Journal of Computational Physics 228 (2009) 3837–3850

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

High-accuracy large-step explicit Runge–Kutta (HALE-RK) schemes for computational aeroacoustics

Vasanth Allampalli [a,*], Ray Hixon [a], M. Nallasamy [b], Scott D. Sawyer [c]

[a] University of Toledo, MIME Department, 2801 W Bancroft, MS 312, Toledo, OH 43606, USA
[b] ASRC Aerospace, Kennedy Space Center, FL 32899, USA
[c] Mechanical Engineering Department, University of Akron, Akron, OH 44325, USA

ARTICLE INFO

Article history:
Received 11 September 2008
Received in revised form 6 February 2009
Accepted 18 February 2009
Available online 26 February 2009

Keywords:
Time marching schemes
Optimized Runge–Kutta method
2N-storage Runge–Kutta scheme
Computational aeroacoustics
High stability
Wave propagation
Explicit methods
Inviscid stability limit
Numerical wave number analysis
Numerical error

ABSTRACT

In many realistic calculations, the computational grid spacing required to resolve the mean flow gradients is much smaller than the grid spacing required to resolve the unsteady propagating waves of interest. Because of this, the high temporal resolution provided by existing optimized time marching schemes can be excessive due to the small time step required for stability in regions of clustered grid. In this work, explicit fourth-order accurate Runge–Kutta time marching schemes are optimized to increase the inviscid stability limit rather than the accuracy at large time steps. Single and multiple-step optimized schemes are developed and analyzed. The resulting schemes are validated on several realistic benchmark problems.

1. Introduction

The field of computational aeroacoustics (CAA) is focused on the accurate simulation of unsteady flow and noise [1,2]. To achieve this goal, highly accurate spatial differencing schemes (e.g., [1,3–5,7]), along with optimized time marching schemes (e.g., [4,8–10]) have been developed. To validate these schemes, a range of benchmark validation problems has been specified, and solutions made available [11–14]. As the CAA schemes have increased in capability, the validation problems have increased in complexity, incorporating realistic nonlinear flows about complex geometries.

As more complex problems are attempted, the underlying assumptions made in the development of the CAA schemes sometimes must be re-evaluated. For example, the optimized spatial differencing and time marching incorporated in CAA methods are designed to propagate accurately unsteady flow phenomena through non-uniform grids wrapped about complex geometries such as the stators in a turbojet engine. Experience has shown that the increased accuracy from the spatial differencing schemes is always beneficial, even near flow discontinuities. However, the use of high-accuracy explicit time marching schemes can result in an excessive number of time steps due to the stability limit on the time step size.

* Corresponding author. Tel.: +1 419 530 6066; fax: +1 419 530 8206.
E-mail address: vallampa@eng.utoledo.edu (V. Allampalli).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.02.015

Nomenclature

A	area of duct
CFL	Courant–Friedrichs–Lewy stability condition for explicit time marching schemes
k	physical wavenumber of propagating wave
$(k\Delta x)^*$	numerical wavenumber of spatial differencing scheme
k	physical wavelength of propagating wave
x	physical frequency of propagating wave
$\mathrm{x}\,\Delta t$	numerical frequency of time marching scheme

In a time marching scheme for an unsteady flow problem, there are two time steps of interest. The first is the largest time step that can be taken while retaining an accurate unsteady solution (the accuracy limit). The second time step of interest is the largest time step that can be taken while retaining a stable calculation (the stability limit).

The inviscid stability limit on an explicit time marching scheme is directly related to the minimum time required for the fastest propagating wave to move from one grid point to the next (the CFL condition). In a time marching scheme using a single time step size throughout the computational domain, such as a Runge–Kutta time marching scheme, this results in the smallest grid spacing determining the time step for the entire calculation, regardless of whether the resulting temporal resolution is necessary for solution accuracy. One method to avoid this problem is to use a multi-step Adams–Bashforth time marching scheme, allowing different grid blocks to march at different time steps while retaining time accuracy [15]. Such methods are effective, but are hard to generalize for use on arbitrary flow problems. Implicit time marching schemes can also allow large time steps [16] at the price of sub iterations to converge the solution at each time step.

A popular explicit Runge–Kutta scheme is the classical four-stage fourth-order scheme, which has a stability limit of CFL = 2.83. In the past, several researchers have optimized Runge–Kutta schemes to increase their stability and/or accuracy. For steady-state calculations, Jameson [17] developed a five-stage second-order scheme, which has a stability limit of CFL = 4.0 but low accuracy for time marching. Hu et al. [8] introduced the low-dispersion and dissipation Runge–Kutta (LDDRK) schemes, which were optimized for accuracy to the stability limit. The most popular of these schemes is the two-step fourth-order RK56 scheme, which has a stability limit of CFL = 2.85. Recently, Calvo et al. [10] have introduced a six-stage fourth-order Runge–Kutta scheme that is partially optimized for accuracy and partially optimized for a large stability limit. The resulting scheme has a stability limit of CFL = 3.82.

In this work, six/seven stage fourth-order Runge–Kutta schemes are optimized to maximize the inviscid stability limit while retaining fourth-order accuracy. The effect of optimizing a single-step scheme is compared to the results obtained by optimizing two-step methods. The resulting schemes have inviscid stability limits of up to a CFL of 5.7, and have better accuracy than a classical fourth-order Runge–Kutta scheme.

2. Linear analysis of numerical schemes

In order to analyze the performance of a time-marching numerical method, a linear model equation for inviscid flow is used:

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

A simple-harmonic solution to the equation is assumed:

$$u(x,t) = e^{i(kx - \mathrm{x}\,t)}, \tag{2}$$

where,

$$\mathrm{x} = ck. \tag{3}$$

This solution represents a simple-harmonic wave propagating at a speed of c in the positive x direction. In applying a numerical scheme to solve Eq. (1), the spatial and temporal derivatives are replaced by numerical approximations, each of which impacts the accuracy of the solution. In the case of a propagating wave, the errors can be classed as *dispersion* (a change in the propagation speed of the wave) and *dissipation* (a change in the amplitude of the wave).

