On the performance of relaxation and adaptive explicit Runge-Kutta schemes for high-order compressible flow simulations

Journal of Computational Physics 464 (2022) 111333

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Short note

On the performance of relaxation and adaptive explicit Runge–Kutta schemes for high-order compressible flow simulations

Rasha Al Jahdali *, Lisandro Dalcin, Matteo Parsani

King Abdullah University of Science and Technology (KAUST), Computer Electrical and Mathematical Science and Engineering Division (CEMSE), Extreme Computing Research Center (ECRC), 23955-6900, Thuwal, Saudi Arabia

ARTICLE INFO

Article history:
Received 26 January 2022
Received in revised form 20 April 2022
Accepted 23 May 2022
Available online 26 May 2022

Keywords:
Entropy stability
Relaxation Runge-Kutta schemes
CFL
Error controller
Turbulent flows

1. Introduction

During the last decade, a substantial effort has been devoted to developing optimized explicit time integration schemes for high-order spatial discretizations in the context of computational fluid dynamics; see, for instance, [9,13] and the reference therein. Several works reported a time-to-solution saving of about 20-to-35% when using adaptive time stepping based on the error [13] or the Courant–Friedrichs–Lewy (CFL) condition [1] for solving industrially relevant turbulent flow problems. Recently, the "relaxation" procedure was introduced to enforce conservation, dissipation, or other solution properties with respect to any convex functional by the addition of a relaxation parameter that multiplies the Runge–Kutta update at each step [14,12]. Since then, the relaxation approach has been adopted and used successfully in many contexts of fluid dynamics and wave propagations.

In this note, we want to report the performance of some standard and optimized explicitly Runge–Kutta (ERK) schemes that are equipped with time step adaptivity when they are coupled with the relaxation procedure to achieve entropy stability for compressible computational fluid dynamics. The ultimate goal is to verify the robustness of the overall adaptivity procedure and the impact on the computational costs of the relaxation for practical and complex flow problems discretized with an *hp*-adaptive collocated entropy stable discontinuous Galerkin (DG) solver [8].

* Corresponding author.
E-mail addresses: rasha.aljahdali@kaust.edu.sa (R. Al Jahdali), dalcinl@gmail.com (L. Dalcin), matteo.parsani@kaust.edu.sa (M. Parsani).

https://doi.org/10.1016/j.jcp.2022.111333

Table 1
Desired CFL number of the ERK schemes for CFL-based controller [1].

Scheme	ERK(4,2)	ERK(8,2)	ERK(5,3)	ERK(4,4)
CFL_d	4.1471	8.3132	4.8111	2.7852

2. Global and local relaxation Runge–Kutta schemes

In many applications, there are smooth energy/entropy/Lyapunov functionals, η, whose evolution in time is important for computing the solution of partial differential equations (PDEs). The approximation of spatially dependent PDEs usually proceeds by partitioning the computational domain, Ω, into K non-overlapping elements and discretizing the spatial derivatives. This process leads to a system of ordinary differential equations (ODEs) which can be integrated, for instance, with a Runge–Kutta method.

A general (explicit or implicit) Runge–Kutta method with s stages can be represented by its Butcher tableau [5] which is composed by a square matrix A of dimensions $s \times s$, and two column vectors b and c of length s. All these coefficients are real numbers. In the context of the relaxation procedure, the basic idea to enforce conservation, dissipation, or other solution properties with respect to a convex functional is to scale the weights b_i of a given Runge–Kutta method by a real-value parameter $\tilde{\gamma}$. Thus, using an s-stage Runge–Kutta scheme, a time step from $u^n \approx u(t_n)$ is given by

$$y_i = u^n + \Delta t \sum_{j=1}^{s} a_{ij} f(t_n + c_j \Delta t, y_j), \qquad u_{\tilde{\gamma}}^{n+1} = u^n + \tilde{\gamma}_n \Delta t \sum_{i=1}^{s} b_i f(t_n + c_i \Delta t, y_i), \tag{1}$$

where y_i are the stage values of the Runge–Kutta method. For the global relaxation procedure, $\tilde{\gamma}_n$ is a root of a global nonlinear algebraic equation for η [14]. On the contrary, for the local relaxation approach, $\tilde{\gamma}_n = min_\kappa\, \tilde{\gamma}_{n,\kappa}$, *i.e.*, the minimum among the roots of a local equation for η_κ for each cell, κ, of the discretized spatial domain [12]. If $\tilde{\gamma}_n = 1$, we recover the standard Runge–Kutta method.

