

Intrastep, Stage-Value Predictors for Diagonally-Implicit Runge–Kutta Methods

Mark H. Carpenter,^{} Christopher A. Kennedy,[§] and Joseph M. Derlaga^{*}*

^{}Langley Research Center, Hampton, Virginia, USA*

[§]Private Professional Consultant, Palo Alto, CA, USA

NASA STI Program Report Series

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

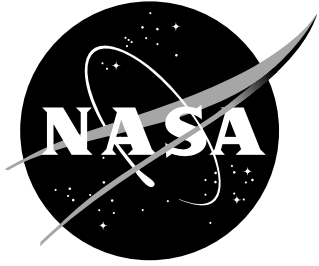
- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA Programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- Help desk contact information: <https://www.sti.nasa.gov/sti-contact-form/> and select the "General" help request type.



Intrastep, Stage-Value Predictors for Diagonally-Implicit Runge–Kutta Methods

Mark H. Carpenter,^{} Christopher A. Kennedy,[§] and Joseph M. Derlaga^{*}*

^{}Langley Research Center, Hampton, Virginia, USA*

[§]Private Professional Consultant, Palo Alto, CA, USA

National Aeronautics and
Space Administration

NASA Langley Research Center
Hampton, Virginia 23681-2199

September 2024

Acknowledgments

Special thanks are extended to Dr. Mujeeb R. Malik for funding this work as part of NASA's Transformational Tools and Technologies (T^3) project.

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

Available from:

NASA STI Program / Mail Stop 050
NASA Langley Research Center
Hampton, VA 23681-2199

Abstract

To better identify the necessary attributes of good stage-value predictors (SVPs), numerous SVPs are designed for an existing: ESDIRK4(3)7L[2]SA [26] and a new: ESDIRK4(3)8L[2]SA scheme.¹ Both are stiffly-accurate, stage-order two, explicit, singly-diagonally implicit Runge–Kutta (ESDIRK) schemes. Tradeoffs are studied in the parameter spaces enforcing the constraints on accuracy, linear stability, nonlinear stability and coefficient size to determine which objectives correlate with effective predictors. The SVPs are tested on three challenging external aerodynamics problems [$10^7 - 10^8$ degrees of freedom (DoFs)], each with a different level of stiffness. The problems include two 3D airfoils simulations and one canonical turbulence simulation. All simulations use the compressible Navier-Stokes equations (CNSE). An entropy stable spectral collocation formulation is used for discretizing the spatial terms in the equations. Simulations are performed at a wide variety of temporal error tolerances. Problems that are sufficiently stiff (e.g., lax temporal error tolerances) benefit from SVPs designed with second-order accuracy and stability properties: A-stability, and L-stability, rather than high accuracy constraints. Simulations with modest stiffness (e.g., strict error tolerances) are better suited for SVPs designed using high accuracy constraints. Designing SVPs with enhanced stability properties is tedious but worthwhile. Simulation times are reduced with optimal SVPs by as much as 100% on some stages, with combined stepwise improvements of between 50 – 100% for both methods. A comparative study is performed with the two aforementioned methods as well as four other ESDIRKs. The newly designed ESDIRK4(3)8L[2]SA with $\gamma \approx 1/10$, proves to be the most efficient of the six tested ESDIRK schemes simulating the CNSE.

Keywords

Diagonally Implicit Runge-Kutta (DIRK), Stage-value-predictor (SVP), Compressible Navier-Stokes, Summation-By-Parts (SBP), Simultaneous-Approximation-Term (SAT), Entropy Stability, Discontinuous Spectral Collocation, Stiffness

Contents

1	Introduction	3
1.1	A brief history of stage-value predictors	5
1.2	Objectives	6
1.3	The Experimental Strategy	6
2	The Base Methods	7
3	Intrastep, Function-Based Predictors	8
3.1	Trivial Predictor	8
3.2	SVP Structure and Implementation	9

¹Nomenclature. Fourth-order accurate: 4, embedded third-order: (3), seven stages: 7, stepwise (A-) L-stability: L, stage order two: [2], and enforces the stiffly accurate assumption: SA.

3.2.1	Dense-Output SVPs for Stage Two	9
3.2.2	SVPs for Stages Three Through s	10
3.3	Order Conditions	10
3.4	Internal Stability	11
4	Predictor Overview and Design Details	12
5	Solver Mechanics	13
5.1	Spatial Discretization	13
5.2	Iterative Solver	13
5.3	Controllers	14
6	Test Problems	14
6.1	SD7003 airfoil	15
6.2	Multi-element Airfoil (MD-30P30N)	15
6.3	Stiffness	16
7	Results	16
7.1	Prologue	16
7.2	ESDIRK4(3)7L[2]SA: H11344 on SD7003 airfoil	17
7.2.1	Stage 2	17
7.2.2	Stage 3	17
7.2.3	Stage 4	18
7.2.4	Stage 5	18
7.2.5	Stage 6	18
7.2.6	Stage 7	19
7.3	ESDIRK4(3)8L[2]SA: H11344 on SD7003 airfoil	19
7.3.1	Stage two	19
7.3.2	Stage three	22
7.3.3	Stage four	22
7.3.4	Stage five	22
7.3.5	Stage 6	22
7.3.6	Stage 7	23
7.3.7	Stage 8	23
7.3.8	Summaries: ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA	23
7.4	ESDIRK4(3)8L[2]SA: H63004 on MD-30P30N ME-airfoil	24

7.5	ESDIRK4(3)8L[2]SA: H4096 on TG-Vortex	24
7.6	Stability contours of performant schemes	27
7.6.1	ESDIRK4(3)7L[2]SA	27
7.6.2	ESDIRK4(3)8L[2]SA	27
7.7	Discussion	28
8	The Efficiency Landscape of ESDIRK Schemes	31
9	Conclusions	34
10	Acknowledgments	34
11	References	35
A	Improving Efficiency with Additional Stages	38
B	Methods	40
B.1	ESDIRK4(3)7L[2]SA	40
B.1.1	Stage-value predictors	40
B.1.2	Dense Output	40
B.2	ESDIRK4(3)8L[2]SA	40
B.2.1	Method Design	41
B.2.2	Butcher Tableau and properties	42
B.2.3	Stage-value predictors	43
B.2.4	Dense Output	43
C	Order Reduction	45
C.1	Van der Pol’s Equation	45

1 Introduction

Engineering solutions to the compressible Navier-Stokes equations (CNSE) are routinely sought and are typically very expensive owing to their size. The dimensionality of a state-of-the-art simulation could be $10^8 - 10^{12}$ degrees of freedom (DoFs), all being advanced temporally using a method of lines approach. The search for improved time-integration methods is one part of the quest to significantly reduce the computational burden of these challenging simulations. Compressible fluid simulations can manifest extreme levels of numerical stiffness from a variety of sources: 1) high Reynolds number boundary-layer elements, 2) trailing-edge singularities, 3) sliver cells that are beyond repair, 4) chemical reactions, 5) acoustic time

scales, 6) turbulence models to name a few. A good "stiff integrator" is vital in these real-world applications.

A stiff integrator for the CNSE is assembled with the following components: 1) a discretization of the right-hand side (RHS) of the equations, 2) an exact or approximate Jacobian matrix, 3) a factorization and storage of said matrix for use as a preconditioner, 4) the nonlinear iteration loop (e.g., Newton's method), and 5) a preconditioned algebraic linear solver (e.g., GMRES [35]). The relative cost of each component can vary dramatically based on algorithmic choices, the simulation hardware, and the level of solver optimization. Modern CPU hardware typically has two gigabytes of RAM per processor, which allows for the storage and reuse of a high-quality preconditioner. Thus, the Jacobian formation, approximate factorization, data compression and preconditioner storage are amortized over many linear solves. As a result, the dominant cost in the "stiff" integration for the CNSE is associated with the nonlinear/linear solvers.

Not surprisingly, the implementation of a given method can have a dramatic effect on the performance of the integration scheme. The quadratic convergence behavior of a Newton iteration often enables only three to four nonlinear iterates to reduce the residual to engineering tolerances. Each iterate requires the solution of an extremely large linear system. The number of search directions required by preconditioned GMRES to converge the linear system is strongly dependent on the temporal method, the timestep, the preconditioner and the starting guess for the solution vector at the beginning of the nonlinear iteration. Optimizing the starting solution vector is therefore a sensible place to search for improving the efficiency of integrators. We refer to the elaborate mechanics which supply these starting guesses as stage-value predictors (SVPs). Relative to the nonlinear/linear iterations, the SVP mechanics is of negligible cost and should always be considered mandatory when designing a stiff integrator. Although effective SVPs are difficult to design for some stiff integrators, the remarkable efficiency improvements demonstrated herein suggest that they should be considered as essential components in all viable stiff integrators.

Stage-value predictors can improve a solver's efficiency in three distinct ways. First, an effective SVP places the starting solution inside the quadratic envelope of Newton's method. The first iteration is much less likely to diverge and is less prone to sporadic convergence as the nonlinear solver searches for Newton's quadratic convergence ball. A second benefit is that fewer Newton iterates are required to achieve the specified residual tolerance. Experience suggests that an effective SVP facilitates convergence in one to two fewer iterations. The third benefit is the impact on the *linear* convergence rate. Nearly all SVPs provide starting solutions that converge more rapidly than the *trivial guess* (i.e., doing nothing). Results presented herein, however, highlight important SVP design properties that lead to more rapid convergence of the linear solver. Note that this third benefit does not manifest in simpler settings where the linear system is solved exactly. For this reason, popular model problems such as the van der Pol equation, or Kap's equation, while important in their own right, provide little guidance on the efficacy of SVPs for the CNSE.

Stage-value predictors may be broadly divided into those which compute the starting guess for the solution at a given stage as either a linear combination of previous solution data or a solution and function data [12]. Within both classes, SVPs may also be distinguished by the degree to which past data is included. We refer to SVPs as *intrastep* if they only employ data from the current step. These intrastep SVPs are exclusive to Runge–Kutta schemes from a subclass of implicit Runge–Kutta methods that are implemented

sequentially, [e.g., Diagonally-Implicit Runge–Kutta methods (DIRKs), Singly-DIRK methods (SDIRKs), or Explicit-SDIRK methods (ESDIRKs)]. They are simpler and require less computer memory, but on the early stages have limited data to exploit and are not defined in the first implicit stage. They are previously studied extensively by Higuera [18]. Alternatively, *interstep* SVPs use additional data from one or more previous steps (e.g., one- or two-step SVPs) to predict starting values in the current step. These predictors are well suited for fully implicit Runge–Kutta schemes (e.g., Radau IIa, Gauss). Note that methods with a lower-diagonal Runge–Kutta $\mathbf{A}$ -matrix are amenable to combinations of interstep and intrastep predictors.

The focus herein is on constructing intrastep SVPs for ESDIRK schemes. All stages with available data are considered, i.e., three and greater. It seems prudent to limit our initial investigations to this simpler class of predictors. Design insights and best practices can be established before extending the scope to the extremely challenging designs that involve intra and interstep data.²

1.1 A brief history of stage-value predictors

A variety of possible stage-value predictors have been developed over the past three decades. Consult the following references (cited chronologically) for some important contributions. [2–8, 12–15, 17, 18, 20, 22, 23, 27, 31, 34, 37, 38, 40].

Single-step predictors appear first to have been considered by Sand [37, 38]. Sand [38] derives order and inverse-order conditions for a class of predictors and distinguishes between cases where the timestep-size ratio, $r = (\Delta t)^{[n+1]}/(\Delta t)^{[n]}$, is either $r < 0.1$, $0.1 \leq r \leq 2.0$, or $2.0 \leq r$. As r increases, order is reduced and inverse order is increased. Laburta [27–30] extends the concept of single-step predictors by including one or more additional function evaluations. Hairer, Lubich and Wanner [15] formalize the work of Laburta in their book. Roldan and Higuera [34] and Higuera and Roldan [18] focus on single- and multiple-step predictors for the subclass of DIRK operators. They utilized available intrastep stage information to augment the data from the previous time step and show that this local information significantly improved the quality of the SVP for DIRK class schemes. Roldan and Higuera [34] extend SVPs to index-1 differential algebraic equations (DAEs) after deriving the general order conditions. No additional functional evaluations are used, but both single- and multiple-step SVPs are constructed with and without intrastep solution data. Higuera and Roldan [18–21] derive the consistency conditions for index-2 DAEs and construct multistep SVPs. They also develop the index-2 DAE SVP theory for the singular coefficient matrices encountered in ESDIRK and Lobatto IIIA methods, and the simplifying assumptions that lead to optimal order of accuracy. Their primary focus is on developing SVPs with the highest possible order-of-accuracy. The numerical stability of the SVPs is largely ignored.

González-Pinto et al. [12–14] made several novel contributions to the development of SVPs. The first is to distinguish between two design approaches. Define the solution values on the internal stages as $X_i = U^{[n]}e_i + \Delta t \sum_{j=1,s} a_{ij}f(t_0 + c_j\Delta t, \mathbf{X}_j)$, ($i = 1, \dots, s$) where e_i is the i th-component of the vector $\mathbf{e} = \{1, 1, \dots, 1\}$, t_0 is time at the beginning of the step and Δt is the time step. The two approaches at the internal stages of the next step are

$$\begin{aligned} \text{Type I : } U_{i,0} &= \gamma_i U^{[n]} + \sum_{j=1,s} (\alpha_{ij} \mathbf{X}_j) \\ \text{Type II : } U_{i,0} &= U^{[n]} + \Delta t \delta_i f(t_0, U^{[n]}) + \Delta t \sum_{j=1,s} (\beta_{ij} f(t_0 + c_j \Delta t, \mathbf{X}_j)). \end{aligned} \quad (1)$$

²A rudimentary *interstep* SVP based on dense output is provided for stage two but is not this investigation’s primary goal.

where α_{ij} , β_{ij} , γ_i and δ_i are coefficients of the predictor. Type I SVPs are essentially *extrapolation methods* based on previous solution data, whereas Type II SVPs adhere more closely to the conventional Runge–Kutta formalism by setting $\gamma_i = 1$ and using solution and function values. A second contribution is the recognition that SVPs, like conventional Runge–Kutta schemes, are susceptible to order reduction by stiff solution components. They define the conventional nonstiff order p as: $|U_i - U_{i,0}| = O(\Delta t^{p+1})$, where $U_{i,0}$ is the starting value and U_i is the converged value. However, they define a stiff order q : $|U_i - U_{i,0}| = O(\Delta t^{q+1})$, if the accuracy of the SVP is independent of the solution stiffness. The model problem by Prothero and Robinson [33] is used to investigate the order reduction of SVPs. A third contribution was the recognition that stiff components could lead to large errors in the predicted starting guesses, despite having high nonstiff-order p . They conclude that at stringent tolerances the formal predictor order is most important, but at more lax tolerances stability properties become increasingly more important. They provide a switching algorithm based on an SVP error estimator that chooses between low-order but stable predictors and high-order, less-stable predictors. Calvo and Portillo [5] also conclude that higher-order starters do not always outperform lower-order ones.

A recent overview of SVPs for DIRKs, SDIRKs and ESDIRKs is given by the present authors [24]. While no SVPs are designed, several recommendations are given on how to design the underlying method to enhance the efficacy of the SVP. Among these are 1) higher stage order (only stage order two can be achieved with ESDIRK schemes), 2) stepwise and stagewise A-stability for all internal stages, 3) stepwise L-stability and stiff-accuracy, 4) stagewise L-stability when possible, and 5) minimized algebraic instability on both internal stages and the step. All of these attributes are not only consistent with the design of good SVPs but are also consistent with the design of a good underlying method.

1.2 Objectives

The primary objective addressed herein is to provide insight into the design landscape of SVPs for use in integrating stiff, nonlinear systems of large dimensions (e.g., CNSE). Characterization of the *important* design principles for SVPs is reasonably well established for systems of small dimensionality. Large dimensional systems introduce an additional layer of complexity to the design of effective SVPs. Guidance is needed to navigate through the inevitable design tradeoffs between available data, accuracy constraints, numerical stability, and their impacts on the efficacy of linear/nonlinear solvers at large scale. Once characterized, these design principles will be applied to construct effective SVPs and dense output capabilities for the existing seven-stage scheme: ESDIRK4(3)7L[2]SA.

1.3 The Experimental Strategy

Engineering simulations of the CNSE are nearly always performed at extremely coarse tolerances for both the temporal error and the nonlinear iteration error, owing to the staggering computational costs involved. SVPs used for the CNSE will necessarily need to minimize the errors resulting from the interpolation or extrapolation of stiff polynomial components. These error components are closely related to the stability properties of the SVP. Thus, an essential attribute of the SVP design procedure is a convenient means of assessing the stability properties of a potential design. This work follows the Type II formalism whereby a unique Runge–Kutta scheme is constructed for every stage based on the currently available intrastep and/or

interstep data. The vast knowledge base on designing implicit Runge–Kutta schemes can be brought to bear when constructing the SVP for each stage. The internal order conditions are well known, and the methodology for designing A-, L- and algebraically stable schemes is well-traversed.

Designing an effective intrastep SVP is usually a compromise in an overly constrained design space, particularly in the early stages that suffer an acute shortage of available data. Suffering a limited design parameter space, beyond first-order consistency which SVP constraints should be treated preferentially? Seeking greater clarity from this study, a new eight-stage base scheme: ESDIRK4(3)8L[2]SA is constructed to lend insight into the prioritization of design objectives. The abscissae locations for the eight-stage scheme are necessarily distinct from those of the ESDIRK4(3)7L[2]SA scheme. The new scheme is internal L-stability on stages three through seven, stiffly accurate and has minimal nonlinear instability on all internal stages and the step. Furthermore, the γ is nearly reduced to 1/10 to improve the condition number of the iteration matrix used in Newton’s method. (The conjecture motivating this design is that improved convergence behavior combined with good SVPs will offset the additional work. The results presented herein, substantiate this conjecture. See A for a further discussion.) SVPs are constructed for both schemes and all the aforementioned questions are tested; many are answered.

The paper is organized as follows: Section 2 presents the Butcher formalism for Runge–Kutta schemes and then simplifies the expressions specific to ESDIRK schemes. Section 3 introduces Type II SVPs, including the order conditions and the internal stability functions. Section 4 presents the fundamentals of intrastep SVP design, followed by general design guidelines for predictors of ESDIRK schemes. Section 5 provides the solver mechanics used in the CNSE simulations, including the discrete operator for the spatial terms, the nonlinear and linear iterative solver algorithms, and the time-step controller. Section 6 describes the test problems being used to compare the efficacy of predictors. Section 7 presents the results of an extensive comparison between fifty distinct SVPs constructed for the two aforementioned ESDIRK schemes. A discussion of important findings as well as outstanding questions is included. Section 8 provides context by comparing the two optimized schemes, with other existing ESDIRK schemes ranging in order from two to four. Section 9 provides the conclusions. Two appendices are included. A motivates the utility of an additional state for an ESDIRK scheme. B provides a design summary and the Butcher tableau for the new ESDIRK4(3)8L[2]SA scheme. SVP and dense-output coefficients for both ESDIRK schemes being studied are also provided.

2 The Base Methods

DIRK-type methods are used to solve ODEs of the form

$$\frac{dU}{dt} = F(t, U(t)), \quad U(a) = U_0, \quad t \in [a, b] \quad (2)$$

and are applied over s -stages as

$$\begin{aligned} F_i &= F(t_i, U_i), & U_i &= U^{[n]} + (\Delta t) \sum_{j=1}^s a_{ij} F_j, & t_i &= t^{[n]} + c_i \Delta t, \\ U^{[n+1]} &= U^{[n]} + (\Delta t) \sum_{i=1}^s b_i F_i, & \hat{U}^{[n+1]} &= U^{[n]} + (\Delta t) \sum_{i=1}^s \hat{b}_i F_i. \end{aligned} \quad (3)$$

where $i = 1, 2, \dots, s$, $F_i = F_i^{[n]} = F(U_i, t^{[n]} + c_i \Delta t)$. Also, $\Delta t > 0$ is the step-size, $U^{[n]} \simeq U(t^{[n]})$ is the value of the U -vector at time step n , $U_i = U_i^{[n]} \simeq U(t^{[n]} + c_i \Delta t)$ is the value of the U -vector on the i th-stage, and $U^{[n+1]} \simeq U(t^{[n]} + \Delta t)$. Both $U^{[n]}$ and $U^{[n+1]}$ are of classical order p . The U -vector associated with the embedded scheme, $\hat{U}^{[n+1]}$, is of order $\hat{p} = p - 1$. This constitutes a $(p, \hat{p})$ pair. Each of the respective Runge–Kutta coefficients a_{ij} (stage weights), b_i (scheme weights), $\hat{b}_i$ (embedded scheme weights), and c_i (abscissae or nodes), $i, j = 1, 2, \dots, s$ are real and are constrained, at a minimum, by certain order of accuracy and stability considerations.