The spatial derivative is considered first. The analytic result for the spatial derivative is:

$$\left.\frac{\partial u}{\partial x}\right|_{j,\mathrm{analytic}} = iku_j = \frac{i(k\Delta x)}{\Delta x}u_j. \tag{4}$$

A finite-difference spatial derivative at grid point j on a uniform grid can be written as:

$$\left.\frac{\partial u}{\partial x}\right|_{j,\mathrm{numerical}} = \frac{1}{\Delta x}\sum_{n=-M}^{N} a_n u_{j+n}. \tag{5}$$

If the spatial derivative is formulated as a central difference, the dissipation errors can be eliminated. The resulting spatial derivative can be written as:

$$\left.\frac{\partial u}{\partial x}\right|_{j,\text{numerical}} = \frac{1}{\Delta x}\sum_{n=1}^{N} a_n(u_{j+n} - u_{j-n}). \tag{6}$$

Substituting the assumed solution into the central difference approximation, the numerical spatial derivative is obtained:

$$\left.\frac{\partial u}{\partial x}\right|_{j,\text{numerical}} = \frac{i}{\Delta x}\left[\sum_{n=1}^{N} a_n \sin(nk\Delta x)\right]u_j = \frac{i(k\Delta x)}{\Delta x}\left[\frac{\sum_{n=1}^{N} a_n \sin(nk\Delta x)}{(k\Delta x)}\right]u_j. \tag{7}$$

Defining the *numerical wavenumber* as:

$$(k\Delta x)^* = \sum_{n=1}^{N} a_n \sin(nk\Delta x). \tag{8}$$

Eq. (7) becomes:

$$\left.\frac{\partial u}{\partial x}\right|_{j,\text{numerical}} = \frac{i(k\Delta x)}{\Delta x}\left(\frac{(k\Delta x)^*}{(k\Delta x)}\right)u_j = \left(\frac{(k\Delta x)^*}{(k\Delta x)}\right)\left.\frac{\partial u}{\partial x}\right|_{j,\text{analytic}}. \tag{9}$$

Due to the Nyquist limit, the highest-wavenumber wave that can be resolved on a uniform grid has two grid points per wavelength. This limits the maximum value of the physical wavenumber that may be resolved in a numerical calculation to:

$$k_{\max} = \frac{2\mathrm{p}}{\mathrm{k}_{\min}} = \frac{2\mathrm{p}}{2\Delta x} = \frac{\mathrm{p}}{\Delta x}. \tag{10}$$

The differences in accuracy of the various schemes can be seen; in particular, it can be seen that each scheme has a different maximum value for the numerical wavenumber. Table 1 lists the maximum numerical wavenumbers from each scheme.

Using the development thus far, the numerical equivalent of Eq. (1) can be written for this assumed solution as:

$$\frac{\partial u}{\partial t} = F(u) = -\frac{i}{\Delta x}[c(k\Delta x)^*]u. \tag{11}$$

Using Eq. (3) to define a *numerical frequency*:

$$\mathrm{x}^* = \frac{c(k\Delta x)^*}{\Delta x}. \tag{12}$$

Eq. (11) can be rewritten as:

$$\frac{\partial u}{\partial t} = F(u) = -i\,\mathrm{x}^* u. \tag{13}$$

To integrate Eq. (13) in time, a Runge–Kutta scheme can be used. A classical four-stage fourth-order Runge–Kutta scheme can be written as:

$$\begin{aligned}
u_0 &= u^n,\\
k_1 &= \Delta t F(u_0, t),\\
k_2 &= \Delta t F\left(u_0 + \frac{1}{2}k_1, t + \frac{\Delta t}{2}\right),\\
k_3 &= \Delta t F\left(u_0 + \frac{1}{2}k_2, t + \frac{\Delta t}{2}\right),\\
k_4 &= \Delta t F(u_0 + k_3, t + \Delta t),\\
u^{n+1} &= u^n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).
\end{aligned} \tag{14}$$

Table 1
Maximum value of numerical wavenumber for spatial differencing methods.

Differencing scheme	Maximum $(k\Delta x)^*$
Explicit second-order	1.0
Explicit DRP	1.664
Explicit sixth-order	1.586
Compact sixth-order	1.989

Substituting in the function from Eq. (13), the fourth-order Runge–Kutta scheme gives:

$$u^{n+1} = \left(1 + (i\mathrm{x}^*\Delta t) + \frac{1}{2}(i\mathrm{x}^*\Delta t)^2 + \frac{1}{6}(i\mathrm{x}^*\Delta t)^3 + \frac{1}{24}(i\mathrm{x}^*\Delta t)^4\right)u^n. \tag{15}$$

The relative error magnitude of the time marching can be evaluated as:

$$\text{Error} = \left|\left(\frac{u^{n+1}}{u^n}\right) - e^{i(\mathrm{x}^*\Delta t)}\right|. \tag{16}$$

For stability, the amplitude of the propagating wave should not increase:

$$\left|\frac{u^{n+1}}{u^n}\right| \leqslant 1. \tag{17}$$

For accuracy, the amplitude of the propagating wave should remain constant. Figs. 1 and 2 show the error magnitude and amplitude results for a Runge–Kutta fourth-order scheme. It is important to note that the time marching accuracy and stability depend only on the product of the time step and the magnitude of the numerical frequency of the scheme. Since the numerical frequency is the product of the physical wavespeed and the numerical wavenumber of the spatial differencing method, the maximum value of the numerical frequency depends on the maximum value of the numerical wavenumber of the spatial differencing scheme. Thus, spatial differencing schemes which have higher maximum numerical wavenumbers result in higher maximum values of the numerical frequency. Because of this, the maximum allowable time step for stability is lower for spatial differencing schemes with higher maximum numerical wavenumbers. For example, from Table 1 it is expected that the use of a second-order central differencing scheme instead of a DRP scheme will result in a time step that is approximately 1.644 times larger due to the lower value of the maximum numerical wave number.

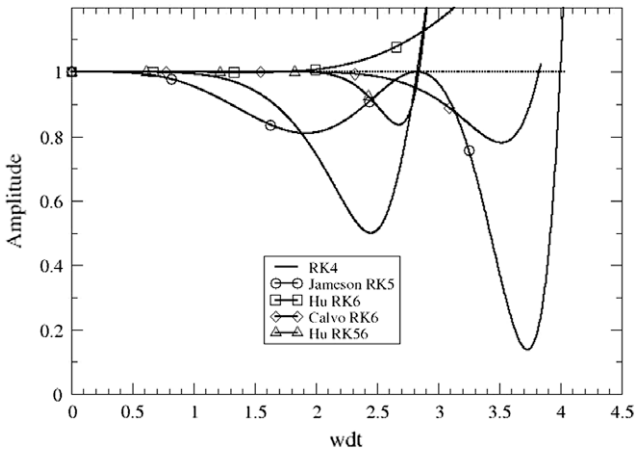

Fig. 1. Amplitude performance of optimized RK schemes.

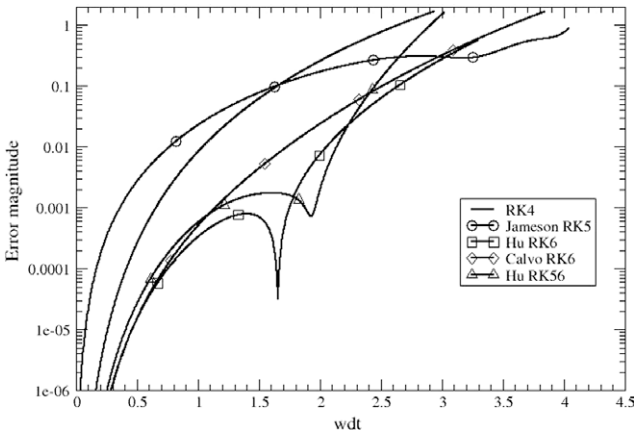

Fig. 2. Error magnitude of optimized RK schemes.