2.1. Selected Runge–Kutta schemes

As required by the relaxation theory [7,14], we select ERK schemes with positive weights, b_i. In particular, we use two second-order schemes with CFL-based timestep controller, *i.e.*, ERK(4,2) and ERK(8,2) [1], one third-order scheme with CFL-based timestep controller, *i.e.*, ERK(5,3) [1], three third-order schemes with error-based timestep controller, *i.e.*, BSRK(4,3), SSP3(2)4[3S*+] and RK3(2)5F[3S*+] [13,9], the classical four-stage fourth-order scheme with CFL-based timestep controller, *i.e.*, ERK(4,4) [5], one fifth-order scheme with error-based timestep controller, *i.e.*, BSRK(8,5) [3], and one sixth-order scheme with error-based timestep controller, *i.e.*, VRK(9,6) [16]. All these schemes have been throughly tested and verified when used in combination with an adaptive timestep strategy [1,13]. In addition, some of these ERK schemes (e.g., ERK(4,4), BSRK(4,3), and BSRK(8,5)) are the work-horses of commercial and national laboratories compressible computational fluid dynamics solvers.

Using the CFL-based controller, we compute the step size as [1]

$$\Delta t_{n+1} = \sigma\, CFL_d \min_i \frac{\Delta x_i}{v_i^n}, \tag{2}$$

where σ is a safety factor, CFL_d is the desired CFL number, Δx_i is a local measure of the mesh spacing, and v_i^n is some measure of the maximal (local) wave speed, related to the largest (in magnitude) eigenvalue of the flux Jacobian. The value of σ is set to 0.95 for all the schemes [1,13,9]. The appropriate choice of CFL_d depends on the details of the space and time discretizations. Here, CFL_d is set to the maximum stable time step [1]; see Table 1. The reason is well-known: in compressible fluid dynamics stability constraints (and not accuracy constraints) limit severely the time step size of explicit time integration schemes when used to integrate system of ODEs arising from the spatial discretization with high-order accurate algorithms [11].

For error-based time adaptivity, we compute the time step using a proportional-integral-derivative (PID) controller [13]:

$$\Delta t_{n+1} = \mathrm{PID}(\beta, \texttt{atol}, \texttt{rtol}, err^{n+1}, err^n, err^{n-1}), \tag{3}$$

where β is a vector of real-value coefficients of the PID controller [13], `atol` and `rtol` are the absolute and relative error tolerances, and err^{n+1}, err^n, and err^{n-1} are the local error estimates. Here, we set $\texttt{rtol} = \texttt{atol} = 10^{-6}$.

2.2. Nonlinear scalar algebraic solver

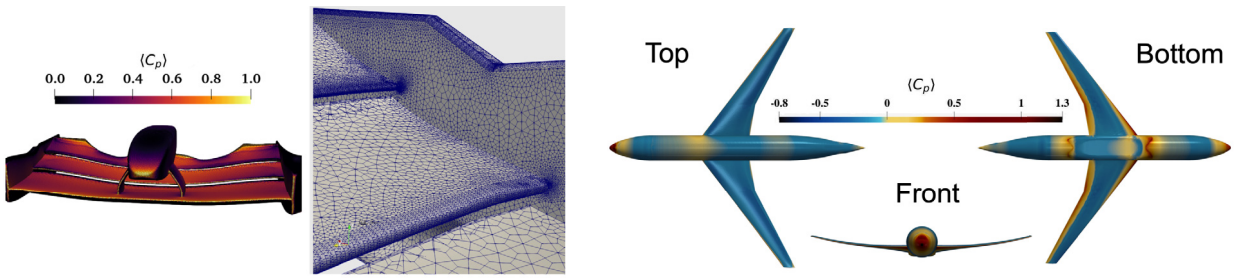
Before presenting the results on the performance of the relaxation Runge–Kutta schemes with time adaptivity, it is critical to compare the capability of the scalar nonlinear solver to converge to a given tolerance, ε. Here, we perform all the computations in double precision and we use a tolerance of $\varepsilon = 10^{-12}$. Tables 2 and 3 show the behavior of the compressible

Table 2
Robustness of the nonlinear solver for the compressible homogeneous isotropic turbulence test case at $Ma_t = 0.6$ and $\mathrm{Re}_\lambda = 100$. The symbols (✗) and (✓) mean failed and successful numerical experiment, respectively.

