

Regular article

Sixth-order exponential Runge–Kutta methods for stiff systems[☆]Vu Thai Luan^{*}, Trky Alhsmey

Department of Mathematics and Statistics, Mississippi State University, 410 Allen Hall, Mississippi State, MS, 39762, USA

ARTICLE INFO

Keywords:

Exponential RK methods
Stiff order conditions
Sixth-order methods

ABSTRACT

This work constructs the first-ever sixth-order exponential Runge–Kutta (ExpRK) methods for the time integration of stiff parabolic PDEs. First, we leverage the exponential B-series theory to restate the stiff order conditions for ExpRK methods of arbitrary order based on an essential set of trees only. Then, we explicitly provide the 36 order conditions required for sixth-order methods and present convergence results. In addition, we are able to solve the 36 stiff order conditions in both their weak and strong forms, resulting in two families of sixth-order parallel stages ExpRK schemes. Interestingly, while these new schemes require a high number of stages, they can be implemented efficiently similar to the cost of a 6-stage method. Numerical experiments are given to confirm the accuracy and efficiency of the new schemes.

1. Introduction

In this paper, we are concerned with time integration of stiff systems of initial value problems of the form

$$u'(t) = Au(t) + g(t, u(t)) = F(t, u(t)), \quad u(t_0) = u_0. \quad (1.1)$$

These problems typically arise from spatial discretization of parabolic partial differential equations (PDEs), such as diffusion–reaction problems. The stiffness here lies in the linear part, where $\|A\|$ exhibits a large norm or even represents an unbounded operator. We assume that the nonlinearity $g(t, u)$ satisfies a local Lipschitz condition with a moderate Lipschitz constant.

Explicit exponential Runge–Kutta (ExpRK) methods have shown to be highly competitive among the viable time integration methods for integrating stiff systems (1.1), see, e.g., [1–9]. They were constructed by approximating the true solution of (1.1) represented via the Duhamel's formula, leading to the following class of s -stage schemes (see [4,5]):

$$U_{ni} = u_n + c_i h \varphi_1(c_i h A) F(t_n, u_n) + h \sum_{j=2}^{i-1} a_{ij}(h A) D_{nj}, \quad 2 \leq i \leq s, \quad (1.2a)$$

$$u_{n+1} = u_n + h \varphi_1(h A) F(t_n, u_n) + h \sum_{i=2}^s b_i(h A) D_{ni}, \quad (1.2b)$$

where $D_{ni} = g(t_n + c_i h, U_{ni}) - g(t_n, u_n)$, $U_{ni} \approx u(t_n + c_i h)$, $h = t_{n+1} - t_n > 0$ is the time step size, and c_i are nodes. The coefficients $a_{ij}(z)$ and $b_i(z)$ are usually linear combinations of the following functions and their scaled versions

$$\varphi_k(z) = \int_0^1 e^{(1-\theta)z} \frac{\theta^{k-1}}{(k-1)!} d\theta, \quad k \geq 1. \quad (1.3)$$

[☆] V.T. Luan was partially supported by National Science Foundation (NSF) awards DMS-2012022 and DMS-2309821.

^{*} Corresponding author.

E-mail address: luan@math.msstate.edu (V.T. Luan).

In [1], stiff ExprK methods with orders up to 4 were derived. Building upon this work, a new stiff order condition theory was developed in [10], enabling the derivation of a fifth-order ExprK method in [5]. None of the previously constructed ExprK methods, however, satisfies all the required stiff order conditions in the strong form. In particular, many of the order conditions have been relaxed in order to reduce complexity and minimize the number of stages when solving them. Very recently, new stiff ExprK methods of orders 4 and 5 were constructed in [9] by relaxing only one order condition as well as allow for parallel implementation of multiple internal stages, resulting in improved accuracy and efficiency. Note that ExprK methods might experience order reduction when applied to PDEs with inhomogeneous time-dependent (i.e., non-vanishing) boundary conditions. However, this issue can be avoided by employing a technique recently proposed and used in [11–14] for both linear and nonlinear problems.

Due to the substantial growth in the number of stiff order conditions for ExprK methods, constructing methods of order higher than 5 is nontrivial as this further increases the complexity in solving these order conditions (which involve matrix functions). For example, it was shown in [10, Table 5.2] that the number of stiff order conditions required for ExprK methods of order 6 is 36, compared to 9 and 16 conditions for fourth- and fifth-order methods, respectively.

In this work, we leverage the exponential B-series theory [10] to explicitly derive the set of 36 stiff order conditions required for ExprK methods of order 6. Notably, we not only achieve the solution to these 36 conditions by weakening just one condition, but also successfully satisfy all the stiff order conditions in their strong forms, i.e., without weakening any conditions. As a result, we derive the first-ever sixth-order constructed stiff ExprK methods. Additionally, while these new schemes require a high number of stages, they can be implemented efficiently similar to a 6-stage ExprK method. This is possible because they are designed to have multiple independent internal stages, allowing for simultaneous or parallel implementation.

The organization of the paper is as follows: In Section 2, we present the stiff order conditions for ExprK methods of any order based on an essential set of trees only. Additionally, we explicitly provide the 36 order conditions required for stiff ExprK methods of order 6. Section 3 presents convergence results for ExprK methods of order 6. With this, in Section 4 we derive the first two families of stiff ExprK methods of order 6, which can be implemented as the cost of a 6-stage method. Finally, a numerical example is presented in Section 5 to illustrate the theoretical results. The primary contributions of this work are the new stiff order conditions presented in Table 1 and the construction of the two families of sixth-order stiff ExprK methods, ExprK6s15 and ExprK6s16.

