ADVANCES IN ATMOSPHERIC SCIENCES, VOL. 34, DECEMBER 2017, 1461–1471

• **Original Paper** •

A High-Order Spatiotemporal Precision-Matching Taylor–Li Scheme for Time-Dependent Problems

Pengfei WANG*[1,2]

[1] *Center for Monsoon System Research, Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing* 100190, *China*
[2] *State Key Laboratory of Numerical Modeling for Atmospheric Sciences and Geophysical Fluid Dynamics,*
Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing 100029, *China*

(Received 18 February 2017; revised 31 May 2017; accepted 2 June 2017)

ABSTRACT

Based on the Taylor series method and Li's spatial differential method, a high-order hybrid Taylor–Li scheme is proposed. The results of a linear advection equation indicate that, using the initial values of the square-wave type, a result with third-order accuracy occurs. However, using initial values associated with the Gaussian function type, a result with very high precision appears. The study demonstrates that, when the order of the time integral is more than three, the corresponding optimal spatial difference order could be higher than six. The results indicate that the reason for why there is no improvement related to an order of spatial difference above six is the use of a time integral scheme that is not high enough. The author also proposes a recursive differential method to improve the Taylor–Li scheme's computation speed. A more rapid and high-precision program than direct computation of the high-order space differential item is employed, and the computation speed is dramatically boosted. Based on a multiple-precision library, the ultrahigh-order Taylor–Li scheme can be used to solve the advection equation and Burgers' equation.

Key words: Taylor–Li scheme, high-order scheme, Burgers' equation

Citation: Wang, P. F., 2017: A high-order spatiotemporal precision-matching Taylor–Li scheme for time-dependent problems. *Adv. Atmos. Sci.*, **34**(12), 1461–1471, https://doi.org/ 10.1007/s00376-017-7018-1.

1. Introduction

Numerical models are powerful tools for probing weather prediction and climate change. Regarding the complex nonlinear interactions between variables, there are large uncertainties in model simulations. One of the main objectives of the Intergovernmental Panel on Climate Change is to gain an improved understanding of these uncertainties (Mastrandrea et al., 2010). The sources of such uncertainties mostly stem from the physical parameters, the framework of models, the observation errors, and the computation errors (von Neumann and Goldstine, 1947). The computation errors of numerical simulations generally exist in atmospheric general circulation models (Wang et al., 2007), coupled models (Song et al., 2012), simple chaotic dynamic systems (Li et al., 2000; Liao, 2009), and quasi-geophysical models (Teixeira et al., 2007). It is important to reduce the accumulation of computational errors in long-term numerical computations to obtain reasonable simulations.

In the field of weather and fluid mechanics, some models are defined as $\partial F/\partial t = LF$, where L is an operator involved in F and the spatial derivatives of F. When a numerical method is used to simulate F based on this type of equation, the accuracy of the algorithm depends on the step size, both spatially and temporally. The computation of spatial derivatives is an important research topic. Previous studies have shown that a second-order spatial difference scheme is enough, but other studies related to direct simulations of turbulent flow further demonstrated that a high-order algorithm is necessary (Lele, 1992). Since then, high-order algorithms have been applied in simulating complex fluid motion (Lele, 1992; Ma and Fu, 1996). Nevertheless, a lower-order (e.g., order of three) time-integration algorithm is often implemented. This causes a mismatch between the spatial and temporal computation precision in the model integration, along with a reduction in the accuracy of computations in long-term simulations.

Additionally, to reduce total errors in numerical simulations, it is necessary to improve the spatiotemporal precision (Tal-Ezer, 1986, 1989). Building a spatiotemporal precision-matched high-order scheme for the integration of an atmospheric numerical model could help reduce the uncertainty associated with computation errors in numerical experiments.

Two types (directions) of computations are used when applying numerical methods to solve time-dependent partial differential equations (PDEs): the spatial direction (differ-

* Corresponding author: Pengfei WANG
Email: wpf@mail.iap.ac.cn

ence method, spectral method, finite volume method, etc.), and the temporal direction (time-integration method). It is possible to reduce the errors in the difference method by increasing the order of schemes or by reducing the grid size. However, increasing the scheme order is generally difficult in a three-dimensional complex fluid dynamical system. Also, this often leads to an increase in the computation by 2^4 times when reducing the grid size by half. While high-precision schemes have been proposed, the computation stability decreases with increasing precision. To ameliorate the deficiencies, Ji et al. (1999) applied a conservation scheme to balance computation precision and stability.

Li (2005) proposed a new method for computing the high order of spatial derivatives. The advantage of this method is that it applies an explicit scheme to perform computations. In addition, the method is time-saving and has the ability to obtain derivatives of 10 orders or higher. Feng and Li (2007) employed the method of Li (2005) in a high-order scheme for a one-dimensional advection equation, the in-viscid Burgers' equation (Hopf, 1950), the barotropic vorticity equation, and shallow-water equation, and found that the sixth-order scheme produced the best result without a large increase in computation time. However, when the scheme order increased (e.g., to 7–10), the improvement in the result was only slight, and even became worse. When the order was higher than seven, the results became weak compared to the sixth order.

There are two ways to reduce the errors of time integration: decrease the time step-size or increase the time integration order. Decreasing the time step-size is the simplest method, as discussed in Li et al. (2000) and Teixeira et al. (2007), who showed the sensitivity of the results to time step-size. However, the result for a chaotic system is not only sensitive to the time step-size, but also the order of the scheme and float-point precision. Additionally, the actual computation time is limited by the cost of the executing program. Wang et al. (2012) examined the cost of a fourth-order Runge–Kutta method and the Taylor method in computation of the Lorenz system, and argued that higher-order methods can decrease the computation time exponentially.

Previous studies have indicated that applying a high-order scheme to reduce the time integration error is important. For example, Li et al. (2000) utilized 2–10-order numerical schemes to investigate the error evaluation rule. In addition, Lorenz (2006) reported that the Taylor series method has high-order characteristics, and is suitable for the study of chaotic dynamical systems. By applying the Moore (1966, 1979) recurrence method to compute coefficients, Barrio et al. (2005, 2011), Liao (2009) and Wang et al. (2014) used an ultrahigh order (in this study, the author regards a 4–9-order scheme as being high order, and an ⩾ 10-order scheme as ultrahigh order) Taylor method to examine the Lorenz system, Kepler system, and Henon–Heiles system (Hénon and Heiles, 1964). These results suggest that, for a certain predefined time, t, the method is capable of generating highly precise numerical solutions, and the Taylor series method has a strong ability and broad application prospects in issues related to high precision or long-term integration.

Base on the high-order Taylor series method, the forward period analysis (FPA) method (Wang, 2016), which is very efficient for the long-term simulation of periodic Hamiltonian systems, was built. Some PDE systems also have periodic properties [e.g., the barotropic vorticity equation on a sphere (Neamtan, 1946)], while others may not [e.g., the Allee effect (Sun, 2016) in population dynamics]. An issue often encountered when applying FPA to the periodic PDE system is that there is no integration scheme available with sufficiently high precision for the integration within one cycle of such PDE and then determine the period value of the PDE system. Nevertheless, Sun et al. (2016) pointed out that the spatial dynamic pattern described by a PDE is ubiquitous in nature, and building a high-order scheme for a PDE dynamical system is also valuable for FPA in dealing with such systems.

