

**MULTIRATE METHODS FOR
HYPERBOLIC SYSTEMS -
NUMERICAL APPROXIMATION OF
FAST WAVES IN WEATHER FORECAST
MODELS**

DISSERTATION

zur Erlangung des akademischen Grades

Doctor rerum naturalium

(Dr. rer. nat.)

vorgelegt

dem Bereich Mathematik und Naturwissenschaften
der Technischen Universität Dresden

von

Andreas Naumann

geboren am 30.01.1985 **in** Jena

Eingereicht am 5. September 2019

Verteidigt am 31. März 2020

Gutachter:

1. Prof. Dr. Axel Voigt
2. Prof. Dr. Jörg Wensch
3. Prof. Dr. Martin Arnold

Diese Dissertation wurde in der Zeit von Oktober 2013 bis September 2019 am Institut für Wissenschaftliches Rechnen angefertigt.

ABSTRACT

The expected temperatures and rainfall in the next days to hours is one of the most important information nowadays. This knowledge is not only of general interest. Disciplines like agriculture and forestry the knowledge of the rain is even more important for a time span of weeks to plan the harvest or protect the plants [3]. Therefore, the possibility to forecast the weather reliably and fast is very important nowadays.

The reliability of weather forecast, or more accurate the numerical weather forecast, depends on several factors. One factor is the complexity of atmosphere models. Whereas the first numerical experiments treat the atmosphere as dry ideal gas with one layer, recent models incorporate the humidity, clouds, precipitation and radiation. But every higher detail in the model come at higher costs for simulation. Hence the development of finer grained models also require more advanced numerical methods to solve them.

The atmosphere models are in general a nonlinear hyperbolic set of partial differential equations (PDEs). In particular the models consist of several waves, traveling with different speeds with nearly no damping. Roughly speaking these varying velocities lead to the multiscale nature of the atmosphere models and a suitable numerical method should respect the different time scales.

The development and analysis of multirate methods for hyperbolic systems remains a challenging problem. Examples for class of hyperbolic systems of PDEs range from the scalar and linear advection equation, the wave equation to nonlinear systems like the shallow water equations or the (inviscid) Euler equations, which are the basis for the atmosphere models.

The hyperbolic PDEs often have an additive split structure, which in turn account for the different time scales. We assume a suitable, often finite volume, discretization in space. Hence we retain the additive splitting from the continuous problem in the semi-discretized ordinary differential equation (ODE). Hence we develop a new numerical method which accounts for the additive split structure and the multiscale nature.

The development of splitting methods is challenging in the analysis of the order conditions and the stability criteria. In particular the interaction between the fast and slow scales render the order conditions often complicated and unstructured. Furthermore every multiscale approach combines the scales in a different way, which is why there is

no unified order condition theory. With these challenges in mind we derive the order conditions in a classical way by differentiation of the numerical method. The splitting in a fast and a slow right hand side leads to several combinations of elementary differentials. And every differential has different non-standard coefficients, without any structure between these combinations. This loss in structure renders the numerical solutions of the order conditions quite complicated, and the analytical solutions are only possible in rare cases.

We develop a new class of multirate methods, which is parameterized by the fast scale solver. That new approach allows for a better generalization and simplifies several steps by unification. Nevertheless this new type of generalization has the advantage to associate the order conditions of the complete (macro scale) method with the structure of the underlying (micro scale) integrator.

The second challenge is the analysis of the (linear) stability of multirate methods. We also analyze the (linear) stability of the newly developed methods. Due to the splitting structure there are many different model problems possible. We deduce a model problem from a simplified system of hyperbolic PDEs. On top of these stability model problems we will construct the new methods. In analogy to the analysis of the order conditions, we unify the construction of the stability functions and highlight the differences due to the different fast scale integrators afterwards.

A good method does not only lead to low errors, but also has a large stability area. Hence we optimize the method parameters with respect to the stability area. In our case, the parameter set contains rational and real parameters. We circumvent the solution of a mixed-integer optimization problem by considering only some rational parameters and optimize for them independently. Nevertheless, we obtain several thousand sub problems.

Finally we consider two nonlinear benchmark problems. With these problems we analyze the accuracy and stability again and compare the efficiency with two reference multiscale methods.

ZUSAMMENFASSUNG

Die zu erwartenden Temperaturen und Regenmengen der folgenden Tage bis Stunden sind heutzutage eine der wichtigsten Informationen. Diese Kenntnis ist nicht nur von allgemeinem Interesse. Insbesondere Bereiche wie die Landwirtschaft und Forstwirtschaft sind die zu erwartenden Regenmengen selbst über einen langen Zeitraum von Wochen

von besonderen Interesse um zum Beispiel die Ernte oder den Schutz von Pflanzen zu planen [3]. Daher ist die Fähigkeit, das Wetter zuverlässig und schnell für ausreichend lange Zeiträume vorher zu sagen, wesentlich.

Die Zuverlässigkeit der Wettervorhersage, oder genau genommen der numerischen Wettervorhersage, hängt von mehreren Faktoren ab. Einer dieser Faktoren ist die Detailliertheit der Atmosphärenmodelle. Während die ersten numerischen Experimente die Atmosphäre als eine Schicht trockenen idealen Gases betrachteten, beinhalten aktuelle Modelle die Feuchte, Wolken, Niederschlag und Strahlung. Mit jedem zusätzlichen Detail steigt natürlich der Simulationsaufwand. Daher müssen parallel zur verbesserten Modellierung auch die numerischen Verfahren erweitert werden.

Im allgemeinen sind die Atmosphärenmodelle Systeme nichtlinearer hyperbolischer Differentialgleichungen (PDEs). Insbesondere beinhalten die Modelle Wellen unterschiedlicher Ausbreitungsgeschwindigkeit, welche nahezu nicht gedämpft werden. Diese unterschiedlichen Geschwindigkeiten sind die Grundlage für den Mehrskalencharakter der Atmosphärenmodelle. Eine effektive numerische Methode muss daher die unterschiedlichen Skalen adäquat behandeln.

Die Entwicklung und Analyse numerischer Mehrskalungsverfahren zur Lösung von Systemen hyperbolischer Differentialgleichungen ist herausfordernd. Beispiele für hyperbolische Systeme beginnen bei der einfachen skalaren linearen Advektionsgleichung, der Wellengleichung und enden bei nichtlinearen Systemen wie den Flachwassergleichungen oder den (reibungsfreien) Eulergleichungen. Letztere sind die Grundlage für alle Atmosphärenmodelle.

Viele hyperbolische PDEs besitzen eine additive Struktur, wobei die Aufteilung gerade den Zeitskalen entsprechen. Wir gehen von einer angepassten Diskretisierung im Raum, in der Regel eine Finite-Volumen Diskretisierung, aus. Diese Diskretisierung erhält die additive Struktur des kontinuierlichen Problems in der (ortsdiskretisierten) gewöhnlichen Differentialgleichung (ODE). Daher entwickeln wir eine neue numerische Methode zur Lösung gewöhnlicher Differentialgleichungen, welche die additive Struktur und gleichzeitig die zugehörigen Zeitskalen ausnutzt.

Die Analyse von Splittingverfahren ist herausfordernd sowohl in der Entwicklung der Ordnungsbedingungen als auch der Stabilitätskriterien. Jeder Mehrskalensatz kombiniert die unterschiedlichen Zeitskalen auf unterschiedliche Art und Weise. Daher gibt es keine einheitliche Ordnungs- und Stabilitätstheorie. Wir entwickeln die Ordnungsbedingungen auf klassischem Wege, durch Differentiation der numerischen Lösung. Die Aufteilung der rechten Seite in schnelle und langsame Terme führt auf zusätzliche Koeffizienten und Kombinationen der elementaren Differentiale. Im Vergleich zu klassischen Verfahren

hat jedes elementare Differential unterschiedliche nicht-klassische Koeffizienten, ohne erkennbare Struktur. Dieser Strukturverlust erschwert die numerische Lösung zusätzlich. Analytische Lösungen gibt es nur in Sonderfällen.

Wir entwickeln und analysieren eine neue Klasse von Mehrskalennethoden, welche mit dem Integrator der schnellen Skala parametrisiert ist. Dieser neue Ansatz erlaubt die Verallgemeinerung der Ausgangsmethode und vereinfacht etliche Schritte in der Herleitung der Ordnungsbedingungen. Zusätzlich hat die Verallgemeinerung auch den Vorteil, die Ordnungsbedingungen des Gesamtverfahrens und die Struktur des darunter liegenden Lösers der schnellen Zeitskala zu assoziieren.

Wir untersuchen ebenfalls die lineare Stabilität der neuen Methoden. Aufgrund der Aufteilung in langsame und schnelle Terme gibt es viele verschiedene Modellprobleme. Wir leiten ein Modellproblem auf Basis eines vereinfachten hyperbolischer PDEs her. Auf Basis dieses Stabilitätsproblems konstruieren wir die neuen Methoden und untersuchen ihre Effizienz anhand zweier nichtlinearer Benchmarkprobleme. Analog zur Herleitung der Ordnungsbedingungen vereinheitlichen wir die Konstruktion der Stabilitätsfunktionen und heben im nachhinein die Unterschiede aufgrund des fast-scale integrators hervor.

Gute numerische Methoden führen nicht nur zu einem kleinen Fehler, sondern haben auch ein großes Stabilitätsgebiet. Daher optimieren wir die Methodenparameter im Hinblick auf die Größe des Stabilitätsgebiets. Unsere neuen Methoden besitzen sowohl reelle, als auch rationale Parameter. Die Lösung des gemischten ganzzahligen-reellen Optimierungsproblems vereinfachen wir durch die Auswahl einzelner rationaler Parameter. Dadurch erhalten wir allerdings einige tausend unabhängige Teilprobleme.

Zum Abschluss analysieren wir die Effizienz der neuen Methoden anhand zweier nichtlinearer Benchmarkprobleme und vergleichen die Genauigkeit und Stabilität mit Referenzverfahren.

ACKNOWLEDGMENTS

First of all I acknowledge the support of my supervisor Jörg Wensch. In particular his introductions and the time for lengthy discussions in this research area. Furthermore this work would not have been done without the funding in the SFB Transregio 96. Many thanks go to the whole institute of scientific computing for the good time and stimulating inspiration. In particular Dr. Simon Praetorius had many hints and ideas for the software and technical parts and Dr. Michael Nestler for his proof readings and comments.

Last, but not least, I want to thank my family for the support all the time.

CONTENTS

I	INTRODUCTION	11
i.1	Motivation	11
i.2	Multi scale additive and partitioned ODEs	13
II	MULTIRATE METHODS	20
ii.1	Multirate infinitesimal step methods	21
ii.2	Multirate finite step methods	30
ii.2.1	Unified notation of MFS methods	31
ii.2.2	Explicit Euler as fast scale integrator	32
ii.2.3	Forward-backward Euler as fast scale integrator	32
ii.2.4	Störmer-Verlet as fast-scale integrator	33
ii.3	Order conditions	34
ii.3.1	Extended notation and common symbols	35
ii.3.2	Derivatives of the numerical solution	37
ii.3.3	Common order conditions of MFS methods	46
ii.3.4	Explicit Euler as fast scale integrator	48
ii.3.5	Forward-backward Euler as fast scale integrator	50
ii.3.6	Störmer-Verlet as fast scale integrator	53
ii.4	Interpretation as GARK method	55
ii.5	Stability	59
ii.5.1	Model problem	60
ii.5.2	Stability function	62
ii.6	Method construction	68
ii.7	Stability optimization for methods of order two	70
ii.7.1	MFS2 methods	71
ii.7.2	Numerical experiments	72
ii.7.3	Conclusions	78
III	NUMERICAL EXPERIMENTS	79
iii.1	Method construction with linear problems	79
iii.1.1	First order fast scale integrators	79
iii.1.2	Second order fast scale integrator	87
iii.1.3	Conclusions	89

III.2 Bubbles and externally driven flows	90
III.2.1 The force driven flow	90
III.2.2 The cold bubble down burst experiment	94
IV CONCLUSIONS AND OUTLOOK	98
IV.1 Conclusions	98
IV.2 Outlook	99
Appendices	106
.1 Order conditions for Störmer-Verlet	107
A METHOD PARAMETERS	110
A.1 explicit Euler	110
A.2 Forward-backward Euler	112
A.3 Störmer-Verlet	115

NOMENCLATURE

- \mathcal{L} Diagonal matrix of total number of steps for every stage.
- $f_{\mathbf{y}\mathbf{y}} \langle F, F \rangle$ Second derivative of the slow tendencies with respect to all variables \mathbf{y} applied to F
- $f_{\mathbf{y}} \langle F \rangle$ First derivative of the slow tendencies applied to F .
- $\hat{Z}_{\hat{i}}$ stacked vector of fast stages, see (II.3.3c) and (II.3.4b)
- $\hat{\mathbf{L}}$ diagonal matrix of (rational) step factors
- \mathbf{e}_i i th basis vector of dimension $s + 1$
- $\mathbf{e}_{N,k}$ k th unit vector of dimension N , i.e. $[\mathbf{e}_{N,k}]_j = \delta_{kj}$ for $k, j \in [1, N]$.
- $F(\mathbf{y})$ right hand side, see equation (I.2.0)
- $f(\mathbf{y})$ slow part of the flux, see equation (I.2.0)
- $g(\mathbf{y})$ fast part of the flux, see equation (I.2.0)
- \mathbf{I} Identity matrix of dimension $(s + 1) \times (s + 1)$
- $\mathbf{y}(t)$ exact solution, see equation (I.2.0)
- IVP initial value problem
- MFS multirate finite steps
- PDE partial differential equation

The sum with diagonal matrix valued indices has a block wise meaning, i.e.

$$\left[\sum_{\hat{l}=1}^{\hat{\mathbf{L}}} g(\hat{Z}_{\hat{l}}) \right]_j = \sum_{l=1}^{\hat{\mathbf{L}}_{jj}} \mathbf{e}_j \otimes g(Z_{n,j,l}).$$

Hence the diagonal entries represent the summation steps, the row in the matrix represents the stage. Then we compute the sum for every stage independently and stack the results on over another.

We also use four different L-shaped symbols, which all relate to the step. The symbol $\hat{\mathbf{l}}$ is a diagonal matrix, which contains the actual step for every stage. The capital L shaped symbols represent the number of steps. They are related by

$$\mathfrak{L} = \mathbf{L}\hat{\mathbf{L}}$$

where \mathbf{L} is the common step factor, $\hat{\mathbf{L}}$ is a diagonal matrix containing the step factors for every stage and \mathfrak{L} contains the total number of steps for every stage on the diagonal.

The derivation of the order conditions requires a lot of symbols. Related symbols have a common base symbol, but differ in shape or size, for example the L-shaped and g-shaped symbols.

symbol	dim	description	
\mathbf{L}	0	common factor of number of steps	step related
$\hat{\mathbf{l}}, \mathfrak{L}, \hat{\mathbf{L}}$	$(s+1)x(s+1)$	actual step, total number of steps, step ratios	
$\alpha, \gamma, \beta, \mathbf{D}$	$(s+1)x(s+1)$	method parameters	derived from method parameters
$\mathbf{R}, \hat{\mathbf{A}}$	$(s+1)x(s+1)$		
$\mathbf{c}, \tilde{\mathbf{c}}, \mathbf{b}, \tilde{\mathbf{b}}$	$s+1$		
$\mathbf{Q}_i(\hat{\mathbf{l}})$	$(s+1)x(s+1)$	matrix valued coefficient	coefficient
\mathbf{Q}_i^S	$(s+1)x(s+1)$	sum of $\mathbf{Q}_i(\hat{\mathbf{l}})$	
\mathbf{Q}_i^I	$(s+1)x(s+1)$	$\mathbf{Q}_i(\hat{\mathbf{l}})$ independent of $\hat{\mathbf{L}}$	
\mathfrak{G}_1^y	-	sum of derivative depending on $\mathbf{f}_y \langle \mathbf{F} \rangle$	derivatives of \mathbf{g}
$\mathfrak{G}_1^{\hat{\mathbf{l}}}$	-	sum of derivative depending on step $\hat{\mathbf{l}}$	
\mathfrak{G}_2	-	sum of second derivatives of \mathbf{g}	

I | INTRODUCTION

I.1 MOTIVATION

The weather forecast is one of the most important tools in recent economies. The success in disciplines like agriculture, forestry and aviation, depend heavily on the reliable knowledge of expected rainfall and temperature in advance. Much more importantly are the expected times and locations of weather extremes like hail, twister and thunderstorms. If these are known in advance, the plants might be protected earlier or the time of harvest could be chosen accordingly.

The detailed evolution of weather phenomena, and on a larger scale the climate, on earth depends on the interaction between very many physical structures. In the simplest model consider only the lowest atmosphere layer, namely the troposphere, as an inviscid dry ideal gas. More advanced models also consider the interaction between the atmosphere, oceans, volcanic activities, vegetation and many more processes [45].

The basic nature of the models are already visible in the simplified one layer atmosphere model. In this model, we treat the air as an ideal gas in one dimension without friction. Then we have conservation laws for the mass, momentum and total energy, i.e.

$$\begin{aligned}\partial_t \rho &= -(\rho u)_x \\ \partial_t(\rho u) &= -(\rho u^2 + P)_x \\ \partial_t E &= -(u(E + P))_x \\ E &= \rho e(T, \rho) + \frac{1}{2} \rho u^2.\end{aligned}$$

The variables ρ , u and P refer to density, velocity and pressure. These equations form a system of nonlinear partial differential equations (PDE) of hyperbolic type. At this point we have four unknowns but only three equations. Hence the full system is closed with the equations of state $P = \rho R T$ and $e(T, \rho) = c_v T$ for an ideal polytropic gas and the constant R is the ideal gas constant. This full PDE system is general known as the Euler equations for a polytropic gas [30]. We get some deeper inside in the time scales by

computing the eigenvalues of the Jacobian of the right hand side. These eigenvalues are $(\mathbf{u}, \mathbf{u} + \mathbf{c}_s, \mathbf{u} - \mathbf{c}_s)$ [30, Sec 14.8], where $\mathbf{c}_s = \sqrt{\frac{c_p}{c_v} RT}$ is the speed of sound [9].

Hence the full system features two time scales. The slow scale belongs to the mean velocity \mathbf{u} and the fast one to the speed of sound \mathbf{c}_s . For dry air the speed of sound $\mathbf{c}_s \approx 300 \frac{\text{m}}{\text{s}}$, whereas winds in Germany seldom approach $50 \frac{\text{m}}{\text{s}}$.

Let us view these numbers in the context of weather forecast. The basis of atmosphere model is the compressible nonhydrostatic Euler equations, but often extended by models for clouds, humidity, precipitation and many more processes [41, 12, 51, 25, 8]. The general aim of weather forecast is the prediction of temperature and winds on a time span of several days and weeks. Hence a suitable numerical solver for atmospheric processes should solve the (semi-discretized) systems on a macro time scale in the order of the wind speeds, which is six to 10 times lower than the speed of sound.

Up to now we considered only a system of hyperbolic PDEs. But also parabolic PDEs (and systems thereof) can contain multiple scales. The thermo-elastic models for machine tools [14] connect two processes. One is the dissipation of heat in a solid, and the other one is the distortion due the change in temperature. Let us focus on the first part. We model the dissipation of heat inside the solid and the heat exchange between coupled machine parts with the system of coupled heat equations, i.e.

$$\begin{aligned} \rho_i C_{p_i} \partial_t T_i - \nabla \cdot \lambda_i \nabla T_i &= S_i \\ \lambda_i \mathbf{n}_{i,j} \nabla T_i &= \dot{Q}_{i,j} \quad \text{on } \Gamma_{ij} \end{aligned}$$

with the material parameters density ρ_i , heat capacity at constant pressure C_{p_i} and heat conduction λ_i . Additionally we have flux boundary conditions on the surfaces Γ_{ij} . Every machine part has its own temperature field T_i with inner thermal sources S_i and boundary sources Q_{ij} on the boundary Γ_{ij} . The aforementioned boundary sources also serve as a thermal connection between the machine tool parts. If two connected parts move relative to each other, we obtain a time dependent source term [35]. In turn we have two time scales. One time scale is given by the time the heat conduction requires to reach steady state. This time $t_{\text{cond}} \approx \frac{d^2 \rho_i C_{p_i}}{\lambda_i}$ is given by a characteristic diameter d of the machine tool part and the material parameters. The other time scale is given by the relative velocity \mathbf{u} between the coupled components. This time we have a characteristic time $t_{\text{move}} \approx \frac{d}{u}$ which depends on the characteristic moving distance. Whereas the movement is quite fast, the heat conduction to a nearly stationary state takes a long time. Hence the temperature fields in the vicinity of the moving surfaces will change considerably faster

then somewhere far away. A problem adapted method, which takes these dependencies into account, can considerably speed up the numerical solution [36].

These two examples give an imagination where multi scale problems might arise. At the same time they also belong to different mathematical problem classes. Whereas the Euler equations form a hyperbolic system, the system of coupled heat equations belong to the parabolic PDEs. Hence the solution of the latter system is in general smoother and small errors during the numerical solution process will be damped. In contrast the hyperbolic system does not damp anything and small errors might accumulate over time. Therefore both systems require different multirate approaches. In the remaining chapters we will concentrate on the hyperbolic system. In more detail we are interested in the numerical solution of the PDE after discretization in space. Hence, we assume that all space operators are discretized with a suitable method, like finite volumes, finite differences or even finite elements. The resulting initial value problems (IVP) will also contain the aforementioned time scales.

On top of these IVPs we will develop a class of multirate methods, which utilize the additive and partitioned structure of the semi discretized ODEs.

1.2 MULTI SCALE ADDITIVE AND PARTITIONED ODES

In the previous motivation, we presented several models containing at least two time scales. From a more general point of view, we started from a PDE with initial and boundary conditions. In turn we obtain an IVP

$$\dot{\mathbf{y}} = \mathbf{F}(\mathbf{y}) = \mathbf{f}(\mathbf{y}) + \mathbf{g}(\mathbf{y}) \in \mathbb{R}^d \quad (2.1a)$$

$$\mathbf{y}(t_0) = \mathbf{y}_0 \quad (2.1b)$$

with an additive structure. The additive splitting of the right hand side in a slow and fast functions \mathbf{f} and \mathbf{g} correspond to different operators in space and at the same time they resemble the time scales. In the following chapters, we assume that the slow function \mathbf{f} is responsible for the long time behavior, whereas the function \mathbf{g} is responsible for the short time scales.

In several applications the model exhibits an additional structure in the states \mathbf{y} . Therefore we will also consider the so called partitioned additive split ODE

$$\mathbf{y} = (\mathbf{y}^p, \mathbf{y}^q) \quad (2.2a)$$

$$\dot{\mathbf{y}}^p = \mathbf{f}^p(\mathbf{y}^p, \mathbf{y}^q) + \mathbf{g}^p(\mathbf{y}^p, \mathbf{y}^q) \quad (2.2b)$$

$$\dot{\mathbf{y}}^q = \mathbf{f}^q(\mathbf{y}^p, \mathbf{y}^q) + \mathbf{g}^q(\mathbf{y}^p, \mathbf{y}^q) \quad (2.2c)$$

where we decomposed the vector \mathbf{y} in two components $\mathbf{y}^p \in \mathbb{R}^{d_1}$ and $\mathbf{y}^q \in \mathbb{R}^{d_2}$ with $d_1 + d_2 = d$. Note the general dependency of the fast tendency \mathbf{g} on both components. In particular the Euler equations from the prevalent section admit a partitioning in the partitions $(\rho\mathbf{u})$ and $(\rho, \rho\mathbf{E})$. Hence in the general case, we have to extend the commonly used partitioned methods *symplectic Euler* and *Störmer-Verlet* to this problem type.

There is an interesting survey on these particular partitioned methods by Hairer et al. [18]. Both methods are strongly connected, because we can write the last one as a composition of two steps of the former. Furthermore both methods remain explicit for decoupled partitioned problems, where the right hand sides depend only on one, but not both components. This property makes them especially efficient. Nevertheless there are extensions to the implicit coupled problems, i.e. $\dot{\mathbf{y}}^p = \mathbf{f}^p(\mathbf{y}^p, \mathbf{y}^q)$. Depending if the “own” component is used in an explicit or implicit way these extensions might lose the advantage of explicit methods and require the solution of non linear systems in every fast step.

The numerical solution of partitioned and additive split problems with one step methods is very well understood. In the case of partitioned problems, the P-trees [17] represent the derivatives of the numerical (and analytical) solution. From these follow the order conditions for partitioned one step methods in analogy to the famous B-trees for classical problems. For additive split problems [24, 38] there are the N-trees [4]. Furthermore both problem structures are very related [5]. A partitioned problem

$$\dot{\mathbf{y}}^i = \mathbf{f}^i(\mathbf{y}^1, \dots, \mathbf{y}^N) \quad (2.3a)$$

consisting of N partitions can be cast into an additive split problem

$$\dot{\mathbf{u}} = \sum_{k=1}^N \mathcal{F}_k(\mathbf{u}) \quad (2.3b)$$

with N right hand side functions f_k . Here we distinguish between the partitioned right hand side and the additive right hand sides by super and subscripts respectively. Assume we have the partitioned problem (2.3a). Then we construct the block vector

$$\mathbf{u} = \sum_{k=1}^N \mathbf{e}_{N,k} \otimes \mathbf{y}^k = \begin{pmatrix} \mathbf{y}^1 \\ \mathbf{y}^2 \\ \vdots \\ \mathbf{y}^N \end{pmatrix} \quad (2.3c)$$

and obtain the (subscripted) additive functions

$$\mathcal{F}_k(\mathbf{u}) = \mathbf{e}_{N,k} \otimes f^k(\mathbf{u}) = \begin{pmatrix} 0 \\ f^k(\mathbf{u}) \\ \vdots \\ 0 \end{pmatrix}. \quad (2.3d)$$

This additive structure is also named *coordinate partitioning* by Sandu and Günther [38].

Instead if we start from an additive split problem (2.3b) we obtain the corresponding partitioned problem by introducing auxiliary variables \mathbf{y}^i , such that

$$\mathbf{u} = \sum_i \mathbf{y}^i \quad (2.3e)$$

and require the IVP

$$\dot{\mathbf{y}}^i = f^i(\mathbf{y}^1, \dots, \mathbf{y}^N) = \mathcal{F}_i \left(\sum_k \mathbf{y}^k \right) \quad (2.3f)$$

for every index $i = 1, \dots, N$. We will use these connections when we compare the derivation of order conditions with different methodologies in section II.4.

The problem (2.1) is merely an algebraic specialization to a general ODE $\dot{\mathbf{y}} = F(\mathbf{y})$. But the splitting itself does not give any qualitative information about the solution. Hence we go one step back and consider linear hyperbolic PDEs. From these PDEs we obtain the physical meaning of slow and fast. A conservation law has the (flux) form

$$\mathbf{q}_t + \nabla \cdot \mathbf{f}(\mathbf{q}) = 0 \quad (2.4)$$

where the quantity q is conserved and the function $f : \mathbb{R} \rightarrow \mathbb{R}^n$ models the flux [30, 9, 53, 13]. Prominent (and by far not exhaustive) examples for these equations are

- conservation of mass using $q = \rho$. The flux function $f(q) = \bar{u}\rho$ is the mass flux. With a constant velocity $\bar{u} = U\bar{e}_x$ the equation reduces to the one dimensional advection equation

$$q_t + Uq_x = 0. \quad (2.5)$$

- conservation of momentum using $q = \rho\bar{u}$. This time the conserved quantity q is a vector and the flux function is $f(\rho\bar{u}) = \rho u^2 + P$ where P is the pressure and in general a function of the density and temperature.