2. Stiff order conditions for ExprK methods of order 6

We begin by denoting and introducing the following sets of trees:

$$T_u = \{\circ\} \cup \{\tau = (\overset{\circ}{k}) : k = 2, 3, \dots\}, \text{ in which } \circ \text{ represents } u'(t), \text{ and } (\overset{\circ}{k}) \text{ represents } u^{(k)}(t) \ (k \geq 2).$$

T_g represents for all the terms involving the time derivatives of $g(u(t))$ and its elementary differentials, which can be defined recursively as the smallest set of trees, satisfying the property if $\tau_1, \dots, \tau_m \in T_u \cup T_g$, then $\tau = [\tau_1, \dots, \tau_m] \in T_g$. Here, $[\cdot, \cdot]$ is the commonly used notation for an operation that combines a finite number of trees $\tau_1, \dots, \tau_m$ by connecting them to a new shared black root node $\bullet$, thereby producing a new tree denoted as $\tau = [\tau_1, \dots, \tau_m]$. With this, one can write

$$T_g = \{\tau = [\tau_1, \dots, \tau_m] : \tau_i \in T_u \cup T_g, \ i = 1, \dots, m\}. \quad (2.1)$$

Next, we consider two disjoint subsets of T_g , denoted by T_1 and T_2 , which are given as:

$$T_1 = \{\tau = [\tau_1, \dots, \tau_m] : \tau_i = \circ, \ i = 1, \dots, m\} = \{\tau = [\underbrace{\circ, \dots, \circ}_{m \text{ times}}] : m \geq 1\} = \left\{ \begin{array}{c} \bullet \\ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \circ \circ \end{array}, \dots \right\}, \quad (2.2)$$

that is used to represent for $g^{(m)}(u(t))(\underbrace{u'(t), \dots, u'(t)}_{m \text{ times}})$, $m \geq 1$, and

$$T_2 = \{\tau = [\tau_1, \dots, \tau_m] : \tau_i \in \{\circ\} \cup T_1 \cup T_2, \ i = 1, \dots, m\} = \left\{ \begin{array}{c} \bullet \\ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \circ \circ \end{array}, \begin{array}{c} \bullet \\ \circ \circ \circ \circ \circ \end{array}, \dots \right\}. \quad (2.3)$$

For a tree $\tau \in T_u \cup T_g$, we also use the standard notions such as the order of τ , denoted by $|\tau|$, and its symmetry coefficient (see, e.g., [15]), denoted by $\sigma(\tau)$, which are defined recursively by

$$|\tau| = \begin{cases} 1, & \tau = \circ \in T_u, \\ k, & \tau = (\overset{\circ}{k}) \in T_u, \\ 1 + \sum_{i=1}^m |\tau_i|, & \tau = [\tau_1, \dots, \tau_m] \in T_g, \end{cases} \quad (2.4)$$

$$\sigma(\tau) = \begin{cases} |\tau|!, & \tau \in T_u, \\ \prod_{i=1}^k n_i! \sigma(\tau_i)^{n_i}, & \tau = [\tau_1^{n_1}, \dots, \tau_k^{n_k}] \in T_g, \end{cases} \quad (2.5)$$

where the notation $[\tau_1^{n_1}, \dots, \tau_k^{n_k}]$ stands for a tree $\tau = [\tau_1, \dots, \tau_m]$ which has k distinct branches among $\tau_1, \dots, \tau_m$, say $\tau_1, \dots, \tau_k$, corresponding to the number of occurrences $n_1, \dots, n_k$, respectively, with $n_1 + \dots + n_k = m$.

Using these set of trees and notations, below we present a simplified version of the main result of [10, Theorem 5.1] which is sufficient to derive stiff order conditions for explicit ExprK methods of arbitrary order.

First, we note that since ExprK methods are invariant under the transformation of (1.1) to its corresponding autonomous version, i.e., replacing $g(t, u)$ by $g(u)$ (see [5]), the order conditions hold the same for both formats. Therefore, for the sake of simplicity in presenting the derivation of order conditions, one can consider (1.1) with $g(t, u) \equiv g(u)$.

Lemma 2.1. Assuming that A is the infinitesimal generator of an analytical semigroup e^{tA} on a Banach space X and the nonlinearity $g: X \rightarrow X$ is sufficiently often Fréchet differentiable in a strip along the exact solution, and that $u: [t_0, t_{\text{end}}] \rightarrow X$ is sufficiently smooth with derivatives in X (all occurring derivatives are assumed to be uniformly bounded). Then, the required order conditions for stiff ExprK methods (1.2) of order p applied to (1.1) are

$$\sum_{i=2}^s b_i(hA) \frac{c_i^{|\tau|-1}}{(|\tau|-1)!} = \varphi_{|\tau|}(hA) \quad \text{for all } \tau \in T_1 \text{ with } |\tau| \leq p, \quad (2.6a)$$

$$\sum_{i=2}^s b_i(hA) g^{(m)}(u)(S_i(\tau_1)(u), \dots, S_i(\tau_m)(u)) = 0 \quad \text{for all } \tau = [\tau_1, \dots, \tau_m] \in T_2 \text{ with } |\tau| \leq p, \quad (2.6b)$$

where $S_i(\tau_k)(u)$ are the elementary differentials given recursively as

$$S_i(\tau_k)(u) = \begin{cases} c_i u'(t), & \text{if } \tau_k = \circ \in T_u \\ \psi_{\ell+1,i}(hA) g^{(\ell)}(u) \underbrace{(u'(t), \dots, u'(t))}_{\ell \text{ times}}, & \text{if } \tau_k = [\circ, \dots, \circ] \in T_1 \\ \frac{\sigma(\tau_{k1}) \dots \sigma(\tau_{k\ell})}{\sigma(\tau_k)} \sum_{j=2}^{i-1} a_{ij}(hA) g^{(\ell)}(u) \underbrace{(S_j(\tau_{k1})(u), \dots, S_j(\tau_{k\ell})(u))}_{\ell \text{ times}}, & \text{if } \tau_k = [\tau_{k1}, \dots, \tau_{k\ell}] \in T_2 \end{cases} \quad (2.7)$$

with

$$\psi_{q,i}(hA) = \sum_{k=2}^{i-1} a_{ik}(hA) \frac{c_k^{q-1}}{(q-1)!} - c_i^q \varphi_q(c_i hA). \quad (2.8)$$