Feng and Li (2007) conducted numerous experiments by applying the Li (2005) method in simple PDE problems. Nevertheless, there are two problems that need to be solved when applying such high-order schemes. Firstly, the higher-order scheme is only used in the spatial direction, with a low-order (third-order Runge–Kutta) method used in the temporal direction. Therefore, errors still accumulate in the numerical results, mainly due to the time-integration error. This means a ⩾ 6-order spatial algorithm cannot produce adequate results. Secondly, the initial values selected are not continuous in the spatial direction, or the spatial derivative is infinite, so that the difference method is unstable and the high-order algorithm is unable to generate results superior to a low-order algorithm. However, these deficiencies do not mean that the Li (2005) method is unsuitable for problems where the derivative is continuous and bounded. Therefore, considering these problems, a high-accuracy algorithm for time-dependent differential equations remains to be established.

In this paper, the author proposes a high-order scheme in both the spatial and temporal direction to compute a kind of time-dependent PDE. Several experiments are also conducted to investigate the type of problems that this high-order scheme is suitable for, and those for which it is not, so as to determine the advantages of such a high-order scheme.

2. Taylor–Li scheme

2.1. *The linear case*

The form of a one-dimensional advection equation is

$$\frac{\partial u}{\partial t}+\frac{\partial u}{\partial x}=0\,, \tag{1}$$

which can be solved by many high-precision algorithms. The Taylor method is the traditional method, and many studies have discussed this method in solving chaotic systems, such as the Lorenz system (Lorenz, 2006; Liao, 2009; Wang et al., 2014). Here, we briefly introduce a τ time step-size and M order Taylor series scheme (in grid position x_i), as

$$u^{n+1}(x_i) = u^n(x_i) + \sum_{k=1}^{M} \alpha_k \tau^k , \tag{2}$$

where

$$\alpha_k = \frac{1}{k!}\frac{\partial^k u(t_n, x_i)}{\partial t^k} \equiv \frac{1}{k!} u_t^{(k)}(t_n, x_i) ,$$

n is the step, k is an integer, and (k) in the superscript means k-th derivatives.

$$u^{n+1}(x_i) = u^n(x_i) + \sum_{k=1}^{M} \frac{\tau^k}{k!} u_t^{(k)}(x_i) , \tag{3}$$

and

$$u_t^{(k)} = \frac{\partial^k u}{\partial t^k} .$$

The core process involved in the Taylor series method is computation of the derivative, $u_t^{(k)}$. According to Eq. (1), the first-order time derivative is

$$u_t^{(1)}(x_i) = -\frac{\partial u(x_i)}{\partial x} \equiv -u_x^{(1)}(x_i) ,$$

and the second-order time derivative is

$$\begin{aligned} u_t^{(2)}(x_i) &= \frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right) = -\frac{\partial}{\partial t}\left(\frac{\partial u(x_i)}{\partial x}\right) \\ &= -\frac{\partial}{\partial x}\left(\frac{\partial u(x_i)}{\partial t}\right) = -\frac{\partial}{\partial x}(u_t^{(1)}(x_i)) . \end{aligned}$$

Since $u_t^{(1)}(x_i)$ is computed in a previous step, the second-order time derivative can be computed from $u_t^{(1)}(x_i)$ by its partial derivative in space, and the kth-order time derivative is

$$u_t^{(k)}(x_i) = (-1)^{k-1} \frac{\partial}{\partial x}(u_t^{(k-1)}(x_i)) . \tag{4}$$

In addition, the arbitrary spatial derivative formula is obtained from Li (2005):

$$f_y^{(m)}(y_i) = \frac{1}{h^m} \sum_{j=0}^{n} d_{n+1,i,j}^{(m)} f(y_j) . \tag{5}$$

The precision of the formula is in the $(n-m+1)$ order, where i is the position number of reference points, j is the position number of coefficients points and the meaning of $d_{n+1,i,j}^{(m)}$ can be found in Li (2005). In this study, we choose $m \equiv 1$, and the nth-order precision spatial derivative requires $(n+1)$ grid points. It should be noted that y_i in the formula is a relative position numbered from 0 to n, and not the absolute grid position of $u(x_i)$ on an actual grid.

The N grid is in the x direction, and each $\partial(u_t^{(k-1)}(x_i))/\partial x$ item can be computed by Eq. (5). The precision involves the nth order using $(n+1)$ grids selected from N grid points $(n \leqslant N-1)$. Equation (4) indicates that $u_t^{(k)}$ can be obtained by $(-1)^{k-1}\partial(u_t^{(k-1)}(x_i))/\partial x$. If x_i is computed and $(n+1)$ is an odd number, then the $(n+1)$ selected grids are $(x_{i-n/2}, \cdots, x_i, \cdots, x_{i+n/2})$ ($x_{i-n/2}$ corresponds to $y_0, \cdots, x_i$ corresponds to $y_{n/2}, \cdots, x_{i+n/2}$ corresponds to y_n) and x_i is located in the center of the $(n+1)$ selected grid. However, when $(n+1)$ is an even number, the selected n grid points are $(x_{i-(n+1)/2}, \cdots, x_i, \cdots, x_{i+(n+1)/2-1})$, which makes x_i approach the center grid points. x_i can be set to a position where the n grid points continue besides the left or right lateral, but it cannot be set at the center when x_i approaches the boundary. Therefore, we select periodical boundary conditions when they are available; x_i is close to the center of all computation points.

The first-order spatial derivative ($m \equiv 1$) is

$$f_y^{(1)}(y_i) = \frac{1}{h} \sum_{j=0}^{n} d_{n+1,i,j}^{(1)} f(y_j) .$$

The computation retains the nth-order precision in the spatial direction, and the computation of temporal derivatives $u^{(M)}(x_i) = (-1)^{M-1}\partial(u_t^{(M-1)}(x_i))/\partial x$ only requires the formula of $f_y^{(1)}(y_i)$; in this way, we can simplify the computation of $d_{n+1,i,j}^{(1)}$. In order to obtain a certain accuracy of the time derivative, a grid with a size of at least $(M+1)$ is required. For example, when $N = 64$ and $M = 5$, six grid points are needed in each computation (and so on). The specific computation formula is as follows:

$$\begin{aligned} & d_{2,0,1}^{(1)} = 1 , \quad d_{2,1,0}^{(1)} = -1 ; \\ & d_{n+1,i,j}^{(1)} = \frac{(-1)^{(1-j)} a_{n-1,i,j}^{(0)}}{j!(n-j)!} , \quad (\text{if } i \neq j) ; \\ & d_{n+1,i,i}^{(1)} = -\sum_{j=0, j\neq i}^{n} d_{n+1,i,j}^{(1)}, \quad (\text{if } i = j) ; \\ & a_{n-1,i,j}^{(0)} = a_0(-i, \cdots, k-i, \cdots, n-i), (k \neq i, k \neq j) \\ & \qquad = (-i)\cdot(\cdots)\cdot(k-i)\cdot(\cdots)\cdot(n-i), (k \neq i, k \neq j) . \end{aligned}$$

The data array in the program is $d[m][n][i][j]$ ($m = 1$), and one dimension can be reduced, i.e., to $d^{(1)}[n][i][j]$. The n depends on M in general, and once n is selected the data array can then be reduced to a two-dimension array $d_{n+1}^{(1)}[i][j]$, which costs a maximum storage space of $(n+1) \times (n+1)$.

