, 9, 11, 13–15]. Especially, along with the development of fast and stable methods for computing or approximating the product of a matrix exponential

*Corresponding author.
Email: liyongzhu@buaa.edu.cn (L. Y. Zhu)

 ©2017 Global Science Press

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

function with a vector (see the review [12]), the exponential integrator methods have become quite efficient and effective in practice. In this paper, we study exponential Runge-Kutta methods for a class of parabolic equations of the following form:

$$\frac{\partial u}{\partial t} = D\Delta u - f(t,u), \quad \mathbf{x}\in\Omega, \quad t\in[t_0,t_0+T], \tag{1.1}$$

where Ω is an open rectangular domain in $\mathbb{R}^d$, $T>0$, and D is a positive diffusion coefficient. Equations of the above form can represent mathematical models in various applications like Allen-Cahn equations [1] or Ginzburg-Landau equations in phase transition modeling. In order to solve the equations like (1.1), some explicit exponential Runge-Kutta methods have been developed and their numerical analysis were carried out [2–8, 11–13]. In [4], the authors analyzed the modifications of the expoential time differencing schemes via complex contour integrations and illustrated that the contour integration could improve the stability of the time integration. In [6], the authors proved convergence of explicit exponential Runge-Kutta methods up to order four and also constructed some new schemes that do not suffer from order reduction. In [13], the conditional stability of exponential Runge-Kutta methods are analyzed and some sufficient conditions are given. In [11], a fifth-order explicit exponential Runge-Kutta method with eight stages was proposed and its convergence was proved for semilinear parabolic problems.

More recently, in [8], by incorporating the linear splitting techniques (examined in [3, 4]) into the multistep approximation with an analytic evaluations of time exponential integrations, explicit compact exponential time differencing multistep methods are proposed for semilinear parabolic equations. These methods combine the decompositions of compact spatial difference operators on a regular mesh and fast discrete Fourier transform techniques to improve computation efficiency. By integrating seamlessly these well-studied techniques, the proposed compact exponential multistep methods can avoid solving nonlinear systems but are still stable and efficient to numerically solve semilinear problems. Because the discretization matrix is irreversible in the cases of periodic and Neumann boundary conditions, it is generally not an easy job to develop fast exponential methods, however, the compact exponential multistep methods developed in [8] can be easily applied to the problems with boundary conditions of different types. In this paper, the exponential Runge-Kutta methods are employed to approximate the temporal integrals. Compared with the multistep approximations, the exponential Runge-Kutta methods can be self-started, and more importantly, they are much easier to be implemented for adaptive time step sizes, that often can help to further improve the overall computation efficiency. In addition, the exponential Runge-Kutta methods are more stable than the corresponding multistep approaches as shown in Section 4.

The structure of this paper is as follows. In Section 2, we present a general compact exponential time integration scheme with a linear splitting parameter. Then, in Section 3, with the Runge-Kutta approximations for temporal integration, we propose fast and stable exponential Runge-Kutta methods in the compact form. In Section 4, linear stabilities

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

are discussed for the proposed schemes. In Section 5, some numerical examples, including linear and nonlinear model problems such as the Allen-Cahn equation, are given to illustrate accuracy, stability and effectiveness of our methods. Finally, some conclusions are given in Section 6.

2 A general compact exponential time integration scheme with a linear splitting parameter

In order to maintain the numerical stability in time integration while avoid solving nonlinear systems, the stabilized semi-implicit methods are developed by incorporating the linear splitting idea into the integration scheme [3, 4, 16]. By introducing a nonnegative splitting parameter κ, the original parabolic problem (1.1) can be rewritten as the following modified form

$$\frac{\partial u}{\partial t}=\mathcal{L}u-\hat{f}(t,u),\quad \mathbf{x}\in\Omega,\quad t\in[t_0,t_0+T], \tag{2.1}$$

where

$$\mathcal{L}u=D\Delta u-\kappa u,\quad \hat{f}(t,u)=f(t,u)-\kappa u.$$

When $f(t,u)$ only depends u, we take the splitting parameter

$$\kappa\geq\frac{1}{2}\max\{0,\max_u f'(u)\}, \tag{2.2}$$

which guaranttes the stability of problem [17]. In this work, we consider the numerical solution of the above modified parabolic problem with a linear splitting parameter.

2.1 Compact spatial semi-discretization schemes

We recall compact spatial semi-discretization schemes developed in [8]. Consider the problem in two dimensions with a Dirichlet boundary condition

$$u=g,\quad (x,y)\in\partial\Omega,\quad t\in[t_0,t_0+T]. \tag{2.3}$$

Here the spatial domain $\Omega=\{x_b<x<x_e,\ y_b<y<y_e\}$ is discretized by a rectangular mesh which is uniform in each direction as follows: $(x_i,y_j)=(x_b+ih_x,y_b+jh_y)$ for $0\leq i\leq N_x$ and $0\leq j\leq N_y$ with $h_x=(x_e-x_b)/N_x$ and $h_y=(y_e-y_b)/N_y$.

Denote the numerical solution $u_{i,j}=u_{i,j}(t)\approx u(t,x_i,y_j)$ for $0\leq i\leq N_x$ and $0\leq j\leq N_y$. The second order accurate central difference discretization scheme is used for the spatial derivative terms. In this case, the set of unknowns is given as

$$\mathbf{U}=(u_{i,j})_{(N_x-1)\times(N_y-1)}=\begin{pmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,N_y-1}\\ u_{2,1} & u_{2,2} & \cdots & u_{2,N_y-1}\\ \vdots & \vdots & \ddots & \vdots\\ u_{N_x-1,1} & u_{N_x-1,2} & \cdots & u_{N_x-1,N_y-1}\end{pmatrix}_{(N_x-1)\times(N_y-1)}.$$