Method	Bisection	Brent's	Newton's	Secant	Simpl. Newton	TOMS 748-1	TOMS 748-2
$p = 2$ $N = 64^3$	✗ (✓)	✗(✓)	✗(✗)	✓(✓)	✗(✓)	✓(✗)	✓(✗)
$p = 2$ $N = 128^3$	✓ (✗)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✗(✓)	✓(✓)
$p = 3$ $N = 64^3$	✓ (✓)	✓(✓)	✗(✓)	✓(✓)	✗(✓)	✓(✓)	✓(✓)
$p = 3$ $N = 128^3$	✓ (✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)
$p = 7$ $N = 32^3$	✓ (✓)	✓(✓)	✗(✓)	✓(✓)	✗(✓)	✓(✓)	✓(✓)
$p = 7$ $N = 64^3$	✓ (✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✓(✓)

Table 3
Robustness of the nonlinear solver for flow past a cylinder at $Ma = 0.1$ and $\mathrm{Re} = 3.9 \times 10^3$, 2.0×10^4, and 1.4×10^5. The symbols (✗) and (✓) mean failed and successful numerical experiment, respectively.

Method	Bisection	Brent's	Newton's	Secant	Simpl. Newton	TOMS 748-1	TOMS 748-2
$Re = 3.9 \times 10^3$	✗ (✓)	✓(✓)	✗(✗)	✓(✓)	✗(✓)	✓(✗)	✓(✓)
$Re = 2.0 \times 10^4$	✓ (✗)	✓(✓)	✓(✓)	✓(✓)	✓(✓)	✗(✓)	✓(✓)
$Re = 1.4 \times 10^5$	✓ (✓)	✓(✓)	✗(✓)	✓(✓)	✗(✓)	✓(✓)	✓(✗)



(a) F1 Imperial front wing.

(b) NASA CRM.

Fig. 1. Time-averaged pressure coefficient, $\langle C_p \rangle$: (a) on the surface of the F1 Imperial front wing, and (b) for the NASA CRM.

homogeneous isotropic turbulence at a turbulent Mach number $Ma_t = 0.6$ and a Reynolds number $\mathrm{Re}_\lambda = 100$ based on the Taylor microscale, and the flow past a three-dimensional circular cylinder at three different Reynolds number [8]. We solve the two problems by using six different configurations of the global and local relaxation procedures. Each problem is solved for $1,000$ time steps with the classical ERK(4,4) [5] and seven methods with global and local relaxation. The results for the local relaxation are shown in parentheses. The *hp*-adaptive entropy stable collocated DG method presented in [8] is used for the spatial discretization. The nonlinear algebraic solvers are bisection, Brent's method [4], original Newton's method, secant, simplified Newton's [6, Section 2.1.2], TOMS748-1 and TOMS748-2 [2, Section 4.1, 4.2]. For the Newton's method and the simplified Newton's method, the derivative of the nonlinear scalar function is evaluated using the analytical expression [14]. Both Newton's and the simplified Newton's methods turned out to be the least reliable and failed to converge in several scenarios. This behavior was reproducible and observed even with the maximum number of allowed iterations as high as $1,000$. Here, we choose the secant methods which have demonstrated a good robustness property for the study reported in Tables 2 and 3 and for other complex test cases presented in [8,1].

3. Numerical results

The performance of the selected explicit ERK is reported and analyzed for two test cases: the flow past the F1 Imperial front wing (IFW) [10], and the transonic flow past the NASA common research model (NASA CRM) with wings and fuselage [15]. The IFW flow problem is characterized by a Mach number of $Ma = 0.036$ and a Reynolds number $Re = 2.2 \times 10^5$ [10]. The computational domain is divided into 3.4×10^6 hexahedral elements with a maximum aspect ratio of approximately 250. The flow configuration of the NASA CRM is characterized by $Ma = 0.85$, $Re = 5 \times 10^6$, and an angle of attack of $AoA = 2.75^0$ [15]. The computational domain contains 9.1×10^6 hexahedral elements with a maximum aspect ratio of approximately 105. The entropy stable collocated DG method presented in [8] is also used here for the spatial discretization. We use a solution polynomial degree $p = 1$ in the far-field region, $p = 3$ in the region surrounding the models, and $p = 2$ elsewhere. Figs. 1a and 1b show the time-averaged pressure coefficient distribution, $\langle C_p \rangle$, for the IFW and NASA-CRM, respectively. In addition, for the IFW, we also show the mesh on the streamwise central part of the wing.

We perform 600 time steps for each problem configuration starting from a fully developed flow. We use 512 CPU cores of Shaheen XC40, the supercomputer hosted at KAUST.

Table 4
Performance of the schemes for the simulation of the flow past the Imperial F1 front wing and the NASA common research model.