For the stiffly-accurate ($a_{sj} = b_j, j = 1, 2, \dots, s$), stage-order two methods considered in this paper, ESDIRK methods are chosen. They are given by the general structure

0	0	0	0	...	0	0	0
2γ	γ	γ	0	...	0	0	0
c_3	a_{31}	a_{32}	γ	$\ddots$	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\vdots$	$\vdots$
c_{s-2}	$a_{s-2,1}$	$a_{s-2,2}$	$a_{s-2,3}$	$\ddots$	γ	0	0
c_{s-1}	$a_{s-1,1}$	$a_{s-1,2}$	$a_{s-1,3}$	...	$a_{s-1,s-2}$	γ	0
1	b_1	b_2	b_3	...	b_{s-2}	b_{s-1}	γ
	b_1	b_2	b_3	...	b_{s-2}	b_{s-1}	γ
	$\hat{b}_1$	$\hat{b}_2$	$\hat{b}_3$	...	$\hat{b}_{s-2}$	$\hat{b}_{s-1}$	$\hat{b}_s$

Some authors prefer to decompose $\mathbf{A} = a_{ij}$, $\mathbf{b} = b_i$ and $\mathbf{c} = c_i$, into

$$\frac{\mathbf{c} \mid \mathbf{A}}{\mathbf{b}^T} = \frac{0 \mid 0 \quad \mathbf{0}^T}{\hat{\mathbf{c}} \mid \mathbf{a} \quad \hat{\mathbf{A}}}{b_1 \mid \hat{\mathbf{b}}^T} \quad \text{with} \quad \mathbf{A} = \begin{bmatrix} 0 & \mathbf{0}^T \\ \mathbf{a} & \hat{\mathbf{A}} \end{bmatrix}, \quad (4)$$

where $\mathbf{0}^T$ (composed of zeros), $\mathbf{a}^T$, $\hat{\mathbf{c}}^T$ and $\hat{\mathbf{b}}^T$ are vectors of length $(s-1)$, $\{0, \mathbf{a}^T\} = a_{\cdot 1}^T$, $\{0, \hat{\mathbf{c}}^T\} = \mathbf{c}^T$, and $\{b_1, \hat{\mathbf{b}}^T\} = \mathbf{b}^T$ and $\hat{\mathbf{A}}$ is a square matrix of dimension $(s-1) \times (s-1)$. Hence, although $\mathbf{A}$ is not invertible, $\hat{\mathbf{A}}$ is often invertible. The motivation for having an explicit first stage is primarily to allow stage-order two methods.

The results presented herein primarily focus on the ESDIRK4(3)7L[2]SA scheme [26], and a newly developed ESDIRK4(3)8L[2]SA. Several secondary methods are also considered: 1) ESDIRK2(1)3L[2]SA [23], 2) ESDIRK3(2)4L[2]SA [23], 3) ESDIRK4(3)6L[2]SA [23], 4) IMEXRK4(3)7L[2]SA₁ [25]. B.2.2 compares the properties of all schemes used in this study. The purpose of the secondary methods is to determine the relative efficiency of all methods. Each method is configured with a rudimentary SVP and all are run using an H321 step-size controller [26] to remove selection bias from the testing procedures.

3 Intrastep, Function-Based Predictors

3.1 Trivial Predictor

The simplest stage predictors for the starting value to stage i are given by [15]

$$U_{i,0} = U^{[n]}, \quad U_{i,0} = U_{i-1}, \quad U_{i,0} = U^{[n]} + (\Delta t)c_i F^{[n]}. \quad (5)$$

The first two are often referred to as the trivial guess or the trivial predictor. The third is the simplest predictor that utilizes both solution and function information, but has proven to provide little benefit relative to the trivial guess. A means of assessing the efficacy of a newly designed SVP is needed. A natural metric is a comparison of the number of iterations required for convergence to some iteration tolerance, between the SVP and the trivial guess. The cost ratio of a proposed SVP might be evaluated as

$$\eta_{\varepsilon_{\text{EC}}, \varepsilon_{\text{IT}}} = \frac{n_{\text{SVP}}}{n_{\text{Trivial}}} \quad (6)$$

where ε_{EC} and ε_{IT} are the error tolerances for the timestep controller and iterative solver, respectively, and n_{SVP} and n_{Trivial} are the number of iterations required to converge the stage. Implicit in this definition is that the optimal predictor will often depend on the target temporal error and the mechanics of the iterative solver as well as the problem being solved.

The order of the predictor is defined as *order* p if p is the largest integer such that

$$\max_{1 \leq i \leq s} \|U_i - U_{i,0}\| = \mathcal{O}(\Delta t)^{p+1}. \quad (7)$$

As such, the *differential component* of the nonlinear equation should converge as $\mathcal{O}(\Delta t)^p$.

Accounting for order-reduction and a stiffness parameter $\epsilon \leq \Delta t$, the SVP is said to have *stiff order* q if $\max_{1 \leq i \leq s} \|U_i - U_{i,0}\| = \mathcal{O}(\Delta t)^{q+1}$, and the accuracy is independent of the stiffness of the problem. The *algebraic stiff-component* of the nonlinear equation should converge as $\mathcal{O}(\Delta t)^q$. The convergence order pair: (p, q) for the SVPs constructed herein is anticipated to be bounded from above by

$$\text{Differential} : (\Delta t)^4 + \epsilon(\Delta t)^3 \quad ; \quad \text{Algebraic} : (\Delta t)^4 + \epsilon(\Delta t)^2.$$

Note that the *design-order* of an SVP: $\|U_i - U_{i,0}\|$ is a relative measure of accuracy and not an absolute measure as would be $\|U_i^{\text{exact}} - U_{i,0}\|$. For example, an SVP for a stage-order two method, could be of design-order four, while only being second-order accurate in an absolute sense.

3.2 SVP Structure and Implementation

3.2.1 Dense-Output SVPs for Stage Two

Dense output may be used to extrapolate function values from the previous step to generate starting guesses for the stage values in the current step. In the dense output formulation, the previous step begins at $t^{[n-1]}$ and ends at $t^{[n]}$ while the current step begins at $t^{[n]}$ and ends at $t^{[n+1]}$. With this, the two step sizes, step-size ratio, $\mathfrak{r}$, and dense-output parameter, θ can be defined as

$$(\Delta t)^{[n]} = t^{[n+1]} - t^{[n]} ; (\Delta t)^{[n-1]} = t^{[n]} - t^{[n-1]} ; \mathfrak{r} = \frac{(\Delta t)^{[n]}}{(\Delta t)^{[n-1]}} ; \theta = 1 + c_i \mathfrak{r}. \quad (8)$$

A third-order dense output is formed from existing derivative data F_i as follows:

$$U_{i,0} = U^{[n-1]} + (\Delta t)^{[n-1]} \sum_{i=1}^s \beta_i^*(\theta) F_i \quad ; \quad \beta_i^*(\theta) = \sum_{j=1}^3 \beta_{ij} \theta^j. \quad (9)$$

The coefficients β_{ij} are provided in B for both ESDIRK schemes. Although equation 9 is valid for arbitrary θ , herein it is used only for predicting $U_{2,0}$, the starting guess at stage two. Dense output methods may be assessed for linear and nonlinear stability properties as well as order of accuracy [24]. A key challenge in designing a good dense-output SVP is that the linear stability function and algebraic stability matrices depend on θ .

3.2.2 SVPs for Stages Three Through s

Intrastep SVPs are designed herein for stages $3 \leq k \leq s$, and use all available stage data from within the current step. Unlike the SVPs advocated by Laburta [27–30] and Hairer, Lubich and Wanner [15], no additional function evaluations are considered. The essential idea of intrastep SVPs is to modify the underlying ESDIRK to create an EDIRK method as the predictor. For a predictor at stage k , we take the $s \times s$, $\mathbf{A}$ -matrix of the ESDIRK and retain the first k rows and columns to form $\tilde{\mathbf{A}}$. We also create $\tilde{\mathbf{c}}$ by taking the first k entries from $\mathbf{c}$. Next, the k th row of $\tilde{\mathbf{A}}$ is replaced with predictor coefficients, β_m , $m = 1, 2, \dots, k-1$. Effectively, a different EDIRK is made for stages three through s . For an SVP at stage k , the ESDIRK (left) and (EDIRK) right take the form

$$\begin{array}{c|cccccc}
 0 & 0 & 0 & \cdots & 0 & 0 \\
 2\gamma & \gamma & \gamma & \cdots & 0 & 0 \\
 c_3 & a_{31} & a_{32} & \ddots & 0 & 0 \\
 \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\
 c_{k-1} & a_{k-1,1} & a_{k-1,2} & \ddots & \gamma & 0 \\
 c_k & a_{k,1} & a_{k,2} & \cdots & a_{k,k-1} & \gamma
 \end{array}
 \quad
 \begin{array}{c|cccccc}
 0 & 0 & 0 & \cdots & 0 & 0 \\
 2\gamma & \gamma & \gamma & \cdots & 0 & 0 \\
 c_3 & a_{31} & a_{32} & \ddots & 0 & 0 \\
 \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\
 c_{k-1} & a_{k-1,1} & a_{k-1,2} & \ddots & \gamma & 0 \\
 c_k & \beta_1 & \beta_2 & \cdots & \beta_{k-1} & 0
 \end{array}$$

The internal stability and accuracy properties of the SVP are evaluated by using $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{c}}$ and treating the SVP as a k -stage EDIRK.

The SVP for stage k is implemented as

$$U_{k,0} = U^{[n]} + (\Delta t) \sum_{j=1}^{k-1} \beta_{kj} F_j, \quad (10)$$

where $U_{k,0}$ is the approximation to U_k .

3.3 Order Conditions

Stage order-conditions are very similar to the order conditions for the step, and are given in Table 1 for the baseline ESDIRK. The order conditions for the EDIRK/SVP are obtained by replacing $\mathbf{A}$ and $\mathbf{c}$ with $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{c}}$, respectively. The *error difference* between the truncation errors of both methods is defined as:

$$\Delta \mathbf{t}_1^{(1)} = \mathbf{t}_1^{(1)} - \tilde{\mathbf{t}}_1^{(1)}, \quad (11)$$

where $\tilde{\mathbf{t}}_1^{(1)}$ denotes an order condition constructed with $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{c}}$. Note the definition for predictor order given in equation 7 is consistent with the definition of $\Delta \mathbf{t}_1^{(1)}$ given by equation 11; i.e., *SVP order of accuracy is a relative quantity at each stage*. Both the *error* and the *error difference* at the i th-stage in a p th-order Runge–Kutta scheme may be quantified in a general way by taking their L_2 principal error norms as follows:

$$A_i^{(p+1)} = \|\mathbf{t}_i^{(p+1)}\|_2 = \sqrt{\sum_{j=1}^{\mathbf{a}_{p+1}} \left(\mathbf{t}_{j,i}^{(p+1)}\right)^2} \quad ; \quad \Delta A_i^{(p+1)} = \|\Delta \mathbf{t}_i^{(p+1)}\|_2 = \sqrt{\sum_{j=1}^{\mathbf{a}_{p+1}} \left(\Delta \mathbf{t}_{j,i}^{(p+1)}\right)^2}. \quad (12)$$

with $\mathbf{a}_{p+1}$ the number of $p+1$ order constraints.

$\mathbf{t}_1^{(1)}$	$\mathbf{A}\mathbf{e} - \frac{\mathbf{c}}{1!}$	$\mathbf{t}_1^{(2)}$	$\mathbf{A}\mathbf{c} - \frac{\mathbf{c}^2}{2!}$
$\mathbf{t}_1^{(3)}$	$\frac{1}{2!}\mathbf{A}\mathbf{c}^2 - \frac{\mathbf{c}^3}{3!}$	$\mathbf{t}_2^{(3)}$	$\mathbf{A}\mathbf{A}\mathbf{c} - \frac{\mathbf{c}^3}{3!}$
$\mathbf{t}_1^{(4)}$	$\frac{1}{3!}\mathbf{A}\mathbf{c}^3 - \frac{\mathbf{c}^4}{4!}$	$\mathbf{t}_2^{(4)}$	$\mathbf{A}\mathbf{C}\mathbf{A}\mathbf{c} - \frac{3\mathbf{c}^4}{4!}$
$\mathbf{t}_3^{(4)}$	$\frac{1}{2!}\mathbf{A}\mathbf{A}\mathbf{c}^2 - \frac{\mathbf{c}^4}{4!}$	$\mathbf{t}_4^{(4)}$	$\mathbf{A}\mathbf{A}\mathbf{A}\mathbf{c} - \frac{\mathbf{c}^4}{4!}$
$\mathbf{t}_1^{(5)}$	$\frac{1}{4!}\mathbf{A}\mathbf{c}^4 - \frac{\mathbf{c}^5}{5!}$	$\mathbf{t}_2^{(5)}$	$\frac{1}{2!}\mathbf{A}\mathbf{C}^2\mathbf{A}\mathbf{c} - \frac{6\mathbf{c}^5}{5!}$
$\mathbf{t}_3^{(5)}$	$\frac{1}{2!}\mathbf{A}(\mathbf{A}\mathbf{c})^2 - \frac{3\mathbf{c}^5}{5!}$	$\mathbf{t}_4^{(5)}$	$\frac{1}{2!}\mathbf{A}\mathbf{C}\mathbf{A}\mathbf{c}^2 - \frac{4\mathbf{c}^5}{5!}$
$\mathbf{t}_5^{(5)}$	$\frac{1}{3!}\mathbf{A}\mathbf{A}\mathbf{c}^3 - \frac{\mathbf{c}^5}{5!}$	$\mathbf{t}_6^{(5)}$	$\mathbf{A}\mathbf{C}\mathbf{A}\mathbf{A}\mathbf{c} - \frac{4\mathbf{c}^5}{5!}$
$\mathbf{t}_7^{(5)}$	$\mathbf{A}\mathbf{A}\mathbf{C}\mathbf{A}\mathbf{c} - \frac{3\mathbf{c}^5}{5!}$	$\mathbf{t}_8^{(5)}$	$\frac{1}{2!}\mathbf{A}\mathbf{A}\mathbf{A}\mathbf{c}^2 - \frac{\mathbf{c}^5}{5!}$
$\mathbf{t}_9^{(5)}$	$\mathbf{A}\mathbf{A}\mathbf{A}\mathbf{A}\mathbf{c} - \frac{\mathbf{c}^5}{5!}$		

Table 1: Stage order-conditions up to fifth-order for Runge–Kutta methods.

3.4 Internal Stability

As the function-based SVPs considered in this paper focus principally on the stages, internal stability of the predictor plays the central role in issues of stability. To determine the vector of internal stabilities of the SVP, one evaluates

$$\begin{aligned}
R_{\text{int}(z)} &= (\mathbf{I} - z\tilde{\mathbf{A}})^{-1}\mathbf{e} = \{R_{\text{int}}^{(1)}(z), R_{\text{int}}^{(2)}(z), \dots, R_{\text{int}}^{(s)}(z)\}^T \\
&= \left\{ \frac{P_{\text{int}}^{(1)}(z)}{Q_{\text{int}}^{(1)}(z)}, \frac{P_{\text{int}}^{(2)}(z)}{Q_{\text{int}}^{(2)}(z)}, \dots, \frac{P_{\text{int}}^{(s)}(z)}{Q_{\text{int}}^{(s)}(z)} \right\}^T.
\end{aligned} \tag{13}$$

Note that the internal stabilities of the baseline is by analogy given by $R_{\text{int}(z)}^b = (\mathbf{I} - z\mathbf{A})^{-1}\mathbf{e}$. In the discussion of linear stability of the internal stages, the respective degrees of $P_{\text{int}}^{(i)}(z)$ and $Q_{\text{int}}^{(i)}(z)$ are important. Define the numerator and denominator polynomial orders of the rational stability polynomials as: $p_n^{(i)} = \deg(P_{\text{int}}^{(i)}(z))$ and $q_n^{(i)} = \deg(Q_{\text{int}}^{(i)}(z))$, respectively. A primary concern for establishing a method's stability is the value of $R_{\text{int}}^{(i)}(-\infty)$; a necessary condition for its boundedness is that $p_n^{(i)} \leq q_n^{(i)}$.

One may also consider the E-polynomial at internal stages to determine stagewise I-stability by using

$$E_{\text{int}}^{(i)}(y) = Q_{\text{int}}^{(i)}(iy)Q_{\text{int}}^{(i)}(-iy) - P_{\text{int}}^{(i)}(iy)P_{\text{int}}^{(i)}(-iy), \tag{14}$$

where $E_{\text{int}}^{(i)}(y) \geq 0$, $y \geq 0$ implies stagewise I-stability.

One could also consider a stagewise analog to the algebraic-stability matrix for irreducible methods where

$$M_{jk}^{(i)} = \tilde{a}_{ij}\tilde{a}_{jk} + \tilde{a}_{ik}\tilde{a}_{kj} - \tilde{a}_{ij}\tilde{a}_{ik} \geq 0, \quad \tilde{a}_{ij} \geq 0, \quad i, j, k = 1, 2, \dots, s, \tag{15}$$

and $M_{jk}^{(i)}$ is the internal algebraic-stability matrix for stage i .

4 Predictor Overview and Design Details

The art of making highly efficient SVPs hinges on determining which EDIRK attributes produce the best predictor, and experience suggests that the hypothetical number of permutations is immense. The number of available free parameters at stage k when designing an intrastep predictor for an ESDIRK scheme, is $k - 1$: one for each previously converged stage. These $k - 1$ degrees of freedom (DoFs) may be applied to enforce order or relative order conditions, internal linear stability requirements, nonlinear stability requirements or any other attribute of the EDIRK that one wishes to control. Navigating the design parameter space requires an answer to the following questions. When should an accuracy constraint be enforced instead of a stability constraint? Is predictor relative order-of-accuracy more important than actual order-of-accuracy? What is the role of imposing different levels of linear stability (A-, L-) and nonlinear stability (algebraic)? Should recent stage data be given preferential weight relative to data extrapolated over great distances? What influence will the abscissae locations of the base scheme have on these design questions? To date, the specific attributes of very good SVPs for large dimensional problems remain an open question. Herein, many predictors are constructed in the interest of using the CNSE to formulate design hypotheses for which design constraints lead to optimality.

The initial stages are seriously lacking available data to enforce desirable constraints. Later stages may have *sufficient* data, but it is not clear which constraints should be enforced or whether linearly-stable predictors even exist. The fortuitous combination of the increased stage order, i.e. two, and the simplifying assumptions used to design the base methods, yields five rather than eight independent order conditions. These are the four bushy tree or quadrature order conditions, $\tilde{t}_1^{(i)}$, $i = 1, 2, 3, 4$, and $\tilde{t}_3^{(4)}$. The order conditions are used to enforce the *error difference* conditions given in equation 11, to force the predictor towards the implicit main method as much as possible.³ Thus, it is theoretically possible to enforce all 4th-order conditions on stages six and later, and in addition impose $L(-\infty) = 0$ on stage seven and later.

There are two constraints that experience suggests should, if possible, be enforced at every stage. The first is $\Delta t_1^{(1)} = \mathbf{0}$ whereby the EDIRK is at least first-order accurate on stage k . A second constraint, $\beta_1 = \beta_2$, is a proxy towards enforcing I-stability via reducing the degree of the $P_{\text{int}}^{(i)}(z)$ polynomial. Serious instability can manifest if the polynomial degree of the numerator, $\deg P_{\text{int}}^{(i)}(z)$, exceeds the polynomial degree of the denominator, $\deg Q_{\text{int}}^{(i)}(z)$, in the internal stability function given in equation 13. Neither A- nor L- stability is possible if $\deg P_{\text{int}}^{(i)}(z) > \deg Q_{\text{int}}^{(i)}(z)$, so it is advisable to enforce $\deg P_{\text{int}}^{(i)}(z) \leq \deg Q_{\text{int}}^{(i)}(z)$. In the context of stage-order two ESDIRKs, enforcing the constraint $\beta_1 = \beta_2$ reduces $\deg P_{\text{int}}^{(i)}(z)$ by one to ensure that minimally, $\deg P_{\text{int}}^{(i)}(z) = \deg Q_{\text{int}}^{(i)}(z)$. Thus beginning with stage three, it is possible to enforce a first-order accurate EDIRK and the constraint $\beta_1 = \beta_2$. If three or more parameters are available, then design decisions become more complex. In general, however, enforcing second-order accuracy: $\Delta t_1^{(1)} = \Delta t_1^{(2)} = 0$, and the proxy I-stability constraint: $\beta_1 = \beta_2$ are nearly always advisable on stages four and greater.

Designing SVPs for stages five and greater begins the quest for optimality from among endless possibilities. Mindful of our stated goal of *understanding the SVP design landscape*, the following strategy

³Although the accuracy conditions are defined in a relative sense, the linear and nonlinear stability attributes are enforced directly on the EDIRK.

is adopted. The important design variables are systematically varied 1) how many and which accuracy constraints, 2) internal A-stability and neutral A-stability, 3) internal $R(-\infty) = 0$, 4) minimizing internal algebraic instability: i.e., maximizing the minimum eigenvalue in $\|\mathcal{M}\|$ with $\mathcal{M}$ the stagewise analogue of the algebraic stability matrix, and 5) keeping the predictor coefficient magnitudes, $\|\beta_j\|$, small. Multiple SVPs are constructed with varying properties for each stage of the two aforementioned ESDIRK schemes. Between five and thirteen individual predictors are constructed on stages six and beyond. Most are designed with what the conventional wisdom would suggest are *desirable properties*. However, some methods are constructed with several desirable properties and one that is suspect (e.g., $\|\mathcal{M}\|$) to measure the impact of that property on the efficacy of the SVP.

One of the more elusive, yet consequential, constraints is that of internal A-stability, which requires a careful traversal of the I-stability polynomial in multi-parameter space. No A-stable SVP is found for ESDIRK4(3)7L[2]SA that is more than second-order accurate (without the assistance of additional function evaluations [28]). Conversely, an internally A-stable and design-order accurate SVP is found for stage seven of the ESDIRK4(3)8L[2]SA scheme, confirming the notion that although identical design objectives are sought, the optimal predictor could be vastly different because of variations in the base scheme. Another illusive constraint is maximizing the minimum eigenvalue in the internal algebraic stability matrix $\|\mathcal{M}\|$, which is a measure of the nonlinear L_2 -stability of the method. Multi-parameter search algorithms using Mathematica [42] are used for this task.

5 Solver Mechanics

5.1 Spatial Discretization

The CNSE are discretized using a classical method of lines approach. The spatial terms of the partial differential equation are discretized using an entropy-stable spectral collocation scheme [9–11], which is provably nonlinearly stable in an integral norm, (provided negative densities are not encountered at shocks). In addition to the physical dissipation, and that generated at boundaries, nonphysical dissipation is added by the approximate Riemann solvers at element interfaces. Conforming element interfaces are used exclusively between elements. Entropy stable solid wall boundary conditions [32] are imposed on the surfaces of the airfoils. Curvilinear hexahedral elements of polynomial order $P1 - P7$ (2nd-order to 8th-order) are used herein.