Despite the simplicity the advection equation (2.5) shows the an important property of hyperbolic problems. The solution $q(t_e, x) = q_0(x - Ut)$ at time $t = t_e$ is the initial profile q_0 , shifted by Ut_e .

Let us now increase the complexity a little bit by considering vector valued problems of size d in 1D, but keep the flux function f linear. In this case we obtain the conservative system

$$q_t + Aq_x = 0 \quad (2.6)$$

where the matrix $A \in \mathbb{R}^{d \times d}$ is diagonalizable with real eigenvalues $\lambda^1, \dots, \lambda^d$ [13]. Hence there exist orthogonal transformations R , such that $RA = \Lambda R$. Let us therefore change the variables by $w = Rq$. These transformed variables satisfy the PDE

$$w_t + \Lambda w_x = 0,$$

which is simply a decoupled system of advection equations with velocities $\lambda^1, \dots, \lambda^d$. In particular the largest in magnitude eigenvalue λ determines the maximal time step size of an explicit time integrator.

The extension to nonlinear hyperbolic system is formally straight forward. A PDE of the form (2.4) is hyperbolic if the Jacobian of f is diagonalizable with real eigenvalues. In analogy to the linear case the eigenvalues are the velocities. But in the nonlinear case these velocities depend on the state, which can lead to shocks, rarefaction waves and other non smooth solutions structures [13].

Meteorological models exhibit several different wave types [2, 31, 10]. Let us consider the two dimensional stratified isentropic Euler equations, i.e

$$\partial_t \rho + \partial_x(\rho u) + \partial_z(\rho w) = 0 \quad (2.7a)$$

$$\partial_t u + u \partial_x u + w \partial_z u + \frac{1}{\rho} \partial_x P = 0 \quad (2.7b)$$

$$\partial_t w + u \partial_x w + w \partial_z w + \frac{1}{\rho} \partial_z P = -g \quad (2.7c)$$

$$\partial_t \theta + u \partial_x \theta + w \partial_z \theta = 0 \quad (2.7d)$$

with the gravitational constant $g = 9.81 \frac{m}{s^2}$. The potential temperature θ is defined as $\theta = T \left(\frac{P}{P_0} \right)^{-\frac{R}{c_p}}$ with a reference pressure P_0 . Usually one uses $P_0 = 100 \text{ hPa}$. This temperature is constant in the absence of heat sources during an adiabatic processes. In particular it does not change due to vertical pressure (and density) variations.

We model the atmosphere as an ideal gas, hence the pressure $P = \rho RT$ is linear in the temperature T and the density ρ . From the definition of the potential temperature θ we obtain the diagnostic equation

$$P = P_0 \left(\frac{R \rho \theta}{P_0} \right)^{\frac{c_p}{c_v}}$$

for the pressure in relation to the density and the potential temperature. The momentum equations depend on the pressure gradient scaled with the inverse density. To overcome this relation we introduce the Exner pressure

$$\pi = \left(\frac{P}{P_0} \right)^{\frac{R}{c_p}}$$

, which relates to the (scaled) pressure gradient

$$\frac{1}{\rho} \nabla P = c_p \theta \nabla \pi.$$

Putting everything together we end up with the two dimensional inviscid Euler equations

$$\begin{aligned}
 \partial_t \mathbf{u} + \mathbf{u} \partial_x \mathbf{u} + w \partial_z \mathbf{u} + C_p \theta \partial_x \pi &= 0 \\
 \partial_t w + \mathbf{u} \partial_x w + w \partial_z w + C_p \theta \partial_z \pi &= -g \\
 \partial_t \theta + \mathbf{u} \partial_x \theta + w \partial_z \theta &= 0 \\
 \partial_t \pi + \mathbf{u} \partial_x \pi + w \partial_z \pi + \frac{R}{C_v} \pi (\partial_x \mathbf{u} + \partial_z w) &= 0
 \end{aligned}$$

where the prognostic equation for the Exner pressure π follows from the mass balance, the prognostic equation (2.7d) and the definition of π . Hence one can view the pressure equation as a replacement for the density. These equations are easier to analyze.

The multiscale character gets best visible with the dispersion relation. Therefore we linearize the system around an isothermally stratified state, i.e.

$$\begin{aligned}
 \theta(\mathbf{t}, \mathbf{x}, z) &= \theta_0 + \theta_1(\mathbf{t}, \mathbf{x}, z) \\
 \mathbf{u}(\mathbf{t}, \mathbf{x}, z) &= \mathbf{U} + \mathbf{u}_1(\mathbf{t}, \mathbf{x}, z) \\
 w(\mathbf{t}, \mathbf{x}, z) &= w_1(\mathbf{t}, \mathbf{x}, z) \\
 \pi(\mathbf{t}, \mathbf{x}, z) &= \pi_0(z) + \pi_1(\mathbf{t}, \mathbf{x}, z)
 \end{aligned}$$

with background horizontal velocity \mathbf{U} and vanishing zeroth order vertical velocity. Furthermore we assume the hydrostatic balance

$$C_p \theta_0 \partial_z \pi_0 = -g$$

at zeroth order. After that we obtain the quasi-linear system

$$\begin{aligned}
 \partial_t \mathbf{u}_1 + \mathbf{U} \partial_x \mathbf{u}_1 &= -C_p \theta_0 \partial_x \pi \\
 \partial_t w_1 + \mathbf{U} \partial_x w_1 &= -C_p \theta_0 \partial_z \pi + g \frac{\theta_1}{\theta_0} \\
 \partial_t \theta_1 + \mathbf{U} \partial_x \theta_1 &= -\frac{N^2 \theta_0}{g} w_1 \\
 \partial_t \pi_1 + \mathbf{U} \partial_x \pi_1 &= -\frac{R \pi_0}{C_v} (\partial_x \mathbf{u}_1 + \partial_z w_1) + w_1 \frac{g}{C_p \theta_0}
 \end{aligned}$$

in the small perturbations \mathbf{u}_1 , w_1 , θ_1 and π_1 of the horizontal, vertical, potential temperature and Exner pressure respectively. The constant N represents the Brunt-Väisälä frequency, i.e. $N^2 = \frac{g}{\theta_0} \partial_z \theta_0$. We moved the terms responsible for the advection to the left of the equation and the remaining terms to the right. To construct the dispersion relation,

we consider wave type solutions of the form $(u_1, w_1, \theta_1, \pi_1)e^{kx+lz-\omega t}$ with frequency ω and the horizontal and vertical wave numbers k and l . In isothermal processes, the speed of sound $c_s = \sqrt{\frac{C_p RT}{C_v}}$ is constant and the linearized system exhibits the dispersion relation

$$(\omega - Uk)^2 = \frac{c_s^2}{2} \left(k^2 + l^2 + \frac{N^2 + S^2}{c_s^2} \right) \pm \frac{c_s^2}{2} \left[\left(k^2 + l^2 + \frac{N^2 + S^2}{c_s^2} \right)^2 - 4 \frac{N^2 k^2}{c_s^2} \right]^{\frac{1}{2}},$$

with $S = c_s \left[\frac{1}{2\rho_0} \frac{d\rho_0}{dz} + \frac{1}{\theta_0} \frac{d\theta_0}{dz} \right]$. Hence the dispersion relation is nonlinear in k and l . In other words, we have different wave speeds for different wavelengths. A wave with wavelengths (k, l) , travels with velocity $\left(\frac{\omega(k,l)}{k}, \frac{\omega(k,l)}{l} \right)$ in horizontal and vertical direction, whereas the wave packages travels with velocity $(\partial_k \omega(k, l), \partial_l \omega(k, l))$. Hence a wave package consisting of several wavelengths disperses with time. Furthermore we also see that the wave velocities do not correspond to the characteristics due to the non differentiated terms on the right.

II

MULTIRATE METHODS

There is a vast variety of numerical methods for multirate problems. Similar to the classical ODE methods, multirate methods fall again in two classes, namely the multistep and one step methods. A very early approach by Gear and Wells [11] bases on multistep methods. Methods from this class make use of the underlying polynomial and can interpolate and extrapolate with sufficient high accuracy.

The other method class are the one step methods. For stiff parabolic systems the special sub class of linearly implicit Rosenbrock methods were extended to so called multirate Rosenbrock methods (MROS) by Günther and Rentrop [15]. Again they used the *slow-first* strategy. The basis for the extension is the extrapolation of the fast values and interpolation of the slow values afterwards. Different extrapolation and interpolation strategies were tested by Savcenca et al. [39], this time on a reaction diffusion equation. Quite recently Kuhn [29] analyzed the stability interpolation approach by defining continuous extensions of Rosenbrock methods. The multirate extension still bases on a component partitioning, in this particular case even a coordinate partitioning. The additive structure of the problems was not exploited, even so a component dependency.

The basis for the order conditions of the multirate Rosenbrock are the P-series by Hairer [17]. Hence the local error, and therefore the order, for the multirate RK method are known for a long time. The crucial and complicated part of these methods are the stability conditions. Whereas classical RK methods can be analyzed with a scalar test equation, see the famous contribution [6], a multirate method requires a system with at least two components. Due to the multirate approach, the method treats every component in a different way. Hence the stability function does not only depend on the eigenvalue of every subsystem and the coupling blocks, but also on their eigenvectors. This fact was explored by Skelboe and Andersen [42] in the context of waveform relaxation methods with the backward Euler method.

The class of PEER method [37, 49] combines the multistep with one-step methods, hence belong to the class of general linear methods. There are at least two extensions of PEER methods to multirate problems. The last chapter in Kuhn [29] uses several sub steps for the fast components. This approach is very sensitive to the interpolation methods of the slower components. On the other hand Jebens et al. [23] exploit the additive

fast-slow structure. Whereas the former approach leads to simpler order conditions, the latter one required additional conditions for the coupling between the fast and slow terms. We extended these methods in two directions in 2017. First we added the coefficients γ to reach order higher than two and second we specialized the method with the several fast explicit Euler steps.

The remaining chapter concentrates on the class of one step methods, particularly the *multirate infinitesimal step* (MIS) and the new *multirate finite step* (MFS) methods. We exploit the additive and partitioned structure of the model problem (I.2.2) and derive the order conditions up to order three. Afterwards we derive a test equation for the stability analysis, which is related to the Euler equations from the introduction.

II.1 MULTIRATE INFINITESIMAL STEP METHODS

The multirate infinitesimal steps (MIS) methods were first introduced by [27]. The basis of MIS methods are explicit RK methods. Whereas a RK method evaluates the fast and the slow terms together, and at the same time points, the MIS approach introduces an auxiliary ODE on the macro scale which accounts for the fast tendencies. Thus an MIS method is given by [52]

$$\mathbf{Z}_{n,i}(0) = \mathbf{y}_n + \sum_j \alpha_{ij}(\mathbf{Y}_{n,j} - \mathbf{y}_n) \quad (1.1a)$$

$$\partial_\tau \mathbf{Z}_{n,i} = \frac{1}{h} \sum_j \gamma_{ij}(\mathbf{Y}_{n,j} - \mathbf{y}_n) + \sum_j \beta_{ij} f(\mathbf{Y}_{n,j}) + \mathbf{D}_{ii} \mathbf{g}(\mathbf{Z}_{n,i}(\tau)) \quad (1.1b)$$

$$\mathbf{Y}_{n,i} = \mathbf{Z}_{n,i}(h) \quad (1.1c)$$

$$\mathbf{y}_{n+1} = \mathbf{Y}_{n,s+1}. \quad (1.1d)$$

The coefficients and in particular the auxiliary IVP for $\mathbf{Z}_{n,i}$ requires a deeper explanation. For the beginning we neglect the coefficients α and γ . In this case the auxiliary IVP for the $\mathbf{Z}_{n,i}$ starts at the old value \mathbf{y}_n and fixes the slow tendencies at the previous stages, but depends on the recent fast tendencies in the whole macro interval $[t_n, t_n + h]$. In the absence of fast tendencies the ODE (1.1b) has the trivial solution $\mathbf{Z}_{n,i}(h) = \mathbf{y}_n + h \sum_j \beta_{ij} f(\mathbf{Y}_{n,j})$ and one recovers the so called underlying RK method with $\hat{\mathbf{A}} = \beta$. Last but not least we scale the fast tendencies using the coefficient \mathbf{D}_{ii} . These coefficients allow for a different

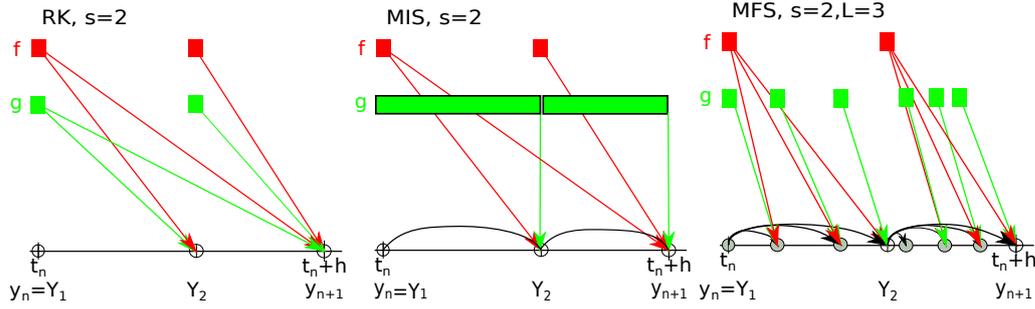


Figure 1: Flow diagrams for two stage RK, MIS and MFS methods from left to right. The rectangle position represents the time point of the stage, whereas the colors green and red correspond to the slow and fast evaluations.

interpretation. Let us scale the fast time τ by $\hat{\tau} \mapsto D_{ii}\tau$ in the stage i . Then the fast scale ODE (1.1b) becomes

$$\partial_{\hat{\tau}} \hat{Z}_{n,i} = \frac{1}{D_{ii}} \sum_j \beta_{ij} F(Y_{n,j}) + g(\hat{Z}_{n,i}(\hat{\tau}))$$

in the absence of γ . The new stage value $Y_{n,i} = \hat{Z}_{n,i}(D_{ii}h)$ is then the result at $\hat{\tau} = D_{ii}h$. In that sense, the coefficients D_{ii} can be seen as a scaling from macro to micro scale.

The coefficients α shift the initial value $Z_{n,i}(0)$ by a (slight) amount. In particular using $\alpha_{ij} = \delta_{ij+1}$ we recover the implicit-explicit method from Knoth and Wolke [27], which starts at the previous stage. Furthermore the coefficients γ look similar to a first order approximation to the sum of the fast and slow terms and for better stability properties.

The flow diagram in Figure 1 visualizes the aforementioned idea on a two stage method. The circles at the time line represent the macro stage values $Y_{n,i}$. We distinguish the slow and fast evaluations with the colors red and green respectively. According to the previous explanation, the red and green arrows correspond to the coefficients β and D respectively. The black arrows between the circles next to the time line represent the γ terms. A classical RK method evaluates the slow and the fast terms at the same time levels, hence the red and green boxes are in the same column. The (original) MIS approach fixes the slow values in the previous stage and solves an auxiliary IVP between the stages exactly. Hence we evaluate, at least theoretically, the fast tendencies on the whole micro scale interval $[0, D_{ii}h]$. The solution of the auxiliary IVP will be the value of the next stage, hence the green vertical arrows.

The last flow diagram already shows the new extension to the MFS methods. Here we discretize the continuous micro interval with a fast scale solver, hence we obtain additional $Z_{n,i}$ values between the stages, which correspond to the filled circles. For an explicit fast

scale integrator the next fast step depends only on the previous \mathbf{g} evaluation, hence the additional green boxes and green arrows from upper left to lower right. These methods will be introduced in the next section and analyzed in details throughout the remaining chapters.

Let us review the construction of the order conditions for MIS methods. Using infinitesimal steps means we require the exact solution of the auxiliary. Hence the order conditions require to have the exact solution of the auxiliary variable $\mathcal{Z}_{n,i}(\tau, \mathbf{h})|_{\tau=\mathbf{H}}$ and treat $\mathcal{Z}_{n,i}$ as a function in two variables. Then one derives an expansion of the numerical solution at $\mathbf{h} = 0$ and derives a recursion for the derivatives of the slow variables \mathbf{Y} in terms of the derivatives of the auxiliary variable $\mathcal{Z}_{n,i}$. The details for this process can be found in [52].

THE SHIFTED AUXILIARY PROBLEM We are interested in the solution with a (real) fast scale solver. Hence we have to consider the error due to the inexact solution of the fast ODEs (1.1b). These perturbed solutions are then the new slow values and enter the ODE (1.1b) for the remaining stages. These connections get clearer when we interpret the step size and the slow stage values \mathbf{Y}_n as parameters and include them in the definition (1.1b). At the same time we shift the IVP (1.1a)-(1.1b) for stage i by the scaled slow values, i.e. by $\sum_j \alpha_{ij} Y_{n,j}$, and obtain

$$\mathcal{Z}_{n,i}(0; \mathbf{h}, \mathbf{Y}_n) = \left(1 - \sum_{j=1}^{i-1} \alpha_{ij} \right) \mathbf{y}_n \quad (1.2a)$$

$$\partial_\tau \mathcal{Z}_{n,i}(\tau; \mathbf{h}, \mathbf{Y}_n) = \mathcal{R}_i(\mathcal{Z}_{n,i}, \mathbf{h}, \mathbf{Y}_n) \quad (1.2b)$$

$$\begin{aligned} \mathcal{R}_i(\mathcal{Z}_{n,i}, \mathbf{h}, \mathbf{Y}_n) = & \frac{1}{\mathbf{h}} \sum_{j=1}^{i-1} \gamma_{ij} (\mathbf{Y}_{n,j} - \mathbf{y}_n) + \sum_{j=1}^{i-1} \beta_{ij} f(\mathbf{Y}_{n,j}) \\ & + \mathbf{D}_{ii} \mathbf{g} \left(\mathcal{Z}_{n,i}(\tau; \mathbf{h}, \mathbf{Y}_n) + \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{Y}_{n,j} \right) \end{aligned} \quad (1.2c)$$

for every stage $i = 1, \dots, s+1$. Hence we recover the original MIS values

$$\mathbf{Z}_i(\tau) = \mathcal{Z}_{n,i}(\tau; \mathbf{h}, \mathbf{Y}_{n,1}, \dots, \mathbf{Y}_{n,i-1}) + \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{Y}_{n,j}$$

by shifting the parameterized \mathcal{Z} and use the exact slow stage values Y_n as parameters. For comparison later, we also write down the slow stages in terms of the shifted function fast solution, i.e.

$$\begin{aligned} Y_n &= \mathcal{Z}_n(\mathbf{h}; \mathbf{h}, Y_n) + \alpha Y_n \\ &= (\mathbf{I} - \alpha)^{-1} \mathcal{Z}_n(\mathbf{h}; \mathbf{h}, Y_n). \end{aligned} \quad (1.3)$$

FAST SCALE INTEGRATOR - EXACT SLOW STAGES Now we apply the fast-scale integrator Φ_i on the shifted and parameterized ODE (1.2b). The subscript i allows for different fast scale integrators in every stage. In particular we might compose a base integrator Φ with different numbers of steps in every stage, i.e. $\Phi_i = \Phi^{\mathcal{L}_{ii}}$ and step size $\frac{\mathbf{h}}{\mathcal{L}_{ii}}$. If we use only exact stage values Y_n we obtain the numerical solution after one step

$$\tilde{\mathcal{Z}}_{n,i}(\mathbf{h}; \mathbf{h}, Y_n) := \left(1 - \sum_{j=1}^{i-1} \alpha_{ij} \right) \mathbf{y}_n + \mathbf{h} \Phi_i(\mathcal{Z}_{n,i,0}, \mathbf{h}; \mathbf{h}, Y_n)$$

in analogy to the notation of Henrici [21]. Please note the additional arguments which correspond to the parameters of the parametrized auxiliary ODE (1.2b). If the fast scale integrator Φ is of order $p < q$, then there exist constants $\mathbf{d}_{i,k}$ such that

$$\tilde{\mathcal{Z}}_{n,i}(\mathbf{h}; \mathbf{h}, Y_n) = \mathcal{Z}_{n,i}(\mathbf{h}; \mathbf{h}, Y_n) + \sum_{k=p+1}^q \mathbf{d}_{i,k}(0; \mathbf{h}, Y_n) \mathbf{h}^k + \mathcal{O}(\mathbf{h}^{q+1}), \quad (1.4)$$

where the coefficients $\mathbf{d}_{i,k}$ depend on the elementary differentials of the shifted ODE (1.2b) and in turn on the parameters Y_n . In particular recognize the first term as the scaled and shifted solution of the MIS method. Also note that the variable \mathbf{h} plays three different roles in this equation. First it is the end time of the fast variable $\tau = \mathbf{h}$ at which we seek the solution $\mathcal{Z}_{n,i}(\tau; \mathbf{h}, Y_n)$. Second we solve the the IVP with only one step and \mathbf{h} is the step size of the fast scale integrator. Due to the error expansion it is therefore also the polynomial variable. Third it is one of the ODE parameters, which is the reason for the dependence of the error coefficient on the macro step size \mathbf{h} . Therefore we denote the error coefficients $\mathbf{d}_{i,p+1}$ with three arguments. The first one corresponds to the independent variable τ , whereas the second and third represents the parametrization with \mathbf{h} and Y_n .

FAST SCALE INTEGRATOR - PERTURBED SLOW STAGES Let us now apply the fast scale integrator Φ on the shifted IVP (1.2b), but this time with perturbed parameters \tilde{Y} . Then the slow values are again the solution at $\tau = h$ and we obtain the following recursion

$$\tilde{Y}_{n,1} = y_n \quad (1.5a)$$

$$\tilde{Y}_{n,2} = \mathcal{Z}_{n,2}(h; \tilde{Y}_{n,1}) + \alpha_{2,1} \tilde{Y}_{n,1} + \sum_{k=p+1}^q d_{2,k}(0; h, \tilde{Y}_{n,1}) h^k + \mathcal{O}(h^{q+1}) \quad (1.5b)$$

$$\tilde{Y}_{n,3} = \mathcal{Z}_{n,3}(h; \tilde{Y}_{n,1}, \tilde{Y}_{n,2}) + \sum_{j=1}^2 \alpha_{3j} \tilde{Y}_{n,j} + \sum_{k=p+1}^q d_{3,k}(0; h, \tilde{Y}_{n,1}, \tilde{Y}_{n,2}) h^k + \mathcal{O}(h^{q+1})$$

⋮

$$\tilde{Y}_{n,i} = \mathcal{Z}_{n,i}(h; \tilde{Y}_{n,1}, \dots, \tilde{Y}_{n,i-1}) + \sum_{j=1}^{i-1} \alpha_{ij} \tilde{Y}_{n,j} + \sum_{k=p+1}^q \hat{d}_{i,k}(0) h^k + \mathcal{O}(h^{q+1}) \quad (1.5c)$$

in the slow stage values $\tilde{Y}_{n,i}$. We added the two operators $\tilde{\cdot}$ and $\hat{\cdot}$ to distinguish between the two error types. With the operator $\tilde{\cdot}$ we denote only the numerical error due to the fast scale integrator Φ . The operator $\hat{\cdot}$ corresponds to the additional error due to the perturbation of the differential equation (1.2c). Hence the error coefficients $d_{i,k}$ also depend on the same perturbation. For a shorter notation we use the symbol $\hat{d}_{i,k}$ to account for the numerical error of the fast scale solver when solving the perturbed auxiliary equation and we neglect the last parameters.

We also observe two exceptions from the rule. The first stage remains exact, without any error because the complete method is explicit and the matrices α , β and γ are strict lower triangular. Furthermore the balanced MIS methods also require $\beta \mathbb{1} = D \mathbb{1}$ which implies $D_{11} = 0$. The second exception is the second stage. Due to the exactness of the first stage, the value $\mathcal{Z}_{n,2}(h, \tilde{Y}_{n,1}) + \alpha_{2,1} y_n$ is the same as the MIS value $Z_{n,2}$ and therefore the second stage $\tilde{Y}_{n,2}$ is only perturbed by the numerical integrator Φ . Hence the error coefficients $d_{2,k}$ in (1.5b) are the same as in the equation (1.4). All remaining stage values $\tilde{Y}_{n,i}$ with $i \geq 3$ are the sum of a perturbed solution $\mathcal{Z}(h, \tilde{Y}_n)$ and the error of the numerical solution of the ODE (1.2b) with perturbed parameters \tilde{Y}_n .

Similar to the slow MIS values Y_n we collect all stages into one vector and rewrite the vector of slow perturbed values

$$\begin{aligned}\tilde{Y}_n &= \mathcal{Z}_n(\mathbf{h}; \mathbf{h}, \tilde{Y}_n) + \alpha \tilde{Y}_n + \sum_{k=p+1}^q \hat{\delta}_k(0) \mathbf{h}^k + \mathcal{O}(\mathbf{h}^{q+1}) \\ &= (\mathbf{I} - \alpha)^{-1} \left(\mathcal{Z}_n(\mathbf{h}; \mathbf{h}, \tilde{Y}_n) + \sum_{k=p+1}^q \hat{\delta}_k(0) \mathbf{h}^k \right) + \mathcal{O}(\mathbf{h}^{q+1})\end{aligned}\quad (1.6)$$

in terms of the vector of the (shifted and perturbed) auxiliary solution \mathcal{Z}_n and the vector of the numerical error coefficients $(\hat{\delta}_k)_i = \hat{\mathbf{d}}_{i,k}$ from the fast scale solver applied to the perturbed ODE.

LOCAL ERROR WITH PERTURBED SLOW STAGES The two equations (1.3) and (1.6) give us the hint how we can split the local error

$$\begin{aligned}e_{s+1}^T \tilde{Y}_n - \mathbf{y}(t_{n+1}) &= \underbrace{e_{s+1}^T Y_n - \mathbf{y}(t_{n+1})}_{\text{MIS}} + \underbrace{e_{s+1}^T (\mathbf{I} - \alpha)^{-1} \sum_{k=p+1}^q \hat{\delta}_k(0) \mathbf{h}^k}_{\text{local error of } \Phi_{s+1}} \\ &\quad + \underbrace{e_{s+1}^T (\mathbf{I} - \alpha)^{-1} \left(\mathcal{Z}_n(\mathbf{h}; \mathbf{h}, \tilde{Y}_n) - \mathcal{Z}_n(\mathbf{h}; \mathbf{h}, Y_n) \right)}_{\text{perturbed exact solution of (1.2b)}} + \mathcal{O}(\mathbf{h}^{q+1})\end{aligned}\quad (1.7)$$

of a s -stage MFS method in three parts. The first difference correspond to the MIS error, the second to the numerical error of the fast scale integrator and the third term to solution of the perturbed auxiliary ODE due to the perturbation of the slow stages.

Let us consider the third term. This product depends on the difference between the exact solution of the perturbed ODE (1.2b) and the exact solution of the unperturbed ODE. The following theorem from Hairer et al. [20, Theorem I.14.2] relates the perturbation of parameters in an ODE to the solution. We specialize the theorem for our purposes and keep the notation simpler.