Table 2
Comparison of optimized Runge–Kutta schemes.

Scheme	c_3	c_4	c_5	c_6	CFL for accuracy limit (2e-3) (CFL per stage)	CFL for stability limit (CFL per stage)
RK4	1/6	1/24	0	0	0.75 (0.1875)	2.83 (0.7075)
Jameson RK5	3/16	1/32	1/128	0	0.45 (0.09)	4.0 (0.798)
Hu RK6	1/6	1/24	0.00781005	0.00132141	1.81 (0.3017)	1.65 (0.275)
Calvo RK6	1/6	1/24	0.00785333	0.00094889	1.30 (0.2167)	3.82 (0.6366)
Hu RK56 (1)	1/6	1/24	0.00361050	0	2.00	2.84
Hu RK56 (2)	1/6	1/24	0.0121101	0.00285919	(0.3636)	(0.5163)

3. Development of optimized Runge–Kutta schemes

In the past, researchers have added additional stages to the Runge–Kutta method in order to gain additional coefficients. For example, the linear performance of a seven-stage Runge–Kutta scheme with a minimum of second-order accuracy can be expressed as:

$$u^{n+1} = \left(1 + c_1(i\mathrm{x}^*\Delta t) + c_2(i\mathrm{x}^*\Delta t)^2 + c_3(i\mathrm{x}^*\Delta t)^3 + c_4(i\mathrm{x}^*\Delta t)^4 + c_5(i\mathrm{x}^*\Delta t)^5 + c_6(i\mathrm{x}^*\Delta t)^6 + c_7(i\mathrm{x}^*\Delta t)^7\right)u^n. \tag{18}$$

To obtain linear fourth-order accuracy, the coefficients c_1, c_2, c_3 and c_4 can be set to 1, 1/2, 1/6 and 1/24, respectively. If fourth-order accuracy is all that is desired, the values of the remaining coefficients are arbitrary and can be determined using an optimization procedure. The goal of the optimization procedure is usually to obtain a high stability limit with little regard for accuracy (e.g., the Jameson RK5), to obtain a more accurate time marching scheme with little regard for the stability limit (e.g., the Hu RK6), or to obtain a more accurate time marching scheme with a larger stability limit (e.g., the Calvo RK6). Table 2 illustrates the effect of optimization on the stability and accuracy of Runge–Kutta schemes. The computational efficiency of each scheme is also given in Table 2; to determine the efficiency, the total accuracy or stability limit of each scheme is divided by the number of stages required by the scheme.

In his work, Hu realized that optimizing Runge–Kutta schemes over two steps rather than one step resulted in much more accurate and efficient schemes [8]. An example of this is the Hu RK56 scheme, which uses a two-step time marching cycle in which a five-stage Runge–Kutta time step is followed by a six-stage time step. Each time step has its own set of coefficients, as shown in Table 2. The resulting scheme increases both accuracy and stability compared to the single-step Hu RK6 scheme, and is more efficient due to the larger time step and lowered computational effort.

The resulting coefficients of these optimized schemes are given in Table 2, and their performance is compared to the classical Runge–Kutta scheme in Figs. 1 and 2. Figs. 1 and 2 show the effectiveness of each optimization strategy in achieving large stability limits or accuracy at larger time steps.

4. Development of HALE-RK schemes

An inherent assumption in the optimization of Runge–Kutta schemes for accuracy is that the unsteady flow dynamics at the length and time scales associated with the smallest grid spacing is important to the accuracy of the overall solution. As CAA codes have matured and have been applied to more realistic flow problems containing a range of length and time scales in the flow, this assumption is not necessarily true.

For example, the steady mean flow about an object can have very steep gradients (in boundary layers, or about sharp leading edges, for example). To accurately resolve the mean flow, computational grid points must be clustered in these regions. The length scales associated with unsteady flow phenomena in the problem may be much larger than the length scales of these mean flow gradients, and thus allow a much coarser grid to be used away from the object.

Due to the inviscid stability limit, the allowable time step in the calculation is associated with the smallest grid spacing and the mean flow velocity and speed of sound. Thus, the clustered grid necessary to resolve steady flow gradients can result in an excessive number of time steps to calculate the unsteady flow.

In this work, an optimization procedure is used to obtain six/seven stage, fourth-order Runge–Kutta schemes. The goal of the optimization is to use the free coefficients c_5, c_6, c_7 to achieve maximum inviscid stability. Once the values of the free coefficients were obtained these schemes are formulated into a 2N-storage format [6]. These schemes in 2N-storage format satisfy the necessary conditions to have nonlinear fourth-order accuracy. This formulation is explained in the next section. The new high-accuracy large-step explicit Runge–Kutta (HALE-RK) schemes obtain an inviscid stability limit of CFL = 4.9–5.7 while obtaining higher accuracy than the classical Runge–Kutta fourth-order scheme.

The coefficients for the HALE-RK schemes are given in Table 3. The optimized coefficients are obtained numerically, using a gradient method starting from zero values for all optimization coefficients. Following Hu's method, the schemes were optimized as multi-step schemes. It was found that optimizing beyond two-step schemes did not improve the results appreciably.

Figs. 3 and 4 show the performance of the resulting schemes compared to the earlier schemes. The new HALE-RK schemes show very large inviscid stability limits compared to the other schemes, and still show a reasonable accuracy range.

Table 3
Comparison of HALE-RK schemes.

Scheme	c_3	c_4	c_5	c_6	c_7	CFL for accuracy limit (2e-3) (CFL per stage)	CFL for stability limit (CFL per stage)
RK4	1/6	1/24	0	0	0	0.75 (0.1875)	2.83 (0.7075)
Jameson RK5	3/16	1/32	1/128	0	0	0.45 (0.09)	3.99 (0.798)
Calvo RK6	1/6	1/24	0.00785333	0.00094889	0	1.30 (0.2167)	3.82 (0.6366)
Hu RK56	1/6	1/24	0.00361050	0	0	2.00 (0.3636)	2.84 (0.5163)
HALE-RK6	1/6	1/24	0.00556351	0.00092671	0	0.945 (0.1575)	4.90 (0.8167)
HALE-RK7	1/6	1/24	0.00770233	0.00087053	0.000107	1.205 (0.1721)	5.67 (0.81)
HALE-RK67	1/6	1/24	0.00509534	0.00092304	0	1.04 (0.16)	5.53 (0.8508)
	1/6	1/24	0.00822319	0.00087153	0.000126		

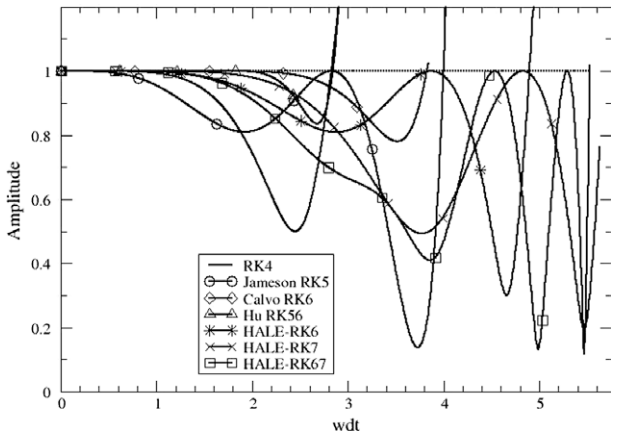

Fig. 3. Amplitude performance of HALE-RK schemes.