Proof. Considering one-step integration (1.2) from t_n to $t_{n+1} = t_n + h$, we denote $\tilde{e}_{n+1} = \hat{u}_{n+1} - u(t_{n+1})$ as the local error of ExprK methods, where $\hat{u}_{n+1}$ represents the numerical solution obtained using the initial value $\tilde{u}_n = u(t_n)$. Using the results of [10, Section 5.2], it is clear that the set of trees $\{\circ\} \cup T_1 \cup T_2$ introduced earlier constitutes a minimal set of trees required for deriving the stiff order conditions (2.6). Specifically, by using [10, eqs. (5.1)], one can express the local error $\tilde{e}_{n+1}$ as

$$\begin{aligned} \tilde{e}_{n+1} = & \sum_{\tau=[\tau_1, \dots, \tau_m] \in T_1} h^{|\tau|} \left(\sum_{i=2}^s b_i(hA) \frac{c_i^{|\tau|-1}}{(|\tau|-1)!} - \varphi_{|\tau|}(hA) \right) g^{(m)}(\tilde{u}_n) \underbrace{(u'(t_n), \dots, u'(t_n))}_{m \text{ times}} \\ & + \sum_{\tau=[\tau_1, \dots, \tau_m] \in T_2} h^{|\tau|} \frac{\sigma(\tau_1) \dots \sigma(\tau_m)}{\sigma(\tau)} \sum_{i=2}^s b_i(hA) g^{(m)}(\tilde{u}_n) (S_i(\tau_1)(\tilde{u}_n), \dots, S_i(\tau_m)(\tilde{u}_n)) \\ & + \underbrace{\sum_{\tau \in T_g \setminus (T_1 \cup T_2)} h^{|\tau|} (\text{same weighted coefficients as trees in } T_1 \text{ or } T_2) \times (\text{corresponding elementary differentials})}_{\text{remainder terms including the trees which have the same order conditions with trees in } T_1 \cup T_2}. \end{aligned} \quad (2.9)$$

The assumption on A implies that the semigroup e^{tA} can be bounded uniformly (see, e.g., [16,17]) and thus $b_i(hA)$, $a_{ij}(hA)$, and $\psi_j(hA)$ are uniformly bounded as well. Together with the remaining assumptions of Lemma 2.1, one can then truncate (2.9) up to any stiff order $p = |\tau|$ which requires the conditions (2.6). $\square$

By using Lemma 2.1, we can explicitly provide the 36 order conditions required for stiff ExprK methods of order 6 by only drawing trees in $T_1 \cup T_2$ up to order 6 and using (2.6), see Table 1 below. There, we note that the 20 order conditions from No. 17 to 36 are new. With this result, we now present convergence results for ExprK methods of order 6.

3. Convergence results for sixth-order stiff ExprK methods

Throughout this section, we denote C as a generic constant that may have different values at different occurrences.

Theorem 3.1. Under the assumptions of Lemma 2.1, an explicit ExprK method (1.2) applied to the IVP (1.1), that fulfills all the 36 order conditions of Table 1 either strictly in the strong sense (without relaxing any conditions) or with the exception that only condition No. 17 holds in a weakened form $\sum_{i=2}^s b_i(0) \frac{c_i^5}{5!} = \varphi_6(0) = \frac{1}{6!}$, converges with order 6. In particular, its global error $e_n = u_n - u(t_n)$ satisfies the bound

$$\|e_n\| = \|u_n - u(t_n)\| \leq C h^6 \quad (3.1)$$

uniformly on time intervals $t_0 \leq t_n = t_0 + nh \leq t_{\text{end}}$ with a constant C that depends on $t_{\text{end}} - t_0$, but is independent of n and h .

Table 1

Order trees and stiff order conditions for explicit Exprk methods up to order 6. The variables Z, J, K, L denote arbitrary square matrices, and B an arbitrary bilinear mapping of appropriate dimensions. The functions $\psi_{q,j}$ are defined in (2.8).