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

Let

$$\mathbf{U}_{x1}=\frac{D}{h_x^2}\begin{pmatrix} g(t,x_0,y_1) & g(t,x_0,y_2) & \cdots & g(t,x_0,y_{N_y-1}) \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ g(t,x_{N_x},y_1) & g(t,x_{N_x},y_2) & \cdots & g(t,x_{N_x},y_{N_y-1}) \end{pmatrix}_{(N_x-1)\times(N_y-1)},$$

$$\mathbf{U}_{y1}=\frac{D}{h_y^2}\begin{pmatrix} g(t,x_1,y_0) & 0 & \cdots & 0 & g(t,x_1,y_{N_y}) \\ g(t,x_2,y_0) & 0 & \cdots & 0 & g(t,x_2,y_{N_y}) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g(t,x_{N_x-1},y_0) & 0 & \cdots & 0 & g(t,x_{N_x-1},y_{N_y}) \end{pmatrix}_{(N_x-1)\times(N_y-1)}.$$

In addition, we define

$$\mathbf{G}_N=\begin{pmatrix} -2 & 1 & 0 & 0 & \cdots & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ & & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 & -2 & 1 \\ 0 & \cdots & 0 & 0 & 1 & -2 \end{pmatrix}_{N\times N},$$

and set

$$\mathbf{A}=\frac{D}{h_x^2}\mathbf{G}_{(N_x-1)},\qquad \mathbf{B}=\frac{D}{h_y^2}\mathbf{G}_{(N_y-1)}.$$

After defining the special operators $\textcircled{x}$ and $\textcircled{y}$

$$(\mathbf{A}\textcircled{x}\mathbf{U})_{i,j}=\sum_{l=1}^{N_x-1}(\mathbf{A})_{i,l}u_{l,j},\qquad (\mathbf{B}\textcircled{y}\mathbf{U})_{i,j}=\sum_{l=1}^{N_y-1}(\mathbf{B})_{j,l}u_{i,l},\tag{2.4}$$

the spatial semi-discretization of (1.1) can be written as in the following compact form:

$$\frac{d\mathbf{U}}{dt}=\mathbf{A}\textcircled{x}\mathbf{U}+\mathbf{B}\textcircled{y}\mathbf{U}-\kappa\mathbf{U}+\mathcal{W}-\mathcal{F}(t,\mathbf{U}),\tag{2.5}$$

where $\mathcal{F}(t,\mathbf{U})=(\hat{f}(t,u_{i,j}))_{(N_x-1)\times(N_y-1)}$ and $\mathcal{W}=\mathbf{U}_{x1}+\mathbf{U}_{y1}$.

2.2 A general exponential time integration approximation

It is well known that the eigenvalues of G_{N-1} are

$$\lambda_N^k=-4\sin^2\frac{k\pi}{2N},\quad k=1,2,\cdots,N-1.$$

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

Let $\mathbf{\Lambda}_N=\text{diag}(\lambda_N^1,\lambda_N^2,\ldots,\lambda_N^{N-1})$ and $\mathbf{P}_N=(p_{i,j})_{(N-1)\times(N-1)}$ with $p_{i,j}$ defined by

$$p_{i,j}=\sin\Big(\frac{2ij\pi}{N}\Big),\quad i,j=1,2,\cdots,N-1.$$

Then we have the eigenvalue decompositions

$$\mathbf{A}=\frac{D}{h_x^2}\mathbf{P}_{N_x}\mathbf{\Lambda}_{N_x}\mathbf{P}_{N_x}^{-1},\quad \mathbf{B}=\frac{D}{h_y^2}\mathbf{P}_{N_y}\mathbf{\Lambda}_{N_y}\mathbf{P}_{N_y}^{-1}.$$

Set $\mathbf{H}=(h_{i,j})_{(N_x-1)\times(N_y-1)}$, $i=1,2,\cdots,N_x-1$, $j=1,2,\cdots,N_y-1$ with

$$h_{i,j}=\frac{D}{h_x^2}\lambda_{N_x}^i+\frac{D}{h_y^2}\lambda_{N_y}^j-\kappa. \tag{2.6}$$

Clearly we have $h_{i,j}\le-\kappa\le 0$.

Let us further discretize the time period as $t_n=t_0+n\Delta t$, $n=1,\cdots,N_t$ with $\Delta t=T/N_t$. Then, a variation of constant formula for the ODE system (2.5) leads to the following general compact exponential time integration scheme [8],

$$\mathbf{U}_{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\Big((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U}_\mathbf{n})+\mathbf{Q}^W-\mathbf{Q}^F\Big), \tag{2.7}$$

where

$$\begin{aligned}&\mathbf{Q}^W=(q_{i,j}^W)_{(N_x-1)\times(N_y-1)}=\int_0^{\Delta t}\mathbf{W}(t_n+\tau)\odot(e^*)^{\mathbf{H}(\Delta t-\tau)}d\tau,\\&\mathbf{W}(t_n+\tau)=(w_{i,j}(t_n+\tau))_{(N_x-1)\times(N_y-1)}=\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathcal{W}(t_n+\tau),\end{aligned}$$

and

$$\begin{aligned}\mathbf{Q}^F&=(q_{i,j}^F)_{(N_x-1)\times(N_y-1)}=\int_0^{\Delta t}\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))\odot(e^*)^{\mathbf{H}(\Delta t-\tau)}d\tau,\\\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))&=\big(f_{i,j}(t_n+\tau,u_{i,j}(t_n+\tau))\big)_{(N_x-1)\times(N_y-1)}\\&=\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathcal{F}(t_n+\tau,\mathbf{U}(t_n+\tau)).\end{aligned}$$

And the operations "(e^*)" and "$\odot$" are defined by

$$(e^*)^{\mathbf{H}}=(e^{h_{i,j}})_{(N_x-1)\times(N_y-1)},\quad(\mathbf{M}\odot\mathbf{L})_{i,j}=(\mathbf{L}\odot\mathbf{M})_{i,j}=(m_{i,j}l_{i,j}).$$

The detailed derivation of (2.7) may be found in [8]. Here, we omit the details. Then exponential integration methods can be developed based on (2.7). Here are some remarks:

Remark 2.1. In [8], the multistep approximation with an analytic evaluation of time exponential integrals is employed to develop the fast explicit integration factor methods for semilinear parabolic equations.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

Remark 2.2. Here, we will use the Runge-Kutta methods to approximate the temporal exponential integrals. Compared with the exponential multistep approximation, the exponential Runge-Kutta method can be self-started and is easily implemented. And it is also much easy for the exponential Runge-Kutta method to adopt the adaptive temporal approximations with variable time step sizes that may help to further improve the overall computation efficiency. Additionally, the exponential Runge-Kutta method is more stable than the exponential multistep method as shown in Section 4.