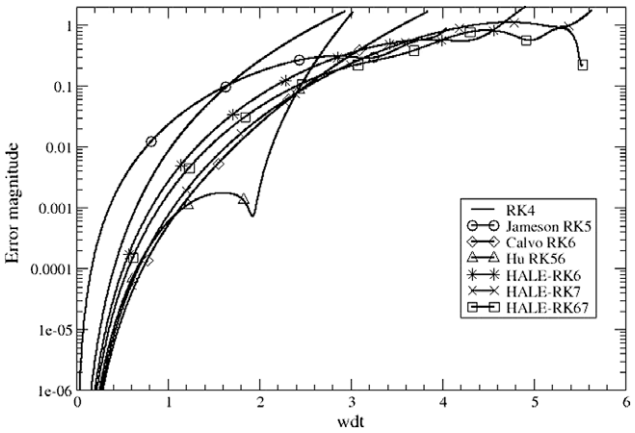

Fig. 4. Error magnitude of HALE-RK schemes.

5. Fourth-order accurate, 2N-storage HALE-RK schemes

For an equation of the form:

$$\frac{\partial u}{\partial t} = F(t, u(t)), \tag{19}$$

with an initial condition:

$$u(t_0) = u_0 \tag{20}$$

the implementation of the pth order of accuracy s-stage RK method, over a time step Δt is given as

$$\begin{aligned} u^{n+1} &= u^n + \Delta t \sum_{i=1}^{s} b_i k_i, \\ k_i &= F\left(t^n + hc_i, u^n + \Delta t \sum_{j=1}^{i-1} a_{ij} k_j \right) \quad i = 1, \ldots, s, \end{aligned} \tag{21}$$

where, $c_i = \sum_{j=1}^{i-1} a_{ij}$

For the scheme in Eq. (21) to have fourth-order accuracy, the coefficients must obey certain conditions [22].These conditions are obtained by equating the coefficients of the Taylor series development of u. The explicit form of these conditions is given below

$$\begin{aligned} &(O1)\ \sum b_i = 1, \quad (O2)\ \sum b_i c_i = \frac{1}{2}, \quad (O3) \sum b_i c_i^2 = \frac{1}{3}, \\ &(O3)\ \sum b_i a_{ij} c_j = \frac{1}{6}, \quad (O4)\ \sum b_i c_i^3 = \frac{1}{4}, \quad (O4)\ \sum b_i c_i a_{ij} c_j = \frac{1}{8}, \\ &(O4)\ \sum b_i a_{ij} c_j^2 = \frac{1}{12}, \quad (O4)\ \sum b_i a_{ij} a_{jk} c_k = \frac{1}{24}. \end{aligned} \tag{22}$$

The sums on all indices extend from 1 to s and each condition is preceded by the order of accuracy it governs. Using the same Taylor series development, the explicit form [22] of the coefficients c_i in Eq. (18) for methods up to seven stages is,

$$\begin{aligned} &c_1 = \sum b_i, \quad c_2 = \sum b_i c_i, \quad c_3 = \sum b_i a_{ij} c_j, \\ &c_4 = \sum b_i a_{ij} a_{jk} c_k, \quad c_5 = \sum b_i a_{ij} a_{jk} a_{kl} c_l, \\ &c_6 = \sum b_i a_{ij} a_{jk} a_{kl} a_{lm} c_m, \quad c_7 = \sum b_i a_{ij} a_{jk} a_{kl} a_{lm} a_{mn} c_n. \end{aligned} \tag{23}$$

It can be noted that some of the sums in Eq. (23) also appear in Eq. (22).The scheme can be formulated to a 2N-storage format to reduce the amount of storage [6]. The implementation of the 2N-storage RK scheme over a time step Δt can be written as

$$\left.\begin{aligned} &u_1 = u(t_0), \\ &t_1 = t_0, \\ &w_{i+1} = a_i w_i + \Delta t F(t_i, u_i), \\ &u_{i+1} = u_i + b_i w_{i+1}, \\ &t_{i+1} = t_0 + c_i \Delta t, \\ &u(t_0 + \Delta t) = u_{s+1} \end{aligned}\right\} \quad i = 1, s. \tag{24}$$

In Eq. (24) s is the number of stages. The relationship between the original RK variables and the 2N-storage variables up to seven stages are given in the Appendix 1. HALE-RK schemes were formulated in 2N-storage format and retain fourth-order accuracy.

5.1. HALE-RK6

To obtain the fourth-order 6 stage HALE-RK scheme in 2N-storage format, the eight order conditions in Eq. (22) and the two additional conditions obtained by specifying c_5 and c_6 in Eq. (23) are expressed in terms of the coefficients $a_2, \ldots, a_6$ and $b_1, \ldots, b_6$.This leads to a system of ten nonlinear equations with eleven unknowns. A solution is chosen such that the scheme has monotonically increasing values of c_i.

5.2. HALE-RK7

For the seven stage scheme, the eight order conditions in Eq. (22) and the three additional conditions obtained by specifying $c_5 c_6$ and c_7 in Eq. (23) are expressed in terms of the coefficients $a_2, \ldots, a_7$ and $b_1, \ldots, b_7$. This leads to a system of eleven nonlinear equations with 13 unknowns. A solution is again chosen such that the scheme has monotonically increasing values of c_i.

5.3. HALE-RK67

This is a two-step scheme. The scheme has fourth-order accuracy in both steps with six and seven stages in the first and second steps. At the end of both steps, the solution marches by a time step of $2\Delta t$. The process of obtaining 2N-storage format for steps 1 and 2 is similar to that used for HALE-RK6 and HALE-RK7 respectively. Tables 4–7 give the 2N-storage coefficients of the three HALE-RK schemes discussed.

Table 4
2N-storage coefficients for HALE-RK6 scheme.

Stage	a_i	b_i	c_i
1	0.000000000000	0.122000000000	0.000000000000
2	−0.691750960670	0.477263056358	0.122000000000
3	−1.727127405211	0.381941220320	0.269115878630
4	−0.694890150986	0.447757195744	0.447717183551
5	−1.039942756197	0.498614246822	0.749979795490
6	−1.531977447611	0.186648570846	0.898555413085

Table 5
2N-storage coefficients for HALE-RK7 scheme.