No	Order	Tree	Stiff order conditions
1	2		$\sum_{i=2}^1 b_i(Z)c_i = \psi_2(Z)$
2	3		$\sum_{i=2}^2 b_i(Z)c_i \frac{Z^2}{2!} = \psi_3(Z)$
3	3		$\sum_{i=2}^1 b_i(Z)J\psi_{2,j}(Z) = 0$
4	4		$\sum_{i=2}^3 b_i(Z)c_i \frac{Z^3}{3!} = \psi_4(Z)$
5	4		$\sum_{i=2}^2 b_i(Z)c_i K\psi_{2,j}(Z) = 0$
6	4		$\sum_{i=2}^1 b_i(Z)J\psi_{3,j}(Z) = 0$
7	4		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{2,j}(Z) = 0$
8	5		$\sum_{i=2}^4 b_i(Z)c_i \frac{Z^4}{4!} = \psi_5(Z)$
9	5		$\sum_{i=2}^2 b_i(Z)c_i^2 L\psi_{2,j}(Z) = 0$
10	5		$\sum_{i=2}^1 b_i(Z)c_i K\psi_{3,j}(Z) = 0$
11	5		$\sum_{i=2}^1 b_i(Z)c_i K\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{2,j}(Z) = 0$
12	5		$\sum_{i=2}^1 b_i(Z)B(\psi_{2,j}(Z), \psi_{2,j}(Z)) = 0$
13	5		$\sum_{i=2}^1 b_i(Z)J\psi_{4,j}(Z) = 0$
14	5		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)c_j K\psi_{2,j}(Z) = 0$
15	5		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{3,j}(Z) = 0$
16	5		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\sum_{k=2}^{j-1} a_{jk}(Z)J\psi_{2,k}(Z) = 0$
17	6		$\sum_{i=2}^5 b_i(Z)c_i \frac{Z^5}{5!} = \psi_6(Z)$
18	6		$\sum_{i=2}^2 b_i(Z)c_i^2 K\psi_{2,j}(Z) = 0$
19	6		$\sum_{i=2}^1 b_i(Z)c_i^2 K\psi_{3,j}(Z) = 0$
20	6		$\sum_{i=2}^1 b_i(Z)c_i^2 K\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{2,j}(Z) = 0$
21	6		$\sum_{i=2}^1 b_i(Z)c_i B(\psi_{2,j}(Z), \psi_{2,j}(Z)) = 0$
22	6		$\sum_{i=2}^1 b_i(Z)c_i K\psi_{4,j}(Z) = 0$
23	6		$\sum_{i=2}^1 b_i(Z)c_i K\sum_{j=2}^{i-1} a_{ij}(Z)c_j K\psi_{2,j}(Z) = 0$
24	6		$\sum_{i=2}^1 b_i(Z)c_i K\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{3,j}(Z) = 0$
25	6		$\sum_{i=2}^1 b_i(Z)c_i K\sum_{j=2}^{i-1} a_{ij}(Z)J\sum_{k=2}^{j-1} a_{jk}(Z)J\psi_{2,k}(Z) = 0$
26	6		$\sum_{i=2}^1 b_i(Z)B(\psi_{2,j}(Z), \psi_{3,j}(Z)) = 0$
27	6		$\sum_{i=2}^1 b_i(Z)B(\psi_{2,j}(Z), \sum_{j=2}^1 a_{ij}(Z)J\psi_{2,j}(Z)) = 0$
28	6		$\sum_{i=2}^1 b_i(Z)J\psi_{5,j}(Z) = 0$
29	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)c_j^2 K\psi_{2,j}(Z) = 0$
30	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)c_j K\psi_{3,j}(Z) = 0$
31	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)c_j K\sum_{k=2}^{j-1} a_{jk}(Z)J\psi_{2,k}(Z) = 0$
32	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)B(\psi_{2,j}(Z), \psi_{2,j}(Z)) = 0$
33	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\psi_{4,j}(Z) = 0$
34	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\sum_{k=2}^{j-1} a_{jk}(Z)c_k K\psi_{2,k}(Z) = 0$
35	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\sum_{k=2}^{j-1} a_{jk}(Z)J\psi_{3,k}(Z) = 0$
36	6		$\sum_{i=2}^1 b_i(Z)J\sum_{j=2}^{i-1} a_{ij}(Z)J\sum_{k=2}^{j-1} a_{jk}(Z)J\sum_{d=2}^{k-1} a_{kd}(Z)J\psi_{2,d}(Z) = 0$

Proof. It was shown in [5, Sect. 4.3] that

$$e_n = u_n - u(t_n) = h \sum_{j=0}^{n-1} e^{(n-j)hA} \mathcal{K}_j(e_j) e_j + \sum_{j=0}^{n-1} e^{jhA} \tilde{e}_{n-j}, \quad (3.2)$$

where the operators $\mathcal{K}_j$ depend on e_j (as well as $b_i(hA), a_{ij}(hA), g^{(m)}(\cdot, \dots, \cdot)$), which can be shown to be bounded on X under the given assumptions (see [5, Lemma 4.1.]). Let us now consider the first scenario where all 36 stiff order conditions for Exprk methods of order 6 are strictly satisfied. In this case, it is clear from (2.9) that, for $|\tau| = 6$, the local error $\tilde{e}_{n+1} = \mathcal{O}(h^7)$. The assumption on A implies $\|e^{tA}\|_{X \leftarrow X} \leq C$, $t \geq 0$. Therefore, one can bound e_n in (3.2) as

$$\|e_n\| \leq h \sum_{j=0}^{n-1} C \|e_j\| + \sum_{j=0}^{n-1} C h^7. \quad (3.3)$$

An application of a discrete Gronwall lemma to (3.3) shows the error bound (3.1) at once.

Finally, we consider the second scenario wherein condition No. 17 is relaxed to be fulfilled with $A = 0$ only (i.e., $\psi_6(0) = 0$), while all other conditions are strictly satisfied in the strong sense. In this case, (2.9) can be simplified to

$$\tilde{e}_{n+1} = h^6 (\psi_6(hA) - \psi_6(0)) g^{(5)}(u(t_n)) (u'(t_n), u'(t_n), u'(t_n), u'(t_n), u'(t_n)) + \mathcal{O}(h^7). \quad (3.4)$$

Clearly, one can show that there exists a bounded operator $\tilde{\psi}(hA)$ such that $\psi_6(hA) - \psi_6(0) = \tilde{\psi}(hA)hA$. Using this, inserting (3.4) into (3.2) and employing the parabolic smoothing property of the analytical semigroup (implying the bound $\|hA \sum_{j=1}^n e^{jhA}\|_{X \leftarrow X} \leq C$), one gets $\|e_n\| \leq h \sum_{j=0}^{n-1} C \|e_j\| + C h^6 + \sum_{j=0}^{n-1} C h^7$, which again proves (3.1) as done for (3.3). $\square$

Remark 3.1. Similar to the convergence proof for Exprk methods of order 5 (see [5, Thm. 4.1]), we note that the result stated in Theorem 3.1 (the error bound (3.1)) also remains valid if all the order conditions of order 6 (from No. 17 to 36) are satisfied

in a weakened form with $b_i(0)$ substituted for $b_i(Z)$. However, we only considered weakening condition No. 17 due to two primary advantages (similar to the approach presented in [9] for fourth- and fifth-order ExpRK methods). Firstly, No. 17 depends solely on $b_i(hA)$, c_i and thus relaxing it leads to a scalar algebraic equation for c_i (which can be solved easily). Secondly, satisfying conditions from No. 18 to 36 strictly in the strong sense (which, in addition, depend also on $a_{ij}(hA)$) not only offers better stability when solving stiff problems but also enables the construction of multiple parallel internal stages U_{ni} , thereby leading to more efficient schemes (if otherwise, i.e., weakening them results in methods where each internal stage depends on preceding stages, restricting them to sequential implementation, see e.g., the construction of ExpRK methods of order 5 in [5]).