3 Efficient and stable exponential Runge-Kutta methods

3.1 Approximations of the boundary integral

For the evaluation of the integral $\mathbf{Q}^W$ resulted from the boundary conditions, as in [8], we take the uniform interpolation points $t_n+\frac{i}{r}\Delta t$, $(0\leq i\leq r$ for $r>0)$ and use the Lagrange interpolation polynomial of degree r to approximate $\mathbf{W}(t_n+\tau)$:

$$P_r^W(\tau)=\sum_{s=0}^{r}\omega_{r,s}(\tau)\mathbf{W}\left(t_n+\frac{s}{r}\Delta t\right) \tag{3.1}$$

with the standard Lagrange basis function

$$\omega_{r,s}(\tau)=\prod_{\substack{l=0\\ l\neq s}}^{r}\frac{r\tau/\Delta t-l}{(s-l)}.$$

Then we have

$$\mathbf{W}(t_n+\tau)\approx P_r^W(\tau)+\mathcal{O}(\Delta t^{r+1}). \tag{3.2}$$

Define

$$\mathbf{S}_{r,s}=(\alpha_{i,j}^{(r,s)})_{(N_x-1)\times(N_y-1)}, \tag{3.3}$$

where

$$\alpha_{i,j}^{(r,s)}=\int_0^{\Delta t}e^{h_{i,j}(\Delta t-\tau)}\omega_{r,s}(\tau)d\tau.$$

We then approximate $\mathbf{Q}^W$ as

$$\mathbf{Q}^W\approx\mathbf{Q}_r^W\triangleq\sum_{s=0}^{r}\mathbf{W}\left(t_n+\frac{s}{r}\Delta t\right)\odot\mathbf{S}_{r,s}, \tag{3.4}$$

which is $(r+1)$-th order accurate.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

3.2 Efficient and stable exponential Runge-Kutta methods

Combining the compact spatial semi-discretization schemes with the exponential Runge-Kutta schemes for time integration as proposed in [2], we can derive the efficient and stable exponential Runge-Kutta methods. The first order efficient and stable exponential Runge-Kutta scheme is as same as the first order exponential multistep scheme presented in [8]. Here the value of $\mathbf{F}(t_n,\mathbf{U}_n)$ is employed to approximate the nonlinear term $\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))$. That is

$$\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))\approx\mathbf{F}(t_n,\mathbf{U}_n)+\mathcal{O}(\Delta t).$$

Define

$$\begin{cases}\phi_0=\displaystyle\int_0^{\Delta t}e^{h_{i,j}(\Delta t-\tau)}d\tau=-\frac{1}{h_{i,j}}(1-e^{h_{i,j}\Delta t}), & h_{i,j}\neq 0,\\ \phi_s=\displaystyle\int_0^{\Delta t}\left(\frac{\tau}{\Delta t}\right)^s e^{h_{i,j}(\Delta t-\tau)}d\tau=-\frac{1}{h_{ij}}\left(1-\frac{s\phi_{s-1}}{\Delta t}\right), \quad s=1,2,3, & h_{i,j}\neq 0,\\ \phi_0=\Delta t,\quad \phi_1=\dfrac{\Delta t}{2},\quad \phi_2=\dfrac{\Delta t}{3},\quad \phi_3=\dfrac{\Delta t}{4}, & h_{i,j}=0.\end{cases} \tag{3.5}$$

Denote $\mathbf{W}_{n+l}=\mathbf{W}(t_n+l\Delta t)$. Then, with (2.7) and (3.4), the first order efficient and stable exponential scheme can be written as

$$\mathbf{U}_{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+\mathbf{W}_n\odot\phi_0-\mathbf{F}_n\odot\phi_0\right), \tag{3.6}$$

where $\mathbf{F}_n=\mathbf{F}(t_n,\mathbf{U}_n)$.

3.2.1 The second order efficient and stable exponential Runge-Kutta scheme

Similar to the "improved Euler" scheme, the second order exponential Runge-Kutta method is a 2-stage method. We first compute the approximation of $\mathbf{U}(t_n+\Delta t)$ by the first order scheme

$$\tilde{\mathbf{U}}_{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+\mathbf{W}_n\odot\phi_0-\mathbf{F}_n\odot\phi_0\right). \tag{3.7}$$

Define $\theta=\tau/\Delta t$. Then, the approximation

$$\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))\approx(1-\theta)\mathbf{F}(t_n,\mathbf{U}_n)+\theta\mathbf{F}(t_{n+1},\tilde{\mathbf{U}}_{n+1})+\mathcal{O}(\Delta t^2) \tag{3.8}$$

is applied on the interval $0\leq\tau\leq\Delta t$, and is substituted into the integral $\mathbf{Q}^F$ to yield the second order efficient and stable exponential Runge-Kutta scheme

$$\begin{aligned}\mathbf{U}_{n+1}=&\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\Big((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+(\mathbf{W}_n-\mathbf{F}_n)\odot\mathbf{S}_{1,0}\\ &+(\mathbf{W}_{n+1}-\mathbf{F}(t_n+\Delta t,\tilde{\mathbf{U}}_{n+1}))\odot\mathbf{S}_{1,1}\Big),\end{aligned} \tag{3.9}$$

where $\mathbf{S}_{1,0}$ and $\mathbf{S}_{1,1}$ are defined in (3.3).