Theorem 1. *Let $\mathcal{Z}_{n,i}(\tau; \mathbf{h}, Y_n)$ be the solution of equation (1.2b) with exact parameters Y_n and consider the Jacobian*

$$A_i(\tau) = \frac{\partial \mathcal{R}_i(\mathcal{Z}_{n,i}(\tau; \mathbf{h}, Y_n), Y_n)}{\partial \mathcal{Z}_{n,i}} = D_{ii} \frac{\partial \mathbf{g}}{\partial \mathcal{Z}_{n,i}}(\mathcal{Z}_{n,i}(\tau; \mathbf{h}, Y_n)).$$

Let $\mathfrak{R}_i(\tau, 0)$ be the resolvent of the equation $\dot{\mathcal{Z}}_{n,i} = A_i \mathcal{Z}_{n,i}$. Then the solution $\hat{\mathcal{Z}}_{n,i}$ of (1.2) with slightly perturbed parameters \hat{Y}_n is given by

$$\hat{\mathcal{Z}}_{n,i}(\tau) = \mathcal{Z}_{n,i}(\tau; \mathbf{h}, Y_n) + \sum_{j=1}^{i-1} \mathfrak{S}_{ij}(\hat{Y}_{n,j} - Y_{n,j}) + \mathfrak{o}(|Y_{n,j} - \hat{Y}_{n,j}|) \quad (1.8a)$$

$$\mathfrak{S}_{ij} := \int_0^\tau \mathfrak{R}_i(\tau, s) \frac{\partial \mathcal{R}_i}{\partial Y_{n,j}}(\mathcal{Z}_{n,i}(s; \mathbf{h}, Y_n), \mathbf{h}, Y_n) ds \quad (1.8b)$$

The proof can be found in the aforementioned source. The symbol \mathfrak{S}_{ij} , defined in equation (1.8b), represents the sensitivity of the exact solutions to small changes in the slow values Y_n and the stability of the fast part due to the resolvent \mathfrak{R}_i .

From the definition (1.2c) of \mathcal{R}_i we obtain the partial derivative

$$\partial_{Y_{n,j}} \mathcal{R}_i|_{\mathcal{Z}_{n,i}(s), \mathbf{h}, Y_n} = \frac{\gamma_{ij}}{\mathbf{h}} + \beta_{ij} f_y(Y_{n,j}) + \alpha_{ij} D_{ii} g_y \left(\mathcal{Z}_{n,i}(s) + \sum_k \alpha_{ik} Y_{n,k} \right) \quad (1.8c)$$

with respect to $Y_{n,j}$. After inserting in the integral and splitting, we obtain the approximation

$$\begin{aligned} \mathfrak{S}_{ij}|_{\tau=\mathbf{h}} &\approx \overbrace{\mathfrak{R}_i(\mathbf{h}, \xi_1)}^{C_{1,i}} \gamma_{ij} + \mathbf{h} \overbrace{\mathfrak{R}_i(\mathbf{h}, \xi_2)}^{C_{2,i}} \beta_{ij} f_y(Y_{n,j}) \\ &\quad + \mathbf{h} D_{ii} \alpha_{ij} \underbrace{\mathfrak{R}_i(\mathbf{h}, \xi_3) g_y \left(\sum_k \alpha_{ik} Y_{n,k} + \mathcal{Z}_{n,i}(\xi_3) \right)}_{C_{3,i}} \end{aligned} \quad (1.8d)$$

with ξ_1, ξ_2 and $\xi_3 \in (0, \mathbf{h})$ from the intermediate value theorem as long as the Jacobian of g is sufficiently smooth. Whereas the first two constants $C_{1,i}$ and $C_{2,i}$ depend only on the resolvent and the intermediate position, the third constant $C_{3,i}$ also depends on the exact values Y_n and exact solution of \mathcal{Z}_n . For convenience we collect all values $\hat{\mathcal{Z}}_{n,i}(\tau)$ in one vector

$$\hat{\mathcal{Z}}_n(\tau) = \mathcal{Z}(\tau; \mathbf{h}, \hat{Y}_n) \quad (1.8e)$$

$$= \mathcal{Z}_n(\tau; \mathbf{h}, Y_n) + [C_1 \gamma + \mathbf{h} C_2 \beta f_y(Y_n) + \mathbf{h} C_3 D \alpha] (\hat{Y}_n - Y_n) \quad (1.8f)$$

and place the constants $C_{1,i}$, $C_{2,i}$ and $C_{3,i}$ on the diagonal of the matrices C_1 , C_2 and C_3 respectively.

At this point we know how arbitrary, but small, perturbations $\hat{Y}_{n,j} - Y_{n,j}$ of the previous stages $j = 1, \dots, i-1$ perturb the new slow value $\hat{Y}_{n,i}$ when using the exact integrator. Now we have to extract the perturbations of the numerical integrator Φ_i . For that we consider MIS methods of order q . Then we can neglect the error terms of order $q+1$ in (1.6) and subtract the exact Y_n , i.e.

$$\tilde{Y}_n - Y_n = (I - \alpha)^{-1} \left(\mathcal{Z}_n(\mathbf{h}; \mathbf{h}, \tilde{Y}_n) + \sum_{k=1}^q \hat{\delta}_k(0) \mathbf{h}^k - \mathcal{Z}_n(\mathbf{h}; \mathbf{h}, Y_n) \right) \quad (1.9a)$$

and use equation (1.8f) with \hat{Y}_n replaced by \tilde{Y}_n to end up with

$$\begin{aligned} &= (I - \alpha)^{-1} [C_1 \gamma + h C_2 \beta f_y(Y_n) + h C_3 D \alpha] (\tilde{Y}_n - Y_n) \\ &+ (I - \alpha)^{-1} \sum_{k=1}^q \hat{\delta}_k(0) \mathbf{h}^k. \end{aligned} \quad (1.9b)$$

The last equation is linear in the difference $\tilde{Y}_n - Y_n$. We multiply $I - \alpha$ to the left and subtract the first term to have the difference only on the left hand side, i.e.

$$(I - [\alpha + C_1 \gamma + h C_2 \beta f_y(Y_n) + h C_3 D \alpha]) (\tilde{Y}_n - Y_n) = \sum_{k=p+1}^q \hat{\delta}_k(0) \mathbf{h}^k. \quad (1.9c)$$

For a sufficiently small macro step size h the coefficients on the left hand side are near $I - \alpha - \gamma$, which is invertible. By virtue of the geometric series, and collecting by powers of h , we end up with the perturbed slow stages

$$\tilde{Y} \approx Y_n + (I + B_0 + h(B_1 + B_0 B_1)) \sum_{k=p+1}^q \hat{\delta}_k(0) \mathbf{h}^k \quad (1.10a)$$

where the matrices B_0 and B_1 are given by

$$B_0 = \alpha + C_1 \gamma \quad (1.10b)$$

$$B_1 = C_2 \beta f_y(Y_n) + C_3 D \alpha. \quad (1.10c)$$

Furthermore we neglected powers of h greater or equal than two in this expansion.

That last approximation (1.10a) is the key result for this section. It relates the approximated slow values to the exact slow values obtained from the MIS method plus some powers of the macro step size h . At the same time we arrived at two different error

coefficients B_0 and B_1 . The first term B_0 depends only on the coefficients α and γ and through the coefficient C_1 also indirectly on the fast tendencies through the resolvent \mathfrak{R} . But it is independent in the slow tendencies f . In contrast the second term depends on the slow and fast tendencies. Furthermore we remind the structure of the vector of error coefficients $\hat{\delta}_k(0)$. As we see from the construction (1.5), the first errors are always zero, because the first stage is always the old value \mathbf{y}_n . In turn the second entry contains only the coefficients due to the numerical integrator Φ_2 . Starting from the third stage the previous perturbations also propagate through the perturbed auxiliary ODE in the error coefficients. From that point of view we interpret the coefficients B_0 and B_1 as the sensitivity of the slow values \tilde{Y} to the previous numerical solution errors and the perturbed auxiliary ODE together. Let us summarize the consequence from formula (1.10a) in the

Theorem 2. *Let the fast scale integrator Φ and the MIS method have the order p and q respectively with $p \leq q$. Then there are no additional order conditions if $p = q$. In particular, there are no MFS conditions for $q = 1$.*

Proof. We multiply equation (1.10a) from the left with the unit vector \mathbf{e}_{s+1}^T and subtract the exact solution $\mathbf{y}(t_n + h)$, i.e.

$$\begin{aligned} \mathbf{e}_{s+1}^T \tilde{\mathbf{Y}}_n - \mathbf{y}(t_n + h) &= \overbrace{\mathbf{e}_{s+1}^T \mathbf{Y}_{n+1} - \mathbf{y}(t_n + h)}^{\mathcal{O}(h^{q+1}) \text{ due to MIS}} \\ &+ \mathbf{e}_{s+1}^T \sum_{k=p+1}^{q+1} [(I + B_0)\hat{\delta}_k(0)h^k + (B_1 + B_0B_1)\hat{\delta}_k(0)h^{k+1}]. \end{aligned} \quad (1.11)$$

Let us consider the cases

- $q = p$: The sums contain only one term of order h^{p+1} , which is also the order of the first term. Hence there are no additional conditions.
- $q = p + 1$: The first sum with coefficient $(I + B_0)\hat{\delta}_{p+1}(0)$ is the only additional power of h . Hence, we have to match only the coefficients $\mathbf{e}_{s+1}^T(I + B_0)\hat{\delta}_{p+1} = 0$.
- $q = p + 2$: We have to match two additional conditions of the form

$$\begin{aligned} \mathbf{e}_{s+1}^T(I + B_0)\hat{\delta}_{p+1}(0) &= 0 \\ \mathbf{e}_{s+1}^T[(B_1 + B_0B_1)\hat{\delta}_{p+2}(0)] &= 0. \end{aligned}$$

□

At this point we know what happens, when we want to increase the order of the fast scale integrator by one or two. But with the same arguments we get the additional MFS order conditions for the adjoint method Φ_i^* . From the theorem Hairer et al. [20, Theorem II.8.5] we see that the leading error coefficient has the same order, but the inverse sign. Hence we finish with the

Theorem 3. *The fast scale integrator Φ and the adjoint integrator Φ^* have the same multirate finite step (MFS) order conditions. In particular the explicit and the implicit Euler method have the same MFS order conditions.*

Proof. We change the signs of $\hat{\delta}_k$ in equation (1.11) and repeat the proof of theorem 2. □

CONCLUSIONS In this section we showed the how the global error of an arbitrary fast scale integrator Φ change the order conditions of MIS methods. In particular low order fast scale integrators lead to many additional order conditions due to the solution of the perturbed auxiliary problem.

II.2 MULTIRATE FINITE STEP METHODS

Let us introduce the details of the new MFS methods. As we roughly laid out in the diagram 1 the MFS methods are MIS methods, which were extended by a numerical fast scale integrator. In analogy to the derivation of theorem 2, we solve (1.1b) with a one step method with \mathfrak{L}_{ii} steps stage i . The theorem 2 shows, that the order conditions depend on the local error of the fast scale integrator. Hence we analyze three one step methods. In [34] we derived the order conditions for the explicit Euler method. This time we generalized the derivation to account also for other fast scale solvers of one step type. Let us now apply every one step method on the ODE (1.1b) with micro step size $\frac{h}{\mathfrak{L}_{ii}}$ and starting value $Z_{n,i,l-1}$. Additionally we will define a unified fast function $g_l^X(Z_{n,i,l-1}, Z_{n,i,l})$ for every fast scale integrator X and step l , hence the subscript l .

II.2.1 Unified notation of MFS methods

Let us first introduce the MFS method utilizing the *unified fast scale evaluation function* g_l^X with s stages, i.e.

$$Z_{n,i,0} = y_n + \sum_j \alpha_{ij}(Y_{n,j} - y_n) \quad (2.1a)$$

$$Z_{n,i,l} = Z_{n,i,l-1} + \frac{1}{\mathfrak{L}_{ii}} \sum_j \gamma_{ij}(Y_{n,j} - y_n) + \frac{h}{\mathfrak{L}_{ii}} \sum_j \beta_{ij} f(Y_{n,j}) \\ + \frac{h}{\mathfrak{L}_{ii}} D_{ii} g_l^X(Z_{n,i,l-1}, Z_{n,i,l}), \quad l = 1, \dots, \mathfrak{L}_{ii} \quad (2.1b)$$

$$Y_{n,i} = Z_{n,i,\mathfrak{L}_{ii}} \quad i = 1 \dots s + 1 \quad (2.1c)$$

and the new coarse scale value at time t_{n+1} , i.e.

$$y_{n+1} = Y_{n,s+1}. \quad (2.1d)$$

Note that this unification not only unifies the usage of the fast scale integrators, but it also unifies the partitioning strategies. This allows us to treat the partitioned methods and the classical method in the same way. The step dependency with the subscript l accounts for one step fast scale integrators with more than one stage. With the superscript X we represent the fast scale integrator. We concentrate on the methods explicit Euler, Forward-Backward Euler and Störmer-Verlet. Hence X is a placeholder for the abbreviations EE, FB or SV respectively. The coefficients α , β , γ and D are the same as in the MIS context.

All fast scale evaluation function g_l^X have the useful property

$$g_l^X(a, a) = g(a). \quad (2.2a)$$

Then we can differentiate this equation in direction $F(y_n)$ at y_n to obtain

$$\partial_1 g_l^X \langle F \rangle + \partial_2 g_l^X \langle F \rangle = g_y \langle F \rangle \quad (2.2b)$$

$$g_{yy} \langle F, F \rangle = \partial_{11}^2 g_l^X \langle F, F \rangle + \partial_{22}^2 g_l^X \langle F, F \rangle + 2\partial_{12} g_l^X \langle F, F \rangle. \quad (2.2c)$$

Also note that we neglected the argument y_n . This symbolism will be explained in the next section in detail. In particular the first relation plays an important role during the derivation of the order condition.

II.2.2 Explicit Euler as fast scale integrator

The explicit Euler is the simplest explicit RK method and of order one. The new fast value $Z_{n,i,l}$ is given by

$$Z_{n,i,l} = Z_{n,i,l-1} + \frac{1}{\mathfrak{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j} - y_n) + \frac{h}{\mathfrak{L}_{ii}} \sum_j \beta_{ij} f(Y_{n,j}) + h \frac{D_{ii}}{\mathfrak{L}_{ii}} g(Z_{n,i,l-1}).$$

Due to the explicit nature the fast tendencies depend only on the old step $Z_{n,i,l-1}$. By comparing the last equation with (2.1b), we directly read the unified fast function g_l^{EE} , i.e.

$$g_l^{EE}(Z_{n,i,l-1}, Z_{n,i,l}) = g(Z_{n,i,l-1}). \quad (2.3a)$$

This unified fast scale evaluation function does not depend directly on the step l . Much more importantly, the derivative with respect to the first argument reduces to the derivative of the fast tendencies.

II.2.3 Forward-backward Euler as fast scale integrator

This time we utilize the partitioning in the partitioned model problem I.2.2 by treating the first equation for y^p explicit and the second equation semi-implicit in terms of the first component y^p . Hence the new step

$$\begin{aligned} Z_{n,i,l}^p &= Z_{n,i,l-1}^p + \frac{1}{\mathfrak{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^p - y_n^p) \\ &\quad + \frac{h}{\mathfrak{L}_{ii}} \sum_j \beta_{ij} f^p(Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{\mathfrak{L}_{ii}} g^p(Z_{n,i,l-1}^p, Z_{n,i,l-1}^q) \\ Z_{n,i,l}^q &= Z_{n,i,l-1}^q + \frac{1}{\mathfrak{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^q - y_n^q) \\ &\quad + \frac{h}{\mathfrak{L}_{ii}} \sum_j \beta_{ij} f^q(Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{\mathfrak{L}_{ii}} g^q(Z_{n,i,l}^p, Z_{n,i,l-1}^q) \end{aligned}$$

depends partially on the value $Z_{n,i,l}$, but only in the second component. Due to the partitioning, the fast unified fast scale function g_l^{FB} is given by

$$g_l^{\text{FB}}(\mathbf{a}, \mathbf{b}) = g_l^{\text{FB}}((\mathbf{a}^p, \mathbf{a}^q), (\mathbf{b}^p, \mathbf{b}^q)) = \begin{bmatrix} g^p(\mathbf{a}^p, \mathbf{a}^q) \\ g^q(\mathbf{b}^p, \mathbf{a}^q) \end{bmatrix}, \quad (2.3b)$$

where the arguments \mathbf{a} and \mathbf{b} correspond to the previous and next fast value respectively. If we neglect the Z^p component in g^p and Z^q component in g^q , we recover the implicit explicit scheme of Mesinger [32].

In contrast to the explicit Euler method, the derivative of g_l^{FB} with respect to the first argument is different from the derivative of the fast tendencies g . Nevertheless it fulfills the relations (2.2b).

II.2.4 Störmer-Verlet as fast-scale integrator

We interpret the method as two halve steps of (an implicit) forward-backward Euler with exchanged roles. Both halve steps are adjoint to each other, therefore the method is of order two. The odd steps l are then given by

$$\begin{aligned} Z_{n,i,l}^p &= Z_{n,i,l-1}^p + \frac{1}{2\mathcal{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^p - y_n^p) \\ &\quad + \frac{h}{2\mathcal{L}_{ii}} \sum_j \beta_{ij} f^p(Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{2\mathcal{L}_{ii}} g^p(Z_{n,i,l}^p, Z_{n,i,l-1}^q) \\ Z_{n,i,l}^q &= Z_{n,i,l-1}^q + \frac{1}{2\mathcal{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^q - y_n^q) \\ &\quad + \frac{h}{2\mathcal{L}_{ii}} \sum_j \beta_{ij} f^q(Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{2\mathcal{L}_{ii}} g^q(Z_{n,i,l}^p, Z_{n,i,l-1}^q) \end{aligned}$$

and for even l

$$\begin{aligned}
 Z_{n,i,l}^q &= Z_{n,i,l-1}^q + \frac{1}{2\mathfrak{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^q - y_n^q) \\
 &\quad + \frac{h}{2\mathfrak{L}_{ii}} \sum_j \beta_{ij} f^q (Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{2\mathfrak{L}_{ii}} g^q (Z_{n,i,l-1}^p, Z_{n,i,l}^q) \\
 Z_{n,i,l}^p &= Z_{n,i,l-1}^p + \frac{1}{2\mathfrak{L}_{ii}} \sum_j \gamma_{ij} (Y_{n,j}^p - y_n^p) \\
 &\quad + \frac{h}{2\mathfrak{L}_{ii}} \sum_j \beta_{ij} f^p (Y_{n,j}^p, Y_{n,j}^q) + h \frac{D_{ii}}{2\mathfrak{L}_{ii}} g^p (Z_{n,i,l-1}^p, Z_{n,i,l}^q)
 \end{aligned}$$

Note that we only exchange the time arguments of the fast function g , whereas all constant parts remain the same. Due to the exchange of the roles and arguments, the unified fast scale function g_l^{SV} is given by

$$g_l^{SV}(a, b) = g_l^{SV}((a^p, a^q), (b^p, b^q)) = \begin{cases} \begin{bmatrix} g^p(b^p, a^q) \\ g^q(b^p, a^q) \end{bmatrix} & l \text{ odd} \\ \begin{bmatrix} g^p(a^p, b^q) \\ g^q(a^p, b^q) \end{bmatrix} & l \text{ even} \end{cases} \quad (2.3c)$$

If we compare the equations for the new step, we see that only the Störmer-Verlet method has a factor two in the denominator. We get rid of this factor by allowing only even numbers of steps and merging the factor into \mathfrak{L}_{ii} .

This time the fast scale evaluation function depends directly on the step l and derivatives with respect to the first argument are again different from the fast scale function g . Hence the derivation of the order conditions for this case are most lengthy.

II.3 ORDER CONDITIONS

In this section we derive the order conditions for a MFS method. In light of the MFS basis, namely MIS and one step Runge-Kutta methods, we expect three sets of conditions. All g -independent derivatives must belong to a classic order condition and all g -dependent derivatives with step L independent coefficients must belong to a MIS condition. All other conditions, depending on the step factor L , belong to the new MFS conditions and depend on the fast scale integrator.

Every sub problem in the partitioned model problem (I.2.2) is vector valued. Hence every stage solution is vector valued and we have to introduce the multivariate derivatives for every component. To unify the notion of the derivatives, we first introduce an extended notation and common symbol. After that we obtain considerably shorter derivative expressions. The following section is the main part of this thesis and we derive the derives of the numerical solution using a general notion for the fast scale integrator. Despite the shortening with the simplified notation, we obtain lengthy coefficients and quite unstructured combinations. Hence there is no "perfect" way for simplification.

The last three subsections the extract the order conditions from the derivatives and specialize them for three fast scale integrators. We will particularly show how the splitting influences the elementary differential coefficients and the effect on the number of order conditions.

II.3.1 Extended notation and common symbols

Before we start with the order conditions, we have to introduce several symbols. Please note that the ODE (I.2.1) is vector valued with \mathbf{d} components. This implies that already the first derivative of the right hand side F is a matrix, hence tensor valued. Thus we use the symbols $I_{\mathbf{d}}$ and $\mathbb{1}_{\mathbf{d}}$ for the identity matrix and the vector of ones in dimension \mathbf{d} respectively. Without subscript we assume the dimension $s + 1$. We use the notations $f_{\mathbf{y}} \langle F \rangle$ and $f_{\mathbf{y}\mathbf{y}} \langle F, F \rangle$ for the first and second derivatives. In fact the notation is a short hand for the vector of sums

$$[f_{\mathbf{y}} \langle F \rangle]_{\mathbf{i}} = \sum_{k=1}^{\mathbf{d}} \partial_k f_{\mathbf{i}} F_k$$

and

$$[f_{\mathbf{y}\mathbf{y}} \langle F, F \rangle]_{\mathbf{i}} = \sum_{k,l} \partial_{k,l}^2 f_{\mathbf{i}} F_k F_l.$$

In general the notation

$$\left[f_{\underbrace{\mathbf{y}, \dots, \mathbf{y}}_k} \langle A^1, \dots, A^k \rangle \right]_{\mathbf{i}} = \sum_{|\alpha|=k} \frac{\partial^{|\alpha|} f_{\mathbf{i}}}{\partial y_{\alpha_1} \partial y_{\alpha_2} \dots \partial y_{\alpha_k}} A_{\alpha_1}^1 A_{\alpha_2}^2 \dots A_{\alpha_k}^k$$

stand for the sum of k th derivatives of the i th f component applied on the vectors A^1, \dots, A^k . So the vector notation of the component wise MFS method (2.1) is of tensor structure and the corresponding product between the vector spaces \mathbb{R}^{s+1} and \mathbb{R}^d is the Kronecker product \otimes , see [46] for an overview on the properties and rules.

We stack the block wise application of the right-hand sides and the fast scale evaluations with the operator $\hat{\cdot}$. For example the stacked slow evaluations at the stacked vector Y are

$$\hat{Y} \in \mathbb{R}^{(s+1)d} \quad \hat{F}(\hat{Y}) := \sum_{i=1}^{s+1} e_i \otimes F(Y_{n,i}) \quad (3.1a)$$

where $Y_{n,i} \in \mathbb{R}^d$ is a sub vector of \hat{Y} , i.e.

$$\hat{Y} = \sum_{i=1}^{s+1} e_i \otimes Y_{n,i}. \quad (3.1b)$$

One can also view Y as a stacked vector of Y_i . The order conditions require the derivative of the numerical solution and therefore the derivatives of the fast and slow tendencies. The fast scale methods may exploit the splitting, which in turn leads to different derivatives for each component. With the notation of the stacked evaluation, the notation of the derivative fits smoothly in the block structure, i.e.

$$\hat{g}_y \langle \hat{F} \rangle = \sum_{i=1}^{s+1} e_i \otimes g_y \langle F(\hat{Y}_{n,i}) \rangle$$

and the partial derivative of the stacked fast scale integrators at the steps \hat{l} are

$$\partial_1 \hat{g}_{\hat{l}} \langle \hat{F} \rangle = \sum_{i=1}^{s+1} e_i \otimes \partial_1 g_{\hat{l}_i}^X \langle F \rangle.$$

Note that we dropped the superscript X for the fast scale integrator type. That will be reintroduced later, when we collect the derivatives of the fast functions g . We collect the steps l for every stage on the main diagonal of the diagonal matrix \hat{L} . In analogy we define the diagonal matrix of the step ratios \hat{L} and the matrix of all steps

$$\mathcal{L} = L\hat{L} \quad (3.2a)$$

such that the fast scale integrator does $\mathfrak{L}_{ii} = \mathbb{L}\hat{\mathbb{L}}_{ii}$ steps in stage i . Let us define the sum of several block vectors B_l , where l is a diagonal matrix. We can write and every B_l in the form

$$B_l = \sum_{i=1}^{s+1} e_i \otimes B_{l_{ii}}. \quad (3.3a)$$

with sub vectors $B_{l_{ii}}$. Then the sum of all B_l represents the stage wise sum of the block vectors B_l , i.e.

$$\sum_{l=1}^{\hat{i}} B_l := \sum_{i=1}^{s+1} \sum_{l=1}^{\hat{l}_{ii}} e_i \otimes B_{l_{ii}}.$$

The block vectors \hat{Y} and $\hat{Z}_{\hat{i}}$ stack all (slow) stage values and fast values respectively, i.e.

$$\hat{Y} := \sum_{i=1}^{s+1} e_i \otimes Y_{n,i} \quad (3.3b)$$

$$\hat{Z}_{\hat{i}} := \sum_{i=1}^{s+1} e_i \otimes Z_{n,i,\hat{l}_{ii}}. \quad (3.3c)$$

Please note the dropped subscript n for the vectors. Every stage vector Y and $Z_{\hat{i}}$ remains to exist only on one macro time interval $[t_n, t_n + h]$.