		Imperial front wing		NASA CRM model	
ERK scheme	Approach	#FE	WTC [s]	#FE	WTC [s]
ERK(4,2)	Without relaxation	6,624	2.1347e+03	1,004	1.4062e+02
	Global relaxation	6,636	2.1358e+03	1,008	1.4108e+02
	Local relaxation	6,628	2.1286e+03	1,008	1.4081e+02
ERK(8,2)	Without relaxation	6,608	2.1441e+03	1,000	1.3995e+02
	Global relaxation	6,632	2.1547e+03	1,024	1.4340e+02
	Local relaxation	6,616	2.1414e+03	1,008	1.4125e+02
BSRK(4,3)	Without relaxation	4,250	1.4105e+03	817	1.1783e+02
	Global relaxation	5,648	1.8323e+03	1,084	1.5248e+02
	Local relaxation	5,652	1.8365e+03	1,088	1.5327e+02
SSP3(2)4[3S*+]	Without relaxation	4,250	1.4175e+03	1,356	1.9057e+02
	Global relaxation	5,648	1.8308e+03	1,352	1.9029e+02
	Local relaxation	5,652	1.8359e+03	1,356	1.9117e+02
RK3(2)5F[3S*+]	Without relaxation	4,250	1.4145e+03	817	1.1780e+02
	Global relaxation	5,648	1.8306e+03	1,084	1.5228e+02
	Local relaxation	5,652	1.8311e+03	1,088	1.5299e+02
ERK(5,3)	Without relaxation	7,135	2.2951e+03	1,080	1.5131e+02
	Global relaxation	7,135	2.2984e+03	1,080	1.5140e+02
	Local relaxation	7,140	2.2970e+03	1,085	1.5187e+02
ERK(4,4)	Without relaxation	9,860	3.1563e+03	1,492	2.0858e+02
	Global relaxation	9,872	3.1603e+03	1,504	2.1078e+02
	Local relaxation	9,864	3.1603e+03	1,496	2.0941e+02
BSRK(8,5)	Without relaxation	4,250	1.4149e+03	904	1.2841e+02
	Global relaxation	5,648	1.8316e+03	952	1.3411e+02
	Local relaxation	5,652	1.8326e+03	952	1.3404e+02
VRK(9,6)	Without relaxation	4,250	1.4116e+03	873	1.2462e+02
	Global relaxation	5,648	1.8297e+03	999	1.4170e+02
	Local relaxation	5,652	1.8276e+03	999	1.4168e+02

In Table 4, we report the number of function evaluations (#FE) and the arithmetic average of the wall-clock time (WCT) of three independent runs. The schemes are listed based on ascending order of accuracy. First, for both test cases, we observe that all the simulations are stable, also when the relaxation procedure is not active, and theoretically, we have no guarantee that the full discretization is entropy stable. Second, we can see that i) the number of function evaluations required by both relaxation approaches is greater than the number of function evaluations required by the schemes without relaxation, and ii) the global and local relaxation approaches require a similar number of function evaluations. As a consequence, the wall-clock time taken by the relaxation approaches is larger than that of the same scheme without relaxation. In addition, we highlight that when the number of function evaluations of the scheme without relaxation is close to those of the schemes equipped with relaxation, the difference in wall-clock time is practically due to the time spent by nonlinear algebraic solver to compute the relaxation parameter, $\tilde{\gamma}_n$, used in Equation (1). Third, comparing the third-order accurate schemes, we observe that the schemes equipped with the error-based controller are substantially more efficient than those using the CFL-based controller, whether or not the relaxation procedure is activated. In addition, we notice that the fifth- and sixth-order schemes with error-based time step adaptivity (*i.e.*, BSRK(8,5) and VRK(9,6)) are even faster than the second- and third-order accurate schemes whose time step is controlled by the CFL number. Finally, the schemes with CFL-based timestep controller (highlighted in gray in Table 4) require a number of function evaluations that is very close to the time integration scheme without relaxation. This is a feature that we do not observe for the schemes where the time step is computed using the error-based controller. This seems to indicate that with the current CFL-based controller leads to a time step that is close to the value required to attain entropy stability – that is, $\tilde{\gamma}_n$ is very close to one. In addition, we do not observe any particular difference when we increase the order of accuracy of the time integration, *i.e.*, when moving down along Table 4.