4. Derivation of sixth-order stiff ExpRK methods

Based on Theorem 3.1, we now derive two families of sixth-order ExpRK methods. While there exist many solutions, our aim is to construct efficient schemes which allow groups of *parallel stages*, i.e., internal stages that are independent of each other.

In [9], it has been shown that solving the first 16 conditions for fifth-order parallel stages ExpRK methods requires $s = 10$ stages (with relaxing one condition, namely, No. 8) and $s = 11$ stages (without relaxing any conditions). To satisfy the 20 additional conditions (from No. 17 to 36) for order 6, a value of $s \geq 11$ is thus necessary. Applying a similar approach to the order conditions outlined in [9], it can be verified (but extremely tedious) that using $s = 11, 12, 13, 14$ is not sufficient to fulfill all the order conditions of Table 1 in the context of Theorem 3.1. We omit the details. On the other hand, it is indeed possible to satisfy all the conditions in the strong sense with $s = 16$ and if condition No. 17 is relaxed, one can use $s = 15$. We will first show the latter case.

For convenience, we use the following abbreviations $b_i = b_i(hA)$, $a_{ij} = a_{ij}(hA)$, $\varphi_i = \varphi_i(hA)$, $\varphi_{j,i} = \varphi_j(c_i hA)$.

4.1. A family of sixth-order parallel-stage ExpRK schemes with $s = 15$

In this case, we assume that condition No. 17 can be relaxed to $\sum_{i=2}^{15} b_i(0) \frac{c_i^5}{5!} = \varphi_6(0) = \frac{1}{6!}$ and will solve for the remaining 35 conditions in Table 1. First, using (2.8) for $j = 2, \dots, 5$ and the fact that $\{\varphi_j\}$ are linearly independent, one can show that $\psi_{j,2} = -c_2^j \varphi_{j,2} \neq 0$, and that $\psi_{j,3}, \psi_{j,4}, \psi_{j,5}$ cannot be zero for all $j = 2, \dots, 5$. Thus, in order for later satisfy consider conditions 3, 5, 6, 9, 10, 12, 13, 18, 19, 21, 22, 26, and 28 (which involve the matrix functions $\psi_{j,i}(Z)$, $j = 2, \dots, 5$) in the strong sense (with arbitrary square matrices Z, J, K, L), it strongly suggests to choose $b_i = 0$ ($i = 2, \dots, 11$). Using this, we next solve conditions 1, 2, 4, 8, and 17 (in weakened form) and get the unique solution

$$b_i = \frac{-c_j c_k c_l \varphi_2 + 2(c_j c_k + c_j c_l + c_k c_l) \varphi_3 - 6(c_j + c_k + c_l) \varphi_4 + 24 \varphi_5}{c_i(c_l - c_j)(c_l - c_k)(c_l - c_j)}, \quad i = 12, 13, 14, 15 \quad (4.1)$$

where $j, k, l \in \{12, 13, 14, 15\}$, $j \neq k \neq l \neq i$ and nodes $c_{12}, c_{13}, c_{14}, c_{15} > 0$ are distinct and satisfy the following relation

$$\frac{1}{5} \sum_{i=12}^{15} c_i - \frac{1}{4} \sum_{\substack{i,j=12 \\ i \neq j}}^{15} c_i c_j + \frac{1}{3} \sum_{\substack{i,j,k=12 \\ i \neq j \neq k}}^{15} c_i c_j c_k - \frac{1}{2} \prod_{i=12}^{15} c_i = \frac{1}{6}. \quad (4.2)$$

Since $b_{12}, b_{13}, b_{14}, b_{15} \neq 0$, one has to enforce $\psi_{i,j} = 0$ for $i = 2, \dots, 5$; $j = 12, \dots, 15$ to strictly satisfy conditions 3, 5, 6, 9, 10, 12, 13, 18, 19, 21, 22, 26, and 28. Using this, requiring $\psi_{i,j} = 0$ for $i = 2, 3, 4$; $j = 8, \dots, 11$, and choosing $a_{ij} = 0$ for $i = 12, \dots, 15$; $j = 2, \dots, 7$, one can fulfill conditions 7, 15 and 33 in the strong form and have the stages $\{U_{nj}\}_{j=12,\dots,15}$ that are independent of $\{U_{nj}\}_{j=2,\dots,7}$. With all the requirements above, conditions 11, 14, 20, 23, 24, 27, 29, 30, 32 are automatically fulfilled.

When solving the linear system $\psi_{2,j} = 0$ ($j = 12, \dots, 15$), several a_{ij} can be taken as free parameters and we choose $a_{ij} = 0$ for $i = 13, 14, 15$; $j = 12, 13, 14$ in order to have the four parallel stages $\{U_{nj}\}_{j=12,\dots,15}$, and thus the remaining a_{ij} appearing in this system can be then uniquely solved as

$$a_{ij} = \frac{-c_i^2 c_d c_k c_l \varphi_{2,i} + 2c_i^3 (c_d c_k + c_d c_l + c_k c_l) \varphi_{3,i} - 6c_i^4 (c_d + c_k + c_l) \varphi_{4,i} + 24c_i^5 \varphi_{5,i}}{c_j(c_j - c_d)(c_j - c_k)(c_j - c_l)}, \quad i = 12, 13, 14, 15 \quad (4.3)$$

with $j, k, l, d \in \{8, 9, 10, 11\}$, $j \neq k \neq l \neq d$ and $c_8, c_9, c_{10}, c_{11} > 0$ are distinct nodes.