II.3.2 Derivatives of the numerical solution

With these changes in mind we obtain the vectorized MFS method

$$\hat{Z}_0 = \mathbb{1} \otimes y_n + \alpha \otimes I_d \cdot (Y - \mathbb{1} \otimes y_n) \quad (3.4a)$$

$$\begin{aligned} \hat{Z}_{\hat{i}} = & Z_{\hat{i}-1} + [\mathfrak{L}^{-1}\gamma] \otimes I_d \cdot (Y - \mathbb{1} \otimes y_n) \\ & + h[\mathfrak{L}^{-1}\beta] \otimes I_d \cdot \hat{f}(Y) + h[\mathfrak{L}^{-1}D] \otimes I_d \cdot \hat{g}_{\hat{i}}(\hat{Z}_{\hat{i}-1}, \hat{Z}_{\hat{i}}) \end{aligned} \quad (3.4b)$$

$$Y = \hat{Z}_{\mathfrak{L}} \quad (3.4c)$$

with vectorized auxiliary initial value Z_0 , the vectorized fast steps $Z_{\hat{l}}$ and the slow stages Y . Now we sum up every all fast steps (3.4b), including the initial value (3.4a), up to \hat{l}

$$\begin{aligned} \hat{Z}_{\hat{l}} &= [I - \alpha - \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot \mathbf{1} \otimes y_n + [\alpha + \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot Y \\ &\quad + h[\hat{l}\mathcal{L}^{-1}\beta] \otimes I_d \cdot \hat{f}(Y) + h[\mathcal{L}^{-1}D] \otimes I_d \cdot \sum_{j=I}^{\hat{l}} \hat{g}_j(\hat{Z}_{j-I}, \hat{Z}_j) \end{aligned} \quad (3.5)$$

and differentiate at $h = 0$. At this point we recognize the value in choosing diagonal matrices as subscripts. With this particular, and unusual, choice we can reuse the same symbol without any transition between vectors and matrices. The stage values coincide with y_n at $h = 0$, so we neglect the argument y_n in the remaining section. Using the Leibniz's rule we get the three derivatives

$$\hat{Z}_{\hat{l}}^{(1)} = [\alpha + \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot Y^{(1)} + [\hat{l}\mathcal{L}^{-1}\beta] \otimes I_d \cdot \mathbf{1} \otimes f + [D\mathcal{L}^{-1}] \otimes I_d \cdot \sum_{j=I}^{\hat{l}} \hat{g}_j \quad (3.6a)$$

$$= [\alpha + \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot Y^{(1)} + [\hat{l}\mathcal{L}^{-1}] \otimes I_d \cdot ([\beta\mathbf{1}] \otimes f + [D\mathbf{1}] \otimes g) \quad (3.6b)$$

$$\begin{aligned} \hat{Z}_{\hat{l}}^{(2)} &= [\alpha + \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot Y^{(2)} + 2[\hat{l}\mathcal{L}^{-1}\beta] \otimes I_d \cdot f(Y)^{(1)} \\ &\quad + 2[D\mathcal{L}^{-1}] \otimes I_d \cdot \sum_{j=I}^{\hat{l}} \hat{g}_j(Z_{j-I}, Z_j)^{(1)} \end{aligned} \quad (3.6c)$$

$$\begin{aligned} \hat{Z}_{\hat{l}}^{(3)} &= [\alpha + \hat{l}\mathcal{L}^{-1}\gamma] \otimes I_d \cdot Y^{(3)} + 3[\hat{l}\mathcal{L}^{-1}\beta] \otimes I_d \cdot f(Y)^{(2)} \\ &\quad + 3[D\mathcal{L}^{-1}] \otimes I_d \cdot \sum_{j=I}^{\hat{l}} \hat{g}_j(\hat{Z}_{j-I}, \hat{Z}_j)^{(2)} \end{aligned} \quad (3.6d)$$

of the fast stage vectors $Z_{\hat{l}}$ at every step \hat{l} . We already summed up all fast steps in equation (3.6a) and collected the result in (3.6b).

Now we analyze the derivatives of the fast steps $Z_{\hat{l}}$ at the last step $\hat{l} = \mathcal{L}$. After that we match the derivatives of the numerical solution with the derivatives of the exact solution. A short view and insertion of $\hat{l} = \mathcal{L}$ in the equations (3.6) suggest a repeated occurrence of $I - \alpha - \gamma$, so we introduce the matrix

$$R^{-1} = I - \alpha - \gamma. \quad (3.7)$$

The remaining section consists of several derivatives applied to vectors or further derivatives applied to vectors and so on. To keep an overview it helps to order the terms in a similar

way. Hence a rule of thumb will be the ordering from slow to fast in a successive way. For example we prefer $f+g$ over $g+f$ and $f_y \langle g_y \langle F \rangle \rangle + g_y \langle f_y \langle F \rangle \rangle$ over $g_y \langle f_y \langle F \rangle \rangle + f_y \langle g_y \langle F \rangle \rangle$.

First derivative of the numerical solution and conditions for order one

Let us start with the first derivative (3.6b) at $\hat{l} = \mathcal{L}$

$$Y^{(1)} = \hat{Z}_{\mathcal{L}}^{(1)} = [R\beta\mathbb{1}] \otimes f + [RD\mathbb{1}] \otimes g \quad (3.8a)$$

and define the matrix

$$\hat{A} := R\beta. \quad (3.8b)$$

The matrix \hat{A} relates the underlying method, i.e. $g = 0$, to the Butcher tableau of the classical one step methods. The first s rows represent the Butcher tableau, whereas the last row corresponds to the update coefficients \mathbf{b} , i.e.

$$\mathbf{b} := \mathbf{e}_{s+1}^T \hat{A}.$$

As usual we define the (underlying) nodes

$$\mathbf{c} = \hat{A}\mathbb{1}$$

as row sums of the matrix \hat{A} . The classic order condition for order one is the vector product

$$\mathbf{b} \cdot \mathbb{1} = 1. \quad (3.9a)$$

Order one requires $Y^{(1)} = \mathbf{c} \otimes F$ and in turn equation (3.8a) implies the so called compatibility condition

$$\beta\mathbb{1} = D\mathbb{1} \quad (3.9b)$$

which guarantees that the last fast step $Z_{n,i,\mathcal{L}i}$ must coincide with the corresponding slow node \mathbf{c}_i . Please note that the first order conditions are independent of the step size ratio. So we insert the derivative $Y^{(1)}$ in the first derivative of the \hat{l} th fast step, i.e.

$$\hat{Z}_i^{(1)} = [\tilde{\mathbf{c}} + \hat{l}\mathcal{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})] \otimes F \quad (3.10)$$

and define the short hand

$$\tilde{\mathbf{c}} = \alpha \mathbf{c} . \quad (3.11)$$

We see a very simple first derivative of all fast scale steps with respect to the macro step size. We will use equation (3.10) several times for the higher derivatives. Also note the usage of the matrix valued subscript $\hat{\mathbf{l}}$ in the formula again, which emphasizes the usefulness of the unusual notation.

Second derivative of the numerical solution

Let us proceed to the second derivative (3.6c). The first two summands are independent of the fast terms, whereas the third one sums up the first derivatives of the unified fast scale function. We expand the first derivative in the two partial derivatives

$$\hat{\mathbf{g}}_j(\hat{\mathbf{Z}}_{j-1}, \hat{\mathbf{Z}}_j)^{(1)} = \partial_1 \hat{\mathbf{g}}_j \langle \hat{\mathbf{Z}}_{j-1}^{(1)} \rangle + \partial_2 \hat{\mathbf{g}}_j \langle \hat{\mathbf{Z}}_j^{(1)} \rangle \quad (3.12a)$$

then insert (3.10) and use equation (2.2b)

$$\begin{aligned} &= \tilde{\mathbf{c}} \otimes \mathbf{g}_y \langle \mathbf{F} \rangle + [(j-1)\mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})] \otimes \mathbf{g}_y \langle \mathbf{F} \rangle \\ &+ [\mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})] \otimes \mathbb{1}_d \cdot \partial_2 \hat{\mathbf{g}}_j \langle \hat{\mathbf{F}} \rangle . \end{aligned} \quad (3.12b)$$

The summands consist of three terms, namely a step independent, a step dependent coefficient of a step independent derivative and a step independent coefficient with the step dependent derivative $\partial_2 \hat{\mathbf{g}}_j \langle \hat{\mathbf{F}} \rangle$.

We sum up the equation (3.12b) and left multiply by $\mathbf{D}\mathfrak{L}^{-1}$, i.e.

$$\begin{aligned} [\mathbf{D}\mathfrak{L}^{-1}] \otimes \mathbb{I}_d \cdot \sum_{j=1}^{\hat{\mathbf{l}}} \hat{\mathbf{g}}_j(\hat{\mathbf{Z}}_{j-1}, \hat{\mathbf{Z}}_j)^{(1)} &= \mathbf{Q}_1(\hat{\mathbf{l}}) \otimes \mathbf{g}_y \langle \mathbf{F} \rangle \\ &+ [\mathbf{D}\mathfrak{L}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})] \otimes \mathbb{1}_d \cdot \sum_{j=1}^{\hat{\mathbf{l}}} \partial_2 \hat{\mathbf{g}}_j \langle \hat{\mathbf{F}} \rangle \end{aligned} \quad (3.13a)$$

and define the coefficient matrix

$$\mathbf{Q}_1(\hat{\mathbf{l}}) = \hat{\mathbf{l}}\mathbf{D}\mathfrak{L}^{-1} \left(\tilde{\mathbf{c}} + \frac{1}{2}(\hat{\mathbf{l}} - 1)\mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) \right) \quad (3.13b)$$

and its sum

$$Q_1^S = \mathfrak{L}^{-1} \sum_{\hat{l}=I}^{\mathfrak{L}} Q_1(\hat{l}-I) \quad (3.13c)$$

$$= \frac{1}{6} D(c + 2\tilde{c} - 3\mathfrak{L}^{-1}c + 2\mathfrak{L}^{-2}(c - \tilde{c})) \quad (3.13d)$$

for later use. Please note the splitting in a step dependent coefficient $Q_1(\hat{l})$ of a step independent derivative and a step independent coefficient of a step dependent sum of the derivative $\partial_2 \hat{g}_j \langle \hat{F} \rangle$.

Together with the relation (3.7) and (3.13a) we simplify the second derivative, i.e.

$$Y^{(2)} = \hat{Z}_{\mathfrak{L}}^{(2)} = 2[\hat{A}c] \otimes f_y \langle F \rangle + 2[RQ_1(\mathfrak{L})] \otimes g_y \langle F \rangle + 2\kappa[R\mathfrak{L}^{-1}D(c - \tilde{c})] \otimes g_y \langle F \rangle \quad (3.14)$$

and introduce the shorthand $g_y \langle F \rangle$ for the sum of partial derivatives in the last term. The derivative $\partial_2 \hat{g}_j \langle F \rangle$ relates the fast scale integrator to the fast tendencies. Although the derivative depends in an nearly arbitrary way on the integrator, the sum should obey a linear relation as

$$\sum_{j=I}^{\mathfrak{L}} \partial_2 \hat{g}_j \langle \hat{F} \rangle = \kappa [\mathfrak{L}\mathbb{1}] \otimes g_y \langle F \rangle \quad (3.15a)$$

with a scalar coefficient κ . The symbol $g_y \langle F \rangle$ represents the actual (summed) derivatives and is not necessarily $g_y \langle F \rangle$ nor $\partial_2 \hat{g}_1 \langle F \rangle$. We will come back to the details, when we discuss the fast scale integrators again. Later we will make use of some additional polynomials. Hence we introduce the short hands

$$\sum_{\hat{l}=I}^{\mathfrak{L}} \sum_{j=I}^{\hat{l}-I} \partial_2 \hat{g}_j \langle \hat{F} \rangle =: \hat{g}_y^{1s} \langle \hat{F} \rangle \quad (3.15b)$$

$$\sum_{\hat{l}=I}^{\mathfrak{L}} [(\hat{l}-I)\mathbb{1}] \otimes \mathbb{1}_d \cdot \partial_2 \hat{g}_l \langle \hat{F} \rangle =: \hat{g}_y^1 \langle \hat{F} \rangle \quad (3.15c)$$

$$\sum_{\hat{l}=I}^{\mathfrak{L}} \partial_2 \hat{g}_l \langle \partial_2 \hat{g}_l \langle \hat{F} \rangle \rangle =: \hat{g}_y^0 \langle \hat{F} \rangle \quad (3.15d)$$

Let us inspect the equation (3.14) in more detail. Obviously it is the sum of three elementary differentials. The first two derivatives are always part of the analytic solution in exactly this way. The third one depends on the fast scale integrator. Due to

the partitioning strategies, the partial derivatives with respect to the component mix differently.

For a first order fast scale integrator we have to sum up the corresponding coefficients and derivatives. Furthermore these coefficients occur in the third derivative. But for a second order method we have a different scenario. Then the second derivative $Z_{\mathcal{L}}^{(2)}$ must match the second derivative of the exact solution of the (auxiliary) ODE. In this case, the additional derivative $\partial_2 \hat{g}_j \langle \hat{F} \rangle$ and the step dependency must vanish. Consequently the number of third order conditions gets much smaller. Remind the theorem 2. It is exactly the same consequence, but this time in terms of the fast functions instead of abstract error coefficients. Now we evaluate the matrix polynomial $Q_1(\mathcal{L})$ in (3.14) and split the coefficients for repeated later use, i.e.

$$Y^{(2)} = 2[\hat{A}c] \otimes f_y \langle F \rangle + \left[Q_2^I - \frac{1}{L} Q_2 \right] \otimes g_y \langle F \rangle + \frac{2\kappa}{L} Q_2 \otimes g_y \langle F \rangle \quad (3.16a)$$

$$Q_2 = RD\hat{L}^{-1}(c - \tilde{c}) \quad (3.16b)$$

$$Q_2^I = RD(c + \tilde{c}). \quad (3.16c)$$

Remind that the first derivative of the numerical solution, and more importantly the first derivative of the fast steps \hat{Z} , are independent in the fast scale function. This time we are in different situation, where we split the derivative in three parts. Both first summands depend only on the elementary differentials of the exact solutions and the third summand is the only one which depends on the derivatives of the fast scale function.

Third derivative of the numerical solution

The third derivative of the numerical solution

$$Y^{(3)} = \hat{Z}_{\mathcal{L}}^{(3)} = 3\hat{A} \otimes I_d \cdot f(Y)^{(2)} + 3[RD\mathcal{L}^{-1}] \otimes I_d \cdot \sum_{j=1}^{\mathcal{L}} \hat{g}_j(\hat{Z}_{j-1}, \hat{Z}_j)^{(2)} \quad (3.17)$$

exhibits a similiar structure as the second derivative. First we have look at the derivatives of f in (3.17). We easily get

$$\hat{A} \otimes I_d \cdot f(Y)^{(2)} = [\hat{A}c^2] \otimes f_{yy} \langle F, F \rangle + \hat{A} \otimes I_d \cdot \hat{f}_y \langle Y^{(2)} \rangle \quad (3.18a)$$

and use equation (3.14) to split the derivative $f(\mathbf{Y})^{(2)}$

$$\begin{aligned} &= [\hat{\mathbf{A}}\mathbf{c}^2] \otimes \mathbf{f}_{\mathbf{y}\mathbf{y}} \langle \mathbf{F}, \mathbf{F} \rangle + 2[\hat{\mathbf{A}}\hat{\mathbf{A}}\mathbf{c}] \otimes \mathbf{f}_{\mathbf{y}} \langle \mathbf{f}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle \\ &+ [\hat{\mathbf{A}}\mathbf{Q}_2^{\hat{\mathbf{l}}}] \otimes \mathbf{f}_{\mathbf{y}} \langle \mathbf{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle - \mathfrak{F} \end{aligned} \quad (3.18b)$$

in four terms. These four summands belong to three groups. The first group consists of derivatives containing only the slow function \mathbf{f} and belong to the underlying method, the second group contains slow and fast terms and has only step independent coefficients. In contrast the last group

$$\mathfrak{F} = \frac{1}{\mathfrak{L}} [\hat{\mathbf{A}}\mathbf{Q}_2] \otimes \mathbf{f}_{\mathbf{y}} \langle \mathbf{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle - \frac{2\kappa}{\mathfrak{L}} [\hat{\mathbf{A}}\mathbf{Q}_2] \otimes \mathbf{f}_{\mathbf{y}} \langle \mathbf{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle \quad (3.18c)$$

contains the slow derivatives applied to fast terms with step dependent coefficients. If the fast scale method is of order two, then the step dependent terms will vanish in equation (3.14) and therefore, we would have $\mathfrak{F} = 0$. Otherwise, the expression \mathfrak{F} depends on the number of steps \mathfrak{L} and unified fast function $\hat{\mathbf{g}}_j$.

Now we proceed to the sum of the fast terms $\hat{\mathbf{g}}_i$ in (3.17). This time it looks useful to split the sum of second derivatives

$$\mathfrak{L}^{-1} \otimes \mathbf{I}_d \cdot \sum_{\hat{i}=1}^{\mathfrak{L}} \hat{\mathbf{g}}_{\hat{i}} (\hat{\mathbf{Z}}_{\hat{i}-1}, \hat{\mathbf{Z}}_{\hat{i}})^{(2)} = \mathfrak{G}_1^{\mathbf{y}} + \mathfrak{L}^{-1} \otimes \mathbf{I}_d \cdot \mathfrak{G}_1^{\hat{\mathbf{l}}} + \mathfrak{G}_2 \quad (3.19a)$$

in the three parts, namely

$$\mathfrak{G}_1^{\mathbf{y}} = \mathfrak{L}^{-1} \otimes \mathbf{I}_d \cdot \sum_{\hat{i}=1}^{\mathfrak{L}} \hat{\mathbf{g}}_{\hat{i}} \langle \hat{\mathbf{Z}}_{\hat{i}-1}^{(2)} \rangle \quad (3.19b)$$

$$\mathfrak{G}_1^{\hat{\mathbf{l}}} = \sum_{\hat{i}=1}^{\mathfrak{L}} \partial_2 \hat{\mathbf{g}}_{\hat{i}} \langle \hat{\mathbf{Z}}_{\hat{i}}^{(2)} - \hat{\mathbf{Z}}_{\hat{i}-1}^{(2)} \rangle \quad (3.19c)$$

$$\mathfrak{G}_2 = \mathfrak{L}^{-1} \otimes \mathbf{I}_d \cdot \sum_{\hat{i}=1}^{\mathfrak{L}} \partial_{11}^2 \hat{\mathbf{g}}_{\hat{i}} \langle \hat{\mathbf{Z}}_{\hat{i}-1}^{(1)}, \hat{\mathbf{Z}}_{\hat{i}-1}^{(1)} \rangle + \partial_{22}^2 \hat{\mathbf{g}}_{\hat{i}} \langle \hat{\mathbf{Z}}_{\hat{i}}^{(1)}, \hat{\mathbf{Z}}_{\hat{i}}^{(1)} \rangle + 2\partial_{12}^2 \hat{\mathbf{g}}_{\hat{i}} \langle \hat{\mathbf{Z}}_{\hat{i}-1}^{(1)}, \hat{\mathbf{Z}}_{\hat{i}}^{(1)} \rangle. \quad (3.19d)$$

The first two parts $\mathfrak{G}_1^{\mathbf{y}}$ and $\mathfrak{G}_1^{\hat{\mathbf{l}}}$ represent the sums of first derivative of $\hat{\mathbf{g}}_j$, hence the subscript 1. The super scripts \mathbf{y} and $\hat{\mathbf{l}}$ indicate the derivative types $\mathbf{g}_{\mathbf{y}}$ and $\partial_2 \hat{\mathbf{g}}_{\hat{i}}$. Note that we have again a splitting in step independent derivative $\hat{\mathbf{g}}_{\mathbf{y}} \langle \dots \rangle$ applied to the step dependent argument $\mathbf{Z}_{i-1}^{(2)}$ and a step dependent fast function $\hat{\mathbf{g}}_i$ applied to step

dependent second derivatives. Last but not least the last symbol \mathfrak{G}_2 sums up all second derivatives applied to first derivatives.

Now we expand the three expressions (3.19b), (3.19c) (3.19d) successively and collect the coefficient of every derivative.

SECOND DERIVATIVES OF THE FAST STEPS Due to linearity it suffices to sum up $Z_{\hat{i}}^{(2)}$

$$\begin{aligned} \sum_{\hat{i}=I}^{\mathfrak{L}} \hat{Z}_{\hat{i}-I}^{(2)} &= \left[\mathfrak{L}\alpha + \frac{1}{2}(\mathfrak{L}-I)\gamma \right] \otimes I_d \cdot Y^{(2)} + [(\mathfrak{L}-I)\beta c] \otimes f_y \langle F \rangle \\ &+ 2[D\mathfrak{L}^{-1}] \otimes I_d \cdot \sum_{\hat{i}=I}^{\mathfrak{L}} \sum_{j=I}^{\hat{i}-I} \hat{g}_j(\hat{Z}_{j-I}, \hat{Z}_j)^{(1)} \end{aligned} \quad (3.20a)$$

and use equation (3.13a) to expand the last sum

$$\begin{aligned} &= [\mathfrak{L}Q_3] \otimes I_d \cdot Y^{(2)} + [(\mathfrak{L}-I)\beta c] \otimes f_y \langle F \rangle + 2 \sum_{\hat{i}=I}^{\mathfrak{L}} Q_1(\hat{i}-I) \otimes g_y \langle F \rangle \\ &+ 2 [D\mathfrak{L}^{-2}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot \hat{g}_y^{1s} \langle \hat{F} \rangle . \end{aligned} \quad (3.20b)$$

Again we ordered the summands, such that the fast functions come after the slow ones. After inserting $Y^{(2)}$ and collecting by derivatives

$$\begin{aligned} &= [\mathfrak{L}Q_3^{fy}] \otimes f_y \langle F \rangle + \left[\mathfrak{L} \left(Q_3 \left(Q_2^I - \frac{1}{L} Q_2 \right) + 2Q_1^S \right) \right] \otimes g_y \langle F \rangle \\ &+ \frac{2\kappa}{L} [\mathfrak{L}Q_3 Q_2] \otimes g_y \langle F \rangle + 2[D\mathfrak{L}^{-2}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot \hat{g}_y^{1s} \langle \hat{F} \rangle \end{aligned} \quad (3.20c)$$

we introduce the coefficients

$$Q_3 = \alpha + \frac{1}{2}(I - \mathfrak{L}^{-1})\gamma \quad (3.20d)$$

$$Q_3^{fy} = (I + \alpha + \mathfrak{L}^{-1}(\alpha - I)) \hat{A}c \quad (3.20e)$$

for the repeated use. We obtain \mathfrak{G}_1^y by applying $\hat{g}_y \langle \dots \rangle$ and left multiplying with \mathfrak{L}^{-1} , i.e.

$$\begin{aligned} \mathfrak{G}_1^y &= Q_3^{fy} \otimes g_y \langle f_y \langle F \rangle \rangle + \left[Q_3 \left(Q_2^I - \frac{1}{L} Q_2 \right) + 2Q_1^S \right] \otimes g_y \langle g_y \langle F \rangle \rangle \\ &+ \frac{2\kappa}{L} [Q_3 Q_2] \otimes g_y \langle g_y \langle F \rangle \rangle + 2[D\mathfrak{L}^{-3}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot \hat{g}_y \langle \hat{g}_y^{1s} \langle \hat{F} \rangle \rangle . \end{aligned} \quad (3.21)$$

Again we ordered the derivatives from slow to fast and moved the fast scale integrator dependent terms to the end.

The second sum $\mathfrak{G}_1^{\hat{f}}$ represents the application of the fast scale integrator on every fast step. Strictly speaking, we apply the first derivative of the fast scale integrator on the difference of two fast steps. After using equation (3.12b) we point out that the difference

$$\begin{aligned} \hat{Z}_i^{(2)} - \hat{Z}_{i-1}^{(2)} &= [\mathfrak{L}^{-1}\gamma] \otimes I_d \cdot Y^{(2)} + 2[\mathfrak{L}^{-1}\beta c] \otimes f_y \langle F \rangle + 2[D\mathfrak{L}^{-1}\tilde{c}] \otimes g_y \langle F \rangle \\ &+ 2[(\hat{I} - I)\mathfrak{L}^{-1}D(c - \tilde{c})] \otimes g_y \langle F \rangle + 2[D\mathfrak{L}^{-2}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot \partial_2 \hat{g}_i \langle \hat{F} \rangle \end{aligned} \quad (3.22a)$$

has two step dependent terms, once the coefficient and then the derivative. Due to the step dependency we have to apply the step dependent derivative $\partial_2 \hat{g}_i \langle \dots \rangle$ first and sum up afterwards to obtain the intermediate result

$$\begin{aligned} \mathfrak{G}_1^{\hat{f}} &= \kappa[\gamma] \otimes I_d \cdot \hat{g}_y \langle Y^{(2)} \rangle + 2\kappa[\beta c] \otimes g_y \langle f_y \langle F \rangle \rangle + 2\kappa[D\tilde{c}] \otimes g_y \langle g_y \langle F \rangle \rangle \\ &+ 2[D\mathfrak{L}^{-2}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot (\hat{g}_y^1 \langle \hat{g}_y \langle \hat{F} \rangle \rangle - \hat{g}_y^0 \langle \hat{F} \rangle) . \end{aligned} \quad (3.22b)$$

After inserting the second derivative $Y^{(2)}$ and collecting by derivatives, we finish with the sum

$$\begin{aligned} &= 2\kappa[(I - \alpha)\hat{A}c] \otimes g_y \langle f_y \langle F \rangle \rangle + \kappa \left[\left(\gamma \left(Q_2^I - \frac{1}{L} Q_2 \right) + 2D\tilde{c} \right) \right] \otimes g_y \langle g_y \langle F \rangle \rangle \\ &+ \frac{2\kappa^2}{L} [\gamma Q_2] \otimes g_y \langle g_y \langle F \rangle \rangle + 2[D\mathfrak{L}^{-2}(c - \tilde{c})] \otimes \mathbb{1}_d \cdot (\hat{g}_y^1 \langle \hat{g}_y \langle \hat{F} \rangle \rangle - \hat{g}_y^0 \langle \hat{F} \rangle) . \end{aligned} \quad (3.22c)$$

All derivatives depend on the fast scale integrator. We order them in such a way, that the first three derivatives and their coefficients are independent of the fast scale step, whereas the last two depend on the steps.

SECOND DERIVATIVES OF THE FAST SCALE FUNCTION \hat{g}_i Last, but not least, we analyze the second derivative expression \mathfrak{G}_2 . We already know the expression for $Z_j^{(1)}$

and it does not depend on the fast scale evaluations. In analogy to the first derivative $\mathbf{g}_y \langle F \rangle$ we use the relation (2.2c) and rewrite \mathfrak{G}_2

$$\mathfrak{G}_2 = Q_4 \otimes \mathbf{g}_{yy} \langle F, F \rangle + \sum_{\hat{i}}^{\mathfrak{L}} Q_5(\hat{i}) \cdot \partial_{22}^2 \hat{g}_{\hat{i}} \langle \hat{F}, \hat{F} \rangle + 2 \sum_{\hat{i}}^{\mathfrak{L}} Q_6(\hat{i}) \cdot \partial_{12}^2 \hat{g}_{\hat{i}} \langle \hat{F}, \hat{F} \rangle \quad (3.23a)$$

as a linear combination of the exact derivative $\mathbf{g}_{yy} \langle F, F \rangle$ and the derivatives of the fast scale evaluations with the coefficients

$$Q_4 = \frac{1}{3}(c^2 + c\tilde{c} + \tilde{c}^2) - \frac{1}{2}\mathfrak{L}^{-1}(c^2 - \tilde{c}^2) + \frac{1}{6}\mathfrak{L}^{-2}(c - \tilde{c})^2 \quad (3.23b)$$

$$Q_5(\hat{i}) = 2\mathfrak{L}^{-2}(c - \tilde{c})\tilde{c} + (2\hat{i} - I)\mathfrak{L}^{-3}(c - \tilde{c})^2 \otimes \mathbb{1}_d \quad (3.23c)$$

$$Q_6(\hat{i}) = \mathfrak{L}^{-2}(c - \tilde{c})\tilde{c} + (\hat{i} - I)\mathfrak{L}^{-3}(c - \tilde{c})^2 \otimes \mathbb{1}_d. \quad (3.23d)$$

Again we moved the step independent coefficient, namely Q_4 to the front. The two remaining derivatives belong to fast scale integrator dependent derivatives with step dependent coefficients.

II.3.3 Common order conditions of MFS methods

At this time we have the second derivative (3.16a) at hand. The third derivative is a linear combination of (3.18b), (3.18c) and (3.19). Therefore we can extract the classical and MIS conditions by inspecting the coefficients of the derivatives.