5.2 Iterative Solver

The nonlinear solver is based on a “Jacobian free” Newton-Krylov approach for each stage of the ESDIRK. The solution to a sequence of linear systems (typically 2-4) is needed for each Newton iteration. Each linear system $Ax = b$ is minimized using GMRES within a right-preconditioned Arnoldi subspace $AP^{-1}PV_n = \bar{V}_{n+1}\bar{H}$, where P is the preconditioner matrix, and V_n is the orthonormal Krylov subspace. The necessary matrix-vector products: $A.w$ are formulated using first-order Frechet derivatives with a perturbation parameter $\epsilon = \text{sqrt}(\epsilon_{machprec})$ taken from Saad [36].

The preconditioner matrix is constructed from an approximate Jacobian of the nonlinear system with all cross-derivative terms removed to decrease the matrix footprint. A high-quality factorization of

P is formed and stored using a dual threshold ILUTP [36] algorithm with 33 (16+1+16) terms on each block-row and a dropping tolerance of $10^{(-3)}$. Two GB/process of RAM is sufficient to store a factorized 32-bit preconditioner for a P7 discretization on 400 hexahedral elements. The factorized preconditioner is reused for 1 – 2 timesteps or approximately 3×10^2 search directions. The extensive reuse of the factorized preconditioner makes negligible its formation and factorization. A full Newton update is applied if it satisfies a threshold criteria based on a realizability constraint and a line search minimization. The line search mechanics is seldom invoked with an effective SVP. All simulations are performed on one node of a Xeon Gold 6148 Skylake cluster having forty cores and 192 GB of RAM. A classical domain-decomposition partitioning: Restricted Additive-Schwarz (0-1 layer overlap) is used to couple the forty computational partitions.

The SVPs are used at the beginning of the nonlinear iteration for each RK stage except stage 2, for which a dense output extrapolation is used. Cumulative timings are recorded for 1) individual stages, 2) Jacobian construction, and 3) preconditioner factorization. The stage timings alone are used to assess the efficacy of the SVPs.

5.3 Controllers

An objective comparison of multiple stiff integrators is frequently difficult. A hands-off approach must be adopted during the comparison, and a temporal error controller provides an ideal means to this end. The embedded error estimator predicts the error and the controller adjusts the timestep to match a specified error tolerance.

6 Test Problems

Two large-scale airfoil problems are used to compare all SVPs and methods. Fully unstructured finite-element method (FEM) grids based on curvilinearly mapped hexahedral elements are used for all simulations. One level of static refinement is performed on a coarse grid to arrive at the final simulation grids. The boundary conditions (BCs) are 1) uniform free stream farfield, 2) periodic flow in the spanwise direction, and 3) entropy-stable solid wall BCs on the surface of the airfoil. A H321 controller [26] is used to adjust step-size based on target errors in L_∞ in the range $10^{(-2)} - 10^{(-5)}$. (An L_∞ error of $10^{(-2)}$ translates into $10^{(-3)}$ variations in the integral quantities of lift and drag.) Nonlinear solver convergence thresholds are set below the target temporal error by factors in the range $10^1 - 10^3$.

A third test case: the Taylor-Green (TG) vortex [41] is included to establish baseline trends for SVPs in the limit of negligible stiffness. The TG vortex is an unsteady canonical test problem characterized by decaying vorticity in a 3D periodic box. A P7 discretization is used on a 16^3 uniformly distributed cube of elements for a total of approximately 10^6 DoFs. A stability bound timestep of $3 \times 10^{(-3)}$ is observed for a fourth-order explicit Runge-Kutta scheme (ERK4), which generates a temporal error of $\mathcal{O}(10^{(-10)})$ per timestep. Typically, implicit methods are ill-advised for this problem owing to the lack of stiffness. (A fourth-order implicit method could potentially run at timesteps $\mathcal{O}(10)$ larger before temporal error becomes problematic.) Nevertheless, it provides a meaningful test for off-body regions in airfoil simulations discretized with nearly isotropic (nonstiff) elements. A desirable requirement in SVP design is performance across the stiffness spectrum, and certainly to *do no harm* in the nonstiff limit.

6.1 SD7003 airfoil

The flow around the SD7003 airfoil at an angle of attack of $\alpha = 8^\circ$, $Re_{chord} = 60000$ and $Mach = 0.2$ is considered. [1, 39] Snapshot visualizations of the (a) vorticity magnitude and (b) *spatial* error estimate are shown in figure 1, taken from a typical P5 simulation. The flow exhibits a laminar separation bubble on the top surface with an oscillating reattachment point. Transition to turbulence takes place in the free shear layer via a Kelvin-Helmholtz instability mechanism. The grids in this study ranged in size from $10^4 - 5 \times 10^4$ elements. Simulation times are on the interval $40.0 \leq T \leq 40.25$ corresponding to vortices traversing about a quarter the distance of one chord.

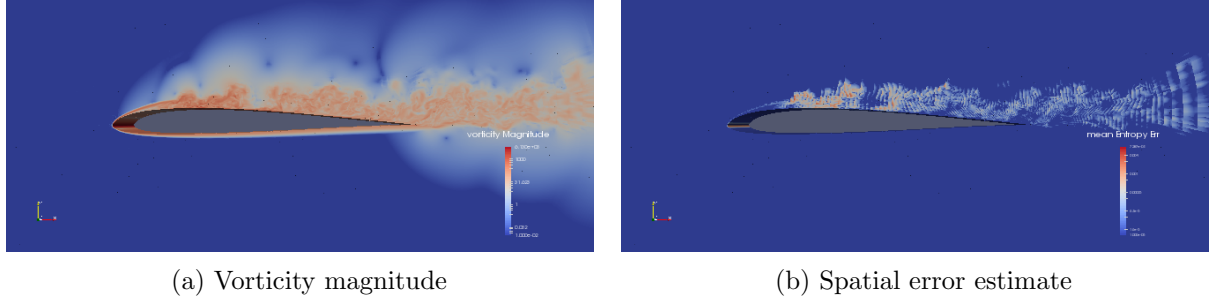


Figure 1: SD7003 Airfoil simulation

6.2 Multi-element Airfoil (MD-30P30N)

The flow around the MD-30P30N multielement airfoil at an angle of attack of $\alpha = 8^\circ$, $Re_{chord} = 60000$ and $Mach = 0.1$ is considered. Visualizations of the (a) vorticity magnitude and (b) *temporal* error estimate are shown in figure 2. Both a coarse: (31251) and a fine: (63004) hexahedral FEM grid are used in the study. Simulation times are on the interval $30.0 \leq T \leq 30.1$ corresponding to vortices traversing about a 10% the distance of one chord.

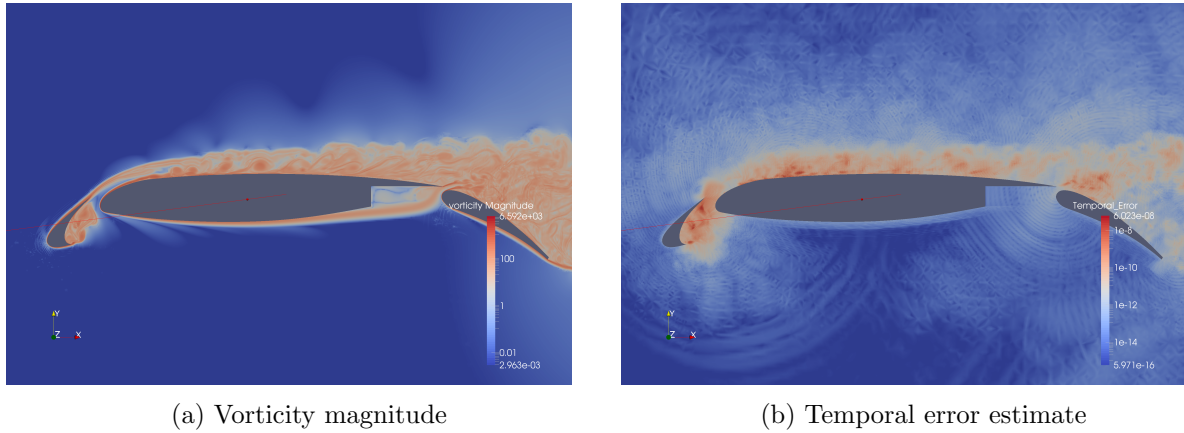


Figure 2: MD-30P30N Airfoil simulation

6.3 Stiffness

Table 2 summarizes the degrees-of-freedom (DoFs) and the stability bound explicit timestep (ERK4) as functions of polynomial order for the two airfoil test cases. Hexahedral FEM meshes with 11344 and 31251 elements are used for the SD7003 airfoil and MD-30P30N multi-element airfoil, respectively. Polynomial orders $P3 - P5$ are used in the comparison studies.

A common metric of stiffness: $\frac{\Delta t_{physics}}{\Delta t_{explicitCFL}}$, quantifies the ratio between the 1) *timestep required to capture the physics of interest*, and 2) *timestep dictated by an explicit stability constraint*. The stiffness of each airfoil is provided in equation 16, with the $\Delta t_{physics}$ assigned a representative value based on simulations targeting a temporal error of 10^{-3} /timestep, and the explicit limit taken from table 2.

$$\text{SD7003} : \frac{3.5 \times 10^{(-3)}}{3.3 \times 10^{(-7)}} \approx 10^4 \quad ; \quad \text{MD-30P30N} : \frac{2.5 \times 10^{(-4)}}{1.6 \times 10^{(-6)}} \approx 10^2 \quad ; \quad \text{TG-vortex} : \frac{3.0 \times 10^{(-2)}}{3.0 \times 10^{(-3)}} \approx 10^1. \quad (16)$$

Note the large difference in timesteps needed to accurately resolve integral quantities (SD7003), and acoustic phenomena (MD-30P30N). (The acoustic signatures captured by the temporal error estimator (figure 2b) in the slat and cove regions of the MD-30P30N airfoil require both a smaller timestep and a stricter tolerance on nonlinear iteration error, and thus decrease the effective stiffness of this test case.)

poly	SD7003 Airfoil		30P30N multielement	
	DoFs	dt_{explct}	DoFs	dt_{explct}
1	0.45×10^6	$1.4 \times 10^{(-5)}$	0.12×10^7	$5.6 \times 10^{(-5)}$
2	1.50×10^6	$3.0 \times 10^{(-6)}$	0.42×10^7	$1.5 \times 10^{(-5)}$
3	3.60×10^6	$1.0 \times 10^{(-6)}$	1.00×10^7	$4.8 \times 10^{(-6)}$
4	7.09×10^6	$5.8 \times 10^{(-7)}$	1.95×10^7	$2.5 \times 10^{(-6)}$
5	12.2×10^6	$3.3 \times 10^{(-7)}$	3.37×10^7	$1.6 \times 10^{(-6)}$
6	19.4×10^6	$1.8 \times 10^{(-7)}$	5.35×10^7	$7.5 \times 10^{(-7)}$

Table 2: Grid stiffness and DoFs for the SD7003 and 30P30N FEM meshes.

7 Results

7.1 Prologue

Figures 3, 4, 5, and 6, present comparative timing studies of the SVPs constructed for the ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA schemes. Figures 3 and 4 present *extensive* results from simulations of the SD7003 airfoil test case, using the ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA schemes, respectively. Figures 5, and 6 present a *representative* subset of results from simulations of the MD-30P30N multi-element airfoil, and the TG-vortex test cases, respectively. Only simulation results from stages 6 and 8 using the ESDIRK4(3)8L[2]SA scheme are presented in the later two figures. Similar conclusions could be drawn from all stages of both ESDIRK schemes in the later two test cases.

The SD7003 airfoil test case (the most stiff of the three cases), proved to best distinguish the performance of competing design attributes for all SVPs. Furthermore, multiple discretizations produced

similar results, i.e., grid densities ranging from 11344 to 31292 elements, discretized with polynomial orders from $P3 - P5$. Thus, for reasons of efficiency, the most comprehensive comparisons presented herein are for simulations of the SD7003 airfoil using 11344 curvilinear hexahedral elements and a $P3$ discretization, with results primarily presented in figures 3 and 4. (Note, however, that subfigures 3e and 3f, present comparisons of stage seven results discretized using $P3$ and $P5$ elements, respectively.)

The results presented in figures 3 - 6 may be interpreted as follows. The method number of the SVP is given on the abscissa of all four figures, while the CPU timing of the simulation is given on the ordinate. A comparison of relative stage timings is facilitated by using identical time scales on all plots (save 3f). The time required for the trivial guess is always presented as *method 0*. Data for distinct *controller error targets*: dt_e and *nonlinear iteration tolerances*: it_e are included on each plot. The preferential data used to assign the relative efficacy of SVPs always corresponds to the coarsest error tolerance used in the study, e.g., $dt_e:1e-2$, $it_e:1e-5$ in the $P3$ results presented in figures 3 and 4. This coarse temporal target error is typical for many engineering simulations. (One could argue these tolerances are too coarse, but answering this question is beyond the scope of this work.)

Tables 3 and 4 summarize the properties of all SVPs designed for the ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA schemes, respectively. The column labeled *Design* provides the design objectives used to build the SVP. A number (e.g., 1,2,3) denotes enforcing/minimizing the order conditions $\Delta t_1^{(1)}, \Delta t_1^{(2)}, \Delta t_1^{(3)}$, respectively. The designations (e.g. 4a,4b,4ab) denotes enforcing/minimizing the fourth-order conditions: $\Delta t_1^{(4)}$, or $\Delta t_2^{(4)}$, or both conditions: $\Delta t_1^{(4)}, \Delta t_2^{(4)}$. Stability properties are denoted with A , L , and M , corresponding to A-, L-stability and algebraic instability: $\|\mathcal{M}\|$, respectively. The designation B denotes minimization of the magnitude $\|\beta_j\|$. The designation N denotes a strictly A-stable predictor. The columns labeled A_{stab} and L_{stab} denote whether the scheme is A-stable and L-stable, respectively. A lowercase n in the L_{stab} column denotes an SVP that is nearly L-stable. The columns labeled $R(-\infty)$, $\|\mathcal{M}\|$ and $\|\beta_j\|$ quantify the amplification at $-\infty$, the minimum eigenvalue in the algebraic stability matrix: $\|\mathcal{M}\|$, and the coefficient magnitude of the SVP, respectively. The columns labeled $\Delta A^{(j)}$ provide the *relative* leading order truncation term for the SVPs at order j . The SVPs that perform well are highlighted in blue for each stage, while those that underperform are highlighted in red. Coefficients for the best performing SVP on each stage for both methods are included in B along with their corresponding number.

An extensive characterization of the data presented in figures 3, 4 is now presented, beginning first with the simulations performed on the SD7003 airfoil with the ESDIRK4(3)7L[2]SA scheme. The optimization strategies used to design the SVPs on each stage is first overviewed, followed by a brief assessment of the performance of the SVPs. Overall trends are then summarized.

7.2 ESDIRK4(3)7L[2]SA: H11344 on SD7003 airfoil

7.2.1 Stage 2

Stage two has no available data to build an intrastep SVP. Thus, the dense-output polynomial from the previous step is used to initialize the nonlinear iteration.

7.2.2 Stage 3

Stage three has two available coefficients β_j that multiply the nonlinear fluxes F_j , $j = 1, 2$, respectively. There are no design choices that are available for this SVP and only one SVP is constructed. Enforcing the

$\Delta t_1^{(1)}$ constraint and $\beta_1 = \beta_2$ yields a first-order method that is A-stable, has nominal damping at $R(-\infty)$ and has a very desirable algebraic stability matrix M . It significantly outperforms the trivial guess and is among the best performers for any SVP on any stage. These desirable properties are likely the consequence of the relative proximity of the abscissa: $c(1) < c(3) < c(2)$, and that stage two is second-order accurate and strictly A-stable (i.e., trapezoidal rule). This SVP does not suffer from first-order accuracy, implying that the favorable stability properties of the SVP contribute to the efficacy of the starting procedure.

7.2.3 Stage 4

Stage four has three available coefficients β_j for SVP design. Enforcing $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$ results in a second-order method that is remarkably strictly-A-stable; the I-stability polynomial becomes identically zero! The algebraic stability matrix is of modest conditioning. This strictly stable SVP is quite anomalous and is observed in only one other circumstance in this study. Tests indicate that this SVP significantly outperforms the trivial guess, and is among the best performers for any SVP on any stage.

7.2.4 Stage 5

Stage five has four available coefficients β_j available for SVP design. Three SVPs are constructed with distinct design objectives. All methods enforce $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Method one enforces $\Delta t_1^{(3)}$ as the final constraint. The resulting method is not A-stable, has poor $R(-\infty)$ damping and large algebraic instability: $\|\mathcal{M}\|$. The second method enforces A-stability and minimizes the accuracy constraint $\Delta t_1^{(3)}$, but has large algebraic instability: $\|\mathcal{M}\|$. The third method enforces A-stability and minimizes the value of $R(-\infty)$. It has a modest level of algebraic instability: $\|\mathcal{M}\|$. Predictor 3 performs the best of the three. Enforcing A-stability and additional stability properties are the best use for the last free design space parameter, i.e., enforcing stability is more important than high stage order.

7.2.5 Stage 6

Stage six has five available coefficients β_j available for SVP design, and the complexity of building an SVP expands considerably. Thirteen SVPs are constructed with distinct design objectives. All SVPs enforce the first two accuracy conditions and the proxy I-stability constraint: $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Two free parameters remain unspecified and are used to construct groups of *accuracy* and *stability* SVPs. Predictors 1-3 and 12-13 enforce the third-order constraint: $\Delta t_1^{(3)}$, while SVP 1 enforces the first fourth-order constraint: $\Delta t_1^{(4)}$. No A-stable method is identified that is third-order accurate at this stage. Predictors 4-11 are A-stable with 7-8 also L-stable. Predictors 10-11 attempt to simultaneously approximate L-stability while minimizing $\Delta t_1^{(3)}$. Predictors 4 and 6 attempt to minimize the $\Delta t_1^{(3)}$ consistency constraint. Predictor 5 is A-stable but is designed with a large negative algebraic stability eigenvalue.

The performances of four SVPs are distinguished: 3, 6, 7, 10. Predictor 3 is third-order consistent but has poor stability properties relative to the others in the group. It is not stable at $R(-\infty)$ and has large algebraic instability: $\|\mathcal{M}\|$. Predictor 3 is an anomaly in that it has reasonably poor properties and yet performs well. Its only virtue is third-order accuracy. Predictors 6, 7, 10, are all A-stable with 7 and 10 being nearly L-stable. Two pairs of SVPs were designed with similar objectives: 7,8 and 10,11. Note

that based on the performance metrics listed in table 3, SVP 8 should have performed better than 7 and that 11 should have performed better than 10, but neither did.

The poor performers are SVP 2 and SVP 5. Both are on par with the trivial guess. SVP 2 is third-order accurate but is not A-stable, while SVP 5 is A-stable. Both 2,5 have a very large algebraic instability metric: $\|\mathcal{M}\|$. Note that the magnitude of the coefficient norm $\|b\|$ is highly correlated with the algebraic instability metric: $\|\mathcal{M}\|$. Also, note the wide variations in $\|\mathcal{M}\|$ when comparing the first 6 stages. While it is appropriate to compare algebraic instability metrics within a common stage, comparisons between different stages are less meaningful.

7.2.6 Stage 7

Stage seven has six available coefficients β_j available for SVP design, and again the complexity of building an SVP expands accordingly. Eleven SVPs are constructed with distinct design objectives. All SVPs enforce the first two accuracy conditions and the proxy I-stability constraint: $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Three free parameters remain unspecified and are used to construct groups of *accuracy* and *stability* SVPs. Predictors 1-4 and 10,11 enforce the third-order constraint: $\Delta t_1^{(3)} = 0$. Predictor 1 enforces both fourth-order constraints $\Delta t_1^{(4)} = \Delta t_2^{(4)} = 0$. Predictors 2 and 3 enforce the fourth-order constraint $\Delta t_1^{(4)} = 0$ and $\Delta t_2^{(4)} = 0$, respectively and use the remaining parameter to optimize the algebraic instability: $\|\mathcal{M}\|$. Predictor 4 uses the two remaining parameters to minimize algebraic instability: $\|\mathcal{M}\|$. No A-stable method is identified that is third-order accurate for stage seven. Predictors 5-7 and 9 enforce A-stability with 9 also enforcing L-stability and minimizing algebraic instability: $\|\mathcal{M}\|$. Predictors 5-7 are strictly A-stable at $-\infty$, although the signs of their amplification is opposite: $R(\infty) = \pm 1$.