4. Conclusions

We have reported and analyzed the performance of nine explicit Runge–Kutta schemes combined with CFL- or error-based controller and the relaxation procedure for the time step adaptivity. According to thorough numerical experiments based on the compressible homogenous isotropic turbulence and the turbulent flow past a three-dimensional cylinder, the secant method is the most robust nonlinear algebraic solver for calculating the relaxation parameter. In general, the

adaptive explicit Runge–Kutta schemes coupled with the global or local relaxation procedure are more expensive than the adaptive schemes without relaxation for two reasons. First, the number of time steps and hence the number of functions evaluations is almost on par or greater than the number of steps taken by the schemes without relaxation. Second, each time step requires the solution of a scalar nonlinear algebraic equation for the global relaxation or a set of scalar nonlinear algebraic equations for the local relaxation procedure. However, the relaxation schemes guarantee entropy stability of the fully-discrete algorithm, a feature that provides indisputable robustness to the solver. Finally, we highlight that the schemes equipped with the error-based controller are substantially more efficient than those using the CFL-based controller, whether or not the relaxation procedure is activated. In addition, we notice that the fifth- and sixth-order schemes with error-based time step adaptivity (*i.e.*, BSRK(8,5) and VRK(9,6)) are even faster than the second- and third-order accurate schemes whose time step is controlled by the CFL number. Thus, if entropy stability is sought, the relaxation high-order time integration schemes with the error-based controller can be used at practically the same cost as lower-order time integration schemes.

CRediT authorship contribution statement

Rasha Al Jahdali: Conceptualization, Data curation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Lisandro Dalcin:** Software, Writing – review & editing. **Matteo Parsani:** Conceptualization, Funding acquisition, Software, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The work described in this paper was supported by King Abdullah University of Science and Technology through the award OSR-2019-CCF-3666. We are thankful to the Supercomputing Laboratory and the Extreme Computing Research Center at King Abdullah University of Science and Technology for their computing resources.

References

[1] R. Al Jahdali, L. Dalcin, R. Boukharfane, I.R. Nolasco, D.E. Keyes, M. Parsani, Optimized explicit Runge–Kutta schemes for high-order collocated discontinuous Galerkin methods for compressible fluid dynamics, Comput. Math. Appl. 118 (2022).
[2] G.E. Alefeld, F.A. Potra, Y. Shi, Algorithm 748: enclosing zeros of continuous functions, ACM Trans. Math. Softw. 21 (3) (1995) 327–344.
[3] P. Bogacki, L.F. Shampine, An efficient Runge–Kutta (4, 5) pair, Comput. Math. Appl. 32 (6) (1996) 15–28.
[4] R.P. Brent, Algorithms for Minimization Without Derivatives, Courier Corporation, 2013.
[5] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, John Wiley & Sons Ltd, Chichester, 2016.
[6] P. Deuflhard, Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms, Springer Publishing Company, Incorporated, 2011.
[7] D.I. Ketcheson, Relaxation Runge–Kutta methods: conservation and stability for inner-product norms, SIAM J. Numer. Anal. 57 (6) (2019) 2850–2870.
[8] M. Parsani, R. Boukharfane, I.R. Nolasco, D.C. Del Rey Fernández, S. Zampini, B. Hadri, L. Dalcin, High-order accurate entropy-stable discontinuous collocated Galerkin methods with the summation-by-parts property for compressible CFD frameworks: scalable SSDC algorithms and flow solver, J. Comput. Phys. 424 (2021) 109844.
[9] M. Parsani, D.I. Ketcheson, W. Deconinck, Optimized explicit Runge-Kutta schemes for the spectral difference method applied to wave propagation problems, SIAM J. Sci. Comput. 35 (2) (2013) A957–A986.
[10] J.M. Pegrum, Experimental study of the vortex system generated by a Formula 1 front wing, Ph.D. thesis, Imperial College London, 2007.
[11] P.-O. Persson, High-Order Accurate Time Integration and Efficient Implicit Solvers, Springer International Publishing, 2021, pp. 239–259.
[12] H. Ranocha, L. Dalcin, M. Parsani, Fully discrete explicit locally entropy-stable schemes for the compressible Euler and Navier–Stokes equations, Comput. Math. Appl. 80 (5) (2020) 1343–1359.
[13] H. Ranocha, L. Dalcin, M. Parsani, D.I. Ketcheson, Optimized Runge–Kutta methods with automatic step size control for compressible computational fluid dynamics, Commun. Appl. Math. Comput. (2021).
[14] H. Ranocha, M. Sayyari, L. Dalcin, M. Parsani, D.I. Ketcheson, Relaxation Runge-Kutta methods: fully-discrete explicit entropy-stable schemes for the compressible Euler and Navier–Stokes equations, SIAM J. Sci. Comput. 42 (2) (2020) A612–A638.
[15] M.B. Rivers, NASA common research model: a history and future plans, AIAA SciTech 2019 Forum 1, pp. 1–36, https://doi.org/10.2514/6.2019-3725.
[16] J.H. Verner, Explicit Runge–Kutta methods with estimates of the local truncation error, SIAM J. Numer. Anal. 15 (4) (1978) 772–790.