The second derivative, given by equation (3.14), reveals the classical condition

$$\mathbf{b} \cdot \mathbf{c} = \frac{1}{2} \quad (3.24)$$

together with the MIS condition

$$\tilde{\mathbf{b}} \cdot (\mathbf{c} + \tilde{\mathbf{c}}) = 1 \quad (3.25a)$$

$$\tilde{\mathbf{b}} := \mathbf{e}_{s+1}^T \mathbf{R} \mathbf{D} \quad (3.25b)$$

for order two. From equation (3.18b) we easily read the remaining classical third order conditions

$$\mathbf{b} \cdot \mathbf{c}^2 = \frac{1}{3} \quad (3.26a)$$

$$\mathbf{b} \cdot \mathbf{A}\mathbf{c} = \frac{1}{6} \quad (3.26b)$$

and the MIS condition

$$\mathbf{b} \cdot \mathbf{Q}_2^I = \frac{1}{3}. \quad (3.26c)$$

We obtain the remaining MIS conditions from the sum of the fast terms in equation (3.19). Keep in mind, that the coefficients from (3.19b), (3.19c) and (3.19d) have to be left multiplied by $3\tilde{\mathbf{b}}$ due to (3.17). The MIS conditions are independent of the fast scale solver. Hence we can neglect all derivatives containing the fast function $\hat{\mathbf{g}}_1$ and concentrate on $\mathbf{g}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle$ and $\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$. We collect their coefficients by powers of \mathfrak{L} and inspect only the \mathfrak{L} free parts (i.e. the zeroth power). The coefficient of the derivative $\mathbf{g}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle$ is already ordered in this way. So we extract the MIS condition

$$\tilde{\mathbf{b}} \cdot (\mathbf{I} + \alpha)\hat{\mathbf{A}}\mathbf{c} = \frac{1}{3}. \quad (3.27a)$$

We expand the coefficient of $\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$

$$\mathbf{Q}_3 \left(\mathbf{Q}_2^I - \frac{1}{\mathbf{L}} \mathbf{Q}_2 \right) + 2\mathbf{Q}_1^S = \mathbf{Q}_7^0 + \frac{1}{\mathbf{L}} \mathbf{Q}_7^1 + \frac{1}{\mathbf{L}^2} \mathbf{Q}_7^2 \quad (3.27b)$$

in three parts according to the powers of \mathbf{L} with the short hands

$$\mathbf{Q}_7^0 := \left(\alpha + \frac{1}{2}\gamma \right) \mathbf{Q}_2^I + \frac{1}{3}\mathbf{D}(\mathbf{c} + 2\tilde{\mathbf{c}}) \quad (3.27c)$$

$$\mathbf{Q}_7^1 := -\hat{\mathbf{L}}^{-1} \left(\frac{1}{2}\gamma \mathbf{Q}_2^I + \mathbf{D}\mathbf{c} \right) - \left(\alpha + \frac{1}{2}\gamma \right) \mathbf{Q}_2 \quad (3.27d)$$

$$\mathbf{Q}_7^2 := \frac{2}{3}\hat{\mathbf{L}}^{-2}\mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}) + \frac{1}{2}\hat{\mathbf{L}}^{-1}\gamma \mathbf{Q}_2 \quad (3.27e)$$

and extract all \mathbf{L} independent coefficients to get the MIS condition.

$$\tilde{\mathbf{b}} \cdot \left(\left(\alpha + \frac{1}{2}\gamma \right) \mathbf{Q}_2^I + \frac{1}{3}\mathbf{D}(\mathbf{c} + 2\tilde{\mathbf{c}}) \right) = \frac{1}{3}. \quad (3.27f)$$

1	$\mathbf{b} \cdot \mathbf{1} = 1$	(3.9a)	F	$\beta \mathbf{1} = D\mathbf{1}$	(3.9b)	F
2	$\mathbf{b} \cdot \mathbf{c} = \frac{1}{2}$	(3.24)	$\mathbf{f}_y \langle F \rangle$	$\tilde{\mathbf{b}} \cdot (\mathbf{c} + \tilde{\mathbf{c}}) = 1$	(3.25a)	$\mathbf{g}_y \langle F \rangle$
3	$\mathbf{b} \cdot A\mathbf{c} = \frac{1}{6}$	(3.26b)	$\mathbf{f}_y \langle \mathbf{f}_y \langle F \rangle \rangle$	$\mathbf{b} \cdot RD(\mathbf{c} + \tilde{\mathbf{c}}) = \frac{1}{3}$	(3.26c)	$\mathbf{f}_y \langle \mathbf{g}_y \langle F \rangle \rangle$
3				$\tilde{\mathbf{b}} \cdot (I + \alpha)\hat{A}\mathbf{c} = \frac{1}{3}$	(3.27a)	$\mathbf{g}_y \langle \mathbf{f}_y \langle F \rangle \rangle$
3				$\tilde{\mathbf{b}} \cdot ((\alpha + \frac{1}{2}\gamma)Q_2^I + \frac{1}{3}D(\mathbf{c} + 2\tilde{\mathbf{c}})) = \frac{1}{3}$	(3.27f)	$\mathbf{g}_y \langle \mathbf{g}_y \langle F \rangle \rangle$
3	$\mathbf{b} \cdot \mathbf{c}^2 = \frac{1}{3}$	(3.26a)	$\mathbf{f}_{yy} \langle F, F \rangle$	$\tilde{\mathbf{b}} \cdot (\mathbf{c}^2 + \mathbf{c}\tilde{\mathbf{c}} + \tilde{\mathbf{c}}^2) = 1$	(3.27g)	$\mathbf{g}_{yy} \langle F, F \rangle$

Table 1: Combined classical and MIS conditions up to order three.

The same procedure applied to the the second derivatives of \mathbf{g} , i.e. equation (3.23a), reveals the tenth MIS condition

$$\tilde{\mathbf{b}} \cdot (\mathbf{c}^2 + \mathbf{c}\tilde{\mathbf{c}} + \tilde{\mathbf{c}}^2) = 1. \quad (3.27g)$$

At this point, we collected all classic and MIS order conditions up to order three in the table 1. Furthermore, we have no additional MFS condition for order one, as predicted by theorem 2. For order two, we get at most one additional MFS condition from equation (3.18c). We obtain the MFS conditions for order three by inspecting the equations (3.21), (3.22c) and (3.23a). For a given unified fast scale function \mathbf{g}_j^X we obtain the MFS order conditions for order two and three with the procedure:

1. Simplify the expressions (3.15) and insert into (3.18c), (3.21), (3.22c) and (3.23a). After simplification, we obtain the coefficients for every derivative.
2. Inspect the coefficients for every derivative. We extract the step factors using $\mathfrak{L} = L\hat{L}$ and collect the terms by powers of L . To obtain an order independent of the number of steps the coefficients of every power of L must vanish. Keep in mind that the coefficients with outer most derivative of \mathbf{f} or \mathbf{g} have to be left multiplied by \mathbf{b} or $\tilde{\mathbf{b}}$ respectively.

In the next subsections we insert the unified fast scale functions from the previous section and derive their corresponding MFS order conditions.

II.3.4 Explicit Euler as fast scale integrator

The function \mathbf{g}_j^{EE} does not depend on the step j , and is independent of the second argument. Hence all expressions $\hat{\mathfrak{G}}_y^1 \langle \hat{F} \rangle$, $\hat{\mathfrak{G}}_y^0 \langle \hat{F} \rangle$, $\mathfrak{G}_y \langle F \rangle$ and κ vanish. The same applies to the second derivatives $\partial_{22}^2 \hat{\mathfrak{g}}_j \langle F, F \rangle$ and $\partial_{12}^2 \hat{\mathfrak{g}}_j \langle F, F \rangle$.

We expand the definition of Q_2 and Q_2^I from equation (3.16a) to obtain the second derivative

$$Y^{(2)} = 2[\hat{A}c] \otimes f_y \langle F \rangle + [RD((c + \tilde{c}) - \mathcal{L}^{-1}(c - \tilde{c}))] \otimes g_y \langle F \rangle \quad (3.28)$$

together with the MFS condition for order two. According to the procedure we continue with the the equation (3.18c) for the slow-fast term

$$\mathfrak{F} = \frac{1}{L} [\hat{A}Q_2] \otimes f_y \langle g_y \langle F \rangle \rangle \quad (3.29a)$$

and the three fast terms (3.21), (3.22c) and (3.23a)

$$\mathfrak{G}_1^y = Q_3^{f_y} \otimes g_y \langle f_y \langle F \rangle \rangle + \left[\sum_{k=0}^2 \frac{1}{L^k} Q_7^k \right] \otimes g_y \langle g_y \langle F \rangle \rangle \quad (3.29b)$$

$$\mathfrak{G}_1^I = 0 \quad (3.29c)$$

$$\mathfrak{G}_2 = Q_4 \otimes g_{yy} \langle F, F \rangle \quad (3.29d)$$

The four expressions in (3.29) contain four different derivatives. After summing up \mathfrak{G}_1^y and \mathfrak{G}_1^I we obtain the conditions

$$0 = \tilde{\mathbf{b}} \cdot \hat{L}^{-1}(c - \tilde{c}) \quad g_y \langle F \rangle \quad (3.30a)$$

$$0 = \mathbf{b} \cdot RD\hat{L}(c - \tilde{c}) \quad f_y \langle g_y \langle F \rangle \rangle \quad (3.30b)$$

$$0 = \tilde{\mathbf{b}} \cdot \hat{L}^{-1}(I - \alpha)\hat{A}c \quad g_y \langle f_y \langle F \rangle \rangle \quad (3.30c)$$

$$0 = \tilde{\mathbf{b}} \cdot (\hat{L}^{-1}\gamma Q_2^I + (2\alpha + \gamma) Q_2 + \hat{L}^{-1}Dc) \quad g_y \langle g_y \langle F \rangle \rangle \quad (3.30d)$$

$$0 = \tilde{\mathbf{b}} \cdot \left(\frac{2}{3}\hat{L}^{-2}D(c - \tilde{c}) + \frac{1}{2}\hat{L}^{-1}\gamma Q_2 \right) \quad g_y \langle g_y \langle F \rangle \rangle \quad (3.30e)$$

$$0 = \tilde{\mathbf{b}} \cdot \hat{L}^{-1}(c^2 - \tilde{c}^2) \quad g_{yy} \langle F, F \rangle \quad (3.30f)$$

$$0 = \tilde{\mathbf{b}} \cdot \hat{L}^{-2}(c - \tilde{c})^2 \quad g_{yy} \langle F, F \rangle \quad (3.30g)$$

These are all MFS conditions for order three with explicit Euler. We present the conditions ordered by the derivatives and powers of L in the Table 2. To summarize, together with the classic, the MIS and the seven MFS conditions, we have to fulfill 17 conditions. This requires at least four stages.

zero	$-\text{pow}(\mathbf{L})$	derivative
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})$	1	$\mathbf{g}_y \langle \mathbf{F} \rangle$
$\mathbf{b} \cdot \mathbf{R} \mathbf{D} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})$	1	$\mathbf{f}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\mathbf{I} - \alpha) \hat{\mathbf{A}} \mathbf{c}$	1	$\mathbf{g}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \left(\frac{1}{2} (\hat{\mathbf{L}}^{-1} \gamma \mathbf{R} \mathbf{D}(\mathbf{c} + \tilde{\mathbf{c}}) + (2\alpha + \gamma) \mathbf{R} \mathbf{D} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})) + \hat{\mathbf{L}}^{-1} \mathbf{D} \mathbf{c} \right)$	1	$\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \left(\frac{1}{2} \hat{\mathbf{L}}^{-1} \gamma \mathbf{R} \mathbf{D} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) + \frac{2}{3} \hat{\mathbf{L}}^{-2} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}) \right)$	2	$\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\tilde{\mathbf{c}}^2 - \mathbf{c}^2)$	1	$\mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})^2$	2	$\mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle$

Table 2: The MFS conditions for order two and order three with explicit Euler. The first column contains the zero expression, the second the negative power of \mathbf{L} and the last column the corresponding derivative.

For completeness we plug in all results into the general formula for the third derivative (3.17)

$$\begin{aligned}
 \mathbf{Y}^{(3)} &= 3[\hat{\mathbf{A}}\mathbf{c}^2] \otimes \mathbf{f}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle + 6[\hat{\mathbf{A}}\hat{\mathbf{A}}\mathbf{c}] \otimes \mathbf{f}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle \\
 &\quad + 3[\hat{\mathbf{A}}\mathbf{R}\mathbf{D}(\mathbf{c} + \tilde{\mathbf{c}} - \mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}))] \otimes \mathbf{f}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle \\
 &\quad + 3[\mathbf{R}\mathbf{D}(\mathbf{I} + \alpha - \mathfrak{L}^{-1}(\mathbf{I} - \alpha))\hat{\mathbf{A}}\mathbf{c}] \otimes \mathbf{g}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle \\
 &\quad + 3 \left[\mathbf{R}\mathbf{D} \left(\alpha + \frac{1}{2}(\mathbf{I} - \mathfrak{L}^{-1})\gamma \right) \mathbf{R}\mathbf{D}(\mathbf{c} + \tilde{\mathbf{c}} - \mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})) \right. \\
 &\quad \left. + \frac{1}{3}\mathbf{R}\mathbf{D}^2(\mathbf{c} + 2\tilde{\mathbf{c}} - 3\mathfrak{L}^{-1}\mathbf{c} + 2\mathfrak{L}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})) \right] \otimes \mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle \\
 &\quad + \left[\mathbf{R}\mathbf{D} \left(\mathbf{c}^2 + \mathbf{c}\tilde{\mathbf{c}} + \tilde{\mathbf{c}}^2 + \frac{3}{2}\mathfrak{L}^{-1}(\tilde{\mathbf{c}}^2 - \mathbf{c}^2) + \frac{1}{2}\mathfrak{L}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})^2 \right) \right] \otimes \mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle
 \end{aligned} \tag{3.31}$$

The excessive length reflects the complicated interaction between the fast terms. If one knows the numbers of fast steps and the step ratios, one can extract the order conditions for the complete method. These conditions then correspond to the GARK methods.

According to theorem 3 the implicit Euler method has $\kappa = 1$ and $\mathbf{g}_y \langle \mathbf{F} \rangle = \mathbf{g}_y \langle \mathbf{F} \rangle$. Then one sees easily the sign change of the MFS part in the second derivative. Furthermore we see the higher order terms from lemma 2 especially in the coefficients of the derivative $\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle$.

II.3.5 Forward-backward Euler as fast scale integrator

We have introduced the forward-backward Euler, also known as split-explicit Euler, function \mathbf{g}_j^{FB} in 2.3b. The forward-backward Euler method computes the component p explicit, and uses the result in the update of the second component. Hence the partial

zero	$-\text{pow}(\mathbf{L})$	derivative
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})$	1	$\mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle$
$\mathbf{b} \cdot \mathbf{RD} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}})$	1	$\mathfrak{f}_{\mathbf{y}} \langle \mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle, \mathfrak{f}_{\mathbf{y}} \langle \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\mathbf{I} - \alpha) \hat{\mathbf{A}} \mathbf{c}$	1	$\mathfrak{g}_{\mathbf{y}} \langle \mathfrak{f}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle, \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathfrak{f}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \left[\left(\alpha + \frac{1}{2} \gamma \right) \mathbf{RD} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) + \frac{1}{2} \hat{\mathbf{L}}^{-1} \gamma \mathbf{RD}(\mathbf{c} + \tilde{\mathbf{c}}) + \hat{\mathbf{L}}^{-1} \mathbf{D} \mathbf{c} \right]$	1	$\mathfrak{g}_{\mathbf{y}} \langle \mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \left[\frac{1}{2} \hat{\mathbf{L}}^{-1} \gamma \mathbf{RD} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) + \frac{2}{3} \hat{\mathbf{L}}^{-2} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}) \right]$	2	$\mathfrak{g}_{\mathbf{y}} \langle \mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot (\hat{\mathbf{L}}^{-1}(\mathbf{I} - \alpha) \mathbf{RD}(\mathbf{c} + \tilde{\mathbf{c}}))$	1	$\partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot (\hat{\mathbf{L}}^{-1} \gamma \mathbf{RD} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) + \hat{\mathbf{L}}^{-2} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}))$	2	$\partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle \rangle,$ $\mathfrak{g}_{\mathbf{y}} \langle \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle,$ $\partial_2 \mathfrak{g}_1^{\text{FB}} \langle \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot ((2\alpha + \gamma) \mathbf{RD} \hat{\mathbf{L}}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) + \hat{\mathbf{L}}^{-1} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}))$	1	$\mathfrak{g}_{\mathbf{y}} \langle \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-1}(\tilde{\mathbf{c}}^2 - \mathbf{c}^2)$	1	$\mathfrak{g}_{\text{pp}} \langle \mathbf{F}, \mathbf{F} \rangle, \mathfrak{g}_{\text{qq}} \langle \mathbf{F}, \mathbf{F} \rangle,$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})^2$	2	$\mathfrak{g}_{\text{pp}} \langle \mathbf{F}, \mathbf{F} \rangle, \mathfrak{g}_{\text{qq}} \langle \mathbf{F}, \mathbf{F} \rangle,$ $\mathfrak{g}_{\text{qq}} \langle \mathbf{F}, \mathbf{F} \rangle$

Table 3: The conditions for order three with forward-backward Euler. The first column contains the zero expression, the second the negative power of \mathbf{L} and the last column the corresponding derivative.

derivative $\partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle$ does not vanish now and differs from the derivative $\mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle$. Again $\mathfrak{g}_j^{\text{FB}}$ is independent of j so we select the first step.

$$\mathfrak{g}_{\mathbf{y}} \langle \mathbf{F} \rangle = \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \quad (3.32a)$$

$$\hat{\mathfrak{g}}_{\mathbf{y}}^1 \langle \hat{\mathbf{F}} \rangle = \hat{\mathfrak{g}}_{\mathbf{y}}^{1s} \langle \hat{\mathbf{F}} \rangle = \frac{1}{2} [\mathfrak{L}(\mathfrak{L} - \mathbf{I}) \mathbf{1}] \otimes \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \quad (3.32b)$$

$$\hat{\mathfrak{g}}_{\mathbf{y}}^0 \langle \hat{\mathbf{F}} \rangle = [\mathfrak{L} \mathbf{1}] \otimes \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \partial_2 \mathfrak{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle \quad (3.32c)$$

and $\kappa = 1$. This time the fast scale integrator depends on the old and new step. The second derivatives

$$\partial_{22}^2 \hat{\mathfrak{g}}_j \langle \mathbf{F}, \mathbf{F} \rangle = \mathbf{1} \otimes \partial_{22}^2 \mathfrak{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle \quad (3.32d)$$

$$\partial_{12}^2 \hat{\mathfrak{g}}_j \langle \mathbf{F}, \mathbf{F} \rangle = \mathbf{1} \otimes \partial_{12}^2 \mathfrak{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle \quad (3.32e)$$

are independent of the step, but do not vanish anymore. We obtain the second derivative

$$\mathcal{Y}^{(2)} = 2[\hat{\mathbf{A}}\mathbf{c}] \otimes \mathbf{f}_y \langle \mathbf{F} \rangle + \left[\mathbf{Q}_2^I - \frac{1}{\mathbf{L}} \mathbf{Q}_2 \right] \otimes \mathbf{g}_y \langle \mathbf{F} \rangle + \frac{2}{\mathbf{L}} \mathbf{Q}_2 \otimes \partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{F} \rangle, \quad (3.33)$$

which differs from the explicit Euler only in the additional derivative $\partial_2 \hat{\mathbf{g}}_1 \langle \mathbf{F} \rangle$.

In particular the four expressions

$$\mathfrak{F} = \frac{1}{\mathbf{L}} [\hat{\mathbf{A}}\mathbf{Q}_2] \otimes \mathbf{f}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle - \frac{2}{\mathbf{L}} [\hat{\mathbf{A}}\mathbf{Q}_2] \otimes \mathbf{f}_y \langle \partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle \quad (3.34a)$$

for the slow term and the two fast terms

$$\mathfrak{G}_1^y = \mathbf{Q}_3^{f_y} \otimes \mathbf{g}_y \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle + \left[\sum_{k=0}^2 \frac{1}{\mathbf{L}^k} \mathbf{Q}_7^k \right] \otimes \mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle \quad (3.34b)$$

$$+ \left[\mathfrak{L}^{-1}(\mathfrak{L}^{-1} - \mathbf{I})\mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}) + \frac{2}{\mathbf{L}} \mathbf{Q}_3 \mathbf{Q}_2 \right] \otimes \mathbf{g}_y \langle \partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle$$

$$\begin{aligned} \mathfrak{G}_1^{\hat{\mathbf{I}}} &= 2[(\mathbf{I} - \alpha)\hat{\mathbf{A}}\mathbf{c}] \otimes \partial_2 \mathbf{g}_1^X \langle \mathbf{f}_y \langle \mathbf{F} \rangle \rangle + 2 \left[\frac{1}{\mathbf{L}} \gamma \mathbf{Q}_2 - \mathbf{D} \mathfrak{L}^{-1}(\mathbf{c} - \tilde{\mathbf{c}}) \right] \otimes \partial_2 \mathbf{g}_1^{\text{FB}} \langle \partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{F} \rangle \rangle \\ &+ \left[\gamma \left(\mathbf{Q}_2^I - \frac{1}{\mathbf{L}} \mathbf{Q}_2 \right) + \mathbf{D}(\mathbf{c} + \tilde{\mathbf{c}}) - \mathfrak{L}^{-1} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}}) \right] \otimes \partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle \end{aligned} \quad (3.34c)$$

$$\mathfrak{G}_2 = \mathbf{Q}_4 \otimes \mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle + \mathbf{Q}_5^S \otimes \partial_{22}^2 \mathbf{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle + \mathbf{Q}_6^S \otimes \partial_{12}^2 \mathbf{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle \quad (3.34d)$$

introduce the additional derivatives $\partial_2 \mathbf{g}_1^{\text{FB}} \langle \mathbf{F} \rangle$ and $\partial_{22}^2 \mathbf{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle$ again. The coefficients of $\partial_{22}^2 \mathbf{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle$ and $\partial_{12}^2 \mathbf{g}_1 \langle \mathbf{F}, \mathbf{F} \rangle$, namely

$$\mathbf{Q}_5^S := \sum_{\hat{\mathbf{l}}=\mathbf{I}}^{\mathfrak{L}} \mathbf{Q}_5(\hat{\mathbf{l}}) = \mathfrak{L}^{-1}(\mathbf{c}^2 - \tilde{\mathbf{c}}^2) \quad \text{and} \quad (3.35a)$$

$$\mathbf{Q}_6^S := 2 \sum_{\hat{\mathbf{l}}=\mathbf{I}}^{\mathfrak{L}} \mathbf{Q}_6(\hat{\mathbf{l}}) = \mathfrak{L}^{-1}(\mathbf{c}^2 - \tilde{\mathbf{c}}^2) - \mathfrak{L}^{-2}(\mathbf{c} - \tilde{\mathbf{c}})^2 \quad (3.35b)$$

reflect the component partitioning. In particular if \mathbf{g}^p does not depend on \mathbf{y}^p and \mathbf{g}^q does not depend in \mathbf{y}^q , the mixed derivative vanishes. But the both terms lead to conditions which are already present in the derivative $\mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle$ through the coefficient \mathbf{Q}_4 .

This time, we have two ways to express the order conditions. We ignore the structure of \mathbf{g}^{FB} and the derivatives. Then every coefficient of the additional derivative $\partial_2 \mathbf{g}_1^{\text{FB}} \langle \dots \rangle$ and every coefficient of \mathbf{L} must vanish. Hence we obtain the 10 MFS conditions in Table

3. Now we have 20 conditions in total, and therefore, we need at least four stages for order three.

II.3.6 Störmer-Verlet as fast scale integrator

The Störmer-Verlet function g_j^{SV} (2.3c) depends on the step j in a piece wise constant way. It is constant for every odd and every even step j . The computation of the expressions (3.15) leads to

$$g_y \langle F \rangle = g_y \langle F \rangle \quad (3.36a)$$

$$\hat{g}_y^{1s} \langle \hat{F} \rangle = \frac{1}{4} [\mathcal{L}^2 \mathbf{1}] \otimes g_y \langle F \rangle - \frac{1}{2} [\mathcal{L} \mathbf{1}] \otimes \partial_2 g_2^{SV} \langle F \rangle \quad (3.36b)$$

$$\hat{g}_y^1 \langle \hat{F} \rangle = \frac{1}{4} [\mathcal{L}^2 \mathbf{1}] \otimes g_y \langle F \rangle - \frac{1}{2} [\mathcal{L} \mathbf{1}] \otimes \partial_2 g_1^{SV} \langle F \rangle \quad (3.36c)$$

$$\hat{g}_y^0 \langle \hat{F} \rangle = \frac{1}{2} [\mathcal{L} \mathbf{1}] \otimes (\partial_2 g_1^{SV} \langle \partial_2 g_1^{SV} \langle F \rangle \rangle + \partial_2 g_2^{SV} \langle \partial_2 g_2^{SV} \langle F \rangle \rangle) \quad (3.36d)$$

and $\kappa = \frac{1}{2}$. The details make use of the odd and even splitting and can be found in the appendix (.1). We see from equation (2.3c) that the mixed second derivatives of g_j^{SV} vanish and the second derivative $\partial_{22}^2 g_1 \langle F, F \rangle$ splits according to the component partitioning. As expected from the theorem 2 the second derivative

$$Y^{(2)} = 2[\hat{A}c] \otimes f_y \langle F \rangle + [RD(c + \tilde{c})] \otimes g_y \langle F \rangle \quad (3.37)$$

is independent of the fast step ratios and $\mathfrak{F} = 0$. We proceed with the procedure for the third derivatives, i.e. we compute the expressions

$$\begin{aligned} \mathfrak{G}_1^y &= Q_3^{fy} \otimes g_y \langle f_y \langle F \rangle \rangle - [D\mathcal{L}^{-2}(c - \tilde{c})] \otimes g_y \langle \partial_2 g_1^{SV} \langle F \rangle \rangle \\ &\quad + \left[Q_3 Q_2^I + 2Q_1^S + \frac{1}{2} \mathcal{L}^{-1} D(c - \tilde{c}) \right] \otimes g_y \langle g_y \langle F \rangle \rangle \end{aligned} \quad (3.38a)$$

$$\begin{aligned} \mathfrak{G}_1^{\hat{I}} &= [(I - \alpha)\hat{A}c] \otimes g_y \langle f_y \langle F \rangle \rangle + \frac{1}{2} [\gamma Q_2^I + D(c + \tilde{c})] \otimes g_y \langle g_y \langle F \rangle \rangle \\ &\quad - [D\mathcal{L}^{-1}(c - \tilde{c})] \otimes (\partial_2 g_1^{SV} \langle g_y \langle F \rangle \rangle + \partial_2 g_1^{SV} \langle \partial_2 g_1^{SV} \langle F \rangle \rangle + \partial_2 g_2^{SV} \langle \partial_2 g_2^{SV} \langle F \rangle \rangle) \end{aligned} \quad (3.38b)$$

$$\mathfrak{G}_2 = Q_4 \otimes g_{yy} \langle F, F \rangle + \sum_{\hat{i}=I}^{\frac{1}{2}\mathcal{L}} Q_5(2\hat{i}) \cdot \mathbf{1} \otimes g_{yy}^p \langle F, F \rangle + Q_5(2\hat{i} - I) \cdot \mathbf{1} \otimes g_{yy}^q \langle F, F \rangle \quad (3.38c)$$

$$= \left[Q_4 + \frac{1}{2} \mathfrak{L}^{-1}(c^2 - \tilde{c}^2) \right] \otimes \mathfrak{g}_{yy} \langle F, F \rangle + \frac{1}{2} [\mathfrak{L}^{-2}(c - \tilde{c})^2] \otimes (\mathfrak{g}_{yy}^p \langle F, F \rangle - \mathfrak{g}_{yy}^q \langle F, F \rangle) \quad (3.38d)$$

$$= \left[\frac{1}{3}(c^2 + c\tilde{c} + \tilde{c}^2) + \frac{1}{6} \mathfrak{L}^{-2}(c - \tilde{c})^2 \right] \otimes \mathfrak{g}_{yy} \langle F, F \rangle \quad (3.38e)$$

$$+ \frac{1}{2} [\mathfrak{L}^{-2}(c - \tilde{c})^2] \otimes (\mathfrak{g}_{yy}^p \langle F, F \rangle - \mathfrak{g}_{yy}^q \langle F, F \rangle)$$

This time the expressions \mathfrak{G}_1^y and $\mathfrak{G}_1^{\hat{1}}$ contain common derivatives. So we sum them up and collect the coefficients by powers of L :

- The fast slow term $\mathfrak{g}_y \langle f_y \langle F \rangle \rangle$:

$$Q_3^{fy} + \mathfrak{L}^{-1}(I - \alpha)\hat{A}c = (I - \alpha)\hat{A}c \quad (3.39a)$$

- The elementary differential $\mathfrak{g}_y \langle g_y \langle F \rangle \rangle$:

$$Q_3 Q_2^I + 2Q_1^S + \frac{1}{2} \mathfrak{L}^{-1} D(c - \tilde{c}) + \frac{1}{2} \mathfrak{L}^{-1} (\gamma Q_2^I + D(c + \tilde{c})) \quad (3.39b)$$

$$= \left(\alpha + \frac{1}{2} \gamma \right) Q_2^I + \frac{1}{6} D(c + 2\tilde{c}) + \frac{2}{3} \mathfrak{L}^{-2} D(c - \tilde{c}) \quad (3.39c)$$

- For the mixed derivatives we have to remind the splitting (3.19). In particular we have to left multiply $\mathfrak{G}_1^{\hat{1}}$ with \mathfrak{L}^{-1} before summing up. Then all mixed derivatives have the same coefficient $\mathfrak{L}^{-2} D(c - \tilde{c})$. The linear combination of elementary differentials consist of many combinations of partial derivatives of odd and even steps and derivatives of the fast function g . We exploit the equation $\partial_2 g_1^{SV} \langle F \rangle + \partial_2 g_2^{SV} \langle F \rangle = g_y \langle F \rangle$ to simplify the required elementary differentials, i.e.

$$3\partial_2 g_1^{SV} \langle \partial_2 g_1^{SV} \langle F \rangle \rangle + \partial_2 g_1^{SV} \langle \partial_2 g_2^{SV} \langle F \rangle \rangle + \partial_2 g_2^{SV} \langle \partial_2 g_1^{SV} \langle F \rangle \rangle + \partial_2 g_2^{SV} \langle \partial_2 g_2^{SV} \langle F \rangle \rangle \quad (3.40)$$

$$= 2\partial_2 g_1^{SV} \langle \partial_2 g_1^{SV} \langle F \rangle \rangle + g_y \langle g_y \langle F \rangle \rangle . \quad (3.41)$$

For the detailed computation we refer to the appendix .1.