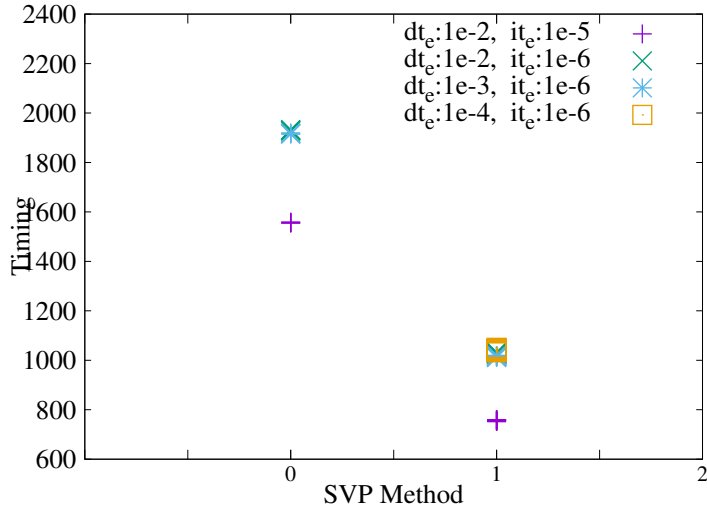
The best performers are SVPs 3, 4, 6, 7, 9. Predictors 3, 4 are both third-order accurate and yet their performance is remarkably good. Enforcing the fourth-order accuracy constraint: $\Delta t_2^{(4)} = 0$, in SVP 3, enables a very small fourth-order error term, but leads to very poor algebraic instability. Predictor 4 minimizes the algebraic instability: $\|\mathcal{M}\|$. Predictors 6, 7 are A-stable with 7 nearly L-stable. The algebraic instability: $\|\mathcal{M}\|$ is quite large when compared with the minimum value found for SVP 4. SVP 9 is A-stable, L-stable, has very small algebraic instability $\|\mathcal{M}\|$, and enforces $\beta_1 = \beta_2 = 0$. The leading order truncation error term, however, is the largest of all SVPs for this stage. Predictors 3, 4 and 7, 9 are optimal for stiff and marginally stiff simulations, respectively.

The poor performers are SVPs 1, 5, 8, 10, 11. Predictors 1, 10, 11, are all third-order methods and all have excessive algebraic instability. Enforcing $R(-\infty) = 0$ for SVP 11 does not improve its performance. Methods 8, attempts to minimize algebraic instability, has $R(-\infty) = 1.022$, and a modest algebraic instability: $\|\mathcal{M}\|$, but still performs poorly. Minimizing algebraic instability alone does not appear to be sufficient to guarantee good performance. Among all poor performers, only SVP 5 is A-stable. It has $R(-\infty) = 1$ and a modest algebraic instability: $\|\mathcal{M}\|$. It is not apparent why this scheme underperforms.

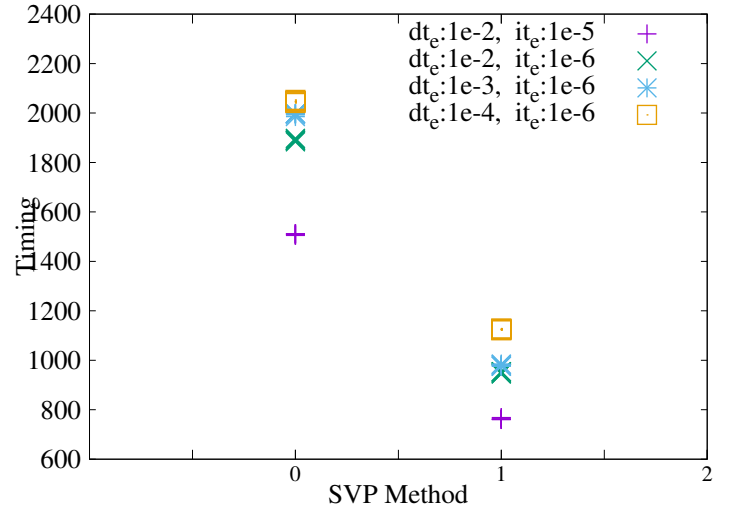
7.3 ESDIRK4(3)8L[2]SA: H11344 on SD7003 airfoil

7.3.1 Stage two

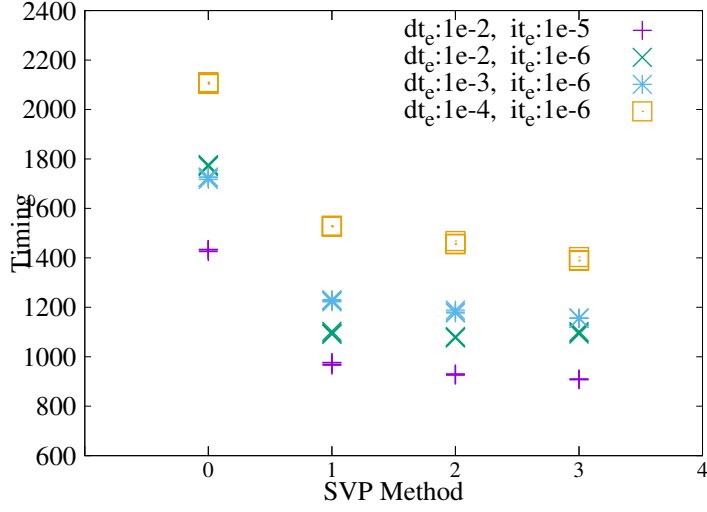
The dense-output polynomial from the previous step is used to initialize the nonlinear iteration.



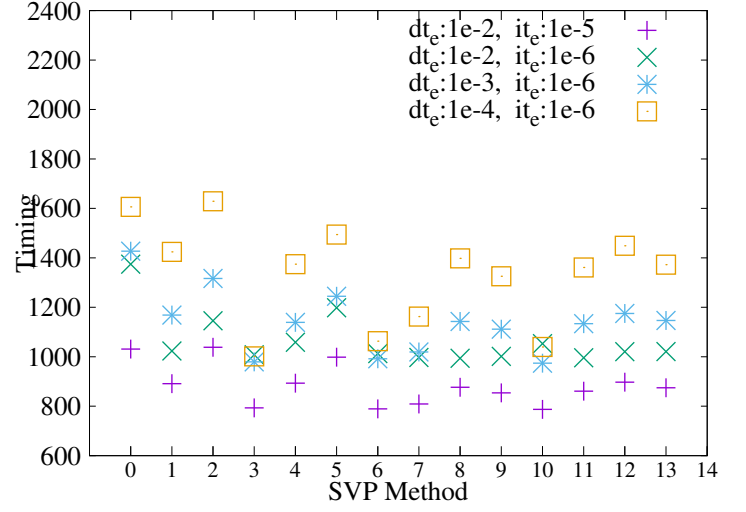
(a) P3: Stage 3



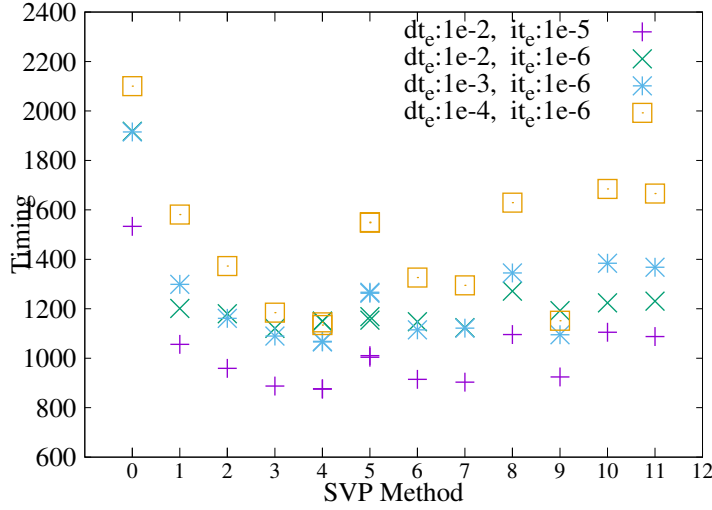
(b) P3: Stage 4



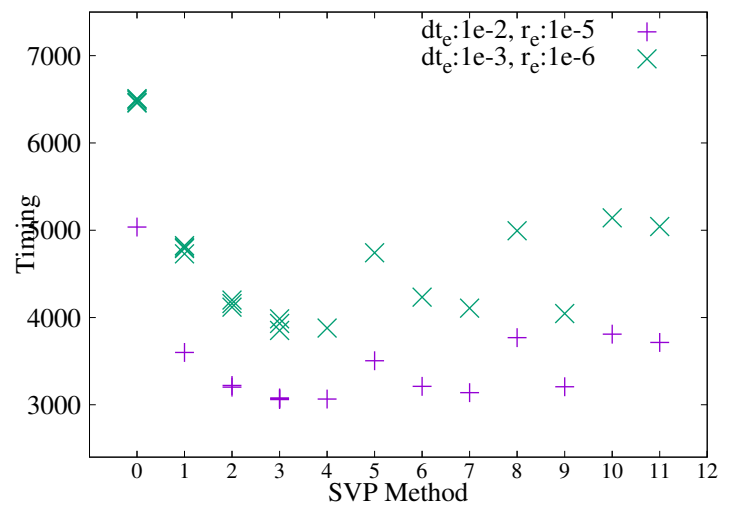
(c) P3: Stage 5



(d) P3: Stage 6



(e) P3: Stage 7



(f) P5: Stage 7

Figure 3: ESDIRK4(3)7L[2]SA simulations of the SD7003 airfoil

SVP	Design	A_{stab}	L_{stab}	$R(-\infty)$	$-\ \mathcal{M}\ $	$\ \beta\ $	$\ \Delta A^{(2)}\ $	$\ \Delta A^{(3)}\ $	$\ \Delta A^{(4)}\ $
3.1	1	Y	N	0.4142	-0.0023	0.30	0.0065	0.0027	0.0043
SVP	Design	A_{stab}	L_{stab}	$R(-\infty)$	$-\ \mathcal{M}\ $	$\ \beta\ $	$\ \Delta A^{(3)}\ $	$\ \Delta A^{(4)}\ $	$\ \Delta A^{(5)}\ $
4.1	A N	Y	N	1.0	-3.0261	0.85	0.0027	0.0043	0.0023
5.1	3	N	N	-2.829	-31.613	2.53	0.0	0.0062	0.0019
5.2	A 3	Y	N	-0.992	-12.217	1.58	0.0051	0.0032	0.0033
5.3	A L	Y	n	-0.056	-5.891	1.09	0.0077	0.0066	0.0053
6.1	3 4	N	N	-0.004	-0.354	0.162	0.0	0.0011	0.0013
6.2	3 M	N	N	+3.137	-28.04	2.156	0.0	0.0068	0.0025
6.3	3 B	N	N	-1.601	-5.591	0.987	0.0	0.0038	0.0016
6.4	A 3	Y	N	1.000	-0.739	0.318	0.0067	0.0082	0.0061
6.5	A M	Y	N	-1.000	-18.256	1.754	0.0170	0.0150	0.0130
6.6	A 3	Y	N	-0.997	-4.867	0.918	0.0049	0.0042	0.0044
6.7	A L	Y	Y	0.0	-0.603	0.333	0.0058	0.0058	0.0051
6.8	A L	Y	Y	0.0	-0.244	0.142	0.0016	0.0022	0.0023
6.9	A L	Y	N	-0.202	-0.255	0.210	0.0016	0.0021	0.0022
6.10	A L	Y	n	10^{-10}	-2.007	0.587	0.0098	0.0093	0.0079
6.11	A L	Y	n	10^{-6}	-0.234	0.147	0.0020	0.0025	0.0025
6.12	3 L	N	N	0.0	-0.357	0.164	0.0	0.0011	0.0013
6.13	3 L	N	N	0.274	-0.241	0.177	0.0	0.0013	0.0013
7.1	3 4ab	N	N	-1.617	-62.54	3.008	0.0	0.0	0.0097
7.2	3 4a	N	N	-3.837	-130.66	4.336	0.0	0.0048	0.0082
7.3	3 4b	N	N	-1.618	-62.91	3.02	0.0	0.0003	0.0098
7.4	3 M	N	N	-2.331	-1.64	0.524	0.0	0.0110	0.0120
7.5	A	Y	N	1.000	-5.51	0.937	0.0074	0.0073	0.0094
7.6	A	Y	N	-1.000	-26.24	1.957	0.0095	0.0110	0.0010
7.7	A L	Y	n	10^{-7}	-12.96	1.389	0.0092	0.0097	0.0099
7.8	M	N	N	1.022	-0.73	0.340	0.0190	0.0220	0.0200
7.9	A L M	Y	Y	0.0	-0.85	0.358	0.0290	0.0290	0.0220
7.10	3	N	N	1.800	-6.00	0.990	0.0	0.0075	0.0120
7.11	3	N	N	0.0	-28.68	2.050	0.0	0.0035	0.0120

Table 3: SVP properties for stages three to seven.

7.3.2 Stage three

The design scenario for stage three of ESDIRK4(3)8SA is identical to that described above for ESDIRK4(3)7SA. Stage three has two available coefficients $\beta_j, j = 1, 2$ and one SVP is designed. It enforces the first-order constraint $\Delta t_1^{(1)} = 0$ and the proxy I-stability condition $\beta_1 = \beta_2$. The abscissa for this stage satisfies $c(1) < c(3) < c(2)$, and is likely the reason why the method has desirable properties despite being only first-order accurate. Note that the following properties are identical between the two ESDIRK schemes considered herein: $A_{stab}, L_{stab}, R(-\infty), \|\mathcal{M}\|$. This results from enforcing identical design constraints (albeit with different γ) on stages two and three in the main methods.

7.3.3 Stage four

Stage four has three available coefficients and two SVPs are designed. The first enforces $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. The resulting scheme is not A-stable but does have modest instability for both $R(-\infty)$ and $\|\mathcal{M}\|$. The second SVP only enforces condition $\Delta t_1^{(1)} = \beta_1 - \beta_2 = 0$. No A-stable schemes exist in the available one-parameter space. The final constraint is used to enforce zero amplification $R(-\infty) = 0$. Both SVPs outperform the trivial guess. Since neither SVP is A-stable, it is not surprising, that SVP 1 is the better of the two.

7.3.4 Stage five

Stage five has four available coefficients β_j available for SVP design. Two SVPs are constructed with both methods enforcing $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Method one enforces $\Delta t_1^{(3)} = 0$ as the final constraint. The resulting method is not A-stable, has poor $R(-\infty)$ damping and large algebraic instability: $\|\mathcal{M}\|$. Methods that satisfy the I-stability constraint do exist in a narrow interval of the one remaining parameter. The second method enforces A-stability and the remaining constraint is used to minimize the value of $R(-\infty)$. Predictor 2 has good algebraic instability: $\|\mathcal{M}\|$ and is noticeably better than SVP 1. Enforcing A-stability and additional stability properties are a better use for the last free design parameter β_j , i.e., stability is more important than high stage order.

7.3.5 Stage 6

Stage six has five available coefficients β_j available for SVP design. Seven SVPs are constructed and all enforce the first two accuracy and the proxy I-stability constraints: $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Two free parameters remain unspecified. Predictors 1-4 enforce $\Delta t_1^{(3)} = 0$, while SVPs 1-2 enforce the $\Delta t_1^{(4)} = 0$ and $\Delta t_2^{(4)} = 0$, respectively. Predictor 2 has large algebraic instability: $\|\mathcal{M}\|$. Predictor 3 minimizes the algebraic instability: $\|\mathcal{M}\|$, while SVP 4 sets the condition $R(-\infty) = 0$. No A-stable method is identified that is third-order accurate for stage six. Predictors 5-7 enforce A-stability and with the remaining parameter minimize $R(-\infty)$, $\Delta t_1^{(3)}$ and algebraic instability: $\|\mathcal{M}\|$, respectively.

Both of the best performers are A-stable: SVPs 5, 6. Predictor 5 is nearly L-stable, although it has a large algebraic stability coefficient. Predictor 6 is a compromise between minimizing $R(-\infty)$ and $\|\mathcal{M}\|$. The poor performers are SVPs 2, 3, and both are very unstable at $R(-\infty)$. In addition, SVP 3 has very poor algebraic instability: $\|\mathcal{M}\|$. Together these attributes no doubt contribute to the poor performance

of SVP 3. Enforcing A-stability and additional stability properties are the best design choices for the free parameters. Again, note the correlation between $\|\beta_j\|$ and the norm of the algebraic instability: $\|\mathcal{M}\|$.

7.3.6 Stage 7

Stage seven has six available coefficients β_j available for SVP design. Three SVPs are constructed and all enforce the first three accuracy and the proxy I-stability constraints:

$\Delta t_1^{(1)} = \Delta t_1^{(2)} = \Delta t_1^{(3)} = \beta_1 - \beta_2 = 0$. Two free parameters remain unspecified. Predictor 1 enforces both available parameters with the accuracy constraints: $\Delta t_1^{(4)} = \Delta t_2^{(4)} = 0$. Astonishingly, *the predictor is A-stable, nearly L-stable and fourth-order accurate*. Predictor 2 enforces A- and L-stability while simultaneously minimizing $\Delta t_1^{(4)}$ and $\Delta t_2^{(4)}$. Predictor 3 enforces A- and L-stability and minimizes the norm of the algebraic instability: $\|\mathcal{M}\|$. All perform well as SVPs with the best being SVPs 1 or 2. These SVPs are the only ones designed herein that are A- and L- stable and of order three or higher. (The third-order barrier can be eclipsed with additional function evaluations as suggested by Laburta [27].) The abscissa for this stage satisfy: $c(5) < c(7) < c(6)$, so the SVP is an interpolation rather than extrapolation. This is the likely reason for the superior performance of all SVPs at this stage.

7.3.7 Stage 8

Stage eight has seven available coefficients β_j available for SVP design. Six SVPs are constructed and all enforce the first two accuracy and the proxy I-stability constraints: $\Delta t_1^{(1)} = \Delta t_1^{(2)} = \beta_1 - \beta_2 = 0$. Four free parameters remain unspecified. Predictor 1 enforces $\Delta t_1^{(3)} = 0$, $R(-\infty) = 0$, and uses the remaining parameters to minimize $\|\mathcal{M}\|$. No A-stable predictor satisfying $\Delta t_1^{(3)} = 0$ is identified despite there being three available parameters. Predictor 2 enforces A-stability and once again is strictly A-stable; the I-stability polynomial vanishes. Predictor 3 minimizes the algebraic instability: $\|\mathcal{M}\|$ but is not A-stable. Predictors 4, 5 enforce A- and L-stability but exhibit excessively large algebraic instability: $\|\mathcal{M}\|$. Predictor 6 enforces all third- and fourth-order accuracy constraints.

The best-performing SVPs are both A-stable: 2, 4. SVP 2 is strictly A- stable: the I-stability polynomial vanishes. SVP 4 is L-stable. Both 2 and 4 suffer from huge algebraic stability coefficients. The worst performer is SVP 6 which is fourth-order accurate but has very poor stability at $R(-\infty)$.

7.3.8 Summaries: ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA

Twenty-nine and twenty-one individual SVPs are constructed for the ESDIRK4(3)7L[2]SA and ESDIRK4(3)8L[2]SA schemes, respectively. *Nearly all SVPs outperform or significantly outperform the trivial guess on each stage for both methods*. Groups of equally performant SVPs emerged on the latter stages e.g., 6, 7 for ESDIRK4(3)7L[2]SA and 6, 8 for ESDIRK4(3)8L[2]SA. Nevertheless, both schemes have at least one performant SVPs on every stage that is 1) A-stable, and fully or partially 2) L-stable. (Stage four is an exception, not having enough parameters to enforce all stability conditions.)

The algebraic instability: $\|\mathcal{M}\|$ metric proves useful in some instances, but in general does not provide significant insight into SVP design. The best performing SVPs all have reasonable algebraic instability compared with other methods within their stage, but are typically not the least unstable. Methods with extremely large algebraic instability: $\|\mathcal{M}\|$ are seldom competitive. Conversely, designing

solely for $\|\mathcal{M}\|$ seldom yields a performant SVP. (Comparing algebraic instability between the two schemes is deceptive because ESDIRK4(3)8L[2]SA has significantly larger values on most stages and over the step.) The $\|\mathcal{M}\|$ correlates well with the norm $\|\beta_j\|$ on all stages of both schemes. This suggests that a simple approach to minimize $\|\mathcal{M}\|$ is to minimize $\|\beta_j\|$. Constraining the remaining free parameters with $\|\mathcal{M}\|$ is advisable provided that it is not the sole design metric. No truly algebraically stable SVPs are encountered during this design study, which is not surprising given that ESDIRK schemes can not be algebraically stable.

Anomalous results appear on stages 6 and 7 of the ESDIRK4(3)7L[2]SA scheme. Some of the SVPs that enforce third-order accuracy $\Delta t_1^{(3)} = 0$ perform quite well: e.g., stages 6: SVP 3, and stage seven: SVPs 3, 4. The majority of schemes that enforce $\Delta t_1^{(3)} = 0$ and especially $\Delta t_1^{(4)} = \Delta t_2^{(4)} = 0$, perform poorly relative to the performant A- and L-stable schemes. The unpredictability of enforcing the $\Delta t_1^{(3)} = 0$ constraint makes it considerably more difficult to design effective SVPs. These ambiguous results related to enforcing $\Delta t_1^{(3)} = 0$ suggests that another important SVPs metric has yet to be identified.

Two SVPs are constructed that are strictly A-stable: SVPs 4.1 on ESDIRK4(3)7L[2]SA and 8.2 on ESDIRK4(3)8L[2]SA. All other stages in either scheme do not support strict A-stability. Neither of the SVPs have small values of algebraic instability: $\|\mathcal{M}\|$. The fact that both SVPs perform well is surprising given the previously noted importance of full or partial L-stability.

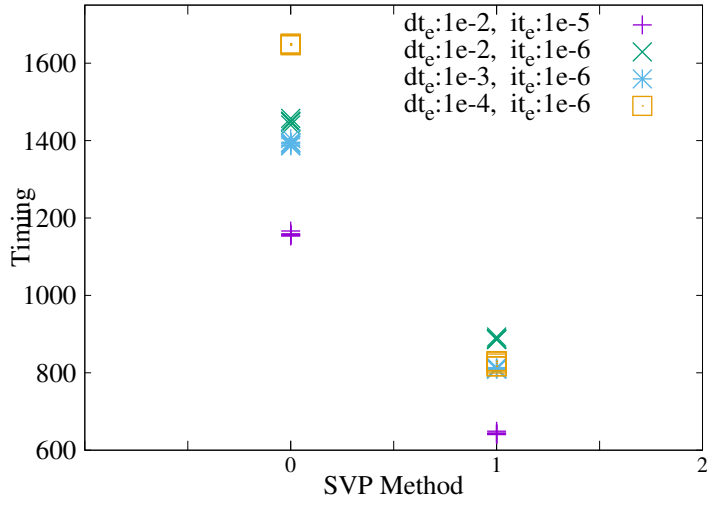
Another anomalous result appears on stage seven of the ESDIRK4(3)8L[2]SA scheme. Predictor 7.1 is nearly an ideal SVP. It is fourth-order accurate, A- and nearly L-stable and performs extremely well in all tests at all error tolerances. It routinely outperforms the trivial guess by a factor $2x$. Establishing the existence of such an SVP is important, as it provides a benchmark against which all other SVPs can be judged. Predictor designs that approach the $2x$ factor are at the point of diminishing returns.