Next, using all the above constraints, one can show that conditions 16 and 35 can be fulfilled in their strong forms if requiring $a_{ij} = 0$ for $i = 8, \dots, 11$; $j = 2, 3, 4$ and $\psi_{2,j} = \psi_{3,j} = 0$ ($j = 5, 6, 7$). Similarly, we then choose free parameters $a_{ij} = 0$ for $i = 9, 10, 11$; $j = 8, 9, 10$ to have four parallel stages $\{U_{nj}\}_{j=8,\dots,11}$. With this, conditions 25, 31 and 34 are now automatically fulfilled. The system $\psi_{i,j} = 0$ for $i = 2, 3, 4$; $j = 8, \dots, 11$ is then solved with the unique solution

$$a_{ij} = \frac{c_i^2 c_k c_l \varphi_{2,i} - 2c_i^3 (c_k + c_l) \varphi_{3,i} + 6c_i^4 \varphi_{4,i}}{c_j(c_j - c_l)(c_j - c_k)}, \quad i = 8, 9, 10, 11; j, k, l \in \{5, 6, 7\}, j \neq k \neq l, \text{ and } c_5, c_6, c_7 \text{ are distinct.} \quad (4.4)$$

Finally, we solve the only remaining condition 36 by taking into account all the above findings. It can be satisfied in the strong sense by enforcing $a_{k2} = 0$ ($k = 5, 6, 7$) and $\psi_{2,3} = \psi_{2,4} = 0$. This linear system and the above $\psi_{2,j} = \psi_{3,j} = 0$ ($j = 5, 6, 7$) can be solved by requiring $a_{43} = a_{65} = a_{75} = a_{76} = 0$ to have two more groups of parallel stages $\{U_{nj}\}_{j=5,6,7}$ and $\{U_{nj}\}_{j=3,4}$, resulting in

$$a_{ij} = \frac{-c_i^2 c_k c_l \varphi_{2,i} + 2c_i^3 \varphi_{3,i}}{c_j(c_j - c_k)}, \text{ for } i = 5, 6, 7; j, k \in \{3, 4\}, j \neq k \text{ and } a_{ij} = \frac{c_i^2}{c_j} \varphi_{j,i} \text{ for } i = 3, 4; j = 2. \quad (4.5)$$

Putting altogether, we obtain the following family of sixth-order 15-stage stiff ExpRK methods, which will be called ExpRK6s15:

$$\begin{aligned}
 U_{n2} &= u_n + \varphi_1(c_2 hA) c_2 hF(t_n, u_n), \\
 U_{n\ell} &= u_n + \varphi_1(c_\ell hA) c_\ell hF(t_n, u_n) + h a_{\ell 2}(hA) D_{n2}, & \ell = 3, 4 \\
 U_{nm} &= u_n + \varphi_1(c_m hA) c_m hF(t_n, u_n) + h(a_{m3} D_{n3} + a_{m4} D_{n4}), & m = 5, 6, 7 \\
 U_{nq} &= u_n + \varphi_1(c_q hA) c_q hF(t_n, u_n) + h \sum_{j=2}^4 \varphi_j(c_q hA) c_q^j \sum_{i=5}^7 \rho_{ji} D_{ni}, & q = 8, \dots, 11 \\
 U_{nk} &= u_n + \varphi_1(c_k hA) c_k hF(t_n, u_n) + h \sum_{j=2}^5 \varphi_j(c_k hA) c_k^j \sum_{i=8}^{11} \mu_{ji} D_{ni}, & k = 12, \dots, 15 \\
 u_{n+1} &= u_n + \varphi_1(hA) hF(t_n, u_n) + h \sum_{j=2}^5 \varphi_j(hA) \sum_{i=12}^{15} \mu_{ji} D_{ni},
 \end{aligned}$$

$$\text{where } \rho_{ji} = \begin{cases} \frac{c_k c_l}{c_l(c_l - c_l)(c_l - c_k)}, & j = 2 \\ \frac{-2(c_k + c_l)}{c_l(c_l - c_l)(c_l - c_k)}, & j = 3 \quad (i, k, l \in \{5, 6, 7\}, i \neq k \neq l), \text{ and } \mu_{ji} = \begin{cases} \frac{-c_d c_k c_l}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 2 \\ \frac{2(c_d c_k + c_d c_l + c_k c_l)}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 3 \\ \frac{-6(c_d + c_k + c_l)}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 4 \\ \frac{24}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 5 \end{cases} \\ \frac{6}{c_l(c_l - c_l)(c_l - c_k)}, & j = 4 \end{cases} \quad \text{with } i \in \{8, 9, \dots, 15\} \text{ for }$$

$d, k, l \in \{8, 9, 10, 11\}$, $i \neq d \neq k \neq l$.

As observed, while ExpRK6s15 uses $s = 15$ stages, it can be implemented with the cost of a 6-stage method since it has 4 groups of parallel stages $\{U_{nj}\}_{j=3,4}$, $\{U_{nj}\}_{j=5,6,7}$, $\{U_{nj}\}_{j=8,\dots,11}$, and $\{U_{nj}\}_{j=12,\dots,15}$, which can be computed simultaneously or in parallel.

For our numerical experiments in Section 5, we take $c_2 = c_3 = c_5 = \frac{1}{2}$, $c_4 = c_9 = c_{13} = \frac{1}{3}$, $c_6 = c_{15} = \frac{1}{5}$, $c_7 = \frac{1}{4}$, $c_8 = \frac{18}{25}$, $c_{10} = c_{14} = \frac{3}{10}$, $c_{11} = \frac{1}{6}$, $c_{12} = \frac{90}{103}$, $c_{15} = \frac{1}{5}$, which satisfy the constraint (4.2) due to relaxing condition 17.