Now we have only quadratic step terms and two MFS conditions. The Table 4 collects the conditions together with their derivatives. As predicted by theorem 2 we have only the quadratic conditions

zero	neg power of L	derivative
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-2} \mathbf{D}(\mathbf{c} - \tilde{\mathbf{c}})$	2	$\mathbf{g}_y \langle \mathbf{g}_y \langle \mathbf{F} \rangle \rangle, \partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_2^{\text{SV}} \langle \mathbf{F} \rangle \rangle$
$\tilde{\mathbf{b}} \cdot \hat{\mathbf{L}}^{-2} (\mathbf{c} - \tilde{\mathbf{c}})^2$	2	$\mathbf{g}_{yy} \langle \mathbf{F}, \mathbf{F} \rangle$

Table 4: The conditions for order three with Störmer-Verlet. The first column contains the zero expression, the second the negative power of L and the last column the corresponding derivatives

II.4 INTERPRETATION AS GARK METHOD

In the previous sections we introduced the MIS and MFS methods for multirate problems (I.2.1). Clearly there are several other approaches to solve multirate problems and in particular overcome the stability restrictions. The group Günther and Sandu [16] specialized their very general approach of the generalized additive Runge-Kutta (GARK) methods [38] to a multirate formulation with exactly one fast and slow function. Hence we review their order conditions and interpret the MFS methods as a GARK instead of a multirate GARK (mGARK). Let us first apply a two stage GARK [38, eq. 2.3] method on the model problem (I.2.1)

$$\mathbf{Y}_i^{\{1\}} = \mathbf{y}_n + h \sum_{j=1}^{s^{(1)}} \mathbf{A}_{i,j}^{\{1,1\}} f(\mathbf{Y}_j^{\{1\}}) + \sum_{j=1}^{s^{(2)}} \mathbf{A}_{i,j}^{\{1,2\}} \mathbf{g}(\mathbf{Y}_j^{\{2\}}) \quad (4.1a)$$

$$\mathbf{Y}_i^{\{2\}} = \mathbf{y}_n + h \sum_{j=1}^{s^{(1)}} \mathbf{A}_{i,j}^{\{2,1\}} f(\mathbf{Y}_j^{\{1\}}) + \sum_{j=1}^{s^{(2)}} \mathbf{A}_{i,j}^{\{2,2\}} \mathbf{g}(\mathbf{Y}_j^{\{2\}}) \quad (4.1b)$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \sum_{i=1}^{s^{(1)}} \mathbf{b}_i^{\{1\}} f(\mathbf{Y}_i^{\{1\}}) + h \sum_{i=1}^{s^{(2)}} \mathbf{b}_i^{\{2\}} \mathbf{g}(\mathbf{Y}_i^{\{2\}}) \quad (4.1c)$$

This formulation allows exactly two stage vectors. Hence, we merge all fast variables $\mathbf{Z}_{i,l}$ into the block vector

$$\mathbf{Y}^{\{2\}} = \sum_{i=1}^{s+1} \mathbf{e}_i \otimes \mathbf{Y}^{\{g_i\}}, \quad (4.2)$$

where every block $\mathbf{Y}^{\{g_i\}} \in \mathbb{R}^{\mathcal{L}_{ii}}$ relates to the fast values by $\mathbf{Y}_l^{\{g_i\}} = \mathbf{Z}_{i,l}$. In other words, the block i has length \mathcal{L}_{ii} . These entries include the initial value $\mathbf{Z}_{i,0}$. Furthermore the coefficients $\mathbf{A}^{\{i,j\}}$ for $i, j = 1, 2$ get an internal structure from the block vectors. In

$\mathbf{A}^{\{1,1\}}$	$\mathbf{A}^{\{1,2,1\}}$	$\mathbf{A}^{\{1,2,2\}}$	$\mathbf{A}^{\{1,2,3\}}$
$\mathbf{A}^{\{2,1,1\}}$	$\mathbf{A}^{\{2,2,1,1\}}$		
$\mathbf{A}^{\{2,1,2\}}$	$\mathbf{A}^{\{2,2,2,1\}}$	$\mathbf{A}^{\{2,2,2,2\}}$	
$\mathbf{A}^{\{2,1,3\}}$	$\mathbf{A}^{\{2,2,3,1\}}$	$\mathbf{A}^{\{2,2,3,2\}}$	$\mathbf{A}^{\{2,2,3,3\}}$
$\mathbf{b}^{\{1\}}$	$\mathbf{b}^{\{2,1\}}$	$\mathbf{b}^{\{2,2\}}$	$\mathbf{b}^{\{2,3\}}$

Table 5: Structure of a three stage MFS method, when viewed as a GARK method. The sub matrices $\mathbf{A}^{\{1,2,j\}}$ connect the fast evaluations $\mathbf{Z}_{i,1}$ to the slow variables, whereas their transposed counter parts $\mathbf{A}^{\{2,1,j\}}$ relate the slow evaluations to the fast variables. In an analogous manner the fast variables are connected to each other by the centered blocks $\mathbf{A}^{\{2,2,i,j\}}$. These centered blocks form a lower triangular structure, which resembles the inherent explicitness of the MFS method.

analogy to the block vector structure of $\mathbf{Y}^{\{2\}}$, we also introduce sub matrices for the GARK matrices regarding the fast steps, i.e.

$$\mathbf{A}^{\{1,2\}} = \sum_{j=1}^{s+1} \mathbf{e}_j^T \otimes \mathbf{A}^{\{1,2,j\}} \quad (4.3a)$$

$$\mathbf{A}^{\{2,1\}} = \sum_{i=1}^{s+1} \mathbf{e}_i \otimes \mathbf{A}^{\{2,1,i\}} \quad (4.3b)$$

$$\mathbf{A}^{\{2,2\}} = \sum_{j=1}^{s+1} [\mathbf{e}_i \cdot \mathbf{e}_j^T] \otimes \mathbf{A}^{\{2,2,i,j\}}. \quad (4.3c)$$

The vectors $\mathbf{b}^{\{1\}}$ and $\mathbf{b}^{\{2,j\}}$ are given by the last rows of their corresponding \mathbf{A} -block, i.e.

$$\mathbf{b}^{\{1\}} = \mathbf{e}_{s+1}^T \mathbf{A}^{\{1,1\}} \quad (4.3d)$$

$$\mathbf{b}^{\{2,j\}} = \mathbf{e}_{\mathfrak{L}_{jj}, \mathfrak{L}_{jj}}^T \cdot \mathbf{A}^{\{1,2,j\}}. \quad (4.3e)$$

The Table 5 visualizes the connection between these sub matrices. Every sub matrix $\mathbf{A}^{\{1,2,j\}}$ is in fact a row with \mathfrak{L}_{jj} columns, and in analogy every sub matrix $\mathbf{A}^{\{2,1,j\}}$ is a column consisting of \mathfrak{L}_{jj} rows. The “fast sub matrices” $\mathbf{A}^{\{2,2,i,j\}} \in \mathbb{R}^{\mathfrak{L}_{ii} \times \mathfrak{L}_{jj}}$ represent the coupling between the fast steps $\mathbf{Z}_{i,1}$. We will derive their origin below. The lower triangular structure of the matrix $\mathbf{A}^{\{2,2\}}$ resembles the explicit nature of the MFS method. In the GARK terminology the fast method is given by the coefficients $\mathbf{A}^{\{2,2\}}$. In our case the coefficients couple (nearly) all fast stages together and we completely loose the simple structure of the explicit fast method. Hence the notion of “fast method” is different between the MFS and GARK context.

Due the complicated connection between the stages in the GARK framework, we concentrate on the explicit Euler as fast scale integrator and derive the GARK coefficients for this particular case. Hence we use $\mathbf{g}_l(\mathbf{a}, \mathbf{b}) = \mathbf{a}$ in the system (3.4)

$$\begin{aligned} \mathbf{Z}_l = & \mathbb{1} \otimes \mathbf{y}_n + (\boldsymbol{\alpha} + \mathbf{l}\boldsymbol{\mathcal{L}}^{-1}\boldsymbol{\gamma}) \otimes \mathbf{I}_d \cdot (\mathbf{Y} - \mathbb{1} \otimes \mathbf{y}_n) + \mathbf{h}[\mathbf{l}\boldsymbol{\mathcal{L}}^{-1}\boldsymbol{\beta}] \otimes \mathbf{I}_d \cdot \hat{\mathbf{f}}(\mathbf{Y}) \\ & + \mathbf{h}[\mathbf{D}\boldsymbol{\mathcal{L}}^{-1}] \otimes \mathbf{I}_d \cdot \sum_{k=1}^l \hat{\mathbf{g}}(\mathbf{Z}_{k-1}) \end{aligned} \quad (4.4a)$$

for all diagonal matrices $\mathbf{l} = \mathbf{I} \dots \boldsymbol{\mathcal{L}}$ and the initial value

$$\mathbf{Z}_0 = \mathbb{1} \otimes \mathbf{y}_n + \boldsymbol{\alpha} \otimes \mathbf{I}_d \cdot (\mathbf{Y} - \mathbb{1}\mathbf{y}_n). \quad (4.4b)$$

The slow values \mathbf{Y}_n are the last fast ones, i.e.

$$\begin{aligned} \mathbf{Y}_n = \mathbf{Z}_{\boldsymbol{\mathcal{L}}} = & \mathbb{1} \otimes \mathbf{y}_n + (\boldsymbol{\alpha} + \boldsymbol{\gamma}) \otimes \mathbf{I}_d \cdot (\mathbf{Y}_n - \mathbb{1} \otimes \mathbf{y}_n) + \mathbf{h}\boldsymbol{\beta} \otimes \mathbf{I}_d \cdot \hat{\mathbf{f}}(\mathbf{Y}_n) \\ & + \mathbf{h}[\mathbf{D}\boldsymbol{\mathcal{L}}^{-1}] \otimes \mathbf{I}_d \cdot \sum_{k=1}^{\boldsymbol{\mathcal{L}}} \hat{\mathbf{g}}(\mathbf{Z}_{k-1}). \end{aligned} \quad (4.4c)$$

If we compare the equations (4.4c) with (4.1a), we recognize the missing $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ terms in the GARK formulation. Hence, we remember the definition $\mathbf{R} = (\mathbf{I} - \boldsymbol{\alpha} - \boldsymbol{\gamma})^{-1}$ and solve for the difference $\mathbf{Y}_n - \mathbb{1}_s \otimes \mathbf{y}_n$ to rewrite the slow solutions

$$\mathbf{Y}_n = \mathbb{1} \otimes \mathbf{y}_n + \mathbf{h}\hat{\mathbf{A}} \otimes \mathbf{I}_d \cdot \hat{\mathbf{f}}(\mathbf{Y}_n) + \mathbf{h}[\mathbf{R}\mathbf{D}\boldsymbol{\mathcal{L}}^{-1}] \otimes \mathbf{I}_d \cdot \sum_{l=1}^{\boldsymbol{\mathcal{L}}} \hat{\mathbf{g}}(\mathbf{Z}_l) \quad (4.4d)$$

in terms of the previous slow solutions and the fast values \mathbf{Z}_l from the MFS formulation. In analogy we get rid of the $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ terms in the fast system (4.4a)

$$\begin{aligned} \mathbf{Z}_l = & \mathbb{1} \otimes \mathbf{y}_n + \mathbf{h} [(\mathbf{l}\boldsymbol{\mathcal{L}}^{-1} + (\mathbf{I} - \mathbf{l}\boldsymbol{\mathcal{L}}^{-1})\boldsymbol{\alpha})\hat{\mathbf{A}}] \otimes \mathbf{I}_d \cdot \hat{\mathbf{f}}(\mathbf{Y}_n) \\ & + \mathbf{h} [(\boldsymbol{\alpha} + \mathbf{l}\boldsymbol{\mathcal{L}}^{-1}\boldsymbol{\gamma})\mathbf{R}\boldsymbol{\mathcal{L}}^{-1}\mathbf{D}] \otimes \mathbf{I}_d \cdot \sum_{k=1}^{\boldsymbol{\mathcal{L}}} \hat{\mathbf{g}}(\mathbf{Z}_{k-1}) + \mathbf{h}[\mathbf{D}\boldsymbol{\mathcal{L}}^{-1}] \otimes \mathbf{I}_d \cdot \sum_{k=1}^l \hat{\mathbf{g}}(\mathbf{Z}_{k-1}). \end{aligned} \quad (4.4e)$$

The two equations (4.4d) and (4.4e) show the coupling structure and give an impression how the GARK sub matrices $\mathbf{A}^{\{i,j,\mu,\nu\}}$ will look like. Equation (4.4d) relates the fast stages to the slow ones. A direct comparison directly gives the coefficients $\mathbf{A}^{\{1,1\}} = \hat{\mathbf{A}}$ and $\mathbf{b}^{\{1\}} = \mathbf{e}_{s+1}^T \hat{\mathbf{A}}$ in terms of the underlying coefficients $\hat{\mathbf{A}}$ and \mathbf{b} . The remaining three sub matrices are ways more cumbersome.

To ease the notation and get rid of the Kronecker product, we consider every single stage of the GARK vectors $Y^{[1]}$ and $Y^{[2]}$ in this order. The slow GARK values are the same as the slow values in the MFS context. Hence we obtain

$$Y_i^{[1]} = y_n + h \sum_j \hat{A}_{ij} f(Y_j^{[s]}) + h \sum_j^{i-1} A_i^{\{1,2,j\}} \sum_{l=1}^{\mathcal{L}_{jj}} g(Y_{l-1}^{[g_j]}) \quad (4.5a)$$

in stage i and read the fast to slow sub row

$$A_i^{\{1,2,j\}} = \mathbb{1}_{\mathcal{L}_{jj}}^T \cdot [RD\mathcal{L}^{-1}]_{ij} . \quad (4.5b)$$

The fast sub vectors $Y^{[g_i]}$ collect all the fast sub steps $Z_{i,l}$ in one vector. Hence we obtain

$$\begin{aligned} Y_l^{[g_i]} = & y_n + h \sum_j A_{lj}^{\{2,1,j\}} f(Y_j^{[1]}) + h \sum_j A^{\{2,2,i,j\}} \sum_{k=1}^{\mathcal{L}_{jj}} g(Y_{k-1}^{[g_j]}) \\ & + h \frac{D_{ii}}{\mathcal{L}_{ii}} \sum_{k=1}^l g(Y_{k-1}^{[g_i]}) \end{aligned} \quad (4.6a)$$

for $l = 0 \dots \mathcal{L}_{ii} - 1$ with the slow to fast coupling coefficients

$$A_{lj}^{\{2,1,f_i\}} = \left\{ \left[\frac{l}{\mathcal{L}_{ii}} + \left(1 - \frac{l}{\mathcal{L}_{ii}}\right) \alpha \right] \hat{A} \right\}_{ij} \quad (4.6b)$$

and the fast to fast coupling coefficients

$$A^{\{2,2,i,j\}} = \sum_{l=1}^{\mathcal{L}_{jj}} e_{\mathcal{L}_{ii},l} \otimes \mathbb{1}_{\mathcal{L}_{jj}}^T \cdot [(\alpha + \gamma)RD\mathcal{L}^{-1}]_{ij} \quad (4.6c)$$

for $j < i$. The diagonal blocks are given by the strict lower triangular matrix

$$A^{\{2,2,i,i\}} = \frac{D_{ii}}{L_{ii}} \sum_{j,k=1}^{j=\mathcal{L}_{ii},k<j} e_{\mathcal{L}_{ii},j} e_{\mathcal{L}_{ii},k}^T . \quad (4.6d)$$

Please note that the case $l = 0$ already contains the initial value $Z_{i,0}$, so we need no special treatment. The fast-fast coupling coefficients $A^{\{2,2,i,j\}}$ for $i \neq j$ do not exist directly in the MFS method. These fast-fast coupling coefficients stem from the α and γ

terms, which incorporate the previous slow stages into the fast evaluations. In turn the new fast stages depend on all previous fast values. From that point of view, they show how a slow coupling amounts to a coupling of fast terms in another framework. Hence again the notion of fast and slow methods differ greatly between the GARK and the MFS framework.

11.5 STABILITY

In general, the convergence of numerical methods for differential equations require consistency and stability. In the previous section we derived the consistency conditions up to order three. Hence we continue and analyze the stability of a MFS method.

The usual tool for the (linear) stability analysis of one step methods is Dahlquist's [6] test problem

$$\dot{\mathbf{y}} = \lambda \mathbf{y}$$

with $\lambda \in \mathbb{C}$ and initial value $\mathbf{y}(0) = \mathbf{y}_0$. A numerical method is stable when the solution after one step remains bounded. The application of a Runge-Kutta method with step size h on the test equation leads to a stability polynomial $R(h\lambda)$ defined by $\mathbf{y}_1 = R(h\lambda)\mathbf{y}_0$ [44, 20].

This concept of stability has to be extended to multirate and splitting methods. The extension to the additively split right hand side is straight forward. We simply split the coefficient $\lambda = \nu + \mu$ in two parts. The first part models the slow process, whereas the second one accounts for the fast processes. The important, and often complicated, challenge is the selection of useful coefficients ν and μ . We select the coefficients according to a hyperbolic model problem.

11.5.1 Model problem

Our starting point is the solution of the compressible Euler equations in one dimension, as already indicated in the motivation. After linearization we end up with the equations of linear acoustics. Hence for the stability we consider the 1D problem

$$\partial_t \mathbf{u} + \mathbf{U} \partial_x \mathbf{u} = -c_s \partial_x \pi \quad (5.1a)$$

$$\partial_t \pi + \mathbf{U} \partial_x \pi = -c_s \partial_x \mathbf{u} \quad (5.1b)$$

in an infinite domain. The unknowns are the Exner pressure $\pi(\mathbf{t}, \mathbf{x})$ [9] and the velocity $\mathbf{u}(\mathbf{t}, \mathbf{x})$. The coefficients are the background velocity \mathbf{U} and the speed of sound $c_s \gg \mathbf{U}$. We discretize the PDE (5.1) on a staggered grid with grid size Δx . Furthermore we discretize the advection and sound terms with different stencils \mathbf{a}_l and s_l respectively, i.e.

$$\mathbf{U} \partial_x \mathbf{u}(\mathbf{t}, x_j) \approx \frac{\mathbf{U}}{\Delta x} \sum_l \mathbf{a}_l \mathbf{u}(\mathbf{t}, x_{j+l}) \quad (5.2a)$$

$$c_s \partial_x \pi(\mathbf{t}, x_j) \approx \frac{c_s}{\Delta x} \sum_l s_l \pi(\mathbf{t}, x_{j+l}). \quad (5.2b)$$

For example the third order upwinding from Hundsdorfer et al. [22]

$$\partial_x \mathbf{u}(\mathbf{t}, x) \Big|_{x=x_j} \approx \frac{1}{6\Delta x} (-\mathbf{u}(\mathbf{t}, x_{j-2}) - 6\mathbf{u}(\mathbf{t}, x_{j-1}) + 3\mathbf{u}(\mathbf{t}, x_j) + 2\mathbf{u}(\mathbf{t}, x_{j+2})) \quad (5.3a)$$

has the coefficients $\mathbf{a}_1 = \frac{1}{3}$, $\mathbf{a}_0 = \frac{1}{2}$, $\mathbf{a}_{-1} = -1$ and $\mathbf{a}_{-2} = -\frac{1}{6}$. For the sound term we use the second order symmetric stencil

$$\partial_x \pi(\mathbf{t}, x) \Big|_{x=x_j} \approx \frac{1}{\Delta x} (\pi(\mathbf{t}, x_{j+1/2}) - \pi(\mathbf{t}, x_{j-1/2})) \quad (5.3b)$$

with the coefficients $s_{l+1/2} = -s_{l-1/2} = 1$.

On this grid we consider a one dimensional wave of the form

$$\mathbf{u}(\mathbf{t}, x) = \mathbf{u}_k(\mathbf{t}) e^{ik\pi \frac{x}{\Delta x}} \quad (5.4a)$$

$$\pi(\mathbf{t}, x) = \pi_k(\mathbf{t}) e^{ik\pi \frac{x}{\Delta x}}. \quad (5.4b)$$

with complex valued coefficients $\mathbf{u}_k(t)$ and $\pi_k(t)$ and real wave number k on the staggered mesh. Due to the discretization with the upwind and symmetric stencils we rewrite the stencils

$$\mathbf{u} \partial_x u(t, x_j) = \frac{\mathbf{u}}{\Delta x} \mathbf{u}_k(t) z_k^j \left(\frac{2z_k^3 + 3z_k^2 - z_k - 1}{6z_k^2} \right) = \frac{\mathbf{u}}{\Delta x} \mathbf{u}_k(t) \mathbf{a}(z_k) z_k^j \quad (5.5a)$$

$$c_s \partial_x u(t, x_j) = \frac{c_s}{\Delta x} \mathbf{u}_k(t) z_k^j \left(\sqrt{z_k} - \frac{1}{\sqrt{z_k}} \right) = \frac{c_s}{\Delta x} \mathbf{u}_k(t) s(z_k) z_k^j \quad (5.5b)$$

in the grid nodes $x_j = j\Delta x$ and introduce $z_k = e^{ik\pi}$. Hence it suffices to consider all wave numbers $k \in [0, 1]$. Also note the approximation of the advection and sound terms with the functions

$$\mathbf{a}(z) = \frac{2z^3 + 3z^2 - z - 1}{6z^2} \quad (5.6a)$$

$$s(z) = \sqrt{z} - \frac{1}{\sqrt{z}}. \quad (5.6b)$$

Hence we end up with the stability model problem

$$\dot{\mathbf{u}}_k(t) + \frac{\mathbf{u}}{\Delta x} \mathbf{a}(z_k) \mathbf{u}_k(t) = - \frac{c_s}{\Delta x} s(z_k) \pi_k(t) \quad (5.7a)$$

$$\dot{\pi}_k(t) + \frac{\mathbf{u}}{\Delta x} \mathbf{a}(z_k) \pi_k(t) = - \frac{c_s}{\Delta x} s(z_k) \mathbf{u}_k(t). \quad (5.7b)$$

This is a homogeneous ODE system with constant coefficients. Thus we rewrite (5.7)

$$\begin{pmatrix} \dot{\mathbf{u}}_k \\ \dot{\pi}_k \end{pmatrix} = \mathbf{A}(z_k) \begin{pmatrix} \mathbf{u}_k \\ \pi_k \end{pmatrix} \quad (5.8a)$$

with the coefficient matrix

$$\mathbf{A}(z_k) = - \frac{1}{\Delta x} \begin{pmatrix} \mathbf{u} \mathbf{a}(z_k) & c_s s(z_k) \\ c_s s(z_k) & \mathbf{u} \mathbf{a}(z_k) \end{pmatrix}. \quad (5.8b)$$

In turn we additive split the matrix in slow and fast parts

$$= - \frac{\mathbf{u}}{\Delta x} \mathbf{a}(z_k) \mathbf{I}_2 - \frac{c_s}{\Delta x} s(z_k) \mathbf{J}_2 \quad (5.8c)$$

and define the matrix

$$J_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \quad (5.8d)$$

Note the vanishing diagonal entries in the fast part. The coefficient matrix $A(z_k)$ is symmetric with two distinct eigenvalues $\lambda_{1/2} = -\frac{1}{\Delta x}(\mathbf{U}\mathbf{a}(z_k) \pm c_s s(z_k))$. This eigenvalue splitting serves for the scalar stability model problem, i.e.

$$\dot{y}_{1,2} = -\underbrace{\frac{\mathbf{U}}{\Delta x} \mathbf{a}(z_k)}_{\nu} y_{1,2} \mp \underbrace{\frac{c_s}{\Delta x} s(z_k)}_{\mu} y_{1,2} \quad (5.9)$$

for every $k \in [0, 1]$. The subscripts 1 and 2 correspond to the positive and negative eigenvalue respectively.