7.4 ESDIRK4(3)8L[2]SA: H63004 on MD-30P30N ME-airfoil

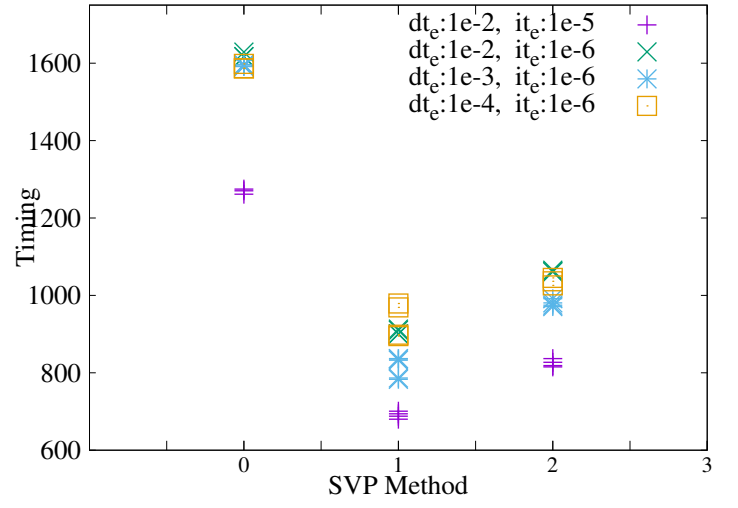
Figure 5 presents a representative subset of results comparing the efficacy of SVPs on simulations of the MD-30P30N multi-element airfoil. The ESDIRK4(3)8L[2]SA scheme is used for these simulations. (Additional simulation results for the MD-30P30N multi-element airfoil simulated with both temporal schemes are available but provide no additional information.) Recall that this airfoil was only modestly stiff (e.g., $\mathcal{O}(10^2)$). The performance of all SVPs on both stages is nearly indistinguishable at the coarse and medium error tolerances. Only at the finest tolerance is there significant deviation in performance and the third-order schemes are the best performers. Note that all SVPs decisively outperform the trivial guess by at least $1.6X$ with increasing benefits achieved at stricter error tolerances (less stiffness). One could argue that this problem doesn't warrant the use of SVPs designed with stability attributes appropriate for problems with significant stiffness.

7.5 ESDIRK4(3)8L[2]SA: H4096 on TG-Vortex

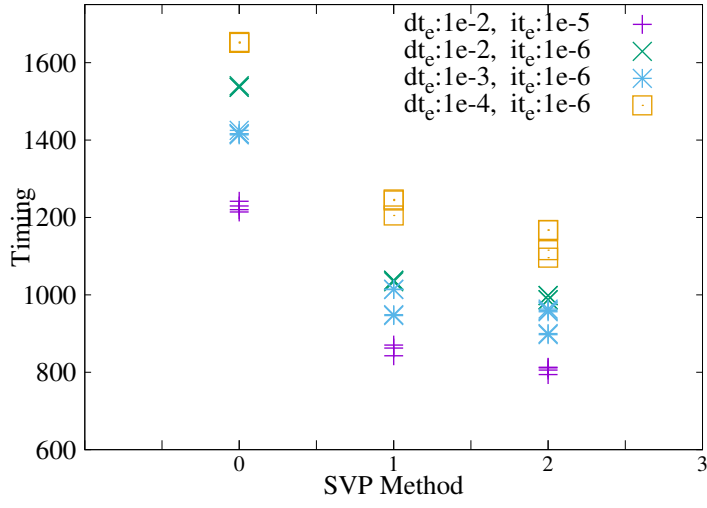
Figure 6 presents a representative subset of results comparing the efficacy of SVPs on TG-vortex simulations using the ESDIRK4(3)8L[2]SA scheme. Recall that this test case is typically simulated using explicit temporal schemes owing to its lack of stiffness. Observe that the trivial guess (method:0) is competitive with all SVP designed for these two stages. Several of the SVPs reduce the efficiency of the simulations relative to the trivial guess. Note that the performant methods in the highly stiff SD7003 simulations (e.g., 6.5, 6.6 and 8.2, 8.4) are as efficient as the trivial guess. Further investigation is warranted in determining whether any SVPs can consistently outperform the trivial guess in the nonstiff limit.



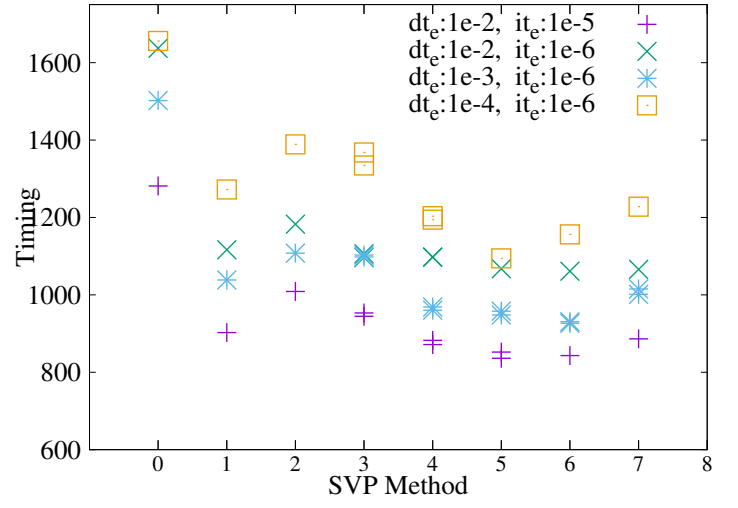
(a) Stage 3



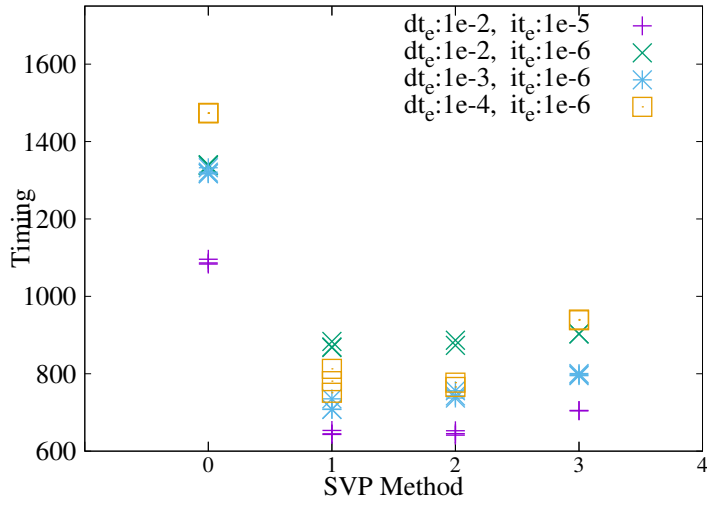
(b) Stage 4



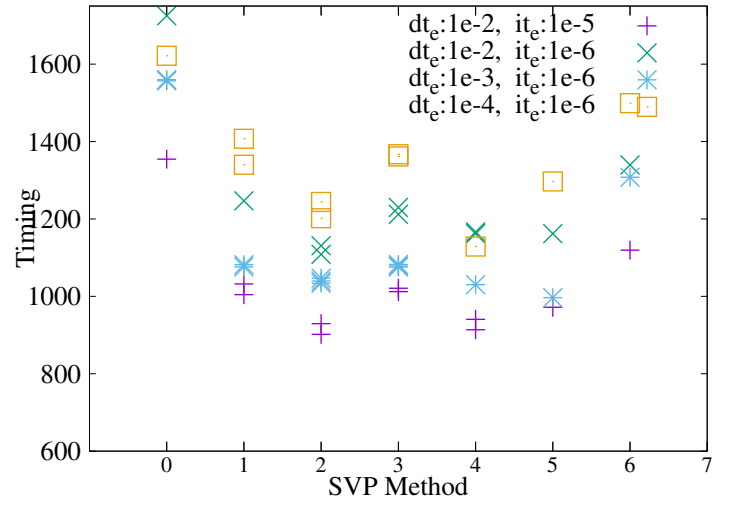
(c) Stage 5



(d) Stage 6



(e) Stage 7



(f) Stage 8

Figure 4: ESDIRK4(3)8L[2]SA simulations of the SD7003 airfoil

SVP	Design	A_{stab}	L_{stab}	$R(-\infty)$	$-\ \mathcal{M}\ $	$\ \beta\ $	$\ \Delta A^{(2)}\ $	$\ \Delta A^{(3)}\ $	$\ \Delta A^{(4)}\ $
3.1	A 1	Y	N	0.4142	-0.0023	0.24	0.0042	0.0012	0.00027
4.1	A 2	N	N	1.21	-2.2629	0.74	0.0	0.0017	0.0021
4.2	A L	N	N	0.0	-1.2990	0.85	0.012	0.0043	0.0023
SVP	Design	A_{stab}	L_{stab}	$R(-\infty)$	$-\ \mathcal{M}\ $	$\ \beta\ $	$\ \Delta A^{(3)}\ $	$\ \Delta A^{(4)}\ $	$\ \Delta A^{(5)}\ $
5.1	3	N	N	-3.408	-15.611	1.78	0.0	0.0021	0.00057
5.2	A L	Y	N	-0.191	-2.071	0.64	0.0051	0.0032	0.0033
6.1	3	N	N	-0.537	-9.403	1.28	0.0	0.0014	0.0032
6.2	3	N	N	+9.993	-31.830	2.29	0.0	0.0056	0.0038
6.3	3	N	N	+3.891	-0.615	0.29	0.0	0.0021	0.0033
6.4	L 3	N	N	+0.000	-12.637	1.47	0.0	0.0014	0.0032
6.5	A L	Y	N	+0.16	-14.05	1.55	0.0045	0.0043	0.0054
6.6	A 3	Y	N	+0.729	-9.992	1.31	0.0036	0.0039	0.0050
6.7	A M	Y	N	+0.999	-8.699	1.23	0.0040	0.0043	0.0052
7.1	A 3 4	Y	N	0.027	-4.434	0.811	0.0	0.0	0.0032
7.2	A 3 L	Y	Y	0.000	-4.441	0.812	0.0	1×10^{-5}	0.0038
7.3	A 3 L M	Y	Y	0.000	-0.329	0.207	0.0	5×10^{-4}	0.0073
8.1	M 3	N	N	0.00	-1.17	0.415	0.0	0.0140	0.0021
8.2	A N	Y	N	-1.00	-43.17	3.415	0.0014	0.0140	0.0052
8.3	M	N	N	+1.00	-2.47	0.415	0.0013	0.0140	0.0031
8.4	A L	Y	Y	+0.0	-53.17	2.569	0.0100	0.010	0.0017
8.5	A L	Y	Y	+0.00	-26.17	1.807	0.0069	0.0120	0.0003
8.6	3 4	N	N	+25.0	-4.32	0.805	0.0	0.0	0.0038

Table 4: SVP properties for stages three to eight.

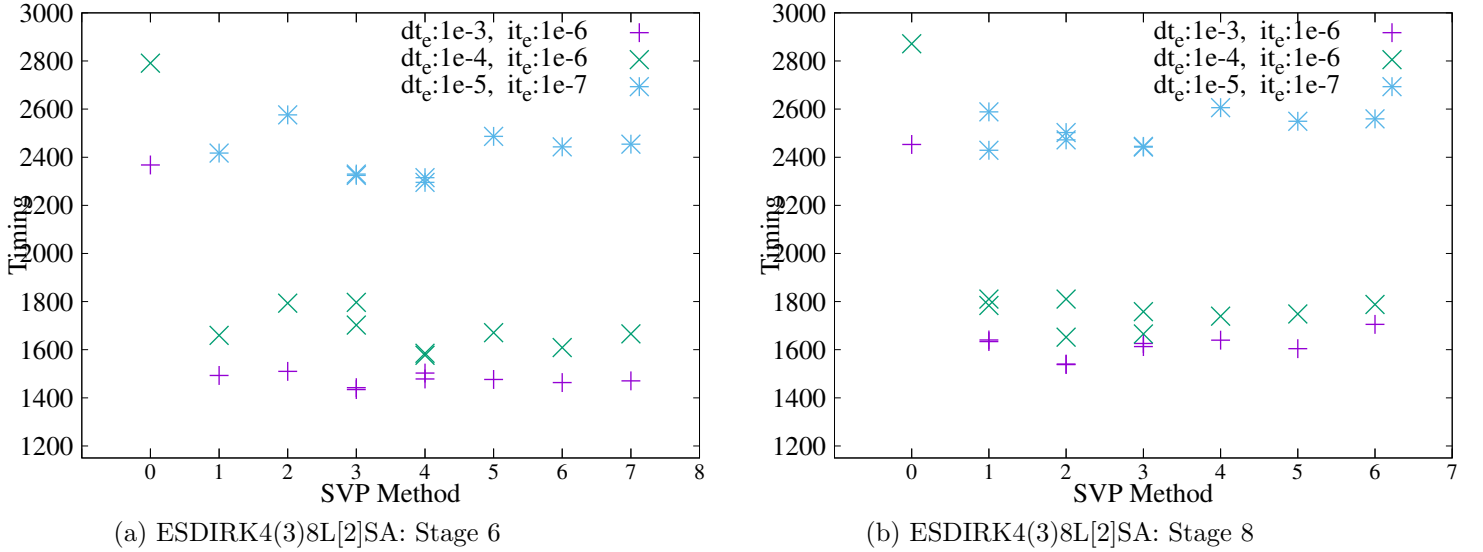


Figure 5: MD-30P30N Airfoil simulation

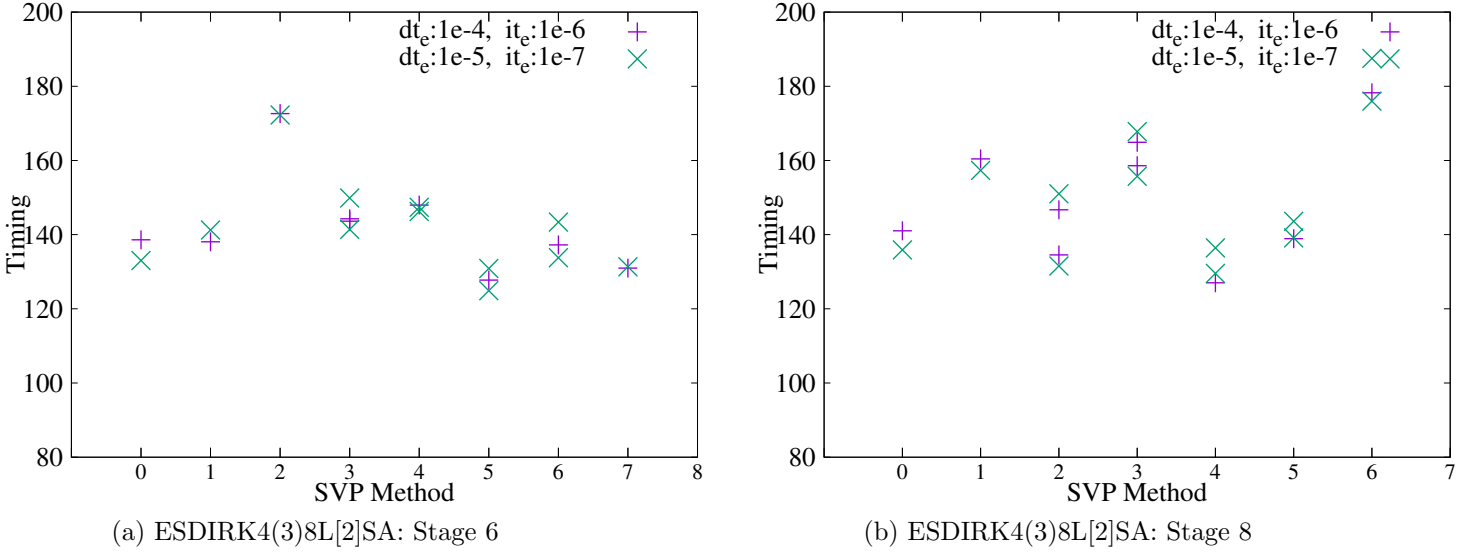


Figure 6: Taylor-Green Vortex simulations

7.6 Stability contours of performant schemes

Figure 8 presents linear stability plots of the amplification factor z for four of the performant SVP designs. The internal amplification $z = \lambda \delta t$ determines the growth of complex modes for a given timestep with $\|z\| \leq 1$ implying stability. (See equation 13 for the definition of z for SVPs.) Recall that A-stability requires the entire left-half of the complex plane satisfy $\|z\| \leq 1$ for a neutrally stable or dissipative system. Only the upper left quadrant is presented as z is symmetric about the real axis. For reference, the stability boundary $\|z\| = 1$ is synonymous with the boundary of the red contours in all plots.

7.6.1 ESDIRK4(3)7L[2]SA

Subfigures 8a, 8c and 8d are all A-stable methods that are highly effective in the SD7003 test simulations. Note that all three stability contours show rapid damping on the negative real axis and reasonably good damping on the imaginary axis. Subfigure 8b has a significant portion around the origin that is stable although clearly it is not A-stable. Modestly stiff problems could be stable for this third order method provided their scaled eigenvalues are within the $\|z\| \leq 1$ lobe. This would explain why some of the third-order methods perform well as SVPs.

7.6.2 ESDIRK4(3)8L[2]SA

Subfigures 9a, 9b, 9c and 9d are all A-stable methods that are highly effective in the SD7003 test simulations. The first three stability contours show rapid damping on the negative real axis and reasonably good damping on the imaginary axis. Subfigure 8d is one of the two strictly A-stable SVPs that performs extremely well in tests despite having no damping at $R(-\infty)$. Note the large lobe that is highly damped along the negative real axis. Scaled eigenvalues along the negative real axis will be adequately damped for modestly stiff problems.

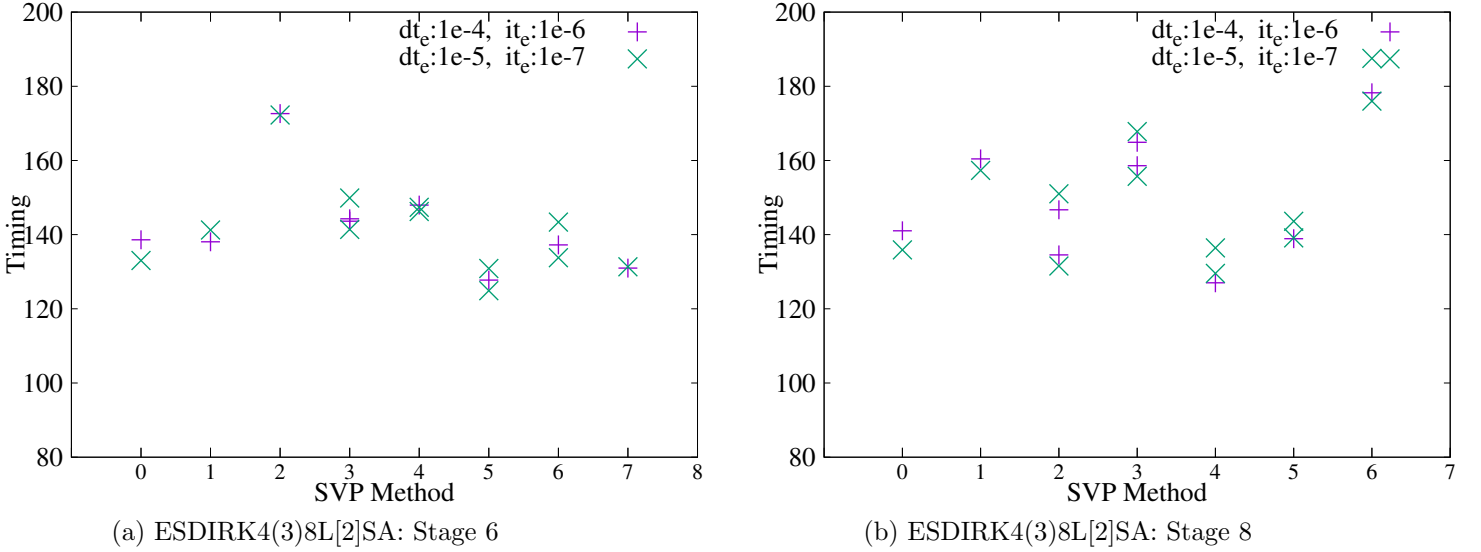


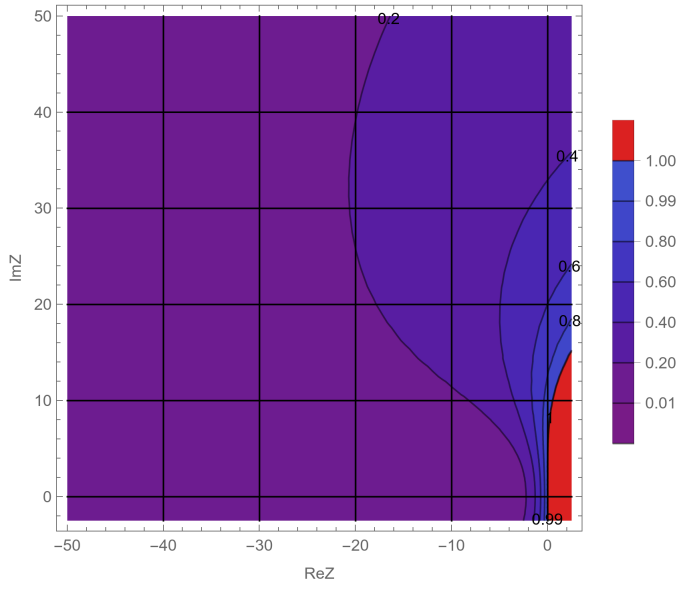
Figure 7: Taylor-Green Vortex simulations

7.7 Discussion

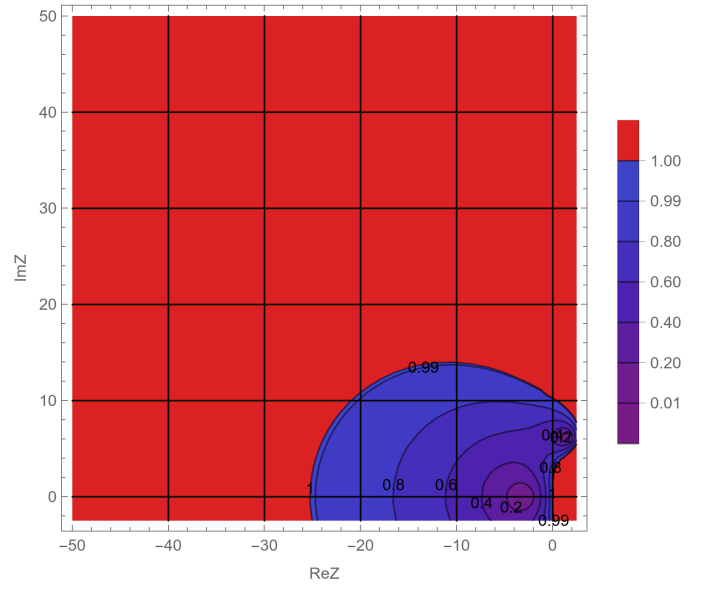
Fifty distinct SVPs are constructed for two ESDIRK schemes, and are tested in simulations of the compressible Navier-Stokes equations at a very large scale: $(1 \times 10^6 - 3 \times 10^7)$ DoFs). An exhaustive study of the SVP design parameter space was conducted to determine the important design constraints. The findings in this study are generally consistent with the “conventional wisdom” in the SVP community interpreted through the lens of computations at a very large scale. Specifically, designing for accuracy is advisable at modest levels of stiffness (e.g., strict error tolerances $\leq 10^{(-4)}/\text{timestep}$). However, if significant stiffness is present (e.g., lax error tolerances: $10^{(-2)}/\text{timestep}$), A-stability and full/partial L -stability are paramount.