Stage	a_i	b_i	c_i
1	0.000000000000	0.117322146869	0.000000000000
2	−0.647900745934	0.503270262127	0.117322146869
3	−2.704760863204	0.233663281658	0.294523230758
4	−0.460080550118	0.283419634625	0.305658622131
5	−0.500581787785	0.540367414023	0.582864148403
6	−1.906532255913	0.371499414620	0.858664273599
7	−1.450000000000	0.136670099385	0.868664273599

Table 6
2N-storage coefficients for HALE-RK67 scheme, step 1.

Stage	a_i	b_i	c_i
1	0.000000000000	0.122000000000	0.000000000000
2	−0.729209403105	0.469478137589	0.122000000000
3	−1.864880899763	0.324481527871	0.249130265107
4	−0.596273618447	0.425379834670	0.409751148470
5	−0.947418171615	0.541445529815	0.709575835016
6	−1.466974856227	0.215637416704	0.889455874056

Table 7
2N-storage coefficients for HALE-RK67 scheme, step 2.

Stage	a_i	b_i	c_i
1	0.000000000000	0.126989033120	0.000000000000
2	−0.680465510779	0.526334177737	0.126989033120
3	−2.913563164146	0.218277359812	0.295170955763
4	−0.413737223754	0.285075800760	0.310235604043
5	−0.497690668921	0.542609381079	0.587171201470
6	−1.891291697445	0.432735912796	0.867440151911
7	−2.091000000000	0.128782687265	0.877440151911

A system of nonlinear and non-autonomous, first order differential equations used in [9] is used to test the accuracy of the HALE-RK schemes. It is given as

$$U' = \frac{1}{U} - \frac{Ve^{t^2}}{t^2} - t; \quad V' = \frac{1}{V} - e^{t^2} - 2te^{-t^2}. \tag{25}$$

The initial conditions are:

$$U(1) = 1, \quad V(1) = e^{-1}. \tag{26}$$

The exact solution for the system is:

$$U(t) = \frac{1}{t}, \quad V(t) = e^{-t^2}. \tag{27}$$

Fig. 5 shows results for a convergence study for the three HALE-RK schemes. They are compared with Hu's RK56 and classical RK4 schemes. The global error norms were calculated as $|U-u| + |V-v|$ at $t = 1.4$. The convergence rate for the schemes approach −4. This is consistent with the theoretical predictions.

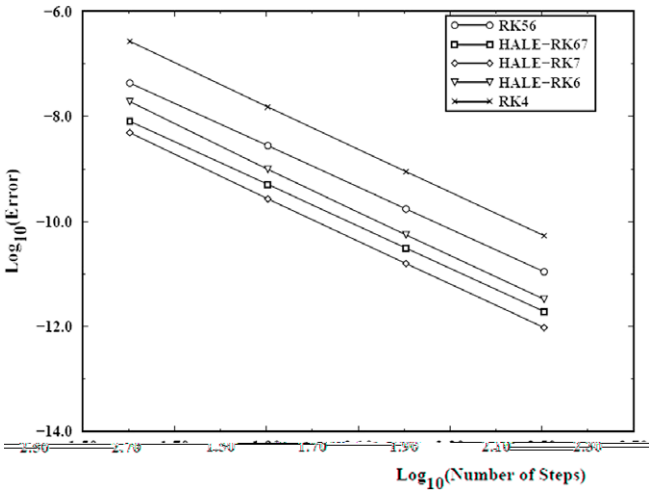

Fig. 5. Convergence study of HALE-RK schemes.

6. Application of HALE-RK67 method to benchmark problems

6.1. Acoustic wave interacting with shock

As a test for the HALE-RK schemes, the schemes were applied to the quasi-1D nonlinear Euler equation benchmark test case from the Third Computational Aeroacoustics Workshop [13]. In this benchmark problem, the quasi-1D Euler equations are solved:

$$\frac{\partial}{\partial t}\begin{Bmatrix} \mathrm{q} \\ \mathrm{q}u \\ E \end{Bmatrix} + \frac{\partial}{\partial x}\begin{Bmatrix} \mathrm{q}u \\ \mathrm{q}u^2 + p \\ u(E+p) \end{Bmatrix} + \frac{1}{A}\frac{\partial A}{\partial x}\begin{Bmatrix} \mathrm{q}u \\ \mathrm{q}u^2 \\ u(E+p) \end{Bmatrix} = 0, \tag{28}$$

where,

$$\begin{aligned} E &= \frac{p}{\mathrm{c}-1} + \frac{\mathrm{q}}{2}(u^2), \\ \mathrm{c} &= 1.4. \end{aligned} \tag{29}$$

The area of the duct is given as:

$$A(x) = \begin{cases} 1.0 - 0.661514e^{-\ln 2\left(\frac{x}{0.6}\right)^2} & x \leqslant 0, \\ 0.536572 - 0.198086e^{-\ln 2\left(\frac{x}{0.6}\right)^2} & x > 0. \end{cases} \tag{30}$$

The computational domain ranges from $-10 < x < 10$. At the inflow boundary ($x = -10$), the mean quantities are:

$$\begin{aligned} \bar{p} &= \frac{1}{\mathrm{c}}, \\ \bar{\mathrm{q}} &= 1, \\ \bar{M} &= 0.2006533, \end{aligned} \tag{31}$$

and at the outflow boundary ($x = 10$):

$$\bar{p} = 0.6071752. \tag{32}$$

At the inflow boundary, a low-amplitude incoming acoustic wave is imposed:

$$\begin{aligned} \begin{Bmatrix} \mathrm{q}' \\ u' \\ p' \end{Bmatrix}_{x=-10} &= \begin{Bmatrix} \mathrm{e} \\ \mathrm{e} \\ \mathrm{e} \end{Bmatrix} \sin\left[\mathrm{x}\left(\frac{x}{1+M} - t\right)\right], \\ \mathrm{e} &= 10^{-5}, \\ \mathrm{x} &= 0.6\mathrm{p}. \end{aligned} \tag{33}$$

This test case illustrates the motivation for the development of the HALE-RK schemes. In this problem, the acoustic wave has a wavelength that is on the order of the length of the nozzle constriction. In the nozzle constriction, the mean flow changes sharply from subsonic to supersonic and back to subsonic via a shock. Upstream and downstream of the nozzle, the mean flow is uniform. Thus, the grid spacing is driven by the resolution requirements of the acoustic wave in the region upstream and downstream of the nozzle (which requires a relatively coarse grid), and by the resolution requirements of the mean flow gradients in the nozzle constriction (which requires a relatively fine grid).

The time step is determined by the fine grid spacing in the nozzle (because of stability concerns), which results in a very fine temporal resolution that is not required by the unsteady low-frequency disturbances that are actually present in the calculation.

For this test case, the HALE-RK schemes were used for time marching. For spatial differencing, Tam and Webb's Dispersion Relation Preserving (DRP) optimized 7-point fourth-order scheme was used. To provide damping for this inviscid nonlinear calculation, an explicit constant-coefficient 10th-order artificial dissipation using Kennedy and Carpenter's boundary stencils [18] was added at each stage.