11.5.2 Stability function

In analogy to the stability analysis of classical Runge-Kutta methods we apply the method on our model problem (5.9). For the classical fast scale integrators explicit Euler it suffices to consider the scalar model problem. Instead for partitioned methods we have to consider the system (5.8). Our starting point is the ODE system (1.1b) for $Z(\tau)$. After inserting the linear model problem (5.8), the IVP (1.1b) becomes linear with analytic solution

$$S_k := C_s c_s(z_k) \quad (5.10a)$$

$$Z(H) = \exp(S_k D \otimes J_2) \cdot Z_0 + \Phi_1(S_k D \otimes J_2) \cdot ([\gamma - C_a \mathbf{a}(z_k) \beta] \otimes I_2 \cdot Y_n - [\gamma \mathbb{1}] \otimes y_n) \quad (5.10b)$$

For simplification we introduce the CFL numbers C_a and C_s with respect to advection and speed of sound respectively, i.e.

$$C_a := h \frac{\mathbf{U}}{\Delta x} \quad (5.10c)$$

$$C_s := h \frac{c_s}{\Delta x}. \quad (5.10d)$$

The equation (5.10b) assumes an exact solution of the fast tendencies, in particular the exact evaluation of the matrix exponential $\exp(\mathbf{S}_k \mathbf{D} \otimes \mathbf{J}_2)$ and the related matrix function

$$\Phi_1(z) = z^{-1}(\exp(z) - 1).$$

The new class of MFS methods have the MIS methods as basis, but use a fast scale integrator, which has to approximate the matrix exponential and the function $\Phi_1(z)$. In other words we can start from the analytic solution and replace the matrix exponential and the matrix function Φ_1 by their approximations Φ_0^M and Φ_1^M respectively. Every stage will use a different number of fast scale steps, therefore we use the stacked notion from the order conditions $\hat{\Phi}_0^M$ and $\hat{\Phi}_1^M$ also for the matrix functions.

With this idea in mind we use the definition (1.1c) of \mathbf{Y}_n

$$\mathbf{W}_k = \mathbf{I} \otimes \mathbf{I}_2 - \hat{\Phi}_0^M(\mathbf{S}_k) \cdot \boldsymbol{\alpha} \otimes \mathbf{I}_2 - \hat{\Phi}_1^M(\mathbf{S}_k) \cdot [\boldsymbol{\gamma} - \mathbf{C}_a \mathbf{a}(z_k) \boldsymbol{\beta}] \otimes \mathbf{I}_2 \quad (5.11a)$$

$$\mathbf{V}_k = \hat{\Phi}_0^M(\mathbf{S}_k) \cdot [\mathbf{I} - \boldsymbol{\alpha}] \otimes \mathbf{I}_2 - \hat{\Phi}_1^M(\mathbf{S}_k) \cdot \boldsymbol{\gamma} \otimes \mathbf{I}_2 \quad (5.11b)$$

$$\mathbf{Y}_n = \mathbf{W}_k^{-1} \cdot \mathbf{V}_k \cdot \mathbf{1} \otimes \mathbf{y}_n \quad (5.11c)$$

The solution at the new time step t_{n+1} , i.e.

$$\mathbf{y}_{n+1} = \mathcal{R}_k \cdot \mathbf{y}_n \quad (5.12a)$$

$$\mathcal{R}_k := \mathbf{e}_{s+1}^T \otimes \mathbf{1}_2 \cdot \mathbf{W}_k^{-1} \mathbf{V}_k \cdot \mathbf{1} \otimes \mathbf{I}_2, \quad (5.12b)$$

is a linear function in \mathbf{y}_n with a complicated coefficient. These coefficients depend on the macro step size h and the fast and slow CFL numbers C_s and C_a respectively. For a shorter notation we neglect the dependency of \mathbf{W}_k on the discretization of the advective terms and the sound terms, $\mathbf{a}(z_k)$ and $\mathbf{s}(z_k)$, but denote the dependency on the wave number in the subscript. But remind that the coefficient \mathbf{V}_k depends on the discretization of the sound terms $\mathbf{s}(z_k)$ only. Now we have everything at hand for

Definition 1 (Stability). *A multirate finite step method is stable for the pair (C_s, C_a) if*

$$\mathbf{R}(C_s, C_a) \leq 1 \quad (5.13a)$$

$$\mathbf{R}(C_s, C_a) := \max_{k \in [0,1]} \mathbf{R}_k(C_s, C_a) \quad (5.13b)$$

$$\mathbf{R}_k(C_s, C_a) := \|\mathcal{R}_k\| \quad (5.13c)$$

With the definition of \mathbf{W}_k and \mathbf{V}_k it suffices to find the function Φ_0^M for the fast scale integrator M . The MIS methods use (theoretically) the exact solution. Thus we have

$\Phi_0^{\text{MIS}}(\mathbf{S}_k) = \exp(\mathbf{S}_k)$ and an approximation of the matrix exponential for all other fast scale integrators. This approximation will be obtained by applying the integrator on the model problem (5.8) with vanishing slow part and arbitrary initial state \mathbf{Z}_n .

Explicit Euler as fast scale integrator

The Explicit Euler method is a classical method with the simple relation

$$\mathbf{Z}_{l_{ii}+1} = [\mathbf{I}_2 - \mathbf{C}_s s(z_k) \mathbf{D}_{ii} \mathbf{J}_2] \mathbf{Z}_{l_{ii}}, \quad (5.14a)$$

for one step. We concatenate all stages into one vector and write the stacked function

$$\hat{\Phi}_0^{\text{EE}}(\mathbf{S}_k) = (\mathbf{I} \otimes \mathbf{I}_2 - \mathbf{S}_k [\hat{\mathbf{L}}^{-1} \mathbf{D}] \otimes \mathbf{J}_2)^{\hat{\mathbf{L}}}. \quad (5.14b)$$

Forward-Backward Euler as fast scale integrator

The Forward-Backward Euler (2.3b) does a forward step for the first component and an implicit step in the second component. Hence we make use of the partitioning matrix

$$\mathbf{P} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

to select the first component, or the second component with \mathbf{P}^T . When we apply this scheme to the fast scale model problem with initial value $\mathbf{Z}_{l_{ii}}$ and macro step size h , we obtain

$$\mathbf{Z}_{l_{ii}+1} = \underbrace{(\mathbf{I}_2 + \mathbf{S}_k \mathbf{D}_{ii} \mathbf{P}^T)^{-1} (\mathbf{I}_2 - \mathbf{S}_k \mathbf{D}_{ii} \mathbf{P})}_{\kappa(\mathbf{S}_k, \mathbf{P})} \mathbf{Z}_{l_{ii}} \quad (5.15a)$$

for the first step. In turn after \mathfrak{L}_{ii} steps and starting from the initial value $\mathbf{Z}_{0_{ii}}$, we have

$$\mathbf{Z}_{\mathfrak{L}_{ii}} = \kappa \left(\frac{\mathbf{S}_k}{\mathfrak{L}_{ii}}, \mathbf{P} \right)^{\mathfrak{L}_{ii}} \mathbf{Z}_{0_{ii}} \quad (5.15b)$$

and the corresponding stacked functions are

$$\hat{\kappa}(\mathbf{S}_k, \mathbf{P}) = \mathbf{I} \otimes \mathbf{I}_2 - \mathbf{S}_k [\mathfrak{L}^{-1} \mathbf{D}] \otimes \mathbf{J}_2 + \mathbf{S}_k^2 [\mathfrak{L}^{-2} \mathbf{D}^2] \otimes [\mathbf{P}^T \mathbf{P}] \quad (5.15c)$$

$$\hat{\Phi}_0^{\text{FB}}(\mathbf{S}_k) = \hat{\kappa}(\mathbf{S}_k, \mathbf{P})^{\mathfrak{L}}. \quad (5.15d)$$

Störmer-Verlet as fast scale integrator

The Störmer-Verlet method can be written as two halve steps of the Forward-Backward Euler scheme with exchanging roles of components. Therefore the single step

$$Z_{l_{ii}+1} = \kappa\left(\frac{1}{2}S_k, P\right) \cdot \kappa\left(\frac{1}{2}S_k, P^T\right) Z_{l_{ii}}$$

is given by the product of two κ functions from equation (5.15a), where the two factors use both the halved fast scale factor S_k but use different partition matrices. In analogy to the Forward-Backward Euler method, we obtain after \mathfrak{L}_{ii} steps the relation

$$Z_{\mathfrak{L}_{ii}} = \left[\kappa\left(\frac{S_k}{\mathfrak{L}_{ii}}, P\right) \cdot \kappa\left(\frac{S_k}{\mathfrak{L}_{ii}}, P^T\right) \right]^{\frac{\mathfrak{L}_{ii}}{2}} Z_{0_{ii}}$$

and we remind, that we merged the even and odd steps in the matrix \mathfrak{L} . Now we have to stack all the stages together. Hence we multiply out

$$\begin{aligned} \kappa\left(\frac{S_k}{\mathfrak{L}_{ii}}, P\right) \cdot \kappa\left(\frac{S_k}{\mathfrak{L}_{ii}}, P^T\right) &= \left(I_2 - \frac{S_k}{\mathfrak{L}_{ii}} D_{ii} J_2 + \frac{S_k^2}{\mathfrak{L}_{ii}^2} D_{ii}^2 P^T P \right) \cdot \left(I_2 - \frac{S_k}{\mathfrak{L}_{ii}} D_{ii} J_2 + \frac{S_k^2}{\mathfrak{L}_{ii}^2} D_{ii}^2 P P^T \right) \\ &= \left(2 \left(\frac{S_k D_{ii}}{\mathfrak{L}_{ii}} \right)^2 + 1 \right) I_2 - 2 \frac{S_k D_{ii}}{\mathfrak{L}_{ii}} J_2 - 2 \left(\frac{S_k D_{ii}}{\mathfrak{L}_{ii}} \right)^3 P^T \end{aligned}$$

and put the blocks together into

$$\Phi_0^{SV}(S_k) = \left[\left(2S_k^2 (\mathfrak{L}^{-1}D)^2 + I \right) \otimes I_2 - 2S_k \mathfrak{L}^{-1}D \otimes J_2 - 2S_k^3 (\mathfrak{L}^{-1}D)^3 \otimes P^T \right]^{\mathfrak{L}}. \quad (5.16)$$

The stability triangle

The stability function $R(C_a, C_s)$ depends on the two real parameters, namely the advective and speed of sound CFL numbers C_a and C_s respectively. We are interested in methods which are stable for wind speeds below $\frac{1}{6}$ of the speed of sound. In terms of the stability function, we construct a triangle by the intersection of the stability area boundary with the two lines $C_a = 0$ and $C_a = \frac{1}{6}C_s$, see Figure 2. The position C_s^6 is the first intersection of the line $C_s = 6C_a$ with the stability area boundary. In analogy the value C_s^0 is the first intersection of the line $C_a = 0$ with the boundary. In equations these definitions read

$$C_s^6 = \min \left\{ C_s > 0 \mid R\left(C_s, \frac{1}{6}C_s\right) = 1 \right\} \quad (5.17a)$$

$$C_s^0 = \min \{ C_s > 0 \mid R(C_s, 0) = 1 \}. \quad (5.17b)$$

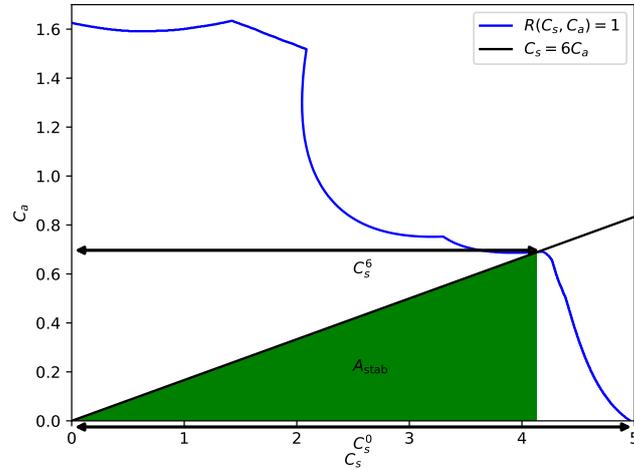


Figure 2: The blue line represents the contour of the stability area for a MFS_{SV} method. The green area depicts the stability triangle and the area A_{stab} . In particular we show the definition of the two meaning of the values C_s^0 and C_s^6 on the contour line.

For the definition of the stability triangle A_{stab} we define the value

$$C_s^* := \min(C_s^0, C_s^6). \quad (5.17c)$$

The stability triangle A_{stab} is defined by the origin and the two points $(C_s^*, 0)$ and $(C_s^*, \frac{1}{6}C_s^*)$, see the dark green area in the Figure 2. This triangle is completely defined by the value C_s^* and it has the size

$$|A_{stab}| = \frac{1}{12} C_s^*. \quad (5.18)$$

Let us review the stability function (5.13c). The advective CFL number C_a enters the stability function only in the matrix W_k , whereas the CFL number with respect to the speed of sound C_s occurs only indirectly through the matrix functions $\hat{\Phi}_0$ and $\hat{\Phi}_1$. These matrix functions not only have a block structure, but in fact are block diagonal matrices. Therefore the computation of the powers of $(s+1) \times 2$ matrices reduces to $s+1$ powers of 2×2 matrices.

Afterwards these block diagonal matrices has to be combined in to the matrices W_k and V_k . From a computational (and implementation) point of view it is more efficient, and easier, to apply the method on the matrix valued test problem once with initial value $(1, 0)$ and once with $(0, 1)$. The results form the columns of the amplification matrix \mathcal{R}_k .

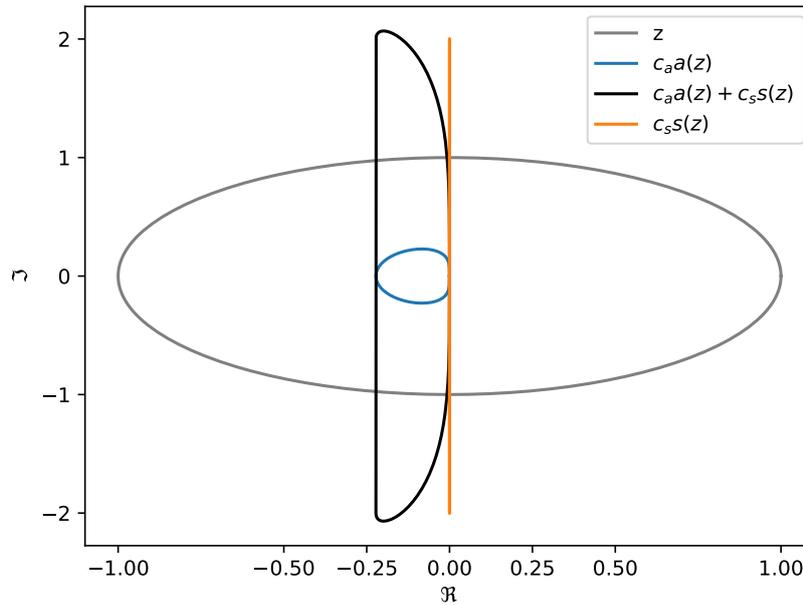


Figure 3: The gray line represents the unit circle and the blue and orange lines the stencils $a(z)$ and $s(z)$ respectively. For a interpretation as (5.1) we use $U = 1$ and $c_s = 6c_a$ in the black line. We clearly see that the explicit Euler method for the whole system and in particular for the fast system is never stable.

The explicit Euler method allows a further simplification. In equation (5.9) we gave the eigenvalues of the coefficient matrix. The stability function for the explicit Euler integrator reduces therefore two scalar problems with the aforementioned coefficients.

Our aim are methods with large stability areas. Clearly, this area depends highly on the model problem. The aforementioned advection problem is of hyperbolic type and has therefore no damping. Furthermore the second order central stencil leads to only pure imaginary eigenvalues as we see in Figure 3. One way to stabilize the scheme is the introduction of a diffusion term with relatively small coefficient, see [40]. Hence the 1D-advection 5.1 changes to

$$\begin{aligned}\partial_t u + U \partial_x u &= -c_s \partial_x \pi + \nu \partial_x^2 u \\ \partial_t \pi + U \partial_x \pi &= -c_s \partial_x u\end{aligned}$$

with a small time step (and grid width) dependent coefficient $\nu = \alpha \frac{(\Delta x)^2}{\tau}$. Common values for $\alpha \in \{0.025, 0.05\}$. Whereas the damping increases the stability area, it also leads an additional error compared to the original problem without damping.

Integrator	class	MIS	MFS	total	$\frac{s}{2}(3s-1)$		s_{\min}
					3	4	
EE	4	5	7	16	12	22	4
FB	4	5	10	19	12	22	4
SV	4	5	2	11	12	22	3

Table 6: Number of order conditions for order three and the fast scale integrators explicit Euler (EE), forward-backward Euler (FB) and Störmer-Verlet (SV). We excluded the compatibility condition 3.9b. The last three columns show the number of real parameters and the deduced the minimal number of stages s_{\min} .

II.6 METHOD CONSTRUCTION

In the previous section we derived the order conditions and defined the stability of the MFS methods. The order conditions consist of three categories, namely the classical (underlying) order conditions, the MIS conditions and the new finite step conditions. Whereas the first two conditions are the same for every fast scale integrator, the last set depends on the integrator in size and structure, see Table 6. The first order MIS condition (3.9b) is trivially fulfilled by choosing

$$D_{ii} = \sum_j \beta_{ij}. \quad (6.1)$$

The remaining conditions must be fulfilled by the real parameters \hat{A}_{ij} , α_{ij} , γ_{ij} and the rational parameters \hat{L}_{ii} . The matrices \hat{A} , α and γ are strictly lower triangular and we set the first column to zero. Hence we have $\frac{s}{2}(3s-1)$ real parameters plus the s rational parameters \hat{L} . The last column in Table 6 shows the smallest number of stages such that the number of parameters is larger than the number of conditions. There are enough parameters with three stages only for the second order fast scale integrator. The two first order integrators require at least four stages.

Our aim are MFS methods with a large stability triangle as defined in the previous section. But the stability area and the size highly depends on the method parameters α , β and γ and the numbers of fast steps in every stage. For a numerical integrator this area also depends on the step factor L , which does not influence the order conditions. We are heading for methods with a large stability area. In practice it is hard to optimize the whole stability contour line. Furthermore the strongest step size bound stems from the sound terms. Therefore we relax to find the coefficients for the largest stability triangle area as defined in (5.18).

The order conditions are nonlinear, but at least polynomial in the coefficients α , γ and β . The number of parameters and the number of conditions is of intermediate size. Nevertheless it was not possible to find symbolic solutions due to high memory usage of the software package Maple©[7]. Therefore we use a three stage numerical process to find optimal parameters. In details our strategy consists of the following steps:

1. First we write the order conditions as a minimization problem. We keep the step factors \hat{L}_{ii} fixed in the minimization. In other words we try to find an optimal method for a given set of step ratios. The cost functional is

$$W(\mathbf{p}) = \frac{1}{2} \sum_{i=1}^{\#\text{cond}} \text{cond}_i(\mathbf{p})^2 \quad (6.2)$$

and the parameters \mathbf{p} are the real coefficients α , γ , β . The function cond_i represents the error in the order condition i . For example the classical condition $\mathbf{b} \cdot \mathbf{1} = 1$ is represented by $\text{cond}_1(\mathbf{p}) = \mathbf{b} \cdot \mathbf{1} - 1$. We constrain the parameters to a useful range $\mathbf{p} \in [-10^5, 10^5]$. We start this first stage with a suitable MIS method from Knoth and Wensch [26], depending on the number of stages. These methods fulfill initially the classical and the MIS conditions and have a sufficiently large stability area for exact integration.

2. Construct the set

$$\mathcal{L}^{\text{initial}} := \left\{ \hat{L} \left| \frac{1}{2} \sum_i |\text{cond}_i|^2 \leq 10^{-15} \ \& \ A_{\text{Stab}} \geq 0.3 \right. \right\}, \quad (6.3)$$

of all step factor combinations, which fulfill the order conditions numerically and have a sufficient large stability area as defined in (5.18).

3. Optimize the method parameters α , β , γ for every $\hat{L} \in \mathcal{L}^{\text{initial}}$ by minimizing the negative stability area, i.e.

$$-A_{\text{stab}} \rightarrow \min \quad (6.4a)$$

constrained by the order conditions

$$|\text{condit}_i| \leq 10^{-13} \quad i \in \text{classical} \quad (6.4b)$$

$$|\text{condit}_i| \leq 10^{-10} \quad i \in \text{other}. \quad (6.4c)$$

We set a sharper bound on the classical order conditions because numerical experiments showed, that these seem to influence the accuracy most drastic. From these solutions, we form the set

$$\mathcal{L}^{\text{opti}} := \left\{ \hat{\mathbf{L}} \left| \frac{1}{2} \sum_{\mathbf{i}} |\text{condit}_{\mathbf{i}}|^2 \leq 10^{-15} \ \& \ \mathbf{A} \geq 0.75 \right. \right\}, \quad (6.5)$$

which selects only these solutions with a larger stability area then the reference method.

We solve the order conditions only numerically using the interior point optimization. Hence the solution depends severely on the initial values. With the selection of a sufficiently optimized MIS method as initial values, we can expect better stability areas later. Furthermore the initial values also fulfill already the classical and MIS order conditions. Our selection criteria in the second step ensures two properties. First, we select only parameters, which fulfill all order conditions up to $\sqrt{\frac{2}{N}}10^{-7}$ where N is the number of order conditions. The second criterion ensures a not too small stability area compared to the reference method RK3.

Let us first consider methods of order two. Most of the MFS conditions arise from the step from order two to three. So for order two we have to consider only two classical, two MIS conditions and one MFS condition.

II.7 STABILITY OPTIMIZATION FOR METHODS OF ORDER TWO

The previous sections introduced the MFS methods, their order conditions and derived the stability function on top of the hyperbolic advection problem. We have seen that the methods for order three require many complicated conditions. Additionally the order condition depend on the real valued coefficients and integer valued step factors. In case of the numerical fast scale integrators, the stability function also depends on the number of steps, whereas the use of the exact solution does not has this dependency.

The derivation also showed, that the order conditions for order two are much simpler. In this case we solve the order conditions analytically. This analytic solution allows us to concentrate on the stability function. We derive two hypothesis for parameters with large stability areas and check them numerically.

II.7.1 MFS2 methods

In this section we construct the methods of order two with two stages. Hence we parametrize the classical parameters \hat{A} with the node c_2 and solve the classic order conditions with

$$\hat{A} = \begin{pmatrix} 0 & 0 & 0 \\ c_2 & 0 & 0 \\ \frac{2c_2-1}{2c_2} & \frac{1}{2c_2} & 0 \end{pmatrix}. \quad (7.1)$$

The matrices γ and α have the only two parameters γ_{32} and α_{32} respectively. At the same time there are only two remaining conditions for MIS and MFS, namely (3.25a) and (3.30a). We write the solution of both equations, i.e.

$$\alpha_{32} = \frac{c_2 - 1}{c_2((1 + \rho_3)c_2 - 1)} \quad (7.4d)$$

$$\gamma_{32} = - \frac{(c_2 - 1)^2 + c_2^2 \rho_3}{c_2(c_2(1 + \rho_3) - 2)(c_2(1 + \rho_3) - 1)}, \quad (7.4e)$$

in terms of the step ratio $\rho_3 = \frac{\hat{L}_{33}}{\hat{L}_{22}}$ and the node c_2 . This solution splits in three branches in terms of the node c_2 , i.e. $c_2 \in \left(0, \frac{1}{1+\rho_3}\right)$, $c_2 \in \left(\frac{1}{1+\rho_3}, \frac{2}{1+\rho_3}\right)$ and $c_2 \in \left(\frac{2}{1+\rho_3}, \infty\right)$. From a practical point of view we can safely reduce the last interval to $\left(\frac{2}{1+\rho_3}, \frac{9}{5}\right)$.

The fast scale integrator solves the auxiliary ODE (1.1b) in every stage on the interval $[0, h]$. By scaling the stage local time interval with D_{ii} on every stage i , we see that the effective local integration interval is hD_{ii} . Hence the first hypothesis is, that we expect the lowest error of the fast scale integrator at minimal coefficients D_{ii} . So we have to minimize the maximum of D_{22} and D_{33} . Luckily the coefficients are given by the simple equations

$$D_{22} = c_2 \quad (7.2a)$$

$$D_{33} = \frac{c_2(1 + \rho_3) - 1}{c_2(1 + \rho_3) - 2}. \quad (7.2b)$$

Both parameters are functions of the node c_2 . Whereas the parameter D_{22} is monotonous increasing, the parameter D_{33} decreases. Therefore we expect the smallest maximal amplification at the intersections $D_{22} = D_{33}$, i.e.

$$c_2^I = \frac{\rho_3 + 3 \pm \sqrt{\rho_3^2 + 2\rho_3 + 5}}{2(\rho_3 + 1)}. \quad (7.3)$$

On the other hand we allow different step sizes in every stage. So we expect different numerical errors in every stage even with the same coefficient D_{ii} on the same macro step size h . The same argument applies to the stability of the fast scale integrator in every stage. The stability of a one step method depends on the product of the step size and the norm of the Jacobian of the right hand side. The auxiliary ODE amplifies the fast scale evaluations g with the coefficient D_{ii} . Therefore as second hypothesis we consider the minimum of the maximum between D_{22} and $\frac{D_{33}}{\rho_3}$. Due to the monotonicity we find this minimum at $D_{22} = \rho_3 D_{33}$, which amounts to the node

$$c_2^F = \frac{3\rho_3 + 1 \pm \sqrt{5\rho_3^2 + 2\rho_3 + 1}}{2\rho_3(\rho_3 + 1)}. \quad (7.4)$$

11.7.2 Numerical experiments

Let us now check the two hypotheses numerically. The stability triangle is bounded by the two lines $\{(\frac{1}{6}C_s, C_s) | C_s \in \mathbb{R}\}$ and $\{(0, C_s) | C_s \in \mathbb{R}\}$, hence we analyze the hypotheses on both lines separately.