2.2. *The nonlinear case*

In view of the nonlinear equation

$$\frac{\partial u}{\partial t} = A \frac{\partial u}{\partial x} , \tag{6}$$

where $A = A(u)$, the initial value is located in N grid points. We select n as the grid point from N grid points and apply Eq. (5) to compute $u_t^{(1)} = A(\partial u/\partial x)$. The result is of nth order $(O(\Delta x^n))$.

If we then let $B = \partial u/\partial x$, then

$$u_t^{(2)} = \frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial t}(AB) = \frac{\partial A}{\partial t}B + A\frac{\partial B}{\partial t} = \frac{\partial A}{\partial u}\frac{\partial u}{\partial t}B + A\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial t}\right).$$

Since $\partial u/\partial t = u_t^{(1)}$, we obtain an nth-order precision numerical value in the previous step, and the item $\partial u_t^{(1)}/\partial x$ can then also be computed by Eq. (5), as

$$u_t^{(3)} = \frac{\partial^3 u}{\partial t^3} = \frac{\partial^2}{\partial t^2}(AB) = \frac{\partial^2 A}{\partial t^2}B + 2\frac{\partial A}{\partial t}\frac{\partial B}{\partial t} + A\frac{\partial^2 B}{\partial t^2} ,$$

where

$$\frac{\partial^2 B}{\partial t^2} = \frac{\partial^2}{\partial t^2}\left(\frac{\partial u}{\partial x}\right) = \frac{\partial}{\partial x}\left(\frac{\partial^2 u}{\partial t^2}\right) = \frac{\partial(u_t^{(2)})}{\partial x} .$$

Similarly, $u_t^{(2)}$ is obtained in the last step, and thus the item $\partial(u_t^{(2)})/\partial x$ can also be computed by Eq. (5). Correspondingly,

$$\begin{aligned} u_t^{(k+1)} &= \frac{\partial^{k+1} u}{\partial t^{k+1}} = \frac{\partial^k}{\partial t^k}(AB) = \sum_{m=0}^{k} C_k^m \frac{\partial^m A}{\partial t^m}\frac{\partial^{k-m} B}{\partial t^{k-m}} \\ &= \sum_{m=0}^{k} C_k^m \frac{\partial^m A}{\partial t^m}\frac{\partial}{\partial x}u_t^{(k-m)} , \end{aligned}$$

where

$$\frac{\partial^0 A}{\partial t^0} = A, \quad \frac{\partial^0 B}{\partial t^0} = B .$$

The item that includes $\partial^m A/\partial t^m$ can be computed using the following procedure:

$$\begin{aligned} \frac{\partial A}{\partial t} &= \frac{\partial A}{\partial u}\frac{\partial u}{\partial t} = A_u^{(1)} u_t^{(1)} ; \\ \frac{\partial^2 A}{\partial t^2} &= \frac{\partial}{\partial t}(A_u^{(1)} u_t^{(1)}) = A_u^{(2)}(u_t^{(1)})^2 + A_u^{(1)} u_t^{(2)} . \end{aligned}$$

The higher-order item can be obtained analogously, which ultimately gives

$$\frac{\partial^m A}{\partial t^m} = \frac{\partial^{m-1}}{\partial t^{m-1}}(A_u^{(1)} u_t^{(1)}) = \sum_{i=0}^{m-1} C_{m-1}^i \frac{\partial^i A_u^{(1)}}{\partial t^i}\frac{\partial^{m-1-i} u_t^{(1)}}{\partial t^{m-1-i}} .$$

Therefore, we can obtain the $1-(k+1)$-order time derivatives, and each of the $(k+1)$-order items can be computed using the previous kth-order numerical values. It is also possible to reduce the overall computation time by repeatedly using the previous result. Therefore, it is necessary to have a proper organizational data storage strategy.

We apply the scheme to solve the one-dimensional inviscid Burgers' equation,

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 . \tag{7}$$

This equation is a special case of Eq. (6) for $A(u) = -u$, and we regard $B = \partial u/\partial x$.

The analytical solution of this equation has properties whereby if the initial solution is $u(x,0) = f(x)$, then $u(x,t) = f(x - u(x,t)t)$ is the solution of time, t.

In the computation, item $\partial^m A/\partial t^m$ can be computed using the following procedure:

$$\frac{\partial A}{\partial u} = -1, \text{ i.e., } A_u^{(1)} = -1,$$

and

$$\begin{aligned} &A_u^{(m)} = 0,\ (m > 1), \text{ i.e., } \frac{\partial A}{\partial t} = \frac{\partial A}{\partial u}\frac{\partial u}{\partial t} = -u_t^{(1)}; \\ &\frac{\partial^2 A}{\partial t^2} = A_u^{(1)} u_u^{(2)} = -u_t^{(2)} , \quad \frac{\partial^m A}{\partial t^m} = A_u^{(1)} u_t^{(m)} = -u_t^{(m)} ; \end{aligned}$$

$$\begin{aligned} u_t^{(1)} &= -u\frac{\partial u}{\partial x} ; \\ u_t^{(2)} &= \frac{\partial A}{\partial t}B + A\frac{\partial B}{\partial t} = -u_t^{(1)}B - u\frac{\partial}{\partial x}(u_t^{(1)}) ; \\ u_t^{(3)} &= \frac{\partial^2 A}{\partial t^2}B + 2\frac{\partial A}{\partial t}\frac{\partial B}{\partial t} + A\frac{\partial^2 B}{\partial t^2} \\ &= -u_t^{(2)}B - 2u_t^{(1)}\frac{\partial}{\partial x}(u_t^{(1)}) - u\frac{\partial}{\partial x}(u_t^{(2)}) ; \\ u_t^{(k+1)} &= \sum_{m=0}^{k} C_k^m \frac{\partial^m A}{\partial t^m}\frac{\partial^{k-m} B}{\partial t^{k-m}} = -\sum_{m=0}^{k} C_k^m u_t^{(m)}\frac{\partial}{\partial x}(u_t^{(k-m)}) . \end{aligned}$$

Where C denotes the combination. It is easy to determine that $u_t^{(k)}(x_i)$ can be obtained from the previous spatial derivative, the time derivative, or their combination.

After we obtain $u_t^{(k)}$ in each grid, we use the formula

$$u^{n+1}(x_i) = u^n(x_i) + \sum_{k=1}^{M}\frac{\tau^k}{k!}u_t^{(k)}(x_i)$$

to compute the solution in the next time step; and the solution precision is $O(h^q + \tau^M)$, where M and q represent the order of the time integration and the spatial difference, respectively. This scheme is tested in section 3.1 and 3.2. It firstly computes the spatial derivatives using the Li (2005) method. Secondly, it computes recurrence of the time derivatives. Thirdly, it applies the Taylor series method to compute time integrations as the Taylor–Li scheme.