Another nugget of SVP “conventional wisdom” is that low-order SVPs can outperform high-order SVPs. While this statement is true, in light of the results presented herein, it should be sharpened to account for the stability properties of both low- and high-order SVPs. Designing an intrastep SVP for any stage nearly always results in a highly over-constrained minimization problem. Frequently, there are no stable and high-order SVPs available for a particular stage. In this case, the efficacy of a stable, low-order SVP will be superior on stiff problems. A more precise statement comparing low- and high-order SVPs might be “low-order, stable SVPs can outperform high-order, unstable SVPs.”

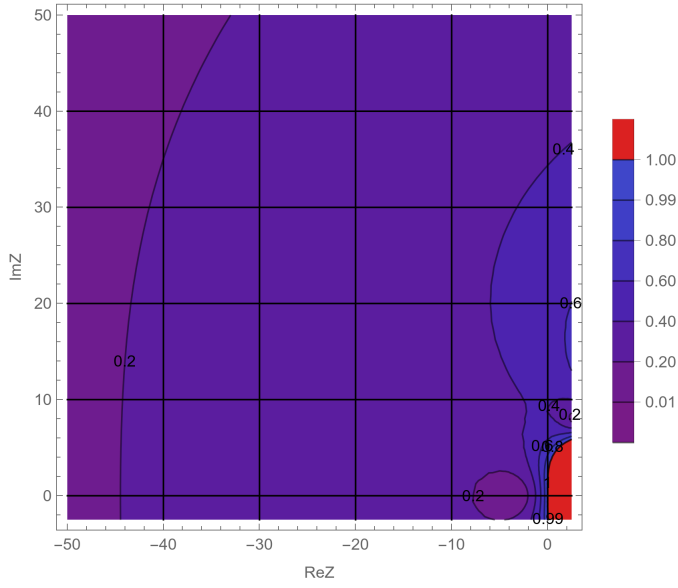
When constructing an intrastep SVP for stage s of an ESDIRK, there are $s - 1$ free parameters in the design space. In the early stages, few choices can be made, and the efficacy of the intrastep SVP is almost completely determined by the properties of the baseline scheme. At later stages, a hierarchy of design constraints emerges for building an intrastep SVP. The following list prioritizes the importance of design constraints: 1) $\Delta t_1^{(1)} = 0$, 2) $\beta_1 - \beta_2 = 0$, 3) $\Delta t_1^{(2)} = 0$, 4) A-stability, 5) L -stability, 6) $\Delta t_1^{(3)} = 0$, 7) $\|\mathcal{M}\|$ (or minimize $\|\beta_j\|$), 8) $\Delta t_1^{(4)} = \Delta t_2^{(4)} = 0$. Recognize, however, that the baseline scheme and the targeted error tolerance dictate the relative importance of the constraints and the design procedure must be adjusted accordingly for every stage of every scheme. A realistic goal for an intrastep SVP is to outperform the trivial guess by a factor $1.5x - 2x$. The upper bound is possible but elusive; it is consistently achieved on stage seven of the ESDIRK4(3)8L[2]SA scheme, but not on the other stages of



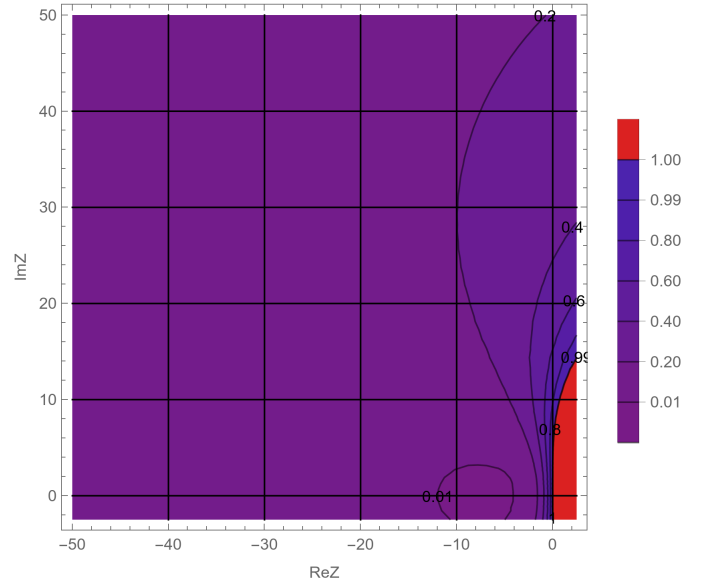
(a) $R_{\text{int}(z)}$: Stage 5.3



(b) $R_{\text{int}(z)}$: Stage 6.3

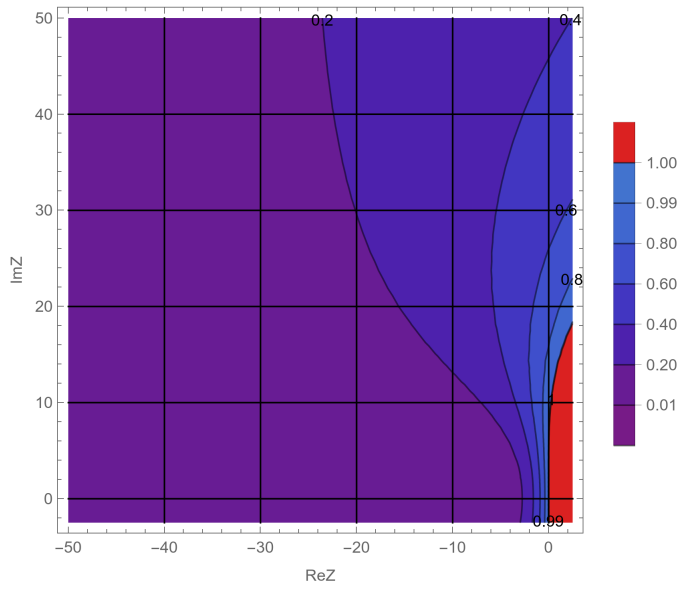


(c) $R_{\text{int}(z)}$: Stage 6.7

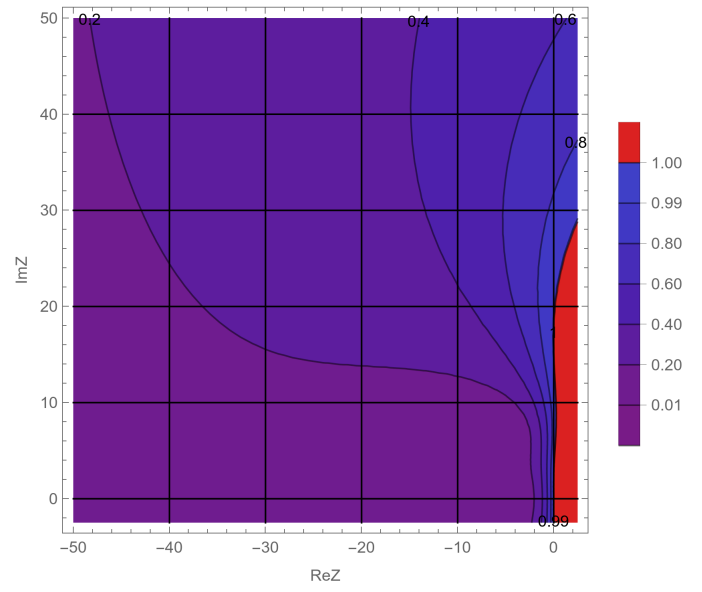


(d) $R_{\text{int}(z)}$: Stage 7.7

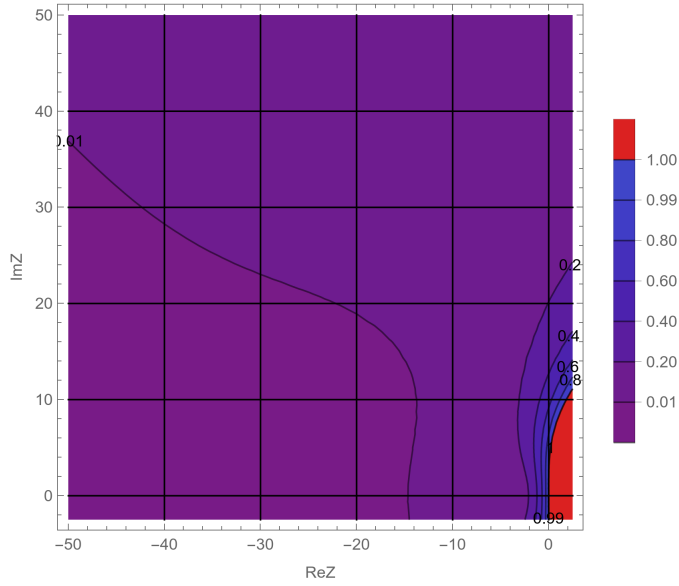
Figure 8: ESDIRK4(3)7L[2]SA: Stability contours of performant SVPs on SD7003 airfoil



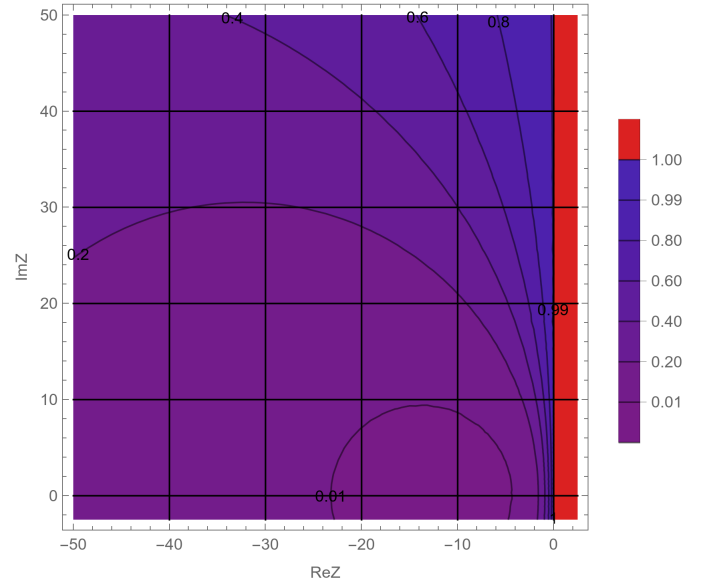
(a) $R_{\text{int}(z)}$: Stage 5.3



(b) $R_{\text{int}(z)}$: Stage 6.3



(c) $R_{\text{int}(z)}$: Stage 6.7



(d) $R_{\text{int}(z)}$: Stage 7.7

Figure 9: ESDIRK4(3)8L[2]SA: Stability contours of performant SVPs on SD7003 airfoil

either ESDIRK scheme considered.

The following insights are drawn from the results of this study. Recall that all ESDIRK schemes used in this study are designed with superior internal and stepwise stability properties and high stage order. Thus, the converged data available at each stage is of high quality. Building an SVP is tantamount to passing a polynomial through said data and evaluating it at the next desired abscissa location. If the problem (and the polynomial) is only modestly stiff, then interpolation/extrapolation error is dictated by design-order accuracy arguments. SVPs constructed in this scenario should focus on high accuracy conditions. As stiffness increases, however, the stiff components of the polynomial (which is subject to order reduction down to second order), will diverge significantly away the polynomial knots. Interpolation is more accurate than extrapolation, but generally both interpolation/extrapolation of data suffer if the polynomial is extended great distances to evaluate the next abscissa. This problem becomes even more acute for interstep predictors that try to ride stiff polynomials even greater distances.

Many questions about the relevance of stability are clarified herein, although several outstanding questions still remain unanswered. 1) Generally, the aforementioned design constraints: A-stability and $\Delta t_1^{(3)} = 0$, are mutually exclusive (with rare exception). Why and how does one know a priori if an A-stable scheme exists that is third-order accurate? 2) Why do the SVPs on stage 4 perform well given their modest design metrics? 3) Why do most schemes that enforce the $\Delta t_1^{(3)} = 0$ constraint perform poorly, but a few work extremely well? 4) Why are SVPs ineffective in the nonstiff limit? And are there SVPs that are effective? 5) Will multiple additional function evaluations (e.g., Laburta [27]) improve the SVPs or will the loss of stability properties negate their efficacy? 6) Will the conclusions herein extend to stage order one ESDIRKs? 7) Will interstep data extrapolated over long distances improve the efficacy of the intrastep SVPs? Stage two is currently using dense output as an SVP. Will stages three and four also benefit from a similar approach?

8 The Efficiency Landscape of ESDIRK Schemes

A new eight-stage base scheme: ESDIRK4(3)8L[2]SA with γ reduced to $\approx 1/10$ is designed for this study of SVPs.⁴ The principal objective is to ensure distinct abscissae locations for the eight-stage scheme relative to the ESDIRK4(3)7L[2]SA scheme. Furthermore, it is conjectured that the improved convergence behavior combined with good SVPs will offset the increased work of an additional stage. To test this hypothesis and in general, to understand the relative efficiency landscape of ESDIRK schemes, an extensive study of simulation costs is performed using six popular ESDIRK schemes.

Table 5 presents a comparison of the simulation times and the SVP gains in efficiency for six popular ESDIRK schemes. Accuracy-based stage-value predictors are constructed for the four schemes not directly the focus of this study. Three target temporal errors $\epsilon_{target} = 10^{(-2)}, 10^{(-4)}, 10^{(-6)}$ are considered. Both the trivial guess and an optimized SVP is used to start the nonlinear simulations. The error in the nonlinear solver is driven below the target temporal error by at least a factor $10^2 - 10^3$. The temporal error controller automates all simulations in this scheme comparison. The test problem is the *P3* (fourth-order) simulation of the SD7003 airfoil using the H11344 hexahedral grid.

Three schemes, all with seven or more stages: ESDIRK4(3)7L[2]SA, ESDIRK4(3)8L[2]SA, and IMEXRK4(3)7L[2]SA, are most efficient over the entire temporal error range: $10^{(-2)} - 10^{(-6)}$, including

⁴The recent papers by Kennedy and Carpenter [25, 26] demonstrate potential improvements in efficiency for seven-stage ESDIRKs. This notion is extended herein to eight stages. A presents this argument in more mathematical terms.

the ridiculously coarse limit of $10^{(-2)}$. Note that the second order scheme: ESDIRK23 is performant in the $10^{(-2)}$ limit, but still not competitive. At strict error tolerances ESDIRK23 is suboptimal by $3x$.

The SVP improves the efficiency $\eta = \frac{t_{SVP}}{t_{triv}}$ in all cases and even more so at strict error tolerances. Reductions in simulation times for the seven and eight-stage schemes is approximately $\eta = \frac{2}{3}$. These simulations do not include dense output prediction in stage two, so further improvements are expected.

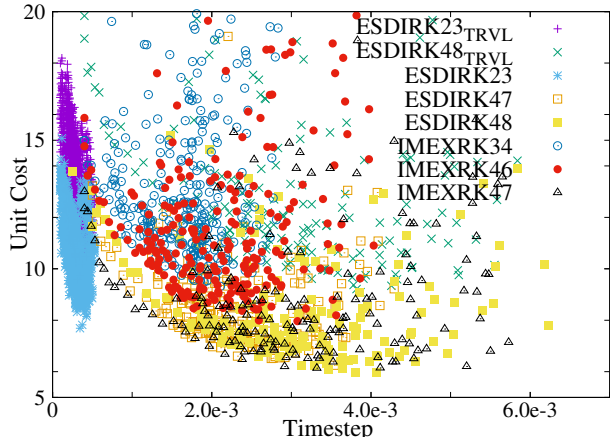
Time Error	$dt_e = \epsilon_{target} = 10^{(-2)}$			$dt_e = \epsilon_{target} = 10^{(-4)}$			$dt_e = \epsilon_{target} = 10^{(-6)}$		
Nonlin resid	$it_e = \epsilon_{resid} = 10^{(-5)}$			$it_e = \epsilon_{resid} = 10^{(-6)}$			$it_e = \epsilon_{resid} = 10^{(-8)}$		
Scheme	SVP	Trivial	η	SVP	Trivial	η	SVP	Trivial	η
ESDIRK23 [24]	16310	19963	0.81	52058	61608	0.84	382780	-	-
ESDIRK47 [26]	12048	17392	0.69	17130	25648	0.66	32195	57466	0.56
ESDIRK48	12058	17594	0.68	16009	23829	0.67	29642	55737	0.53
IMEXRK34 [23]	20693	25189	0.82	29941	44539	0.67	71655	108195	0.66
IMEXRK46 [23]	18133	27548	0.65	22299	36692	0.60	43928	82467	0.53
IMEXRK47 [25]	12637	18667	0.67	15155	24199	0.62	29961	56064	0.53

Table 5: Timings for three target errors ϵ_t

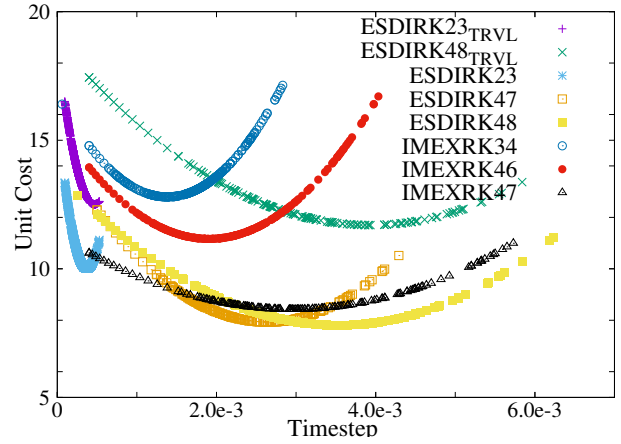
The simulation data that produced the coarse level results shown in table 5 is now presented in much finer granularity in 10 to gain insight into the performance characteristics of the seven and eight-stage schemes. Figures 10a, 10c show scatter plots of the normalized cost of all six methods, with each symbol corresponding to a single timestep. Two additional baseline results are included corresponding to the ESDIRK23 and ESDIRK48 schemes run with the trivial guess. (The “normalized cost” is defined as the total Krylov vectors required per step, divided by the timestep.) Data scatter results from variations in the timestep as the temporal controller adjusts to instantaneous flow phenomena (e.g., vortex shedding), which in turn strongly influences the efficiency of the linear and nonlinear iterations. Figures 10b, 10d show a quadratic fit to the scatter data of all eight methods. Figures 10a, 10b correspond to a target timestep errors of $10^{(-2)}$ and a nonlinear residual reduction of $10^{(-5)}$. Figures 10c, 10d correspond to a target timestep errors of $10^{(-4)}$ and a nonlinear residual reduction of $10^{(-6)}$.

The efficiency of all schemes is concave with respect to timestep; i.e., decreasing the timestep could improve efficiency if the integrator is “overstepping” the capability of the linear/nonlinear solver machinery. All schemes achieve a minimum unit cost (the optimal timestep for a given target accuracy) at the lax tolerance: $dt_e = 10^{(-2)}$. At the strict error tolerance: $dt_e = 10^{(-4)}$ the ESDIRK23 scheme lacks sufficient accuracy to achieve the minimum, while the IMEXRK46 ($\gamma = 1/4$) scheme oversteps the capabilities of the nonlinear/linear solver mechanics in trying to achieve the temporal error target. Note that the optimal timestep of the ESDIRK48 ($\gamma \approx 0.1$) is larger than ESDIRK47 ($\gamma = 0.125$), by approximately the ratio in γ s. Also, note that the optimal timestep for ESDIRK48 *decreases* when the SVP is in use.