The calculation was run on a 3201 point evenly spaced grid to ensure that the maximum CFL conditions for the HALE-RK schemes were reached; it was found that the use of a stretched grid of 185 grid points actually allowed a higher CFL condition to be used. The grid spacing was determined by the sharp mean flow gradients in the throat region of the nozzle; the acoustic wave was massively over-resolved with over 600 points per wavelength.

The flow solution was initialized at zero flow and run 1500 cycles to convergence. For HALE-RK67 and HALE-RK7 schemes, each cycle of oscillation had 350 time steps; this resulted in a final CFL of 3.21. For the HALE-RK6 scheme, each cycle of oscillation had 390 time steps; this resulted in a final CFL of 2.8. For Figs. 6 and 7 compare the results using the HALE-RK schemes to the benchmark solutions for the mean pressure and the instantaneous disturbance pressure distributions at the start of the cycle. Both compare very well. In particular, note the disparity between the acoustic wavelengths shown in Fig. 7 with the sharp mean flow gradients in the nozzle shown in Fig. 6. The results in Fig. 7 show that the HALE-RK schemes are accurately resolving this low-frequency disturbance.

6.2. Vortical gust impinging on airfoil

At this point, the HALE-RK schemes were implemented in the NASA Glenn Research Center Broadband Aeroacoustic Stator Simulation (BASS) CAA code [19], and tested on a realistic 2D benchmark gust-airfoil interaction problem [20]. The results are compared to previously validated results from this code using Stanescu and Habashi's [9] low-storage extension of Hu's LDDRK56 method [8].

The BASS code solves the nonlinear 3D Navier–Stokes equations, and can be switched down to solve the 2D nonlinear Euler equations. For general applicability to complex flow geometries, the curvilinear coordinate form of the governing equations are solved on a structured multiblock grid. For portability and efficiency, BASS is coded in Fortran90 and parallelized using the Message Passing Interface (MPI).

In this test case, the nonlinear Euler equations are solved for the unsteady flow about a cambered Joukowski airfoil. The free stream Mach number for this case is 0.5. A two-dimensional vortical gust with a reduced frequency of 3.0 is introduced at the inflow boundary and convects with the flow. When the gust impinges on the airfoil, noise is radiated. The BASS code has been extensively tested for this problem; details of these calculations are given in Ref. [21].

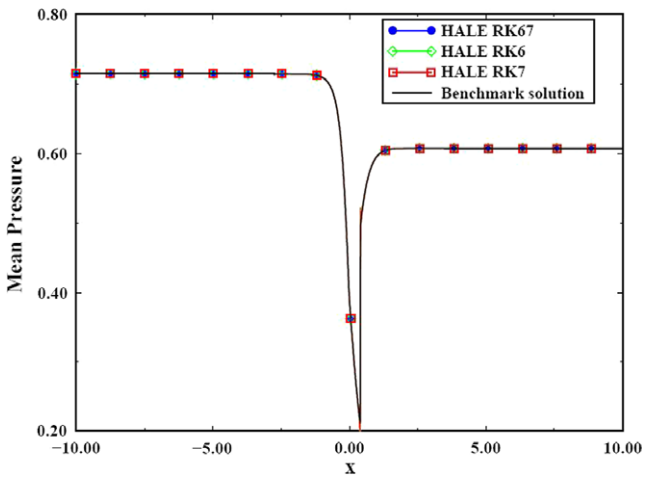

Fig. 6. Comparison of mean pressure.

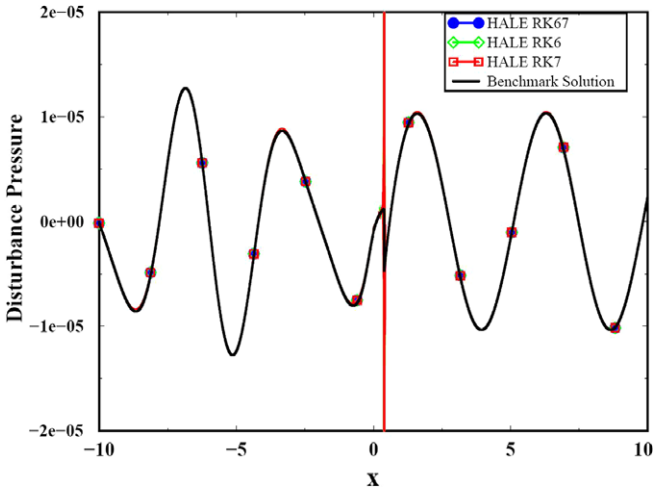

Fig. 7. Comparison of disturbance pressure at start of cycle.

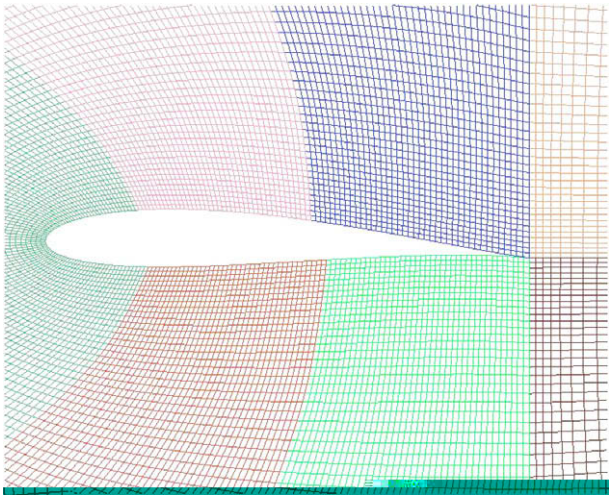

Fig. 8. Joukowski airfoil grid.

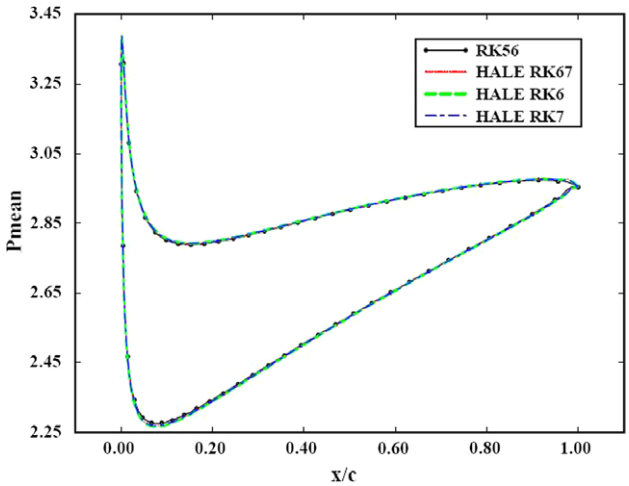

Fig. 9. Comparison of Pmean on airfoil.