2.3. *Implementation of multiple-precision Taylor–Li scheme*

The scheme in this study is distinct from that of Feng and Li (2007) in the computation of item $f_y^{(m)}(y_i)$, where $d_{n+1,i,j}^{(1)}$ is repeated m times as opposed to directly applying $d_{n+1,i,j}^{(m)}$. This procedure is a recurrence type of computation, the benefits of which are as follows: Firstly, when m and n are large integers, then the computation of $d_{n+1,i,j}^{(m)}$ inside the program is time-consuming, particularly when m is higher than 10 orders. The memory required for direct computation is $m \times (n+1) \times (n+1) \times (n+1)$, which is a large amount of memory that may decrease the CPU cache rate and ultimately slow down the computation speed. Secondly, even if the coefficients $d_{n+1,i,j}^{(m)}$ are applied as previous data inputs, it is still not possible to reduce the memory usage. Thirdly, the item $d_{n+1,i,j}^{(1)}$ is simple and fast in the actual computation. Once the result is stored, it can be used many times. In addition, the memory storage size is one to two orders of magnitude smaller than that of $d_{n+1,i,j}^{(m)}$.

The next problem related to the high-order scheme is the round-off error. As the computation of spatial difference is at orders higher than 6–10 in the experiments of this study, the solution value is $O(1)$. When only the double-precision computation is used, the absolute error of the high-order scheme oscillates between 10^{-15} and 10^{-16}, which is the minimal relative error of double-precision float computation. In order to analyze the performance of the high-order scheme, multiple-precision computations are necessary. In this study, the multiple-precision (MP) library is used. In the computation, we apply a float precision of 1024 bits, which corresponds to above 200 significant digits in the decimal system. This precision can identify the absolute error at around 10^{-200} in the numerical results. It is worth noting that the speed of the MP version is 100 times slower than that of double precision (see Table 1). Therefore, the author plans to reduce the wall-time in future versions by using parallel computation.

Table 1 lists the time cost of the computation for the advection equation. This benchmark starts from the 6th-order (for the non-recurrence method), and adds 1 each time until the 16th order. The initial order is 10 for the recurrence method, and the interval is 10 until the 100th order. The benchmark results indicate that, for the non-recurrence method, when (M,n) increases by 1 order, the time cost is increased by a factor of 4.25. For the recurrence method, when (M,n) increases by 10 orders, the time cost is only increased by a factor of 1.5. Therefore, the increase in speed when using the recurrence method is much more remarkable than that of the non-recurrence method.

Table 1. Time cost (wall-time; units: s) in Expt. 1 from different methods. DP refers to double precision and MP multiple precision. The platform for the benchmark is an Intel E5-2640 2.6 GHz CPU running on a Linux system.

Temporal and spatial order	Non-recurrence method (DP)	Recurrence method (DP)	Recurrence method (MP)
$M = 6, n = 6$	0.01	< 0.01	-
$M = 6, n = 10$	0.02	< 0.01	-
$M = 10, n = 10$	0.42	< 0.01	2.37
$M = 11, n = 11$	1.74	-	-
$M = 12, n = 12$	7.44	-	-
$M = 13, n = 13$	31.96	-	-
$M = 14, n = 14$	136.80	-	-
$M = 15, n = 15$	583.36	0.01	3.21
$M = 16, n = 16$	2476.49	-	-
$M = 20, n = 20$	-	0.03	4.39
$M = 30, n = 30$	-	0.06	7.30
$M = 40, n = 40$	-	0.11	11.78
$M = 50, n = 50$	-	0.17	17.11
$M = 60, n = 60$	-	0.24	24.46
$M = 70, n = 70$	-	0.33	33.54
$M = 80, n = 80$	-	0.43	41.86
$M = 90, n = 90$	-	0.55	51.53
$M = 100, n = 100$	-	0.69	63.41

3. Linear advection experiments using the Taylor–Li scheme

3.1. *Expt. 1—Appling the square initial wave to solve [Eq. (1)]*

We apply the square initial wave conditions to solve Eq. (1),

$$u(x)|_{t=0} = \begin{cases} 0, & x < \dfrac{3}{32}, \\ 1, & \dfrac{3}{32} \leqslant x \leqslant \dfrac{9}{32}, \\ 0, & \dfrac{9}{32} < x \leqslant 1, \end{cases}$$

where the spatial step-size in the x direction is $h = 1/64$, the time step-size is $\tau = 1/128$, the computed region is $x \in [0,1]$, and the integration is 128 steps (just one period). Theoretically, the final status is the same as the initial status. In this experiment, the order of time integration is from three to six.

Based on Takacs (1985), we define the total error $E = \sqrt{\sum_{i=1}^{N}(u_D - u_T)^2/N}$, where u_D means the numerical solution, u_T is the analytical solution, and N is the grid numbers in the x direction. Here, we apply the square root to enable the error to have the same units as the original variables.

The simulated results are shown in Fig. 1b. When the time integration order is $M = 3$ and the spatial difference order is

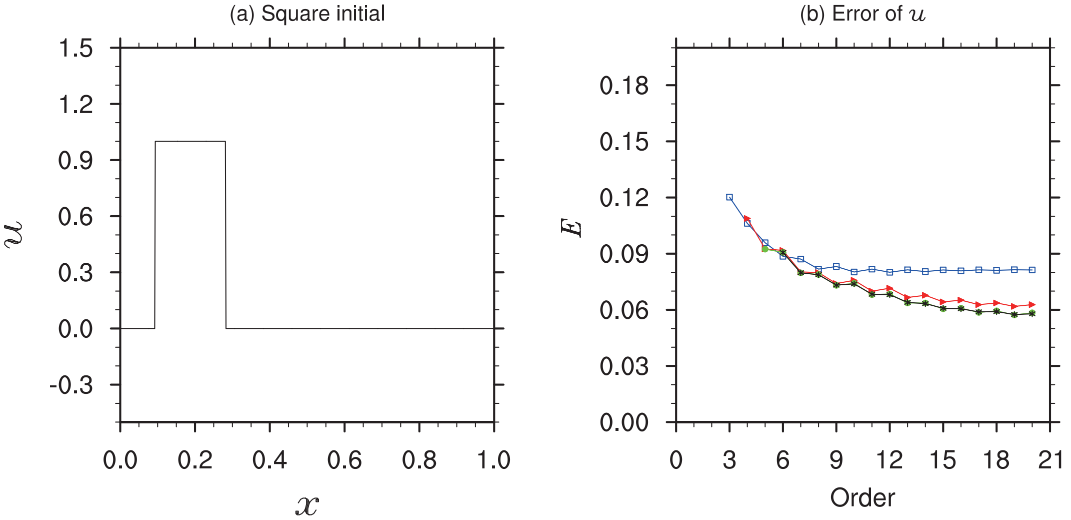

Fig. 1. (a) Square initial. (b) Error versus spatial difference order, where the abscissa is the spatial difference order, the ordinate is error, and curves in blue, red, green and black correspond to the third, fourth, fifth and sixth time-integration orders, respectively.