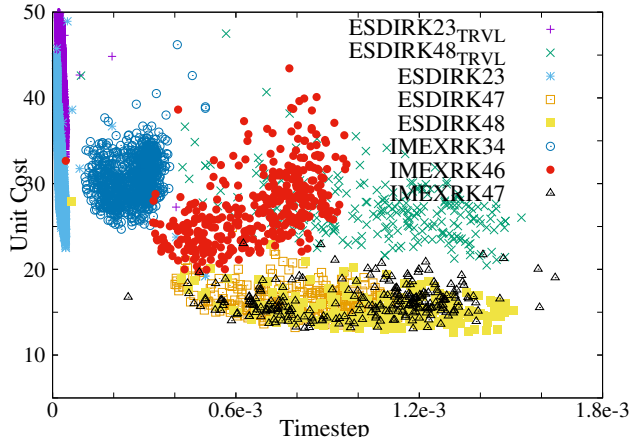
The absolute ranking at coarse tolerance of $10^{(-2)}$ is 1) ESDIRK48, 2) ESDIRK47, 3) IMEXRK47, 4) ESDIRK23, 5) IMEXRK46, 6) IMEXRK34, while at the strict error tolerance ($10^{(-4)}$) the relative position of the 4) IMEXRK46, 5) ESDIRK23 is reversed. The fourth-order, ESDIRK47, ESDIRK48 and IMEXRK47 schemes are the most efficient of those tested at both error tolerances. The fourth-order ESDIRK46 scheme is notably less efficient than the other fourth-order schemes, the IMEXRK34 scheme is the least efficient. While the ESDIRK23 scheme is advantageous at the lax error tolerances ($10^{(-2)}$), its poor accuracy severely limits its efficacy at strict tolerances. Note that at $10^{(-4)}$ /timestep the ESDIRK23



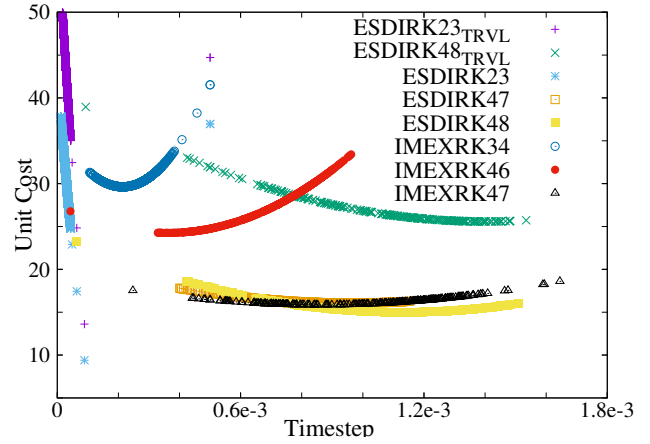
(a) P3 Scatter: $dt_e = 10^{-2}, it_e = 10^{-5}$



(b) P3 Data fit: $dt_e = 10^{-2}, it_e = 10^{-5}$



(c) P3 Scatter: $dt_e = 10^{-4}, it_e = 10^{-6}$



(d) P3 Data fit: $dt_e = 10^{-4}, it_e = 10^{-6}$

Figure 10: Method comparison on normalized costs: scatter plots and data fits

scheme is accuracy limited and never achieves an optimal timestep. Thus, the potential efficiency of only two implicit stages can never be exploited.

9 Conclusions

To better identify the necessary attributes of good stage-value predictors (SVPs), numerous SVPs are designed for an existing: ESDIRK4(3)7L[2]SA [26] and a new: ESDIRK4(3)8L[2]SA (eight stages) scheme. Both are stiffly-accurate, stage-order two, explicit, singly-diagonally implicit Runge–Kutta (ESDIRK) schemes. As many as thirteen SVPs are constructed for different stages of each method designed with different objectives. Tradeoffs are studied in the parameter spaces enforcing the constraints on accuracy, linear stability, nonlinear stability and coefficient size to determine which objectives correlate with effective predictors. The SVPs are tested in challenging external aerodynamics simulations (10^7 DoFs) using the compressible Navier-Stokes equations (CNSE). Three test cases with varying degrees of stiffness are investigated. They are the SD7003 airfoil, the MD-30P30N multielement airfoil and the Taylor-Green vortex. Curvilinearly mapped hexahedral FEM grids are used in all simulations. An entropy-stable spectral collocation formulation discretizes the spatial terms in the equations. Simulations are performed at a wide variety of temporal error tolerances. Problems that are sufficiently stiff (e.g., lax temporal error tolerances), benefit from SVPs designed with second-order accuracy and stability properties: A-stability, and L-stability, rather than high accuracy constraints. Simulations with modest stiffness (e.g., strict error tolerances) are better suited for SVPs designed using high accuracy constraints. Minimizing the algebraic instability of the SVPs does not contribute significantly to their efficacy. However, methods with exceedingly large algebraic instability are consistently poor performers. Designing SVPs with enhanced stability properties is tantamount to constructing an individual Runge–Kutta scheme at every stage of the ESDIRK. Although this approach is tedious it is worthwhile. Simulation times are reduced with optimal SVPs by as much as 100% on some stages, with combined stepwise improvements of between 50 – 100% for both methods. A comparative study is performed with the two aforementioned methods as well as four other ESDIRKs. Optimized SVPs are used in all cases. The most efficient ESDIRK methods are those with six or more implicit stages including the two used in this study. The newly designed ESDIRK4(3)8L[2]SA with $\gamma \approx 1/10$, proves to be the most efficient of the six tested ESDIRK schemes simulating the CNSE.

10 Acknowledgments

We gratefully acknowledge the support received from the NASA Transformational Tools and Technologies Project (TTT) under the supervision of Dr. Mujeeb Malik. Special thanks are extended to those individuals involved in building the software used in this project including, Travis C. Fisher, Matteo Parsani, Eric Nielsen, and David Del Rey Fernandez.

11 References

References

1. Andrea P. C. Bresciani, Large Eddy simulation of the Transitional Flow around the SD7003 Airfoil and Application to Blade-Vortex Interaction, Master's Thesis, Scuola di Ingegneria Industriale e Dell'Informazione Dipartimento di Scienze e Tecnologie Aerospaziali, 2019.
2. M.P. Calvo and C. Palencia, Avoiding the order reduction of Runge–Kutta methods for linear initial boundary value problems, *Math. Comp.*, 71(240) (2002) 1529-1543.
3. M. Calvo, M.P. Laburta and J.I. Montijano, Starting algorithms for Gauss Runge–Kutta methods for Hamiltonian systems, *Comp. and Math. w/ Appl.*, 45(1-3) (2003) 401-410.
4. M. Calvo, M.P. Laburta and J.I. Montijano, Two-step high order starting values for implicit Runge–Kutta methods, *Adv. Comp. Math.*, 19(4) (2003) 401-412.
5. M. Calvo, J.I. Montijano and L. Rández, On the solution of discontinuous IVPs by adaptive Runge–Kutta codes, *Numer. Alg.*, 33(1-4) (2003) 163-182.
6. M.P. Calvo, J. de Frutos and J. Novo, An efficient way to avoid the order reduction of linearly implicit Runge–Kutta methods for nonlinear IBVP's, In: *Mathematical Modelling, Simulation and Optimization of Integrated Electrical Circuits*, K. Antreich, R. Bulirsch, A. Gilg and P. Rentrop, Eds., Inter. Series of Numer. Math. Vol. 146, Birkhäuser, Basel (2003) 321-332.
7. M.P. Calvo and A. Portillo, Are high order variable step equistage initializers better than standard starting algorithms?, *J. Comp. and Appl. Math.*, 169(2) (2004) 333-344.
8. F. Cameron, *Low-order Runge–Kutta Methods for Differential-Algebraic Equations*, Ph.D. Thesis, Dept. of Math., Tampere Univ. of Tech., Tampere (1999).
9. M.H. Carpenter, T.C. Fisher, E.J. Nielsen, S.H. Frankel, Entropy stable spectral collocation schemes for the Navier-Stokes equations: Discontinuous interfaces. *SIAM J. Sci. Comput.*, Vol 36(5) B835-B867.
10. T.C. Fisher, M.H. Carpenter, J. Nordstrom, N.K. Yamaleev, C. Swanson, Discretely conservative finite-difference formulations for nonlinear conservation laws in split form: Theory and boundary conditions, *J. Comput. Phys*, Vol 234, pp 353-375, 2013.
11. T.C. Fisher, M.H. Carpenter, High-order entropy-stable finite difference schemes for nonlinear conservation laws: finite domains, *J. Comput. Phys*, Vol 252, pp 518-557, 2013.
12. S. González-Pinto, J.I. Montijano and S. Pérez-Rodríguez, On starting algorithms for implicit RK methods, *BIT*, 40(4) (2000) 685-714.
13. S. González-Pinto, J.I. Montijano and S. Pérez-Rodríguez, Variable-order starting algorithms for implicit RK methods on stiff problems, *Appl. Numer. Math.*, 44(1-2) (2003) 77-94.
14. S. González-Pinto, J.I. Montijano and S. Pérez-Rodríguez, Stabilized starting algorithms for collocation Runge–Kutta methods, *Comp. and Math. w/ Appl.*, 45(1-3) (2003) 411-428.

15. E. Hairer, Ch. Lubich and G. Wanner, *Geometric Numerical Integration, Structure-Preserving Algorithms for Ordinary Differential Equations*, 2nd Ed., Springer-Verlag, Berlin (2006).
16. E. Hairer, S.P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I, Nonstiff Problems*, 2ed., Springer-Verlag, Berlin (1993).
17. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems*, 2ed., Springer-Verlag, Berlin (1996).
18. I. Higueras and T. Roldán, Starting algorithms for some DIRK methods, *Numer. Alg.*, 23(4) (2000) 357-369.
19. I. Higueras and T. Roldán, Starting algorithms for a class of RK methods for index- DAEs TechRpt Universidad Publica de Navarra, No 4 (2001).
20. I. Higueras and T. Roldán, ESDIRK methods for index-2 DAE: Starting algorithms, In: *Monografias del Sem. Matem. García de Galdeano*, 27, M. Madaune-Tort, D. Trujillo, M.C. López de Silanes, M. Palacios and G. Sanz, Eds., Prensus Univ. Zaragoza, Zaragoza (2003) 345-352.
21. I. Higueras and T. Roldán, Consistency of a class of RK methods for index-2 DAE's *Monografias del Seminario Matematico García de Galdeano* 31, 179-189 (2004).
22. I. Higueras and T. Roldán, Stage value predictors for additive and partitioned Runge–Kutta methods, *Appl. Numer. Math.*, 56(1) (2006) 1-18.
23. C.A. Kennedy and M.H. Carpenter, Additive Runge–Kutta schemes for convection-diffusion-reaction equations, *Appl. Numer. Math.*, 44(1-2) (2003) 139-181.
24. C.A. Kennedy and M.H. Carpenter, Diagonally Implicit Runge–Kutta Methods for Ordinary Differential Equations. A Review, NASA/TM-2016-219173, NASA Langley Research Center (2016) 1-162.
25. C.A. Kennedy and M.H. Carpenter, Higher-order additive Runge–Kutta schemes for ordinary differential equations, *Appl. Numer. Math.*, 136 (2019) 183-205.
26. C.A. Kennedy, M.H. Carpenter, Diagonally implicit Runge–Kutta methods for stiff ODEs, *Appl. Numer. Math.*, 146 (2019) 221-244.
27. M.P. Laburta, Starting algorithms for IRK methods, *J. Comp. and Appl. Math.*, 83(2) (1997) 269-288.
28. M.P. Laburta, Starting algorithms for IRK methods, *J. Comp. and Appl. Math.*, Construction of starting algorithms for the RK-Gauss methods, *J. Comp. and Appl. Math.* 83 (1998) 239-261.
29. M.P. Laburta, Starting algorithms for implicit Runge–Kutta-Nyström methods, *Appl. Numer. Math.* 27 (1998) 233-251.
30. M.P. Laburta, Starting algorithms for the iterations of the RKN-Gauss methods, *Appl. Numer. Math.* 27 (1999) 81-101.
31. H. Olsson and G. Söderlind, Stage value predictors and efficient Newton iterations in implicit Runge–Kutta methods, *SIAM J. Sci. Comp.*, 20(1) (1998) 185-202.

32. M. Parsani, M.H. Carpenter, E.J. Nielsen, Entropy stable wall boundary conditions for the three-dimensional compressible Navier-Stokes equations, *J. Comput. Phys*, Vol 292, pp 88-113.
33. Prothero, A., and A. Robinson. On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations. *Math. of Comp.* 28.125 (1974): 145-162.
34. T. Roldán and I. Higuera, IRK methods for DAE: starting algorithms, *J. Comp. and Appl. Math.*, 111(1) (1999) 77-92.
35. Y. Saad and M. H. Schultz, A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. stat. Comput.*, Vol (7) 856-869.
36. Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd edition. SIAM, Philadelphia, PA, 2003.
37. J. Sand, RK-predictors: Extrapolation methods for implicit Runge–Kutta formulae, DIKU Report Nr. 88/18, Datalogisk Institut, Københavns Universitet, Copenhagen (1988) 1-18.
38. J. Sand, Methods for starting iteration schemes for implicit Runge–Kutta formulae, In: *Computational Ordinary Differential Equations*, J.R. Cash and I. Gladwell, Eds., Clarendon Press, New York (1992) 115-126.
39. H. Sarlak Large Eddy Simulation of an SD7003 Airfoil: Effects of Reynolds number and subgrid-scale modeling IOP Confer. Series: Journal of Physics: Confer. Series 854 (2017) 012040.
40. L.M. Skvortsov, An efficient scheme for the implementation of implicit Runge–Kutta methods, *Comp. Math. and Math. Phys.*, 48(11) (2008) 2007-2017.
41. Taylor, G. I. and Green, A. E., Mechanism of the production of small eddies from large ones, *Proc. R. Soc. Lond. A*, 158, (1937) 499–521.
42. S. Wolfram, *Mathematica*, V.13, Wolfram Research. (2021).

Appendix A

Improving Efficiency with Additional Stages

Adding a stage to a six stage ESDIRK formulation is shown elsewhere to be beneficial [25, 26]. The improvements manifest from 1) better overall design characteristics (e.g., minimized algebraic instability and leading order truncation error (L.O.T.E.) terms, better embedded scheme), 2) reducing by $(2x)$ the diagonal coefficient γ of the method. This reduction strongly influences the conditioning of the preconditioner and the overall convergence characteristics.

The following simple observation motivates the design of these ESDIRKs. Consider the linear system that must be solved at every Newton iteration on all stages of an ESDIRK. It is of the form

$$(\mathbf{I} - \gamma \Delta t \mathbf{J})(\delta U_{i,k}) = \mathbf{N}(\delta U_{i,k}) = -\mathbf{r}_{i,k} \quad (\text{A1})$$

where the subscripts i and k denote the i^{th} stage and the k^{th} iteration, respectively, Δt is the timestep, γ is the weight of the diagonal term arising from the DIRK, $\mathbf{I}$ is the identity matrix, $(\partial F / \partial U) = \mathbf{J}$ is the Jacobian of the RHS of the CNSE, $(\mathbf{I} - (\Delta t)\gamma \mathbf{J}) = \mathbf{N}$ is the (Newton) iteration matrix, $(\delta U_{i,k})$ is the update of the solution and the vector $\mathbf{r}_{i,k}$ of (A1) is the residual of the nonlinear system of equations.

Definition A.1 Let $\mathbf{N}$ be an $n \times n$ matrix with entries n_{ij} , with row-wise Gershgorin circle radius $R_i = \sum_{j \neq i} |n_{ij}|$, $i = 1, n$. Next define a Gershgorin row i metric as $\mathbf{G}_i = R_i / n_{ii}$. The L_∞ -norm of $\mathbf{G}_i$ yields a “Gershgorin metric”; $\mathbf{G} = \max_i \mathbf{G}_i$.

The convergence rate of the linear system for the CNSE is strongly dependent on the $\mathbf{G}$ of the iteration matrix $\mathbf{N}$. Observe that reducing either Δt or γ will reduce the Gershgorin metric by decreasing the off-diagonal contributions relative to the diagonal, and thus improving the convergence behavior of the linear solver.

Table A1 provides important properties for several popular schemes including γ , the temporal order, the number of implicit stages, and L.O.T.E. of the 4th-order schemes to facilitate comparison.

	Scheme	order	γ	Impl Stages	L.O.T.E.
1	BDF1	1	1.0000	1	N.A.
2	BDF2	2	0.6667	1	N.A.
3	ESDIRK3(2) 4 L[2]SA [23]	3	0.4359	3	N.A.
4	ESDIRK4(3) 6 L[2]SA [23]	4	0.2500	5	0.01224
5	ESDIRK4(3) 7 L[2]SA [25]	4	0.1250	6	0.00026
6	IMEXRK4(3) 7 L[2]SA [26]	4	0.1235	6	0.00163
7	ESDIRK4(3) 8 L[2]SA	4	0.1009	7	0.00033

Table A1: Diagonal coefficients γ for popular schemes.

Consider the four fourth-order schemes 4, 5, 6, 7. Schemes 5, 6, 7 are significantly more accurate than 4 and could potentially run at timesteps large by a factor of two. Since the γ is smaller by a factor of

two, the Gershgorin metric of both linear systems is effectively equal. The additional cost of schemes $5,6,7$ relative to scheme 4 , is easily absorbed by the much larger timestep. Additional stages can potentially increase efficiency.

Appendix B

Methods

B.1 ESDIRK4(3)7L[2]SA

B.1.1 Stage-value predictors

The SVPs are implemented using the formulae given in equation (10). The β_{kj} coefficients for the ESDIRK4(3)7L[2]SA scheme are give in table B1. Lower precision is adopted owing to the approximate nature of the SVP: $U_{k,0}$.

c_3	β_{31}	$\frac{3998607}{109216786}$	0	0	0	0	1
c_4	β_{41}	$\frac{46611179}{54608393}$	$-\frac{178447502}{147830751}$	0	0	0	1
c_5	β_{51}	$-\frac{47797639}{50219660}$	$\frac{120185484}{62590349}$	$\frac{17}{25}$	0	0	3
c_6	β_{61}	$-\frac{22672606}{107793547}$	$\frac{48949423}{70512297}$	$\frac{94971561}{371244478}$	$\frac{177}{1000}$	0	7
c_7	β_{71}	$-\frac{181872246}{122088097}$	$\frac{215909468}{73524603}$	$\frac{3579}{10000}$	$\frac{2749}{5000}$	$\frac{1351}{10000}$	7

Table B1: SVP for ESDIRK4(3)7L[2]SA, with $\beta_{i1} = \beta_{i2}$ for $i = 3, 7$.

The method corresponding to the reported coefficients is given in the last column of table B1.

B.1.2 Dense Output

Dense output is used to extrapolate function values from the previous step to generate starting guesses for the stage values. The implementation coefficients: $\beta_{i,j}$ needed in equations (8) and (9) are given in table B2.

B.2 ESDIRK4(3)8L[2]SA

β_{1j}	$\frac{-266426472506}{7112241585}$	$\frac{+331477915752}{5127050801}$	$\frac{-170359219871}{6173927403}$
β_{2j}	$\frac{-266426472506}{7112241585}$	$\frac{+331477915752}{5127050801}$	$\frac{-170359219871}{6173927403}$
β_{3j}	$\frac{+431609494593}{6579011485}$	$\frac{-727141547929}{6408476411}$	$\frac{+521433824925}{10684957954}$
β_{4j}	$\frac{+148211146869}{10218122302}$	$\frac{-230866228517}{11439148937}$	$\frac{+199192141}{32148900}$
β_{5j}	$\frac{-12953801331}{12143793896}$	$\frac{+33225514585}{8647497831}$	$\frac{-2}{1}$
β_{6j}	$\frac{-58429657621}{14620201597}$	$\frac{+3}{2}$	$\frac{+97}{50}$
β_{7j}	$\frac{+7}{8}$	$\frac{-1}{1}$	$\frac{+1}{4}$

(B1)

Table B2: Dense output for ESDIRK4(3)7L[2]SA.