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

3.2.2 The third order efficient and stable exponential Runge-Kutta scheme

The third order exponential Runge-Kutta method is a 3-stage method. We first compute the approximation of $\mathbf{U}(t_n+\Delta t/2)$ by the first order scheme

$$\tilde{\tilde{\mathbf{U}}}_{n+1/2}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t/2}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+\mathbf{W}_n\odot\hat{\phi}_0-\mathbf{F}_n\odot\hat{\phi}_0\right),\tag{3.10}$$

where

$$\hat{\phi}_0=\int_0^{\Delta t/2}e^{h_{i,j}(\Delta t/2-\tau)}d\tau=-\frac{1}{h_{i,j}}(1-e^{h_{i,j}\Delta t/2}),\quad h_{i,j}\neq 0,\quad \hat{\phi}_0=\frac{\Delta t}{2},\quad h_{i,j}=0.\tag{3.11}$$

Then, with the value of $\tilde{\tilde{\mathbf{U}}}_{n+1/2}$, we compute the approximation of $\mathbf{U}(t_n+\Delta t)$ by the first order scheme

$$\begin{aligned}\tilde{\tilde{\mathbf{U}}}_{n+1}=&\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})\right.\\&\left.+(2\mathbf{W}_{n+1/2}-\mathbf{W}_n)\odot\phi_0-(2\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2})-\mathbf{F}_n)\odot\phi_0\right),\end{aligned}\tag{3.12}$$

where ϕ_0 is defined in (3.5) and $\mathbf{F}_n=\mathbf{F}(t_n,\mathbf{U}_n)$. Then, we have the approximation

$$\begin{aligned}\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))\approx&(1-2\theta)(1-\theta)\mathbf{F}(t_n,\mathbf{U}_n)+2\theta(1-2\theta)\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2})\\&+\theta(2\theta-1)\mathbf{F}(t_{n+1},\tilde{\tilde{\mathbf{U}}}_{n+1})+\mathcal{O}(\Delta t^3).\end{aligned}\tag{3.13}$$

Substituting the above approximation into the integral $\mathbf{Q}^F$ yields to the compact third order efficient and stable exponential Runge-Kutta scheme given by

$$\begin{aligned}\mathbf{U}_{n+1}=&\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+(\mathbf{W}_n-\mathbf{F}_n)\odot\mathbf{S}_{2,0}\right.\\&\left.+(\mathbf{W}_{n+1/2}-\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2}))\odot\mathbf{S}_{2,1}+(\mathbf{W}_{n+1}-\mathbf{F}(t_{n+1},\tilde{\tilde{\mathbf{U}}}_{n+1}))\odot\mathbf{S}_{2,2}\right),\end{aligned}\tag{3.14}$$

where $\mathbf{S}_{2,0}$, $\mathbf{S}_{2,1}$ and $\mathbf{S}_{2,2}$ are defined in (3.3).

3.2.3 The fourth order efficient and stable exponential Runge-Kutta scheme

A straightforward extension of the standard fourth order Runge-Kutta method yields a scheme which is only third order. By varying the scheme and introducing further parameters, a fourth order exponential Runge-Kutta scheme is presented in [2]. Here we derive its efficient and stable form.

First we compute $\tilde{\tilde{\mathbf{U}}}(t_{n+1/2})$ by (3.10). Then, with the value of $\tilde{\tilde{\mathbf{U}}}_{n+1/2}$, we compute another approximation of $\mathbf{U}(t_n+\Delta t/2)$ by the following first order scheme

$$\begin{aligned}\tilde{\tilde{\tilde{\mathbf{U}}}}_{n+1/2}=&\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\left((e^*)^{\mathbf{H}\Delta t/2}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})\right.\\&\left.+\left(\mathbf{W}_{n+1/2}-\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2})\right)\odot\hat{\phi}_0\right).\end{aligned}\tag{3.15}$$

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

In the above two schemes, $\hat{\phi}_0$ is defined in (3.11). Next, we compute the approximation of $\mathbf{U}(t_n+\Delta t)$ by the following first-order scheme

$$\tilde{\tilde{\mathbf{U}}}_{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\Big((e^*)^{\mathbf{H}\Delta t/2}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\tilde{\tilde{\mathbf{U}}}_{n+1/2})+(2\mathbf{W}_{n+1/2}-\mathbf{W}_n)\odot\hat{\phi}_0-(2\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2}-\mathbf{F}_n))\odot\hat{\phi}_0\Big),$$

where ϕ_0 is defined in (3.5) and $\mathbf{F}_n=\mathbf{F}(t_n,\mathbf{U}_n)$.

Then, with the above approximation, we get the approximation

$$\mathbf{F}(t_n+\tau,\mathbf{U}(t_n+\tau))\approx(1-2\theta)(1-\theta)\mathbf{F}(t_n,\mathbf{U}_n)+\theta(1-2\theta)\Big((\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2})+\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2})\Big)+\theta(2\theta-1)\mathbf{F}(t_{n+1},\tilde{\tilde{\mathbf{U}}}_{n+1})+\mathcal{O}(\Delta t^3). \quad (3.16)$$

Substituting the above approximations into the integral $\mathbf{Q}^F$ yields to the efficient and stable fourth order exponential Runge-Kutta scheme given by

$$\mathbf{U}_{n+1}=\mathbf{P}_y\textcircled{y}\,\mathbf{P}_x\textcircled{x}\Big((e^*)^{\mathbf{H}\Delta t}\odot(\mathbf{P}_y^{-1}\textcircled{y}\,\mathbf{P}_x^{-1}\textcircled{x}\,\mathbf{U_n})+(\mathbf{W}_n\odot\mathbf{S}_{3,0}+\mathbf{W}_{n+1/3}\odot\mathbf{S}_{3,1}+\mathbf{W}_{n+2/3}\odot\mathbf{S}_{3,2}+\mathbf{W}_{n+1}\odot\mathbf{S}_{3,3})-\Big(\mathbf{F}_n\odot\mathbf{S}_{2,0}+\frac{1}{2}(\mathbf{F}(t_{n+1/2},\tilde{\mathbf{U}}_{n+1/2})+\mathbf{F}(t_{n+1/2},\tilde{\tilde{\mathbf{U}}}_{n+1/2}))\odot\mathbf{S}_{2,1}+\mathbf{F}(t_{n+1},\tilde{\tilde{\mathbf{U}}}_{n+1})\odot\mathbf{S}_{2,2}\Big)\Big), \quad (3.17)$$

where $\mathbf{S}_{r,s}$ are defined in (3.3).