$n = 6$, the error decreases to a minimum. In addition, when the spatial difference order, n, is above six, there is no significant decrease in the error. When the time integration order increases to ($M = 4,5,6$), the error continues to decrease with an increase of n, but the error value does not decrease to such a large extent. Furthermore, the curves of $M = 5$ and 6 overlap. The results suggest that, with the square initial wave, it is not possible to improve the result with a time integration order above five.

3.2. *Expt. 2—Applying the Gaussian initial wave* $u(x)|_{t=0} = e^{-400(x-0.5)^2}$ *to initial conditions to solve the advection equation [Eq. (1)]*

The spatial step-size in the x direction is $h = 1/200$, the time step-size is $\tau = 1/400$, the computed region is $x \in [0,1]$, and there are 400 steps involved in the computation.

In Fig. 2b, when $M = 3$ and n reaches the sixth order, the error becomes invariant; this result is in broad agreement with that of Feng and Li (2007). However, for the $M = 4$ order, when n reaches the eighth order, the error becomes minimal; for the curve of $M = 5$, when n reaches the 10th order, the error become minimal; for the curve of $M = 6$, and when n reaches the 12th order, the error become minimal. These results indicate that, for these types of initial values, the total error of the result is greatly influenced by the time integration. When the time integration order is high enough, the order of the spatial difference exceeds far beyond six.

A comparison of the results from Expt. 1 and Expt. 2 shows that, for the initial value of the square wave, the high-order scheme is only able to produce a result close to that of the third-order time integration scheme. However, for the smooth periodical initial values, it is easy to obtain a good result. The reason for this difference is that the first type of initial values is non-continuous, which also makes the derivative in the x direction discontinuous, and the computation result is not very accurate. For the second type of initial values, all the derivatives are continuous and the computation error is very small, leading to a better result.

4. In-viscid Burgers' equation experiments using the Taylor–Li scheme

4.1. *Expt. 3—Applying the initial condition of* $u(x,0) = -\sin x$ *to solve the in-viscid Burgers' equation [Eq. (7)]*

The nonlinear case is different from the linear one, and $u_t^{(k+1)} = \sum_{m=0}^{k} C_k^m (\partial^m A/\partial t^m)(\partial^{k-m} B/\partial t^{k-m}) = -\sum_{m=0}^{k} C_k^m u_t^{(m)} \partial(u_t^{(k-m)})/\partial x$ uses all the previous mth-order time derivatives, $u_t^{(m)}$, and not only the kth-order time derivatives. In the program, to maintain precision of the time derivatives and avoid the error accumulation from spatial derivatives, n is usually set to be larger than M, and the actual value of n can be determined from experiments.

In the experiments, the x-axis computation region is $[-\pi,\pi]$, the grid number $N = 800$, the time step-size is $\tau = 0.001$, and the lateral periodical boundary conditions are applied. The target time of experiments is $t = 0.8$, with a total integration of 800 steps.

The results of Expt. 3 are shown in Fig. 3. Since the total error is smaller than 10^{-15}, the experiment is conducted using the MP version of the program. Figure 3c indicates that, when $M = 3$, and n reaches the sixth order, the error is almost unchanged (thus order six is called the maximal effective spatial difference order corresponding to $M = 3$). In addition, when the time integration order is the fourth, fifth and sixth order, the effective spatial difference order is 11, 18 and 33. The results illustrate that, for these types of initial values, the in-viscid Burgers' equation can use the spatial difference scheme far beyond six, as long as the time integration

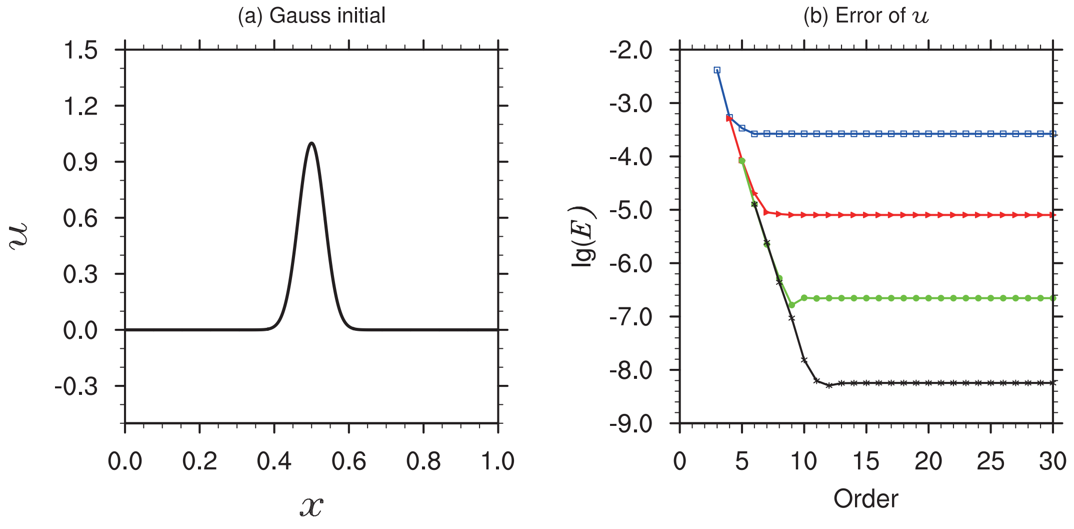

Fig. 2. (a) Gaussian-type initial. (b) Error versus spatial difference order, where the abscissa is the spatial difference order, the ordinate is the logarithm of error ($\log_{10}$), and the blue, red, green and black curves denote the third, fourth, fifth and sixth time-integration orders, respectively.

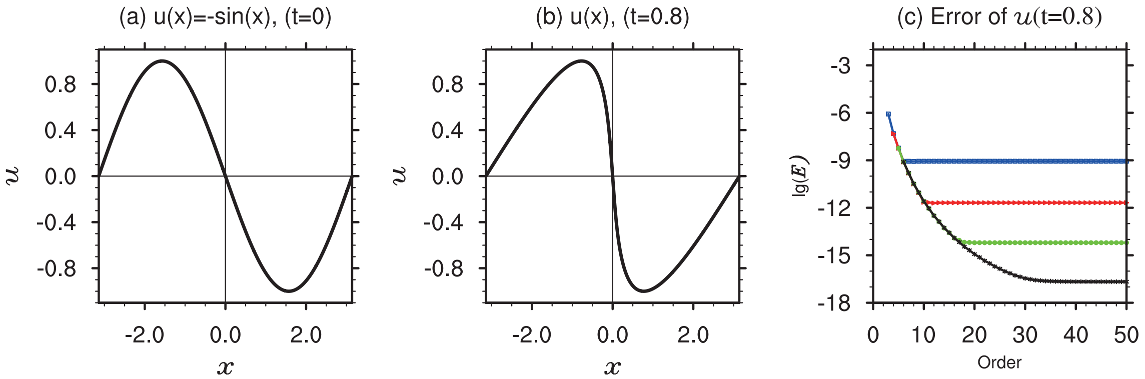

Fig. 3. (a) Initial u at $t = 0$. (b) Analytical solution of u at $t = 0.8$. (c) Error versus spatial difference order, where the abscissa is the spatial difference order and the ordinate is the logarithm of error ($\log_{10}$); curves in blue, red, green and black correspond to the third, fourth, fifth and sixth time-integration order, respectively.

order is high enough.