B.2.1 Method Design

One way to make an ESDIRK more efficient is to reduce the burden on the solver mechanics used for the nonlinear system. A key way to do this is to reduce the magnitude of γ . The ESDIRK4(3)7L[2]SA method is a fourth-order method with $\gamma = 1/8$. With this in mind, how low could γ be driven if another stage were to be added while still producing a useful method? To do this, we enforce

$$0 = \tau_1^{(1,2,3,4)} = q_{2,3,4,5,6,7}^{(1)} = q_{2,3,4,5,6,7}^{(2)} = \tau_3^{(4)} = p_7 = R_{\text{int}}^{(3,4,5,6,7)}(-\infty), \quad (\text{B2})$$

$$0 = \hat{\tau}_1^{(1,2,3)} = \hat{p}_{6,7}, \quad (\text{B3})$$

for the main method and for the embedded method. The shorthand notation may be understood with

$$\mathbf{q}^{(k)} = \mathbf{A}\mathbf{c}^{k-1} - \frac{1}{k}\mathbf{c}^k, \quad R_{\text{int}}(z) = [\mathbf{I} - z\mathbf{A}]^{-1} \cdot \mathbf{e} = \frac{P_{\text{int}}(z)}{Q_{\text{int}}(z)}, \quad (\text{B4})$$

$$P(z) = \sum_{i=0}^s p_i z^i = \text{Det} [\mathbf{I} - z\mathbf{A} + z\mathbf{e} \otimes \mathbf{b}^T], \quad \hat{P}(z) = \sum_{i=0}^s \hat{p}_i z^i = \text{Det} [\mathbf{I} - z\mathbf{A} + z\mathbf{e} \otimes \hat{\mathbf{b}}^T]. \quad (\text{B5})$$

For the abscissae, we choose $c_1 = 0$, $c_2 = 2\gamma$, $c_8 = 1$ and

$$c_3 = (2 - \sqrt{2})\gamma, \quad c_4 = \frac{402}{971}, \quad c_5 = \frac{250}{439}, \quad c_6 = \frac{993}{1283}, \quad c_7 = \frac{256}{345}, \quad \gamma = 59/585; \quad (\text{B6})$$

Imposing various degrees of freedom for the main and embedded methods, we have

$$\begin{aligned} \tau_4^{(5)} &= -\frac{1}{22906}, \quad \tau_5^{(5)} = -\frac{1}{26393}, \quad \tau_1^{(6)} = \frac{1}{47678}, \quad \tau_4^{(6)} = \frac{1}{11409}, \\ \tau_6^{(6)} &= \frac{1}{16303}, \quad \tau_7^{(6)} = -\frac{1}{9308}, \quad \tau_1^{(7)} = \frac{1}{161688}, \\ \hat{\tau}_1^{(4)} &= -\frac{1}{11725}, \quad \hat{\tau}_1^{(5)} = \frac{1}{73400}, \quad \hat{\tau}_1^{(6)} = \frac{1}{73950}. \end{aligned} \quad (\text{B7})$$

where $\tau_1^{(k)} = \frac{1}{(k-1)!} \mathbf{b}^T \mathbf{c}^{k-1} - \frac{1}{k}$, $\hat{\tau}_1^{(k)} = \frac{1}{(k-1)!} \hat{\mathbf{b}}^T \mathbf{c}^{k-1} - \frac{1}{k}$ and

$$\begin{aligned} \tau_3^{(4)} &= \frac{1}{2} \mathbf{b}^T \mathbf{q}^{(3)} + \tau_1^{(4)}, & \tau_4^{(5)} &= \frac{1}{2} \mathbf{b}^T \mathbf{C} \mathbf{q}^{(3)} + 4\tau_1^{(5)}, & \tau_5^{(5)} &= \frac{1}{6} \mathbf{b}^T \mathbf{q}^{(4)} + \tau_1^{(5)} \\ \tau_4^{(6)} &= \frac{1}{4} \mathbf{b}^T \mathbf{C}^2 \mathbf{q}^{(3)} + 10\tau_1^{(6)}, & \tau_6^{(6)} &= \frac{1}{6} \mathbf{b}^T \mathbf{C} \mathbf{q}^{(4)} + 5\tau_1^{(6)}, & \tau_7^{(6)} &= \frac{1}{24} \mathbf{b}^T \mathbf{q}^{(5)} + \tau_1^{(6)}. \end{aligned} \quad (\text{B8})$$

The ESDIRK4(3)8L[2]SA method is L-stable, internally L-stable on stages three through eight and strictly A-stable on the second stage. Relative to ESDIRK4(3)7L[2]SA, the ESDIRK4(3)8L[2]SA scheme reduces γ by 19.3% at the price of adding an implicit stage, a 16.7% increase in implicit stages. The efficiency of the seven- and eight-stage methods are nearly identical on all problems tested to date.

B.2.2 Butcher Tableau and properties

A new stiffly-accurate, stage-order two ESDIRK method in eight stages: ESDIRK4(3)8L[2]SA is developed herein to study the behavior of SVPs. The abscissae of the new method are distinct from those of the existing ESDIRK4(3)7L[2]SA scheme. The Butcher tableau for the new method is given in table B3 with the remaining coefficients given in table B4.

0	0	0	0	0	0	0	0	0
$\frac{118}{585}$	a_{21}	$\frac{59}{585}$	0	0	0	0	0	0
c_3	a_{31}	$\frac{-19518028676870}{934427233300157}$	$\frac{59}{585}$	0	0	0	0	0
$\frac{402}{971}$	a_{41}	$\frac{344729309340395}{1131933348968038}$	$\frac{-341351779839085}{1153422898589157}$	$\frac{59}{585}$	0	0	0	0
$\frac{250}{439}$	a_{51}	$\frac{-407310541348277}{1457416150858249}$	$\frac{825797892681077}{1108830414526536}$	$\frac{347150461205827}{1227445856948264}$	$\frac{59}{585}$	0	0	0
$\frac{993}{1283}$	a_{61}	$\frac{1365085473788065}{2144135753095052}$	$\frac{-1182497954870351}{1420056438593455}$	$\frac{-63695567441873}{1007972570448412}$	$\frac{553123701809414}{1870580602846629}$	$\frac{59}{585}$	0	0
$\frac{256}{345}$	a_{71}	$\frac{-526494814415147}{1342446036971084}$	$\frac{972489732556969}{1041901655162605}$	$\frac{231710015292815}{710040785046631}$	$\frac{149813302106005}{784935650003848}$	a_{76}	$\frac{59}{585}$	0
1	b_1	$\frac{43330198141423}{1552245574212436}$	$\frac{126920317765990}{976320234585877}$	$\frac{144252338374735}{235812665300824}$	$\frac{-461586332999218}{981082973953595}$	b_6	b_7	$\frac{59}{585}$
b_j	b_1	$\frac{43330198141423}{1552245574212436}$	$\frac{126920317765990}{976320234585877}$	$\frac{144252338374735}{235812665300824}$	$\frac{-461586332999218}{981082973953595}$	b_6	b_7	$\frac{59}{585}$
$\hat{b}_j$	$\hat{b}_1$	$\frac{63525278823359}{589073924187652}$	$\frac{-1215341952797}{169743795871373}$	$\frac{568324990202744}{980157605573067}$	$\frac{-260265382870227}{560889253908905}$	$\hat{b}_6$	$\hat{b}_7$	$\hat{b}_8$

Table B3: ESDIRK4(3)8L[2]SA, with $a_{i1} = a_{i2}$, $b_1 = b_2$, $\hat{b}_1 = \hat{b}_2$, and $a_{8,j} = b_j$.

Five ESDIRK methods are compared herein: three fourth-order, a third-order and a second-order. The properties of the new eight-stage scheme as compared with the other four ESDIRK schemes are given in table B5. The methods being compared in the table B5 (and in table A1) are as follows:

$$c_3 = \frac{156018921355884}{2640838318719043} \quad a_{76} = \frac{-33068834936140}{1321803926597241} \quad b_6 = \frac{-274883779192603}{365924002944524} \quad b_7 = \frac{624128017493557}{471650707219883}$$

$$\hat{b}_6 = \frac{-700237699821775}{93074423579744} \quad \hat{b}_7 = \frac{1054294140731335}{793259632340454} \quad \hat{b}_8 = \frac{76832074920277}{776473806427012}.$$

Table B4: ESDIRK4(3)8L[2]SA: remaining coefficients

Name	ESDIRK 2(1)3L[2]SA	ESDIRK 3(2)4L[2]SA	ESDIRK 4(3)6L[2]SA_1	ESDIRK 4(3)7L[2]SA	ESDIRK 4(3)8L[2]SA
s	3	4	6	7	8
p	2	3	4	4	4
γ	$(2 - \sqrt{2})/2$	0.435867	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{59}{585}$
$A^{(p+1)}$	0.05719	0.03663	0.003401	0.000260	0.000337
$A^{(p+2)}$	0.07944	0.07870	0.005405	0.001177	0.001024
$\hat{A}^{(p)}$	0.02513	0.02552	0.000824	0.000301	0.000271
$\hat{A}^{(p+1)}$	0.07801	0.07418	0.004517	0.000977	0.000305
$\{B^{(5)}, C^{(5)}, E^{(5)}\}$	{3.11, 0.828, 2.28}	{2.91, 1.64, 1.44}	{5.48, 1.38, 4.13}	{3.24, 3.07, 0.861}	{1.13, 0.902, 1.25}
D	1.0	1.27	0.810	1.00	1.32
$\{\lambda_{\text{Min}}^{\text{M}}, \lambda_{\text{Min}}^{\text{M}}\}$	{-0.127, -0.114}	{-1.13, -2.34}	{-0.492, -0.565}	{-1.990, -1.361}	{-2.902, -2.971}
$\{b_{i,\text{Min}}, a_{ij,\text{Min}}\}$	{+0.293, -0.000}	{-0.595, -0.595}	{-0.275, -0.592}	{-0.557, -0.557}	{-0.751, -0.833}
$\lambda_{\text{Min}}^{\text{M}(2)}$	-0.086	-0.190	-0.312	-0.016	-0.010
$\lambda_{\text{Min}}^{\text{M}(3)}$	-0.127	-0.988	-0.097	-0.009	-0.006
$\lambda_{\text{Min}}^{\text{M}(4)}$	—	-1.13	-0.972	-0.314	-0.274
$\lambda_{\text{Min}}^{\text{M}(5)}$	—	—	-2.30	-1.225	-0.783
$\lambda_{\text{Min}}^{\text{M}(6)}$	—	—	-1.04	-0.204	-1.765
$\lambda_{\text{Min}}^{\text{M}(7)}$	—	—	—	-1.990	-1.267
$\lambda_{\text{Min}}^{\text{M}(8)}$	—	—	—	—	-2.902
$\{R(-\infty), \bar{R}(-\infty)\}$	{0.0, 0.0.293}	{0.0, 0.218}	{0.0, 0.0}	{0.0, 0.0}	{0.0, 0.0}

Table B5: ESDIRK methods

1) ESDIRK2(1)3L[2]SA from [24], 2) Implicit portion of ARK3(2)4L[2]SA [23], 3) Implicit portion of ARK4(3)6L[2]SA [23] and ESDIRK4(3)6L[2]SA_1 from [26], 4) ESDIRK4(3)7L[2]SA from [26], 5) ESDIRK4(3)8L[2]SA designed in this paper. The ESDIRK4(3)8L[2]SA scheme has better properties overall, except the algebraic instability on internal stages and the step.

B.2.3 Stage-value predictors

The β_{kj} coefficients for the ESDIRK4(3)8L[2]SA scheme are give in table B6. The SVPs are implemented using the formulae given in equation (10). The method corresponding to the reported coefficients is given in the last column of table B6.

B.2.4 Dense Output

The implementation coefficients: $\beta_{i,j}$ needed in equations (8) and (9) for the ESDIRK4(3)8L[2]SA scheme are given in table B7.

c_3	β_{31}	$\frac{1812329}{61352403}$	0	0	0	0	0	1
c_4	β_{41}	$\frac{50245319}{68549022}$	$-\frac{29595219}{28133372}$	0	0	0	0	1
c_5	β_{51}	$-\frac{85334134}{164083875}$	$\frac{520239157}{462388393}$	$\frac{38482782}{79429241}$	0	0	0	2
c_6	β_{61}	$-\frac{81820811}{52047104}$	$\frac{224142662}{74716127}$	$\frac{537}{1000}$	$\frac{7623}{20000}$	0	0	5
c_7	β_{71}	$-\frac{30577813}{36373682}$	$\frac{108338209}{62935626}$	$\frac{33363543}{84030943}$	$\frac{19865774}{71044047}$	$\frac{2740356}{108459265}$	0	1
c_8	β_{81}	$\frac{196155495}{88245572}$	$-\frac{362835506}{104934831}$	$-\frac{79435259}{57861274}$	$\frac{129849}{100000}$	$\frac{324093}{500000}$	$-\frac{56177}{100000}$	4

Table B6: SVP for ESDIRK4(3)8L[2]SA, with $\beta_{i1} = \beta_{i2}$ for $i = 3, 8$.

β_{1j}	$\frac{+4111165927}{17552424484}$	$\frac{-3065939197}{13865167531}$	$\frac{+93934989}{6339375476}$
β_{2j}	$\frac{+4111165927}{17552424484}$	$\frac{-3065939197}{13865167531}$	$\frac{+93934989}{6339375476}$
β_{3j}	$\frac{+2675205767}{11272080602}$	$\frac{-10926757293}{12093844160}$	$\frac{+685437919}{860923542}$
β_{4j}	$\frac{+22780857425}{6249027518}$	$\frac{-18619344673}{7554573043}$	$\frac{-7544868238}{13256738257}$
β_{5j}	$\frac{-63709336598}{11730941487}$	$\frac{+43463834873}{7292108227}$	$\frac{-1}{1}$
β_{6j}	$\frac{-31566902283}{13091756221}$	$\frac{+0}{1}$	$\frac{+83}{50}$
β_{7j}	$\frac{+61648759756}{14426552075}$	$\frac{-29}{20}$	$\frac{-3}{2}$
β_{8j}	$\frac{+509}{2340}$	$\frac{-7}{10}$	$\frac{+7}{12}$

(B9)

Table B7: Dense output for ESDIRK4(3)8L[2]SA.

Appendix C

Order Reduction

C.1 Van der Pol's Equation

Van der Pol's (vdP) equation describes a nonlinear oscillator with solutions that are damped (amplified) for large (small) values of y_1 , [16, 17]

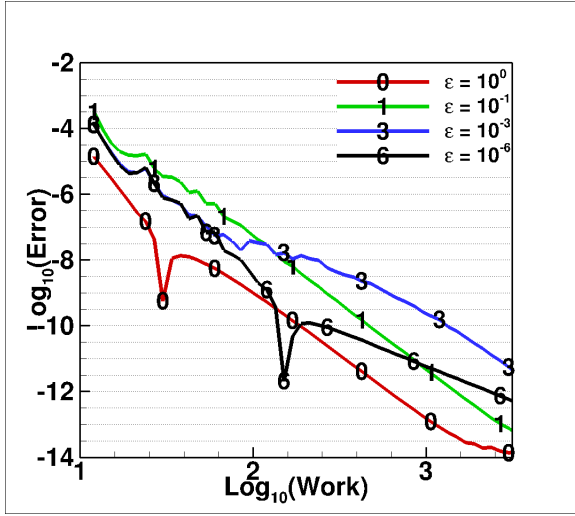
$$\dot{y}_1(t) = y_2(t), \quad \dot{y}_2(t) = \varepsilon^{-1} \left((1 - y_1(t)^2)y_2(t) - y_1(t) \right). \quad (\text{C1})$$

Unperturbed initial conditions are given by $y_1(0) = 2$, $y_2(0) = -0.6666654321121172$. Van der Pol's equation develops a very challenging boundary layer at time $T \approx 0.8$ based on these initial conditions, and is an effective test problem to quantify a method's susceptibility to order reduction [23, 25, 26].

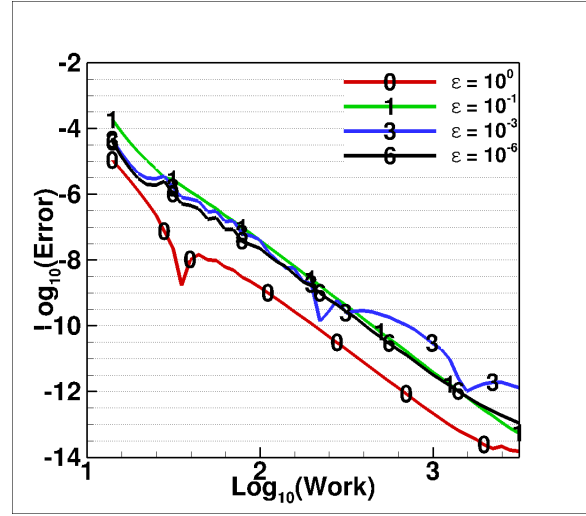
The stiffness parameter: ε appearing in Van der Pol's equation is used to arbitrarily adjust the dynamics of the system. As ε decreases, the ODE gradually transitions in behavior toward an index-1 differential-algebraic equation (DAE). For Runge-Kutta methods, accompanying this transition is an order-reduction phenomena where the observed convergence rates of the methods fall below the classical order of accuracy. Some differential variables transition to algebraic variables, displaying different convergence rates. Hairer et al. [17] determine the convergence rates of SDIRK methods with and without the stiffly accurate assumption. Global error for both differential and algebraic variables are of the form $\epsilon_{\text{global}} = c_1(\Delta t)^\alpha + c_2\varepsilon(\Delta t)^\beta$ for $\varepsilon \leq \text{Const.}(\Delta t)$. Independent of stiff accuracy, SDIRK methods have $\alpha = q$ and $\beta = q_{\text{so}} + 1$ for the differential variable, where q and q_{so} are the classical and stage-orders of the method. The convergence rate of the algebraic variables is $\alpha = q$ and $\beta = q_{\text{so}}$ for stiffly accurate methods.

Figure C1 compares the convergence behavior and order reduction of the ESDIRK437 and ESDIRK438 schemes. Figures C1a and C1b quantify the convergence behavior of the differential variable, while figures C1c and C1d quantify the convergence of the algebraic variable. Four stiffness parameters are shown in each plot. Note that $\varepsilon = 10^{-3}$ and 10^{-6} experience the most severe reduction in accuracy. Figures C1e and C1f show the rate of convergence of the differential and algebraic variables as a function of the stiffness parameter ε . Note that the differential variables of the ESDIRK437 and ESDIRK438 schemes order reduce in the DAE limit of small ε , to a rates of ≈ 3.0 and ≈ 3.6 , respectively. The algebraic variable convergence rates are similar between the two schemes with the ESDIRK438 scheme experiencing slightly less order reduction in the DAE limit.

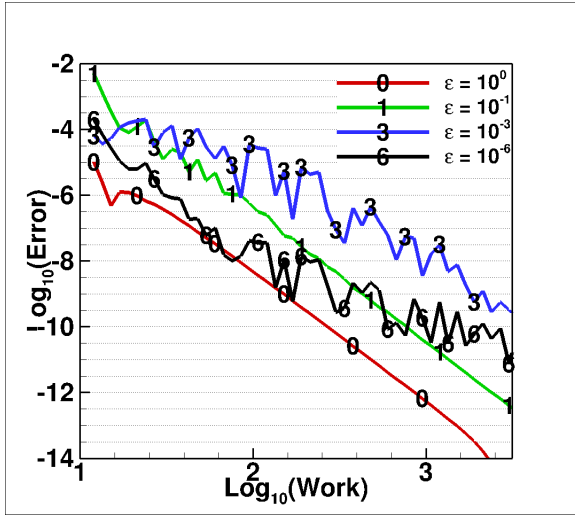
It was shown in [25, 26] that the ESDIRK437 scheme was best among several tested seven stage ESDIRK437 schemes and was clearly better than the ESDIRK436 scheme introduced in [23]. The ESDIRK438 scheme is as accurate, and is comparable if not less susceptible to order reduction than the ESDIRK437 scheme.



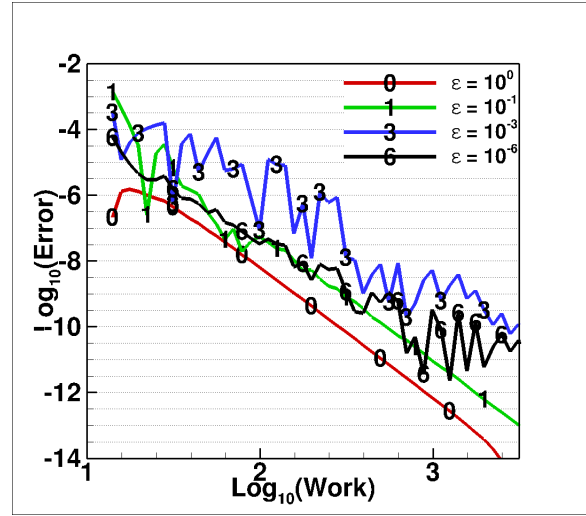
(a) Differential Variable Error: ESDIRK437



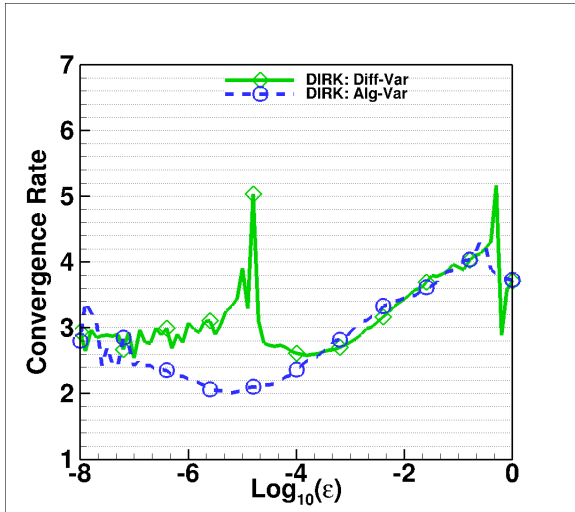
(b) Differential Variable Error: ESDIRK438



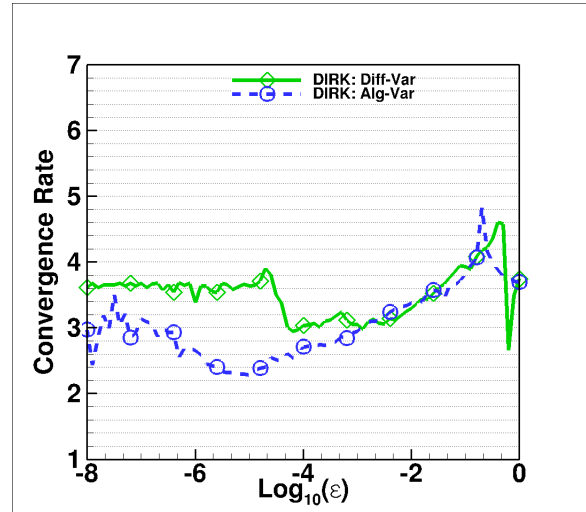
(c) Algebraic Variable Error: ESDIRK437



(d) Algebraic Variable Error: ESDIRK438



(e) Order Reduction: ESDIRK437



(f) Order Reduction: ESDIRK438

Figure C1: Comparison of order reduction: ESDIRK437 vs. new ESDIRK438