As in [8], with the discrete fast Fourier transforms, the overall computation cost of the efficient and stable exponential Runge-Kutta schemes in two dimensions can be reduced from $\mathcal{O}(N_xN_yN)$ to $\mathcal{O}(N_xN_y\log_2 N)$ per time step where $N=\max\{N_x,N_y\}$, which is very important especially when N is quite large. The total required memory is obviously $\mathcal{O}(N^2)$. With the similar derivations in [8], the efficient and stable exponential Runge-Kutta schemes can be extended to three dimensional problems and the problems with other boundary conditions, such as periodic and Neumann boundary conditions.

4 Linear stability

In this section, we consider the linear stability of the proposed exponential Runge-Kutta method and discuss its comparison with the multistep scheme developed in [8]. We test the stability on the following linear equation

$$u_t=\mathrm{L}u+\lambda u, \quad (4.1)$$

with $\mathrm{L}u=-qu$ and a homogeneous Dirichlet boundary condition. We follow [3,4] and consider the cases where λ is complex-valued and q is a positive real number that corresponds to a Fourier mode of the self-adjoint and elliptic operator L.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

In the efficient and stable exponential Runge-Kutta schemes, for the above test equation, we take the splitting parameter κ in an adaptive way as

$$\kappa=\alpha\max\{0,-Re(\lambda)\}, \tag{4.2}$$

where $Re(\lambda)$ is the real part of λ and α is a non-negative constant to control the scale of splitting. Then, the test equation becomes

$$u_t=(\mathrm{L}u-\kappa u)+(\lambda u+\kappa u).$$

For the first order scheme, the efficient and stable exponential Runge-Kutta scheme is as same as the multistep scheme developed in [8]. Thus, they have the same stability regions as shown in [8]. In the case of homogeneous Dirichlet boundary condition, the efficient and stable exponential Runge-Kutta schemes is invertible. For the test equation, the second order Runge-Kutta scheme with the splitting parameter becomes

$$\begin{aligned}\mathbf{U}_{n+1}=\Big(&e^{(-q-\kappa)\Delta t}+(\lambda+\kappa)\Delta t^{-1}(-q-\kappa)^{-2}(e^{(-q-\kappa)\Delta t}-1)^2\\&+(\lambda+\kappa)^2\Delta t^{-1}(-q-\kappa)^{-3}\big(e^{(-q-\kappa)\Delta t}-1-(-q-\kappa)\Delta t\big)(e^{(-q-\kappa)\Delta t}-1)\Big)\mathbf{U}_n.\end{aligned} \tag{4.3}$$

In the case of $Re(\lambda)<0$, we have the splitting parameter $\kappa=-\alpha Re(\lambda)$ according to (4.2). Set $x=Re(\lambda\Delta t)$, $y=Im(\lambda\Delta t)$. Here, $Im(\lambda)$ is the imaginary part of λ. It is easy to get the amplification factor

$$\begin{aligned}\xi=&e^{-(q\Delta t-\alpha x)}+((1-\alpha)x+yi)\frac{(e^{-(q\Delta t-\alpha x)}-1)^2}{(q\Delta t-\alpha x)^2}\\&+((1-\alpha)x+yi)^2\frac{(e^{-(q\Delta t-\alpha x)}-1+(q\Delta t-\alpha x))(1-e^{-(q\Delta t-\alpha x)})}{(q\Delta t-\alpha x)^3}.\end{aligned} \tag{4.4}$$

We then obtain the boundary locus curve equation of the stability region in the left half plane ($Re(\lambda)<0$) as

$$|\xi|=1, \tag{4.5}$$

where $Re(\lambda\Delta t)$ and $Im(\lambda\Delta t)$ are implicitly defined. Different values of $q\Delta t$ will give us a family of boundary curves for stability regions. In the case of $Re(\lambda)\geq 0$, it is easy to see that $\kappa=\alpha\cdot 0=0$ from (4.2) for our scheme. Thus, set $\alpha=0$ in (4.5), we will get the boundary curve of the stability region in the right half plane ($Re(\lambda)\geq 0$).

Fig. 1(a) illustrates the stability regions (determined by (4.5)) of the second order efficient and stable exponential Runge-Kutta method with $\kappa=\max\{0,-Re(\lambda)\}$ for different values of $q\Delta t=1.0,2.0,4.0$. It is easy to find that the stability region gets larger when the value of $q\Delta t$ gets bigger. Fig. 2(b) shows the stability regions of the second order efficient and stable exponential Runge-Kutta scheme with different values of $\kappa=\alpha\max\{0,-Re(\lambda)\}$ with $\alpha=0,0.5,1.0,2.0$, for a fixed $q\Delta t=1.0$. We observe that the stability region grows larger

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

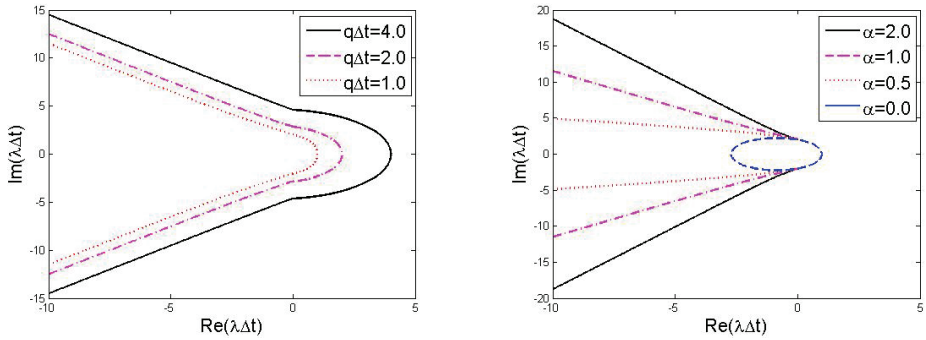


Figure 1: Stability regions (the left part or interior of the stability boundary cures) of the second order efficient and stable exponential Runge-Kutta schemes. (a) $\kappa=\max\{0,-Re(\lambda)\}$ with $q\Delta t=1.0,2.0,4.0$, respectively; (b) $q\Delta t=1.0$ and $\kappa=\alpha\max\{0,-Re(\lambda)\}$ with $\alpha=0,0.5,1.0,2.0$, respectively.