4.2. A family of sixth-order parallel-stage ExpRK schemes with $s = 16$

When $s = 16$, it becomes feasible to solve all 36 stiff order conditions strictly using a very similar approach to the $s = 15$ case. We note that in this case such a restriction on the nodes c_l like (4.2) is no longer required. While we omit the specific details, we present the final result for the following family of sixth-order 16-stage stiff ExpRK methods, which will be referred to as ExpRK6s16:

$$\begin{aligned}
 U_{n2} &= u_n + \varphi_1(c_2 hA) c_2 hF(t_n, u_n), \\
 U_{n\ell} &= u_n + \varphi_1(c_\ell hA) c_\ell hF(t_n, u_n) + h a_{\ell 2}(hA) D_{n2}, & \ell = 3, 4 \\
 U_{nm} &= u_n + \varphi_1(c_m hA) c_m hF(t_n, u_n) + h(a_{m3} D_{n3} + a_{m4} D_{n4}), & m = 5, 6, 7 \\
 U_{nq} &= u_n + \varphi_1(c_q hA) c_q hF(t_n, u_n) + h \sum_{j=2}^4 \varphi_j(c_q hA) c_q^j \sum_{i=5}^7 \rho_{ji} D_{ni}, & q = 8, \dots, 11 \\
 U_{nk} &= u_n + \varphi_1(c_k hA) c_k hF(t_n, u_n) + h \sum_{j=2}^5 \varphi_j(c_k hA) c_k^j \sum_{i=8}^{11} \mu_{ji} D_{ni}, & k = 12, \dots, 16 \\
 u_{n+1} &= u_n + \varphi_1(hA) hF(t_n, u_n) + h \sum_{j=2}^6 \varphi_j(hA) \sum_{i=12}^{16} \theta_{ji} D_{ni},
 \end{aligned}$$

$$\text{where } \rho_{ji} = \begin{cases} \frac{c_k c_l}{c_l(c_l - c_l)(c_l - c_k)}, & j = 2 \\ \frac{-2(c_k + c_l)}{c_l(c_l - c_l)(c_l - c_k)}, & j = 3 \quad (i, k, l \in \{5, 6, 7\}, i \neq k \neq l), \mu_{ji} = \begin{cases} \frac{-c_d c_k c_l}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 2 \\ \frac{2(c_d c_k + c_d c_l + c_k c_l)}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 3 \\ \frac{-6(c_d + c_k + c_l)}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 4 \\ \frac{24}{c_l(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 5 \end{cases} \\ \frac{6}{c_l(c_l - c_l)(c_l - c_k)}, & j = 4 \end{cases} \quad (i, d, k, l \in \{8, 9, \dots, 11\},$$

$$i \neq d \neq k \neq l), \text{ and } \theta_{ji} = \begin{cases} \frac{-c_j c_d c_k c_l}{c_l(c_l - c_j)(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 2 \\ \frac{2(c_j c_k c_d + c_j c_k c_l + c_k c_d c_l + c_d c_l c_j)}{c_l(c_l - c_j)(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 3 \\ \frac{-6(c_j c_k + c_j c_d + c_j c_l + c_k c_d + c_k c_l + c_d c_l)}{c_l(c_l - c_j)(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 4 \\ \frac{24(c_j + c_k + c_l + c_d)}{c_l(c_l - c_j)(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 5 \\ \frac{-120}{c_l(c_l - c_j)(c_l - c_d)(c_l - c_k)(c_l - c_l)}, & j = 6 \end{cases} \quad (i, j, d, k, l \in \{12, \dots, 16\}, i \neq j \neq d \neq k \neq l).$$

Similarly, ExpRK6s16 also has 4 groups of parallel stages $\{U_{nj}\}_{j=3,4}$, $\{U_{nj}\}_{j=5,6,7}$, $\{U_{nj}\}_{j=8,\dots,11}$, and $\{U_{nj}\}_{j=12,\dots,16}$. This again offers a great advantage is that it can be implemented with the cost of a 6-stage method. For our numerical experiments, we choose $c_2 = c_3 = c_5 = c_8 = c_{12} = \frac{1}{2}$, $c_4 = c_{11} = c_{15} = \frac{1}{3}$, $c_6 = c_9 = c_{13} = \frac{1}{5}$, $c_7 = c_{10} = c_{14} = \frac{1}{4}$, $c_{16} = 1$.

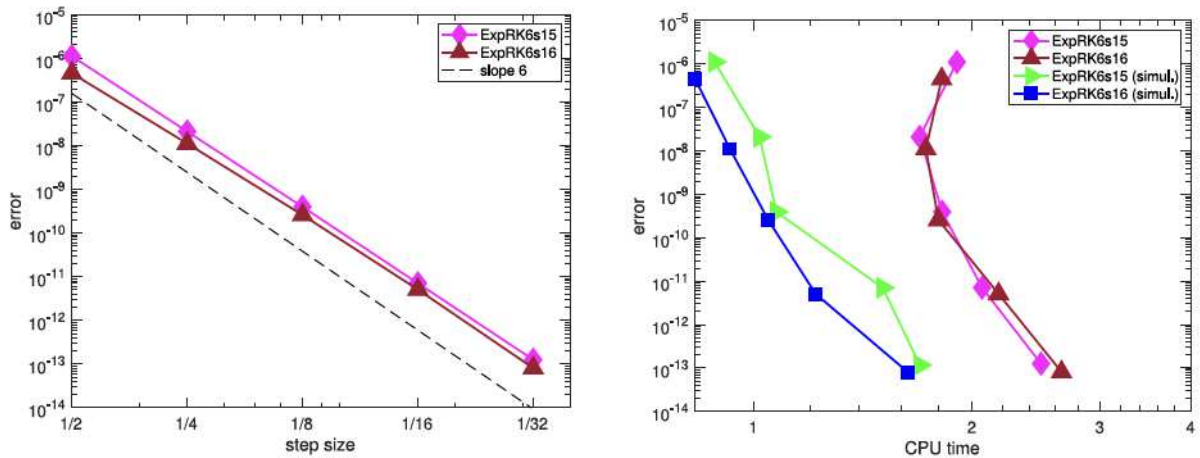


Fig. 1. Order plots (left) and total CPU times (right) of ExpRK6s15 and ExpRK6s16 when applied to Example 5.1.