4.2. *Expt. 4—Applying the initial condition of* $u(x,0) = -\arctan(x)$ *to solve the in-viscid Burgers' equation [Eq. (7)]*

In the experiments, the x-axis computation region is $[-1,1]$, the grid number $N = 64$, and the time step-size is $\tau = 0.005$. The lateral boundary conditions are the analytical conditions $u(-1,t)$ and $u(1,t)$, resulting from the algebraic equation $u + \arctan(x - ut) = 0$. The target computation times for this experiment are $t = 0.5$, $t = 0.8$, and $t = 1.0$.

Figure 4d shows the solution at $t = 0.5$, when the shock wave has not appeared. The high-order scheme is capable of generating solutions with good precision. In this experiment, when $M = 3$, the spatial difference order reaches the sixth to eighth orders, and the error trend increases. When the time integration order is 4, 5 and 6, the minimal total error position moves to the right at the horizontal axis and the value of the total error becomes smaller. This result indicates that an increase in the spatial difference order can reduce the total error, and the corresponding effective spatial difference orders are 10, 16 and 16. These effective spatial difference orders are all above six, and the results are distinct from those of Feng and Li (2007).

The shock wave does not occur when the integration time reaches $t = 0.8$ (Fig. 4e). However, it is closer to the time of shock-wave occurrence. At this moment, the high-order scheme obtains a certain precision solution, but the precision is lower than that of time $t = 0.5$. The total error is at a magnitude of 10^{-5}, which is bigger than the error $10^{-10} - 10^{-8}$ of $t = 0.5$. Although an increase in the time integration order can cause the minimal error point to move right at the horizontal axis (spatial difference order increase), there is no obvious decrease in the minimal error.

If the integration time approaches $t = 1.0$ (Fig. 4f), a singularity point appears and the shock wave occurs. In this situation, any of the numerical methods used to compute the derivatives is not very accurate, and the error accumulates from the singularity. Figure 4 indicates that the high-order scheme is able to generate similar precision as the third-order scheme, consistent with the result of Feng and Li (2007).

In summary, for the in-viscid Burgers' equation, the ability of the Taylor–Li scheme to generate good results depends strongly on the form of the initial values, mainly because of the target computation time. When the derivative is continuous (and does not appear as an infinite value) at the target computation time, the high-order scheme is effective. In contrast, if the derivative is discontinuous, or is a derivative to infinity, shock waves occur and the higher-order scheme is unable to obtain a high-precision numerical solution.

Furthermore, once the numerical computation is applied to periodical boundary conditions, the precision of the solution is generally higher than the non-periodical one. A possible reason for this is that, when the periodical boundary conditions are applied, the objective computed point can be placed at, or next to, the center of the selected grid points. The derivative error is therefore smaller than that when the objective point is not located near the center. This issue indicates that when the formula of Li (2005) is applied to the edge points, it is not as accurate as when applied to the center points.

5. Ultrahigh-order experiment using the Taylor–Li scheme

5.1. *Expt. 5—ultrahigh-order result of Expt. 2*

This experiment is the same as Expt. 2, but the time integration order is set to $M = [10,40]$, interval of 10, and the spatial difference order (n) is from M to 100 or above.

Figure 5 shows that the error decreases at a time integration order of 10. When the spatial difference order increases (order 11–23) up until the 23rd order, the error remains constant. This implies that, for $M = 10$, the maximal effective spatial differential order is 23. In the case of $M = 20$, the range of the spatial difference order causing the error to decrease is 21–75, and a near-constant error is then maintained. When the time integration order increases to $M = 30$ and

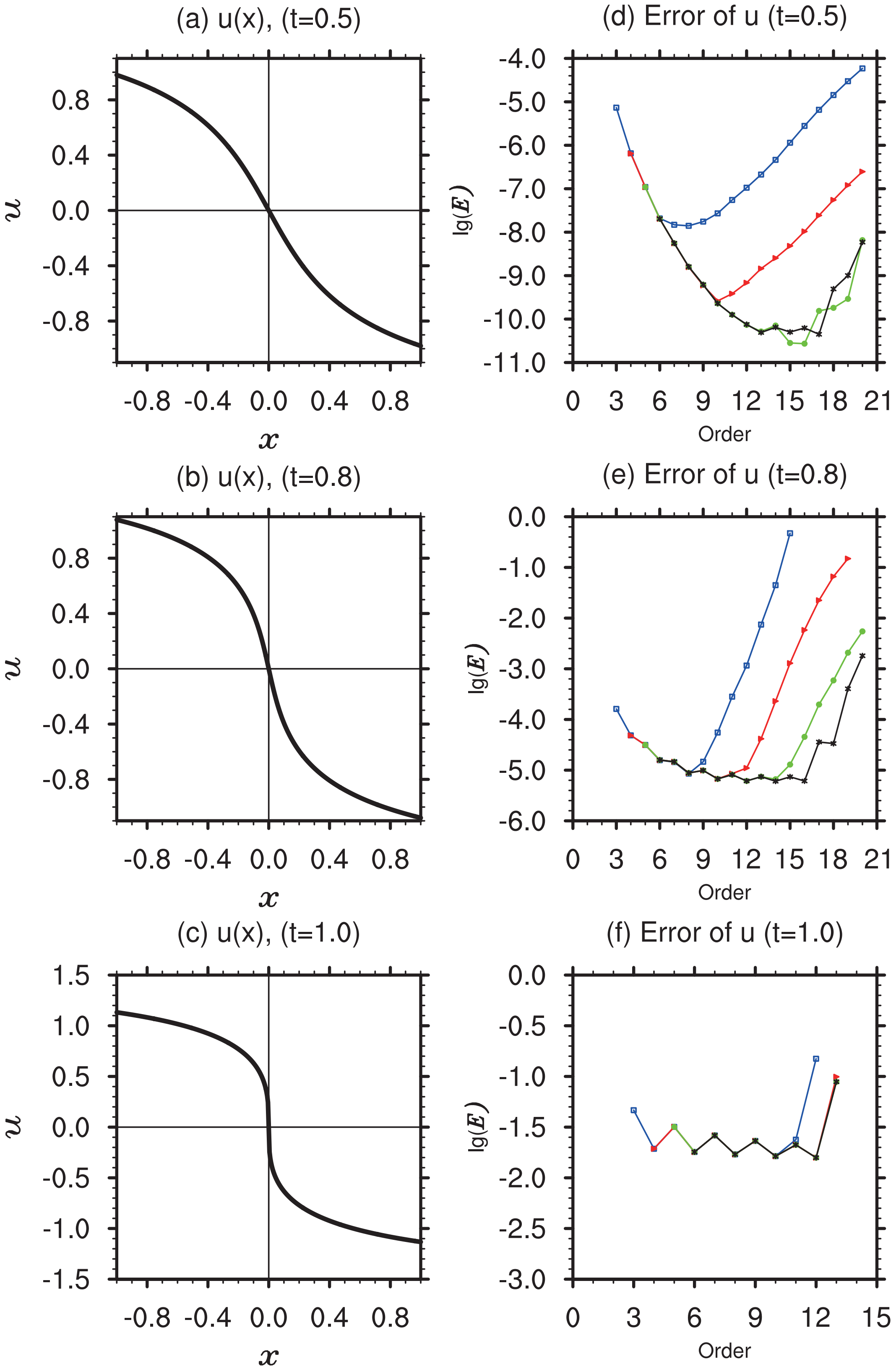

Fig. 4. (a) Analytical solution for u at $t = 0.5$. (b) Analytical solution for u at $t = 0.8$. (c) Analytical solution for u at $t = 1.0$. (d) Error versus spatial difference order at $t = 0.5$, where the abscissa is the spatial difference order and the ordinate is the logarithm of error ($\log_{10}$). The blue, red, green and black curves signify the third, fourth, fifth and sixth time-integration order, respectively. (e) As in (d) but for $t = 0.8$. (f) As in (d) but for $t = 1.0$.