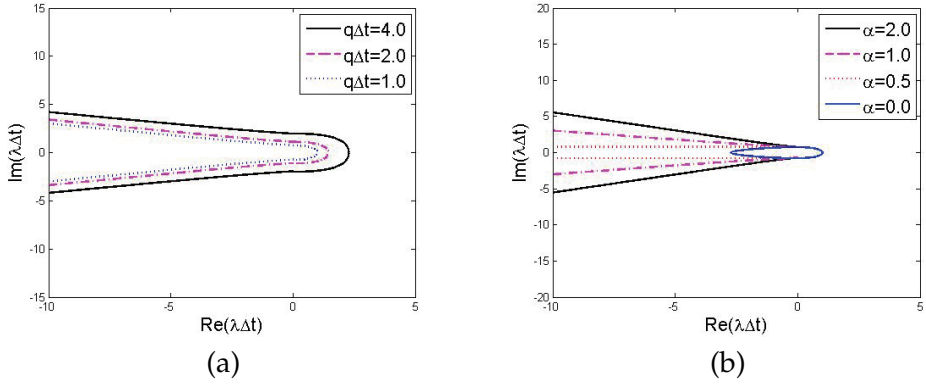


Figure 2: Stability regions (the left part or interior of the stability boundary cures) of the second order exponential muti-step schemes. (a): $\kappa=\max\{0,-Re(\lambda)\}$ with $q\Delta t=1.0,2.0,4.0$, respectively; (b): $q\Delta t=1.0$ and $\kappa=\alpha\max\{0,-Re(\lambda)\}$ with $\alpha=0,0.5,1.0,2.0$, respectively.

monotonically along with the increasing of κ in the second order exponential Runge-Kutta scheme.

In [8], the stability regions for the second order exponential multistep schemes were plotted (as shown in Fig. 2). From the above figures, we can see that the efficient and stable exponential Runge-Kutta scheme has bigger stability region than the multistep version, which implies that the former is more stable than the later. In order to show clearly this, we plot the stability regions of both the efficient and stable exponential Runge-Kutta scheme and the exponential multistep scheme in Fig. 3. Once again, Fig. 3 show that the efficient and stable exponential Runge-Kutta schemes are more stable than the exponential multistep scheme developed in [8].

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

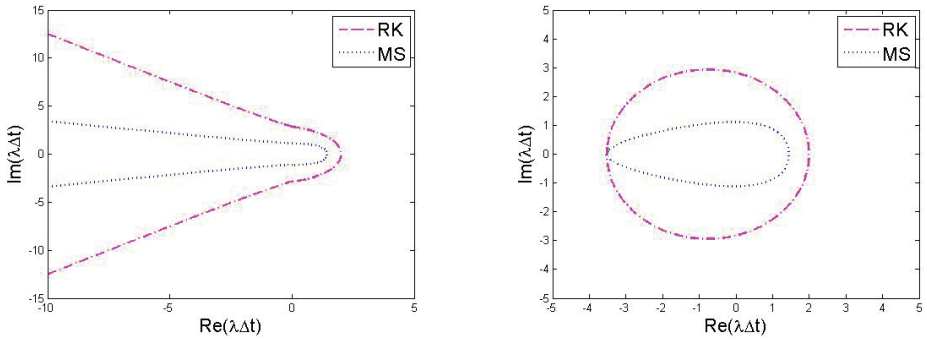


Figure 3: Comparision of the stability regions (the left part or interior of the stability boundary cures) of the second order efficient and stable exponential Runge-Kutta schemes with the multistep schemes. (a): $\kappa = \max\{0,-Re(\lambda)\}$ with $q\Delta t = 2.0$; (b): $\kappa = 0$ with $q\Delta t = 1.0$.

5 Numerical experiments

In this section, we test the accuracy, stability and robustness of the efficient and stable Runge-Kutta methods through both linear and nonlinear model examples.

5.1 Linear diffusion problems

The first example is a linear diffusion problem as follows. For this simple problem, we set the splitting parameter $\kappa = 0$.

Example 5.1. Let $\Omega = [-1,1]^2$ and $T = 1$, we consider

$$\begin{cases} \dfrac{\partial u}{\partial t} = \dfrac{9}{\pi^2}\Delta u, & (x,y)\in\Omega, \quad t\in[0,T], \\ u(0,x,y) = \cos(\pi x)+\sin(\pi y), & (x,y)\in\Omega. \end{cases} \tag{5.1}$$

Let us set the exact solution to be

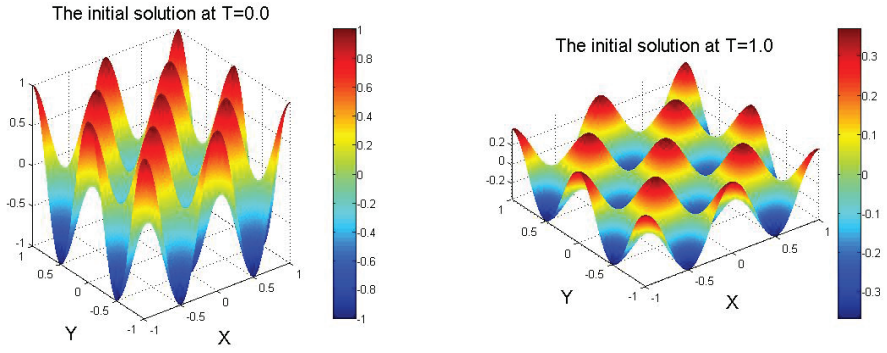
$$u(t,x,y) = e^{-9t}(\cos(\pi x)+\sin(\pi y)), \tag{5.2}$$

which satisfies the equation and initial condition. We test the case of Dirichle boundary condition which is determined accordingly from the exact solution. Clearly we have non-zero boundary conditions, $u|_{\partial\Omega} \neq 0$.

In Fig. 4, both the initial solution surface at $T = 0.0$ and the numerical solution surface at $T = 1.0$ on the mesh $N_x \times N_y \times N_t = 256\times256\times64$ are shown. For this simple and smooth solution, it is easy to test the numerical accuracy in both space and time.