Fig. 8 shows a close-up of the body-fitted grid used about the Joukowski airfoil. The grid is clustered near the leading edge of the airfoil in order to resolve the sharp flow gradients in this region. In this calculation, the physical wavelength of the unsteady gust is approximately one chord length; thus, the time step limit is based on the stability restriction from the clustered grid rather than on the accuracy restriction for resolving the unsteady gust dynamics. To quantify this further, the RK56 time stepping method takes 960 time steps per cycle of unsteady gust oscillation, while the HALE-RK schemes take 384.

For comparison, the BASS code is run on the same grid using the same spatial differencing, dissipation, and boundary conditions. On the first run, the RK56 time marching scheme is used with a CFL of 1.27. On the remaining runs, the HALE-RK schemes are used with a CFL of 3.17. The observed speedup is a factor of 2.1.

As an initial condition for both runs, the airfoil is impulsively started in a uniform flowfield, and the gust convects into the domain from the upstream boundary. Figs. 9–11 compare the results from the time marching schemes used. On the airfoil surface, Figs. 9 and 10 show that the schemes are getting very similar answers for the mean pressure and RMS pressure dis-

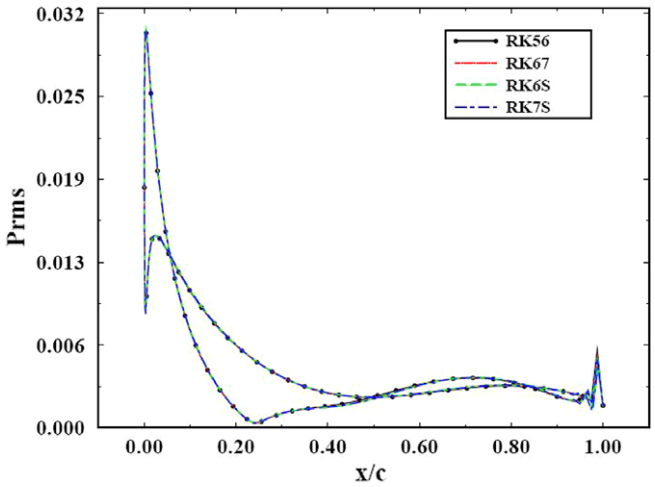

Fig. 10. Comparison of Prms on airfoil.

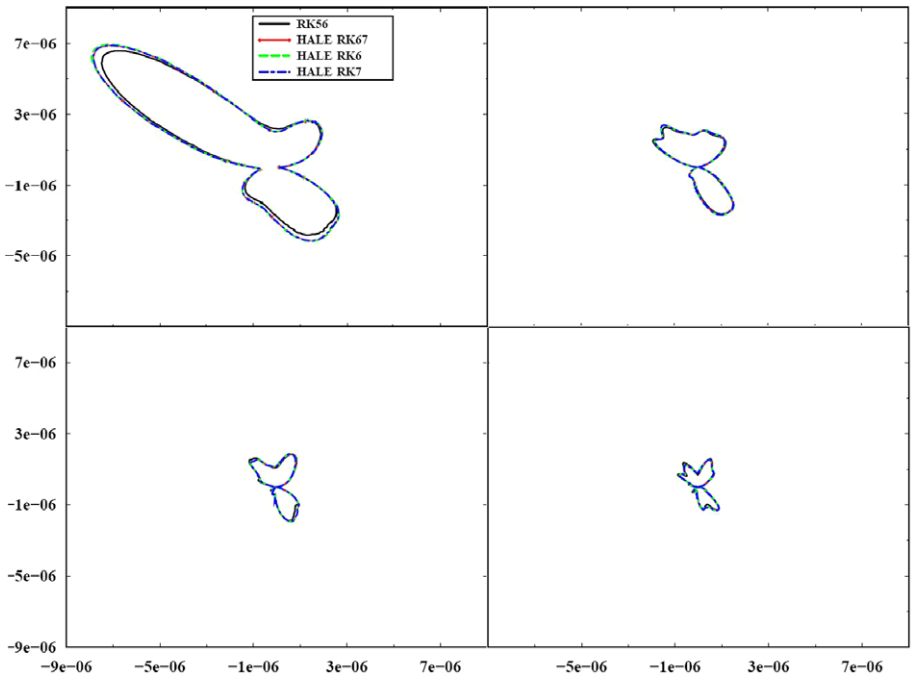

Fig. 11. Comparison of far field acoustic intensity.

turbance. In Fig. 9, the HALE-RK schemes are predicting a slightly higher mean airfoil lift than the RK56 scheme; however, the unsteady pressure disturbance on the airfoil surface is almost identical between the schemes. Fig. 11 shows the acoustic intensity in the far field. Again, the solutions are very similar, with slight differences in the amplitude of the radiated acoustic field.

7. Conclusions and future directions

In this work, a new series of high-accuracy large-step Runge–Kutta schemes were developed and validated for unsteady inviscid flow calculations. These schemes are designed to provide the maximum inviscid stability limit while retaining fourth-order nonlinear time accuracy. These schemes were tested on several benchmark nonlinear computational aeroacoustic problems, with very good results. It must be noted, however, that these schemes were tailored for inviscid flow calculations and lose their advantages for viscous flow problems. Based on these results, research is continuing to develop similar methods for use in Large Eddy Simulations.

Acknowledgments

This work is supported by NASA Glenn Research Center under the Quiet Aircraft Technology project. The authors would like to thank Dr. Edmane Envia, the technical monitor for this effort.

Appendix

The relationship between the original RK variables $a_{i,j}, b_i, c_i$ and the 2N-storage variables $\mathrm{a}_i, \mathrm{b}_i$ up to seven stages are presented in this section. Formulae given in [6] were used to obtain these relationships

$$
\begin{aligned}
a_{2,1} &= \mathrm{b}_1,\\
a_{3,1} &= \mathrm{a}_2\mathrm{b}_2 + \mathrm{b}_1,\\
a_{3,2} &= \mathrm{b}_2,\\
a_{4,1} &= \mathrm{a}_2(\mathrm{a}_3\mathrm{b}_3 + \mathrm{b}_2) + \mathrm{b}_1,\\
a_{4,2} &= \mathrm{a}_3\mathrm{b}_3 + \mathrm{b}_2,\\
a_{4,3} &= \mathrm{b}_3,\\
a_{5,1} &= \mathrm{a}_2(\mathrm{a}_3(\mathrm{a}_4\mathrm{b}_4 + \mathrm{b}_3) + \mathrm{b}_2) + \mathrm{b}_1,\\
a_{5,2} &= \mathrm{a}_3(\mathrm{a}_4\mathrm{b}_4 + \mathrm{b}_3) + \mathrm{b}_2,\\
a_{5,3} &= \mathrm{a}_4\mathrm{b}_4 + \mathrm{b}_3,\\
a_{5,4} &= \mathrm{b}_4,\\
a_{6,1} &= \mathrm{a}_2(\mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5\mathrm{b}_5 + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2) + \mathrm{b}_1,\\
a_{6,2} &= \mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5\mathrm{b}_5 + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2,\\
a_{6,3} &= \mathrm{a}_4(\mathrm{a}_5\mathrm{b}_5 + \mathrm{b}_4) + \mathrm{b}_3,\\
a_{6,4} &= \mathrm{a}_5\mathrm{b}_5 + \mathrm{b}_4,\\
a_{6,5} &= \mathrm{b}_5,\\
a_{7,1} &= \mathrm{a}_2(\mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6\mathrm{b}_6 + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2) + \mathrm{b}_1,\\
a_{7,2} &= \mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6\mathrm{b}_6 + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2,\\
a_{7,3} &= \mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6\mathrm{b}_6 + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3,\\
a_{7,4} &= \mathrm{a}_5(\mathrm{a}_6\mathrm{b}_6 + \mathrm{b}_5) + \mathrm{b}_4,\\
a_{7,5} &= \mathrm{a}_6\mathrm{b}_6 + \mathrm{b}_5,\\
a_{7,6} &= \mathrm{b}_6,
\end{aligned}
$$