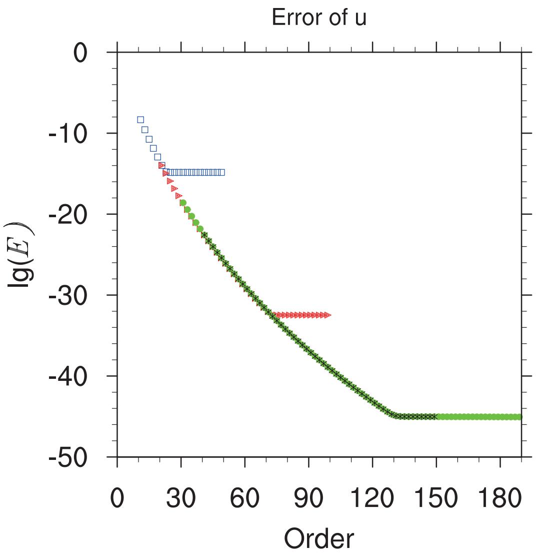

Fig. 5. Error versus spatial difference order for solving the advection equation. The abscissa is the spatial difference order and the ordinate is the logarithm of error ($\log_{10}$). The blue, red, green and black curves correspond to the 10th, 20th, 30th and 40th time-integration orders, respectively.

$M = 40$, the minimal error appears until the spatial difference order (n) approaches 130. When the grid number at the x-axis is 200, the 190th-order spatial difference is close to the upper bound of the grid points. The curves of the 30th and 40th orders overlap, which means that the 30th time-integration order is sufficient for this problem, and the $\geqslant$ 40th-order time-integration scheme can be omitted (experiments of $M = 40$ are only computed to an order of $n = 150$). Finally in this experiment, the ($M = 30$, $n = 151$) Taylor–Li scheme is able to control the computation error at a bound of about 10^{-46}, with a relatively high-precision result.

5.2. *Expt. 6—ultrahigh-order result of Expt. 3*

This experiment is the same as Expt. 3, but the time integration order is set to $M = [5, 20]$ with an interval of five, and the spatial difference order (n) ranges from M to 100 or above. The grid number in this experiment is $N = 1600$, i.e., Δx is half of that in Expt. 3.

In Fig. 6, when the time integration order is $M = 5$, the error decreases and the spatial difference order increases to 13. The error remains unchanged, suggesting that the maximal effective spatial difference order is 13. In the case $M = 10$, the range of spatial difference order causing the error to decrease is 11 to 39, where the error then becomes constant. When the time integration order increases to $M = 15$ and $M = 20$, a minimal error (of about 10^{-35}) appears until the spatial difference order (n) approaches 150. The curves of the 15th and 20th orders overlap, implying that the 15th time-integration order is enough (current parameters), and the $\geqslant$ 20th-order time integration scheme experiments could be omitted. If a more precise result is required, the grid step-size (Δx) will decrease (for example, when $\Delta x = 1/3200$, $M = 20$, and $n = 150$, the error can be reduced to 10^{-50}).

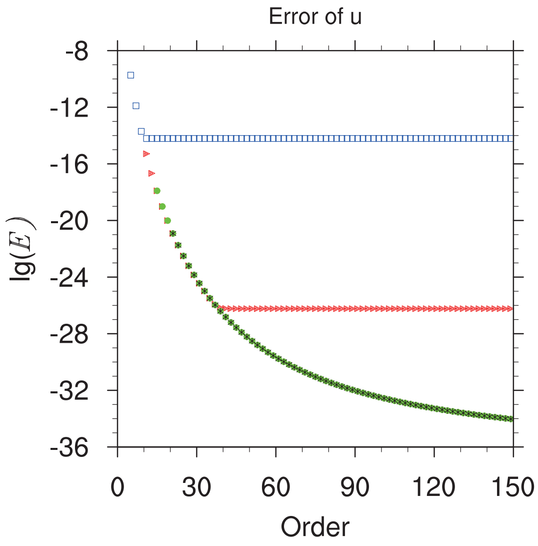

Fig. 6. Error versus spatial difference order in the computation of the in-viscid Burgers' equation. The abscissa is the spatial difference order and the ordinate is the logarithm of error ($\log_{10}$). The blue, red, green and black curves correspond to the 5th, 10th, 15th and 20th time-integration orders, respectively.

6. Conclusions

The spatial differential formula of Li (2005) and the Taylor series method are used to implement a new high-order scheme to deal with a type of time-dependent PDE. The new scheme is applied to solve the one-dimensional advection equation. By comparing numerical solutions with the theoretical solution, variations in the computation error and scheme orders are investigated. The results suggest that, when the time integration order is three for the smooth and periodical initial conditions [as used in Feng and Li (2007)], the error will be saturated after the spatial differential order increases to six. However, for a time integration order above three, such as four, five and six, at a time after the spatial differential order increases to above six, the error continues to decrease, and the corresponding effective maximal spatial differential order is beyond six. These experiments interpret the phenomenon, as reported in Feng and Li (2007), that a spatial difference order above six is unable to improve the computation result. This is because they didn't apply a time-integration scheme which order high enough. In addition, for the case with square initial wave conditions, the results of our work are the same as those of previous studies.

Furthermore, the ultrahigh-order Taylor–Li scheme produces accurate results for the advection example, and bounds the error within 10^{-46}. This is a superior result compared to other types of schemes and the standard spectral method (the general spectral method is limited by a round-off error, and thus it only controls the error at a magnitude of 10^{-15}).

A smooth and periodical test case for the in-viscid Burgers' equation results demonstrates that the Taylor–Li scheme is capable of producing good results. Compared to a previous study (Feng and Li, 2007), which applied the third-order time integration and sixth-order spatial difference scheme, the new Taylor–Li scheme easily generates the 5th-order, 10th-order, and even 20th-order time integration schemes, and the corresponding spatial difference order can be extended from the 6th to the 30th, 50th, 100th, and even the 500th order. The computation in Expt. 3 shows that a 15th-order time integration and 150th-order spatial difference Taylor–Li scheme reduces the error to a magnitude of about 10^{-35}. Once the grid size is further reduced, the scheme can reduce the error to an even smaller value (10^{-50}). Therefore, all the experiments identify the effectiveness of the new scheme.