Numerical results at the final time T of Example 5.1 with *Dirichlet boundary condition* produced by the efficient and stable exponential Runge-Kutta schemes are reported in Table 1. As expected, we clearly observe the second order accuracy of space discretization, and first, second, third and fourth order accuracy of time discretization.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press



(a) (b)

Figure 4: (a): the initial solution surface at $T=0.0$; (b): the numerical solution surface at $T=1.0$.

Table 1: Errors and convergence rates at the final time T of Example 5.1 using the efficient and stable exponential Runge-Kutta schemes.

Accuracy test of space discretization ($N_t=2048$)								
$N_x\times N_y$	16×16		32×32	64×64	128×128	256×256	512×512	
L_∞ Error	1.5522e-04		3.8595e-05	9.6254e-06	2.4049e-06	6.0114e-07	1.5029e-07	
CR	-		2.01	2.00	2.00	2.00	2.00	
Accuracy test of time discretization								
$N_x\times N_y\times N_t$	Ord=1	CR	Ord=2	CR	Ord=3	CR	Ord=4	CR
$(2048^2)\times4$	1.2139e-02	-	2.6366e-03	-	5.1490e-05	-	2.4170e-05	-
$(2048^2)\times8$	4.4314e-03	1.45	6.2999e-04	2.07	3.1893e-06	4.01	1.4607e-06	4.04
$(2048^2)\times16$	1.9088e-03	1.22	1.5572e-04	2.02	2.0675e-07	3.95	8.1819e-08	4.16
$(2048^2)\times32$	8.8791e-04	1.10	3.8810e-05	2.00	2.1604e-08	3.26	3.8197e-09	4.42

5.2 Allen-Cahn equation

We now consider the Allen-Cahn equation (with a normalized diffusion coefficient) which is nonlinear and in the whole space has a travelling wave solution. We use this to do the benchmark test [10].

Example 5.2. Let $\Omega=[-0.5,1.5]^2$, we consider

$$\begin{cases} \dfrac{\partial u}{\partial t}=\Delta u-\dfrac{1}{\epsilon^2}(u^3-u), & (x,y)\in\Omega, \quad t\in[0,T], \\ \dfrac{\partial u}{\partial \mathbf{n}}\Big|_{\partial\Omega}=0, & (x,y)\in\partial\Omega, \quad t\in[0,T], \\ u(0,x,y)=\dfrac{1}{2}\Big(1-\tanh\Big(\dfrac{x}{2\sqrt{2}\epsilon}\Big)\Big), & (x,y)\in\Omega. \end{cases} \tag{5.3}$$

The zero-Neumann boundary condition is imposed to allow for an approximate exact

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

solution (for $\epsilon << 1$) of the form

$$u(t,x,y) = \frac{1}{2}\left(1-\tanh\left(\frac{x-st}{2\sqrt{2}\epsilon}\right)\right) \tag{5.4}$$

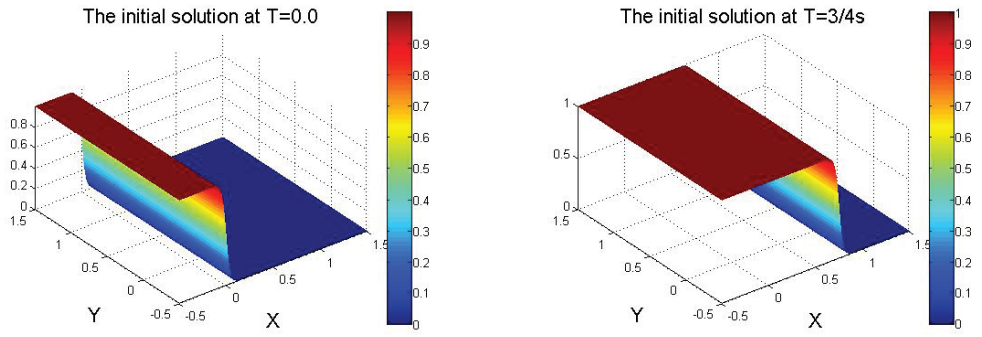
with $s = 3/\sqrt{2}\epsilon$. We set $\epsilon = 0.015$ (which is small enough) and $T = 3/4s$.

For the Allen-Cahn equation, as in [8], we set the splitting parameter $\kappa = 2/\epsilon^2$ to ensure the numerical stability. Then we apply the proposed Runge-Kutta schemes to solve

$$\frac{\partial u}{\partial t} = \Delta u - \frac{2}{\epsilon^2}u - \frac{1}{\epsilon^2}(u^3-3u), \quad \mathbf{x}\in\Omega, \quad t\in[t_0,t_0+T]. \tag{5.5}$$

The initial solution surface at $T = 0.0$ of this Allen-Cahn problem is shown in Fig. 5(a). From this figure we see the the solution decreases rapidly from 1 to -1 at the small range of ϵ. The wave will travels along with time while maintaining it shape. Fig. 5(b) shows the numerical solution surface of the Allen-Cahn equation at $T = 1.0$ on the mesh $N_x \times N_y \times N_t = 512 \times 512 \times 128$, which is consistent with what is expected. For the Allen-Cahn equation, due to the rapid change in a small range, it is a challenge to compute effectively its numerical solution.

In Table 2, we present the numerical results at the final time T of Example 5.2 produced by the efficient and stable exponential Runge-Kutta schemes. As expected we observe the second order accuracy of space discretization, and first, second, third and fourth order accuracy of time discretization when the time step size gets smaller. This shows that our method is robust and can be applied to solve effectively the complicated nonlinear parabolic problems.



(a) (b)

Figure 5: (a): the initial solution surface at $T = 0.0$; (b): the numerical solution surface at $T = 3/4s$.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

Table 2: Errors and convergence rates at the final time T of Example 5.2 using the efficient and stable exponential Runge-Kutta schemes.