5. Numerical experiments

In this section, we demonstrate the sharpness of the error bound in Theorem 3.1 when applying the two newly constructed sixth-order schemes ExpRK6s15 and ExpRK6s16. We also compare their efficiency when implemented with and without the simultaneous computation of parallel stages. To implement the new schemes, we use the code `phimp_simul_iom.m`, see [9,18].

Example 5.1. Consider the following semilinear parabolic PDE in $X = L^2([0, 1])$ (see [1]) for $u(x, t)$, $x \in [0, 1]$, $t \in [0, 1]$, subject to homogeneous Dirichlet boundary conditions,

$$\frac{\partial u(x, t)}{\partial t} - \frac{\partial^2 u(x, t)}{\partial x^2} = \frac{1}{1 + u^2(x, t)} + \Phi(x, t). \quad (5.1)$$

With a suitable choice of the source function $\Phi(x, t)$, the exact solution is $u(x, t) = x(1 - x)e^t$.

For a spatial discretization of (5.1), we use standard second order finite differences with 200 grid points, leading to a very stiff system (with $\|A\| \approx 1.6 \times 10^5$) of the form (1.1). The resulting system is then integrated on the time interval $[0, 1]$ using constant step sizes $h = \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$. The errors are measured in a discrete L^2 norm at the final time $t_{\text{end}} = 1$.

As seen from Fig. 1, the left diagram clearly shows that they fully achieve order 6 even at large time steps, thereby verifying the sharpness of our error bound in Theorem 3.1. With the same set of step sizes, ExpRK6s16 exhibits slightly higher accuracy compared to ExpRK6s15, while both schemes require similar CPU times. As anticipated, the simultaneous computation of parallel stages significantly reduces the CPU times for the same level of accuracy, as shown in the right diagram.

When considering the PDE in Example 5.1 with inhomogeneous boundary conditions, our experiments show that, depending on each specific case, ExpRK6s15 and ExpRK6s16 might or might not suffer from an order reduction. For instance, with the time-dependent boundary values $u(0) = t$, $u(1) = t + 1$ (e.g., corresponding to $u(x, t) = x(1 - x)e^t + x + t$), we do not observe any order reduction. However, with $u(0) = \cos(t)$, $u(1) = \cos(t + 1)$ (e.g., corresponding to $u(x, t) = x(1 - x)e^t + \cos(x + t)$, where the solution involves oscillating components), we observe significant order reduction in both 6th-order integrators. Interestingly, this phenomenon is not observed with lower-order (≤ 3) ExpRK schemes which strictly satisfy all the stiff order conditions. Since studying order reduction for ExpRK methods is not the primary focus of this work, we do not elaborate further details here, but instead refer to some recent results [11–14].

Data availability

No data was used for the research described in the article.

References

- [1] M. Hochbruck, A. Ostermann, Explicit exponential Runge–Kutta methods for semilinear parabolic problems, *SIAM J. Numer. Anal.* 43 (2005) 1069–1090.
- [2] G. Dujardin, Exponential Runge–Kutta methods for the Schrödinger equation, *Appl. Numer. Math.* 59 (8) (2009) 1839–1857.
- [3] G. Dimarco, L. Pareschi, Exponential Runge–Kutta methods for stiff kinetic equations, *SIAM J. Numer. Anal.* 49 (5) (2011) 2057–2077.
- [4] V.T. Luan, A. Ostermann, Stiff order conditions for exponential Runge–Kutta methods of order five, in: H. Bock, et al. (Eds.), *Modeling, Simulation and Optimization of Complex Processes - HPSC 2012*, Springer, 2014, pp. 133–143.
- [5] V.T. Luan, A. Ostermann, Explicit exponential Runge–Kutta methods of high order for parabolic problems, *J. Comput. Appl. Math.* 256 (2014) 168–179.
- [6] V.T. Luan, High-order Exponential Integrators (Ph.D. thesis), University of Innsbruck, 2014.

- [7] Q. Li, L. Pareschi, Exponential Runge–Kutta for the inhomogeneous Boltzmann equations with high order of accuracy, *J. Comput. Phys.* 259 (2014) 402–420.
- [8] V.T. Luan, R. Chinomona, D.R. Reynolds, A new class of high-order methods for multirate differential equations, *SIAM J. Sci. Comput.* 42 (2) (2020) A1245–A1268.
- [9] V.T. Luan, Efficient exponential Runge–Kutta methods of high order: Construction and implementation, *BIT Num. Math.* 61 (2) (2021) 535–560.
- [10] V.T. Luan, A. Ostermann, Exponential B-series: The stiff case, *SIAM J. Numer. Anal.* 51 (2013) 3431–3445.
- [11] I. Alonso-Mallo, B. Cano, N. Reguera, Analysis of order reduction when integrating linear initial boundary value problems with Lawson methods, *Appl. Numer. Math.* 118 (2017) 64–74.
- [12] B. Cano, M.J. Moreta, Exponential quadrature rules without order reduction for integrating linear initial boundary value problems, *SIAM J. Numer. Anal.* 56 (3) (2018) 1187–1209.
- [13] B. Cano, N. Reguera, How to avoid order reduction when Lawson methods integrate nonlinear initial boundary value problems, *BIT Numer. Math.* 62 (2) (2022) 431–463.
- [14] B. Cano, et al., Avoiding order reduction with explicit Runge–Kutta exponential methods in nonlinear initial boundary value problems, 2022, arXiv preprint arXiv:2211.11318.
- [15] E. Hairer, S. Norsett, G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, Springer, 1987.
- [16] D. Henry, *Geometric Theory of Semilinear Parabolic Equations*, in: *Lecture Notes in Mathematics*, Springer-Verlag Berlin Heidelberg.
- [17] A. Pazy, *Semigroups of Linear Operators and Applications to Partial Differential Equations*, Springer, New York, 1983.
- [18] V.T. Luan, J.A. Pudykiewicz, D.R. Reynolds, Further development of efficient and accurate time integration schemes for meteorological models, *J. Comput. Phys.* 376 (2019) 817–837.