$$
\begin{aligned}
b_1 &= \mathrm{a}_2(\mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6(\mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6) + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2) + \mathrm{b}_1,\\
b_2 &= \mathrm{a}_3(\mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6(\mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6) + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3) + \mathrm{b}_2,\\
b_3 &= \mathrm{a}_4(\mathrm{a}_5(\mathrm{a}_6(\mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6) + \mathrm{b}_5) + \mathrm{b}_4) + \mathrm{b}_3,\\
b_4 &= \mathrm{a}_5(\mathrm{a}_6(\mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6) + \mathrm{b}_5) + \mathrm{b}_4,\\
b_5 &= \mathrm{a}_6(\mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6) + \mathrm{b}_5,\\
b_6 &= \mathrm{a}_7\mathrm{b}_7 + \mathrm{b}_6,\\
b_7 &= \mathrm{b}_7,
\end{aligned}
$$

$$
\begin{aligned}
c_1 &= 0,\\
c_2 &= b_1,\\
c_3 &= b_1 + b_2(a_2 + 1),\\
c_4 &= b_1 + b_2(a_2 + 1) + b_3(a_3(a_2 + 1) + 1),\\
c_5 &= b_1 + b_2(a_2 + 1) + b_3(a_3(a_2 + 1) + 1) + b_4(a_4(a_3(a_2 + 1) + 1) + 1),\\
c_6 &= b_1 + b_2(a_2 + 1) + b_3(a_3(a_2 + 1) + 1) + b_4(a_4(a_3(a_2 + 1) + 1) + 1) + b_5(a_5(a_4(a_3(a_2 + 1) + 1) + 1) + 1),\\
c_7 &= b_1 + b_2(a_2 + 1) + b_3(a_3(a_2 + 1) + 1) + b_4(a_4(a_3(a_2 + 1) + 1) + 1)\\
&\quad + b_5(a_5(a_4(a_3(a_2 + 1) + 1) + 1) + 1)\\
&\quad + b_6(a_6(a_5(a_4(a_3(a_2 + 1) + 1) + 1) + 1) + 1).
\end{aligned}
$$

References

[1] C.K.W. Tam, Computational aeroacoustics: issues and methods, AIAA J. 33 (10) (1995) 1788–1796.
[2] S.K. Lele, Computational Aeroacoustics: A Review, AIAA Paper 97-0018, Reno, NV, 1997.
[3] S.K. Lele, Compact finite-difference schemes with spectral-like resolution, J. Comput. Phys. 103 (1992) 16–42.
[4] C.K.W. Tam, J.C. Webb, Dispersion-relation-preserving finite-difference schemes for computational acoustics, J. Comput. Phys. 107 (1993) 262–281.
[5] J.W. Kim, D.J. Lee, Optimized compact finite-difference schemes with maximum resolution, AIAA J. 34 (1996) 887–893.
[6] J.H. Williamson, Low storage Runge–Kutta schemes, J. Comput. Phys. 35 (1980) 48–56.
[7] M.R. Visbal, D.V. Gaitonde, Very high order spatially implicit schemes for computational acoustics on curvilinear meshes, J. Comput. Acoust. 9 (2001) 1259–1286.
[8] F.Q. Hu, M.Y. Hussaini, J. Manthey, Low dissipation and dispersion Runge–Kutta schemes for computational acoustics, J. Comput. Phys. 124 (1996) 177–191.
[9] D. Stanescu, W.G. Habashi, 2N-storage low dissipation and dispersion Runge–Kutta schemes for computational acoustics, J. Comput. Phys. 143 (1998) 674–681.
[10] M. Calvo, J.M. Franco, L. Randez, A new minimum storage Runge–Kutta scheme for computational acoustics, J. Comput. Phys. 201 (2004) 1–12.
[11] J.C. Hardin, J.R. Ristorcelli, C.K.W. Tam (Eds.), ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics (CAA), NASA CP-3300, Hampton, VA, 1994.
[12] C.K.W. Tam, J.C. Hardin (Eds.), Second Computational Aeroacoustics (CAA) Workshop on Benchmark Problems, NASA CP-3352, Hampton, VA, 1996.
[13] M. Dahl (Ed.), Third Computational Aeroacoustics (CAA) Workshop on Benchmark Problems, NASA CP-2000-209790, Cleveland, OH, 2000.
[14] M. Dahl, (Ed.), Fourth Computational Aeroacoustics (CAA) Workshop on Benchmark Problems, NASA CP-2004-212954, Cleveland, OH, 2004.
[15] H. Shen, C.K.W. Tam, Three dimensional numerical simulation of the jet screech phenomenon, AIAA J. 40 (2002) 33–41.
[16] D. Rizzetta, M. Visbal, Numerical Simulation of Separation Control for a Transitional Highly-Loaded Low-Pressure Turbine, AIAA Paper 2004-2204, Portland, OR, June 2004.
[17] A. Jameson, Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings, AIAA Paper 91-1596, Honolulu, HI, June 1991.
[18] C.A. Kennedy, M.H. Carpenter, Several new numerical methods for compressible shear-layer simulations, Appl. Numer. Simulat. 14 (1994) 397–433.
[19] R. Hixon, M. Nallasamy, S. Sawyer, R. Dyson, Comparison of numerical schemes for a realistic computational aeroacoustics benchmark problem, Int. J. Aeroacoust. 3 (2004) 379–398.
[20] J.R. Scott, Single airfoil gust response problem, in: Proceedings of the Fourth CAA Workshop on Benchmark Problems, NASA CP-2004-212954, October 2003, pp. 45–58.
[21] R. Hixon, V. Golubev, R.R. Mankbadi, J.R. Scott, S. Sawyer, M. Nallasamy, Application of a nonlinear computational aeroacoustics code to the gust-airfoil problem, AIAA J. 44 (2006) 323–328.
[22] J.C. Butcher, The Numerical Analysis of Ordinary Differential Equations, Wiley, New York, 1987.