Accuracy test of space discretization ($N_t=2048$)						
$N_x\times N_y$	32×32	64×64	128×128	256×256	512×512	1024×1024
L_∞ Error	8.4046e-01	3.2353e-01	9.5292e-02	2.4370e-02	6.1126e-03	1.5064e-03
CR	-	1.38	1.76	1.98	2.00	2.02
Accuracy test of time discretization ($ord=1$)						
$N_x\times N_y\times N_t$	$(2048)^2\times32$	$(2048)^2\times64$	$(2048)^2\times128$	$(2048)^2\times256$	$(2048)^2\times512$	$(2048)^2\times1024$
L_∞ Error	9.9985e-01	9.9654e-01	9.5799e-01	7.9049e-01	5.1438e-01	2.8453e-01
CR	-	0.01	0.06	0.28	0.62	0.85
Accuracy test of time discretization ($ord=2$)						
$N_x\times N_y\times N_t$	$(2048)^2\times32$	$(2048)^2\times64$	$(2048)^2\times128$	$(2048)^2\times256$	$(2048)^2\times512$	$(2048)^2\times1024$
L_∞ Error	9.8783e-01	8.0559e-01	3.6949e-01	1.1501e-01	3.1382e-02	7.9482e-03
CR	-	0.29	1.12	1.68	1.87	1.98
Accuracy test of time discretization ($ord=3$)						
$N_x\times N_y\times N_t$	$(2048)^2\times16$	$(2048)^2\times32$	$(2048)^2\times64$	$(2048)^2\times128$	$(2048)^2\times256$	$(2048)^2\times512$
L_∞ Error	9.8954e-01	7.4766e-01	2.2979e-01	4.1448e-02	5.9049e-03	4.8068e-04
CR	-	0.40	1.70	2.47	2.81	3.61
Accuracy test of time discretization ($ord=4$)						
$N_x\times N_y\times N_t$	$(2048)^2\times8$	$(2048)^2\times16$	$(2048)^2\times32$	$(2048)^2\times64$	$(2048)^2\times128$	$(2048)^2\times256$
L_∞ Error	9.984286e-01	8.9068e-01	2.9939e-01	3.8231e-02	3.0551e-03	1.2472e-04
CR	-	0.16	1.57	2.97	3.65	4.61

6 Conclusions

In this work, by integrating seamlessly the linear splitting technique and the high-order exponential Runge-Kutta integration approximations in time and compact spatial difference schemes on a regular mesh, we develop efficient and stable exponential Runge-Kutta methods for the solution of a class of parabolic equations. The exponential Runge-Kutta methods are easily implemented and easy to adopt the adaptive temporal approximations with variable time steps while they are applied to various different types of boundary data. In the future, we will develop the adaptive exponential Runge-Kutta methods in time (and/or space) based on the proposed schemes. In addition, to apply these methods to solve three-dimensional realistic nonlinear parabolic problems is also very interesting works to be further explored.

Acknowledgments

The work is supported in part by China Fundamental Research of Civil Aircraft under grant number MJ-F-2012-04 and and the Fundamental Research Funds for the Central Universities (YWF-15-SXXY-017).

References

[1] S. ALLEN AND J. W. CAHN, *A microscopic theory for antiphase boundary motion and its applica-*

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press

tion to antiphase domain coarsening, Acta Metall., 27 (1979), pp. 1084–1095.
[2] S. Cox and P. Matthews, *Exponential time differencing for stiff systems,* J. Comput. Phys., 176 (2002), pp. 430–455.
[3] Q. Du, and W.-X. Zhu, *Stability analysis and applications of the exponential time differencing schemes and their contour integration modifications,* J. Comput. Math., 22 (2004), pp. 200–209.
[4] Q. Du, and W.-X. Zhu, *Analysis and applications of the exponential time differencing schemes,* BIT Numer. Math., 45 (2005), pp. 307–328.
[5] M. Hochbruck, C. Lubich and H. Selhofer, *Exponential integrators for large systems of differential equations,* SIAM J. Sci. Comput., 19 (1998), pp. 1552–1574.
[6] M. Hochbruck and A. Ostermann, *Explicit exponential Runge-Kutta methods for semilinear parabolic problems,* SIAM J. Numer. Anal., 43 (2005), pp. 1069–1090.
[7] M. Hochbruck and A. Ostermann, *Exponential integrators,* Acta Numer., 19 (2010), pp. 209–286.
[8] L. Ju, J. Zhang, L. Zhu and Q. Du, *Fast explicit integration factor methods for semilinear parabolic equations,* J. Sci. Comput., 62 (2015), pp. 431–455.
[9] S. Krogstad, *Generalized integrating factor methods for stiff PDEs,* J. Comput. Phys., 203 (2005), pp. 72–88.
[10] Y. Li, H.-G. Lee, D. Jeong, and J. Kim, *An unconditionally stable hybrid numerical method for solving the Allen-Cahn equation,* Comput. Math. Appl., 60 (2010), pp. 1591–1606.
[11] V. Luan and A. Ostermann, *Explicit exponential Runge-Kutta methods of high order for parabolic problems,* J. Comput. Appl. Math., 256 (2014), pp. 168–179.
[12] C. Moler and C. V. Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later,* SIAM Rev., 45 (2003), pp. 3–49.
[13] S. Maset and M. Zennaro, *Stability properties of explicit exponential Runge-Kutta methods,* IMA J. Numer. Anal., 33 (2013), pp. 111–135.
[14] Q. Nie, F. Wan, Y.-T. Zhang and X.-F. Liu, *Compact integration factor methods in high spatial dimensions,* J. Comput. Phys., 227 (2008), pp. 5238–5255.
[15] D. Wang, L. Zhang and Q. Nie, *Array-representation integration factor method for high-dimensional systems,* J. Comput. Phys., 258 (2014), pp. 585–600.
[16] X. Yang, J. Feng, C. Liu and J. Shen, *Numerical simulations of jet pinching-off and drop formation using an energetic variational phase-field method,* J. Comput. Phys., 218 (2007), pp. 417–428.
[17] J. Zhang and Q. Du, *Numerical studies of discrete approximations to the Allen-Cahn equation in the sharp interface limit,* SIAM J. Sci. Comput., 31 (2009), pp. 3042–3063.

https://doi.org/10.4208/aamm.2015.m1045 Published online by Cambridge University Press