Although we can increase the time integration precision by reducing the time step-size, this method is not as effective as that which increases the time integration order, as previously demonstrated in Wang et al. (2012). Compared to the previous scheme, which only increased the spatial difference order, the first improvement in this scheme is related to an increase in the scheme order in two directions (spatially and temporally), where the author ensures the precision in both directions within the computation. The second improvement is that the author proposes a recurrent method to estimate the time derivatives, which improves the speed of the Taylor–Li scheme, and makes it hundreds of times faster than that of the direct computation method. The third improvement involves controlling the round-off error in the procedure's computation, where the MP library is introduced and the ultrahigh-order program is implemented to solve the advection and inviscid Burgers' equation.

In addition to the above three improvements, the program of the Taylor–Li scheme is flexible in specifying the order and step-size of the time integration and spatial difference; it is therefore convenient for probing the complex relationships between orders of the scheme, the temporal and spatial step-size, and computational errors.

Acknowledgements. The author gratefully acknowledges Prof. Jianping LI for his suggestions and discussions. The work is supported by the National Natural Sciences Foundation of China (Grant Nos. 41375112 and 41530426) and the Chinese Academy of Sciences Key Technology Talent Program.

REFERENCES

Barrio, R., 2005: Performance of the Taylor series method for ODEs/DAEs. *Applied Mathematics and Computation*, **163**, 525–545, doi: 10.1016/j.amc.2004.02.015.

Barrio, R., M. Rodríguez, A. Abad, and F. Blesa, 2011: Breaking the limits: The Taylor series method. *Applied Mathematics and Computation*, **217**, 7940–7954, doi: 10.1016/j.amc.2011.02.080.

Feng, T., and J. P. Li, 2007: A comparison and analysis of high order upwind-biased schemes. *Chinese Journal of Atmospheric Sciences*, **31**, 245–253, doi: 10.3878/j.issn.1006-9895.2007.02.06. (in Chinese with English abstract)

Hénon, M., and C. Heiles, 1964: The applicability of the third integral of motion: Some numerical experiments. *Astronomical Journal*, **69**, 73–79, doi: 10.1086/109234.

Hopf, E., 1950: The partial differential equation $u_t + uu_x = \mu_{xx}$. *Commun. Pure Appl. Math.*, **3**, 201–230, doi: 10.1002/cpa.3160030302.

Ji, Z. P., J. Li, and B. Wang, 1999: Construction and test of compact scheme with square-conservation. *Chinese Journal of Atmospheric Sciences*, **23**, 323–332, doi: 10.3878/j.issn.1006-9895.1999.03.08. (in Chinese with English abstract)

Lele, S. K., 1992: Compact finite difference schemes with spectral-like resolution. *J. Computat. Phys.*, **103**, 16–42, doi: 10.1016/0021-9991(92)90324-R.

Li, J. P., 2005: General explicit difference formulas for numerical differentiation. *Journal of Computational and Applied Mathematics*, **183**, 29–52, doi: 10.1016/j.cam.2004.12.026.

Li, J. P., Q. C. Zeng, and J. F. Chou, 2000: Computational uncertainty principle in nonlinear ordinary differential equations (I)-Numerical results. *Science in China Series E*, **43**, 449–460.

Liao, S. J., 2009: On the reliability of computed chaotic solutions of non-linear differential equations. *Tellus A*, **61**, 550–564, http://dx.doi.org/10.1111/j.1600-0870.2009.00402.x.

Lorenz, E. N., 2006: Computational periodicity as observed in a simple system. *Tellus A*, **58**, 549–557, http://dx.doi.org/10.1111/j.1600-0870.2006.00201.x.

Ma, Y., and D. Fu, 1996: Super compact finite difference method (SCFDM) with arbitrary high accuracy. *Computational Fluid Dynamics*, **5**, 259–276.

Mastrandrea, M. D., and Coauthors, 2010: Guidance note for lead authors of the IPCC fifth assessment report on consistent treatment of uncertainties. Intergovermental Panel on Climate Change (IPCC). [Available online from http://www.ipcc.ch]

Moore, R. E., 1966: *Interval Analysis*. Prentice-Hall, Englewood Cliffs, NY, USA.

Moore, R. E., 1979: *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, USA, 190 pp.

Neamtan, S. M., 1946: The motion of harmonic waves in the atmosphere. *J. Meteor.*, **3**, 53–56, doi: 10.1175/1520-0469(1946)003<0053:TMOHWI>2.0.CO;2.

Song, Z., F. Qiao, X. Lei, and C. Wang, 2012: Influence of parallel computational uncertainty on simulations of the Coupled General Climate Model. *Geoscientific Model Development*, **5**, 313–319, doi: 10.5194/gmd-5-313-2012.

Sun, G. Q., 2016: Mathematical modeling of population dynamics with Allee effect. *Nonlinear Dynamics*, **85**, 1–12, doi: 10.1007/s11071-016-2671-y.

Sun, G. Q., Z. Y. Wu, Z. Wang, and Z. Jin, 2016: Influence of isolation degree of spatial patterns on persistence of populations. *Nonlinear Dynamics*, **83**, 811–819, doi: 10.1007/s11071-015-2369-6.

Takacs, L. L., 1985: A two-step scheme for the advection equation with minimized dissipation and dispersion errors. *Mon. Wea. Rev.*, **113**, 1050–1065, doi: 10.1175/1520-0493(1985)113<1050:ATSSFT>2.0.CO;2.

Tal-Ezer, H., 1986: Spectral methods in time for hyperbolic equations. *SIAM Journal on Numerical Analysis*, **23**, 11–26, doi: 10.1137/0723002.

Tal-Ezer, H., 1989: Spectral methods in time for parabolic problems. *SIAM Journal on Numerical Analysis*, **26**, 1–11, doi: 10.1137/0726001.

Teixeira, J., C. A. Reynolds, and K. Judd, 2007: Time step sensitivity of nonlinear atmospheric models: Numerical convergence, truncation error growth, and ensemble design. *J. Atmos. Sci.*, **64**, 175–189, doi: 10.1175/JAS3824.1.

von Neumann, J., and H. H. Goldstine, 1947: Numerical inverting of matrices of high order. *Bulletin of the American Mathematical Society*, **53**, 1021–1099, doi: 10.1090/S0002-9904-1947-08909-6.

Wang, P. F., 2016: Forward period analysis method of the periodic hamiltonian system. *PLoS One*, **11**, e0163303, doi: 10.1371/journal.pone.0163303.

Wang, P. F., Z. Z. Wang, and G. Huang, 2007: The influence of round-off error on the atmospheric general circulation model. *Chinese Journal of Atmospheric Sciences*, **31**, 815–825, doi: 10.3878/j.issn.1006-9895.2007.05.06. (in Chinese with English abstract)

Wang, P. F., J. P. Li, and Q. Li, 2012: Computational uncertainty and the application of a high-performance multiple precision scheme to obtaining the correct reference solution of Lorenz equations. *Numerical Algorithms*, **59**, 147–159, doi: 10.1007/s11075-011-9481-6.

Wang, P. F., Y. Liu, and J. P. Li, 2014: Clean numerical simulation for some chaotic systems using the parallel multiple-precision Taylor scheme. *Chinese Science Bulletin*, **59**, 4465–4472, doi: 10.1007/s11434-014-0412-5.