Stability on the line $C_a = 0$

Along the line $\{(C_a, C_s) | C_a = 0\}$ the slow tendencies in the auxiliary ODE (1.1b) vanish. Therefore the stability of the full MFS scheme depends only on the stability function of the underlying fast scale integrator. The plots in Figure 4 depict values C_s^0 using the fast scale integrators exact, explicit Euler, forward-backward Euler and Störmer-Verlet from top left to bottom right. Please note the semi logarithmic scale for the exact integrator, whereas all other fast scale integrators use two linear axes. We denote the optimal nodes c_2^I and c_2^F with a point and a star respectively. Clearly the exact integrator leads to the largest intersections C_s^0 and the nodes c_2^I perfectly match the maxima locations. Also note the non-smooth dependency, but this might be an effect due to the bisection method used to find the value C_s^0 . From that plot there is no unique optimal step ratio for the exact integrator. But for every ratio seems to exist a node, such that the method is stable

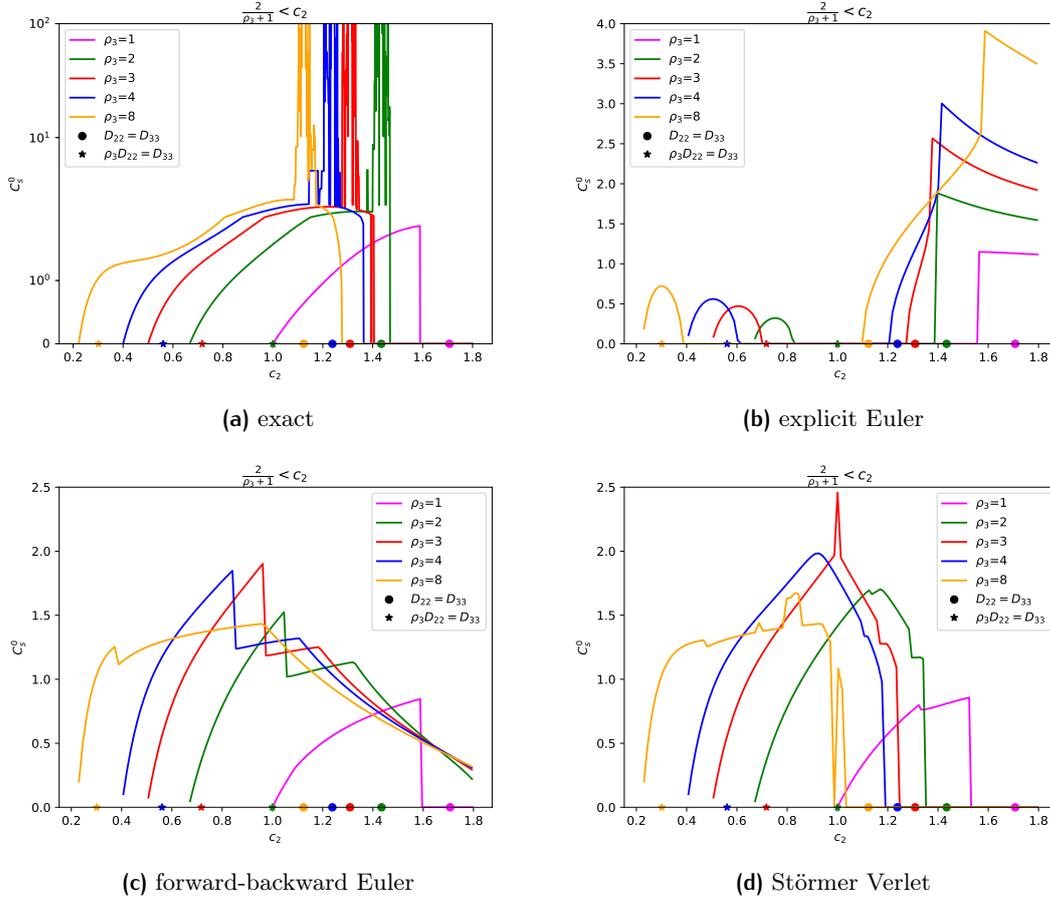
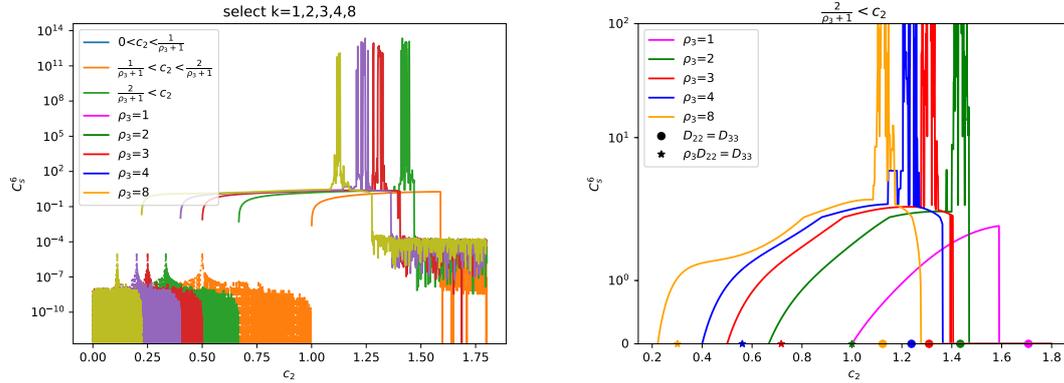


Figure 4: The values C_s^0 with respect to the node c_2 . The markers point and star denote the two hypothetical points c_2^I and c_2^F respectively. All numerical integrators used the step factor $L = 2$.

for speed of sound CFL numbers larger than 100. In contrast the numerical integrators have always by magnitudes smaller C_s^0 values. The maxima are sometimes quite sharp and next to jump. Furthermore the predicted optimal values c_2^I and c_2^F do not have any clear relation to the maximal nodes. Note that the plot 4b, which uses the explicit Euler, has an increasing intersection point with increasing step ratio ρ_3 . This indicates a high, possibly infinite step ratio for the last stage.

The more accurate, and more importantly stable, fast scale integrators FB-Euler and Störmer-Verlet are depicted in the bottom left and right plots in Figure 4 respectively. This time we have bounded intersections C_s^0 , and very surprisingly, the largest maximum for both stable methods is smaller than for the unstable explicit Euler in plot 4b. Both integrators lead to optimal nodes c_2 somewhere in the middle between the nodes c_2^I and



(a) All branches for the node c_2 with different line styles (b) Only the branch $\frac{2}{\rho_3+1} < c_2$, which leads to the largest lengths.

Figure 5: Both plots show the length C_s^6 with respect to the node c_2 for MFS2 methods with exact integration. The colors represent the factors, whereas the line styles correspond to the branches. So we easily see the small lengths for the first two branches. We highlight the third branch with the same line style in the right plot. Additionally we show the nodes c_2^I and c_2^F as dot and star respectively.

c_2^I . More interestingly the plots indicate an optimal step ratio $\rho_3 = 3$, in contrast to the explicit Euler and exact integration.

Both plots together indicate a rule for the step factor L in combination with the optimal method parameters. If we minimize the coefficients D_{ii} , then we have to choose the common factor L such that the fast scale integrator is always stable. Furthermore we can achieve a larger stability area with larger step factors. If we minimize $\frac{D_{ii}}{L_{ii}}$ we can choose a smaller factor L and use an unstable fast scale integrator. But we have a stability bound regardless of the accuracy, and stability, of the fast scale integrator.

Stability on the line $C_s = 6C_a$

The previous discussion concentrated on only one point of the stability triangle, namely the bottom intersection C_s^0 . But the length of the bottom line also depends on the intersection of the line $C_s = 6C_a$ with the stability area boundary $R(C_s, C_a) = 1$. This time there is no simple relation of the method parameters to the accuracy of the fast scale integrator. Hence we apply the previous procedure with the same step size ratios on the length of the bottom side.

Let us first check the lengths of the bottom side with exact integration. The previous discussion neglected the other branches for c_2 . This time we include the corresponding lengths for completeness in the plot 5a. The line styles dotted, dashed and straight

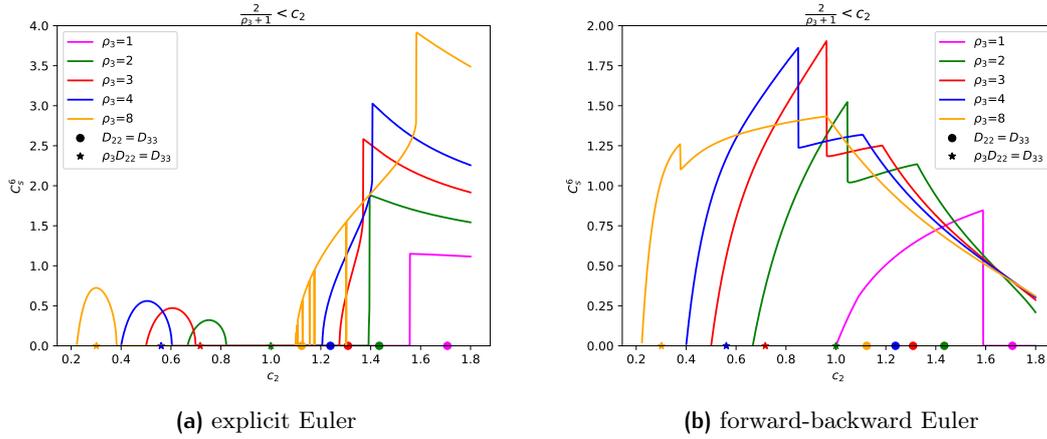


Figure 6: The length C_s^6 of the bottom line with explicit Euler (left) and forward-backward Euler (right) integration using the step factor $L = 2$. Additionally we show the nodes c_2^I and c_2^F as points and stars respectively.

correspond to the branches, whereas the colors represent the step factors ρ_3 . The dotted and dashed lines are hard distinguishable and always below 10^{-12} . In contrast the third branch with solid lines show a high dependence on the node and the step factor ρ_3 . Therefore we neglect all other nodes.

The solid lines in plot 5b correspond to the same lines in the plot 5a but with a linear C_s^0 axis this time. Every line shows clear maximal values at different heights. A second look reveals a maximal stability area for $\rho_3 = 3$ with $c_2 \approx 1.16$. The parameter ρ_3 scales the step size in the last integration step compared to the second step. So with a higher numbers of steps in the last stage we would solve the last auxiliary more accurate. It is therefore surprising to see a decreasing stability area with increasing step ratio. This behavior stems exclusively from the MFS order conditions, which connect the step ratios to the remaining coefficients. Otherwise the step ratios do not influence the complete method with exact integration. From that point of view, the Figure 5 shows the best areas one can expect from a MFS2 method.

In analogy to the C_s^0 analysis before, the plot 5b shows also the optimal nodes c_2^I and c_2^F using points and stars respectively. Again the colors correspond to the step factors ρ_3 . In contrast to the analysis for the intersection positions C_s^0 this time there is no direct relation between the optimal nodes and the maximal lengths visible.

Using the the first order numerical integrators explicit Euler and forward-backward Euler we obtain the left and right plot in Figure 6 respectively. Whereas the left plot 6a shows a good agreement of the points with some extrema, we see the stars next to the

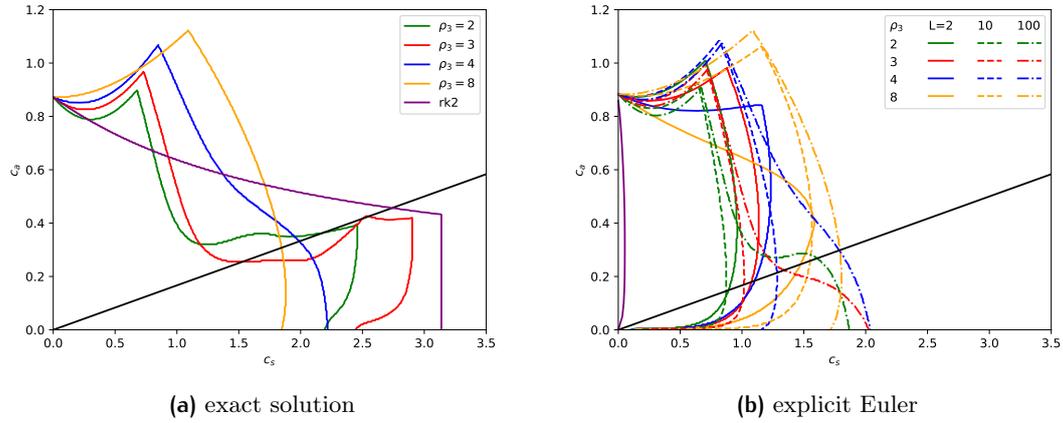


Figure 7: Stability contour lines of the MFS2 methods using the optimal values for the node c_2 from plot 5b. The black line shows the top boundary $C_s = 6C_\alpha$. The colors green, red, blue and orange, correspond to the step ratios ρ_3 . In the right plot we also add three step factors L and distinct them using the line styles 2, 10 and 100.

extrema in the right plot 6b. Hence there is again no clear relation between the claimed optimal nodes and the real behavior. Although the forward-backward Euler is stable for all nodes c_2 , we do not see a much larger stability area compared to the unstable explicit Euler method. This result is in accordance with the maximal intersection points C_s^0 .

Stability contours with optimal parameters

After studying the qualitative behavior of the stability areas through their intersection points C_s^0 and C_s^6 we proceed to their contours lines in Figure 7. The left plot 7a shows the contours using exact integration with the optimal node c_2 and the same step ratios, but neglect $\rho_3 = 1$, as in the previous discussions. Using $\rho_3 = 1$ corresponds to a method with the same number of steps in each stage, and at the same time, shows the smallest stability areas in all experiments. As reference we also add the method RK2 from Wicker and Skamarock [54], which based on the forward-backward Euler, in purple. But for comparison we replaced the forward-backward Euler by the exact solution. The straight black line represents the ratio between the speed of the winds and the speed of sounds, i.e. $C_s = 6C_\alpha$. The colors green, red blue and orange correspond to the step ratios two, three, four and eight respectively. In accordance with the previous discussion we see the smallest stability area with $\rho_3 = 8$. The base of stability triangle for $\rho_3 = 2$ is only slightly larger than for $\rho_3 = 4$ with $C_s^0 \approx 2.19$ and $C_s^6 \approx 1.99$.

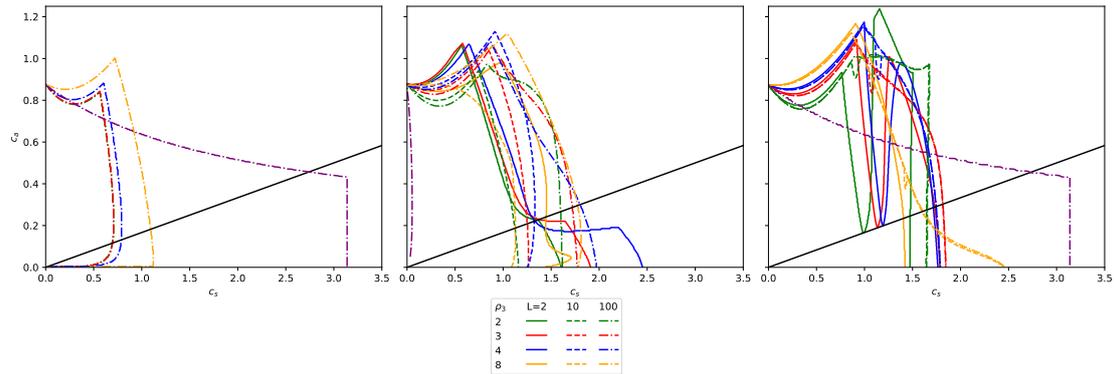


Figure 8: Stability area of the MFS2 methods using the optimal values for the node c_s from the left Figure 6 with exact integration (left) and explicit Euler (center) and forward-backward Euler (right). In the centered Figure we interpreted the method RK2 as a MIS method and used the explicit Euler method with factor $L = 100$. The left one uses the exact solution for the linear problem.

Let us have look at the right Figure 7. These line correspond to the stability area contour using the same (optimal) node c_2 and the same step ratios ρ_3 as in the left Figure. But this time we use the explicit Euler method as fast scale integrator. Therefore the solutions depend also on the step factor L . The different step factors two, 10 and 100 correspond to the line styles straight, dashed and dotted respectively. For reference we use again the RK2 method, but now with the explicit Euler with $L = 100$ as the fast scale integrator. On the first view, the stability areas are sufficiently smaller than with the exact solution. Even with 100 steps the contour is far away from the optimal solution in the left Figure. From this result we can conclude that the optimization with the exact solutions leads to methods, which are very sensitive to the number of steps. Even in the case of the MFS conditions, which already take the step ratios into account.

But what happens if we optimize the method parameters the other way around? This strategy corresponds to the optimal nodes in plot 6a. Now we can also use the numerical method, which should lead to the largest areas, or again the exact solution as a measure for the largest possible stability area with these parameters. In Figure 8 we depict the contour lines for this case. Again the left plot uses the exact solution and the centered one the explicit Euler method as fast scale integrator. Additionally we use the forward-backward Euler method for the right plot. The same applies also to the reference base method RK2. This time the stability areas using the exact integration are far smaller and the largest area is not the original one with ratio $\rho_3 = 3$, but the previously smallest one with $\rho_3 = 8$.

The right plot then shows the stability areas using forward-backward Euler, which belong to the optimization using $L = 2$. First we concentrate on the straight lines. The green, red and blue intersect together with the black line near $C_s \approx 1.33$. Only the orange line has the intersection position near $C_s \approx 1.45$ and leads therefore to the largest stability triangle. A close look also shows that the intersections at $C_a = 0$ are larger for the red and blue lines corresponding to $\rho_3 = 3$ and $\rho_3 = 4$. Do not get confused by the first Figures, showing that $\rho_3 = 8$ has the largest intersection at $C_a = 0$. They were optimal at again different nodes \mathbf{c}_2 .

This confusion alone already shows the discrepancy in different heuristics for optimal MFS (and MIS) methods. Whereas the exact integration strikes out the fast scale step size, it leads in general to smaller stability areas with the explicit Euler method. Or in other words the corresponding methods require many steps to obtain a sufficiently large stability area. But at least the behavior is somewhat predictable in the sense that more steps lead to a larger area. On the other hand, the optimization with the numerical method at hand leads in general to methods with useful stability areas for exactly this one step size. But then there is no clear behavior with respect to the step factor L . Increasing the step factor, hence use more steps, resolve the fast scale more exactly, might lead to a less stable method.

11.7.3 Conclusions

The previous section discussed the parameter dependency for the stability areas. We solved the MFS conditions analytically and obtained a one-parameter family of solutions. But even in that simplified case leads to a complicated dependency of the stability area on the free parameter. Due to this complicated dependency, we checked two heuristic hypotheses. The numerical experiments then showed, that these heuristics predict the parameter next to the optimal node for the exact integrator. In case of the numerical ones, the predictions are worse. From an optimization point of view, it would be valuable to find good heuristics, which predict good parameters and can be extended to more stages.

III

NUMERICAL EXPERIMENTS

This chapter presents the numerical experiments. First we construct MFS_{EE} , MFS_{FB} and MFS_{SV} methods of order three and optimize their stability area according the procedure in section II.6. All optimizations were done with the interior point optimization implemented in the software package IPOPT by Wächter and Biegler [48] in version 3.11.9. As described in the section II.6 we maximize the stability triangle size subject to the order conditions. To speed up this process, we generate an optimized code for the order conditions and their derivatives using Maple©[7].

The benchmark problems are ordered from simple to more advanced. Hence the first benchmark is the advection problem II.5.1 with an initial pulse. This problem is linear and serves for a rough hint on the obtainable order and stability. The last two benchmark problems are nonlinear in the slow tendencies. These benchmark problems therefore serve as hints for the applicability on more advanced real world models of the atmosphere.

III.1 METHOD CONSTRUCTION WITH LINEAR PROBLEMS

III.1.1 First order fast scale integrators

We consider the first scale integrators explicit Euler and forward-backward Euler. Whereas the first method leads to 16 order conditions, the second requires already 19 conditions. Hence we will consider only methods with $s = 4$ stages in this paragraph. Keep in mind that we already fulfill the compatibility condition II.3.9b by construction. As described in the section II.6, we use a three stage process. Our initial MIS method is MIS4 from Table 4 in Knoth and Wensch [26]. Our analysis on second order methods in section II.7 already showed an unstructured dependency of the stability on the step factors \hat{L}_{ii} . Hence we study all methods with the step factors $1 \leq \hat{L}_{ii} \leq 10$.

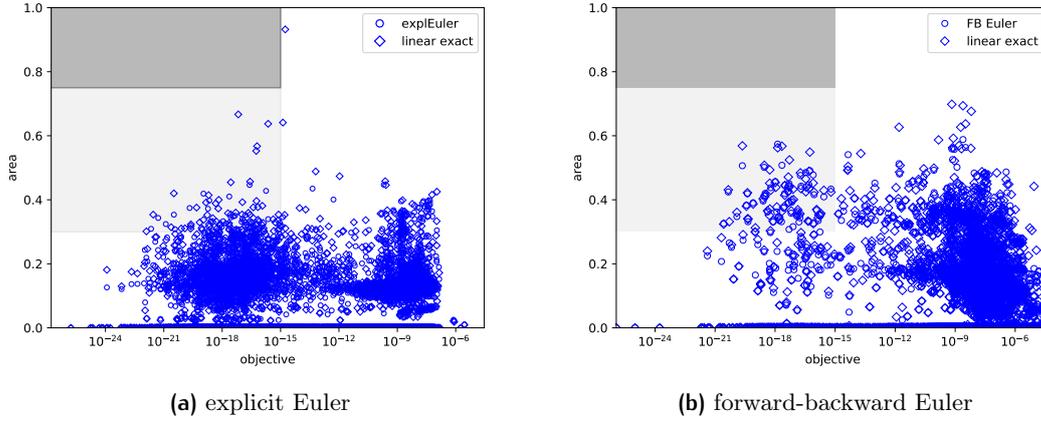


Figure 1: The stability triangle area with respect to order condition error after step one. The left and right plot correspond to the order conditions MFS_{EE} and MFS_{FB} respectively. Both Figures mark the numerical integrator with a circle and the exact one with a diamond. The numerical solvers used the common step factor $L = 2$. Every point corresponds to a step size ratio. Points in the bright and dark shaded area fulfill the selection criteria for the sets $\mathcal{L}^{\text{initial}}$ and $\mathcal{L}^{\text{opti}}$ respectively.

Optimal parameters with first order integrators, stage I

Let us first consider the conditions MFS_{EE} . Then we obtain the objectives and areas in the left plot 1a. We distinct the explicit Euler with $L = 2$ and the (linear) exact solver with the circles and diamonds respectively. All points in the bright shaded rectangle correspond to the set $\mathcal{L}^{\text{initial}}$. The stability triangle area is strongly influenced by the accuracy, and stability, of the fast scale integrator. Whereas the diamond markers even reach an area of 0.9, the circles seldom approach an area of 0.4. But we aim at a numerical method with inexact integration, hence the exact integration serves more as a hint for the largest possible areas.

Furthermore we also recognize that there are many step ratios, which lead to a large minimal sum (II.6.2). This indicates that there are many fast scale step ratios which do not satisfy the whole set of order conditions. In particular we did not find a solution for the ratio $\hat{L}_{ii} = 1$ for $i = 1, \dots, s + 1$. Hence we have to use different numbers of steps in every stage.

The right plot 1b depicts the areas and objectives again, but this time for the order conditions MFS_{FB} . Once again there are many markers to the right of the shaded area. This indicates that there are much more step ratios, for which the order conditions do not posses a solution. Similar to the left Figure using the explicit Euler, obtain larger stability areas with the exact integrator. But this time the differences are not that pronounced.

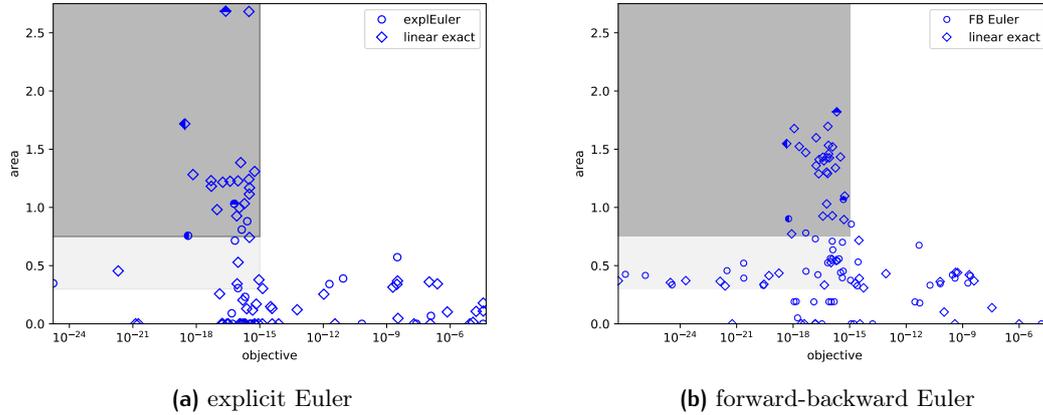


Figure 2: The stability triangle area with respect to order condition error after optimization step three. The markers and axis correspond to the Figure 1. We also see, that many optimization procedures did not converge or even left the constraint set. As a helper for the eye we colored four points, which refer to selected best methods. The top filled marker leads to the highest area, whereas the left filled one corresponds to the lowest condition error.

The figures also differ from a qualitative view. Whereas in the left, most markers are below the shaded areas, in the right we see most markers in the right half of the plot. Translated to the axis, this indicates a strong requirement at least for a stable fast scale integrator. But the stable method leads to a higher number of (and more complicated) order conditions. In turn, we obtained slightly larger stability triangles. But have in most times there is no good solution to the order conditions with small residuals. But these are only rough initial parameters.

Optimal parameters with first order integrators, stage II

Hence we proceed to optimize this parameter subset with respect to the stability triangle area. We minimize the negative stability triangle area and constrain the parameters by the order conditions. After the optimization procedure, we consider only these parameters, which have a larger stability area as the reference method RK3, but fulfill the order conditions at least as good as the initial parameters. The left Figure 2a depicts these results for the MFS_{EE} methods. Again we computed the stability areas with the numerical solver (circles) and exact solutions to the underlying linear fast scale problem (diamonds). This time, four step sequences remain in the numerical case and 18 when using the exact solution. For the remaining step sequences, the optimizer left the constraint set, or failed to optimize the area. In most bad cases we see both effects together.

Φ	MFS _{EE}				MFS _{FB}			
exact	(1,2,1,10)	◆	(3,1,7,2)	◆	(8,1,10,8)	◆	(7,2,3,8)	◆
numerical	(5,4,1,8)	●	(3,3,1,10)	●	(2,7,2,10)	●	(6,2,3,10)	●

Table 1: Step factors for the optimal methods derived from the Figures 2.

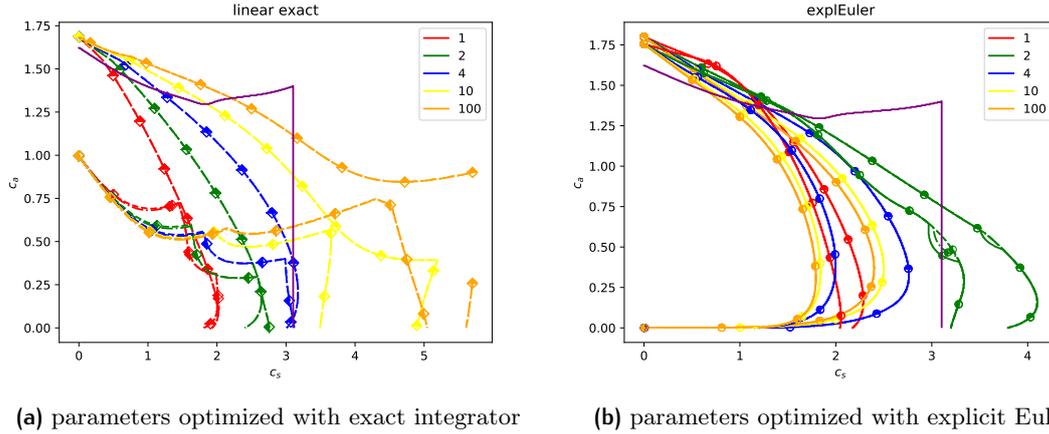


Figure 3: Contours of the stability areas of the MFS_{EE} method with the step sequences from Table 1 after optimization with exact and numerical integration respectively. The colors correspond to the step factors L , whereas the markers correspond to the plot 2a. We see, that this optimization strategy leads to methods with large stability areas for many fast steps. But for a small factor, the stability area is ways too small.

After the optimization we select the best methods. Ideally the best method would be found in the upper left corner in both Figures 2. In reality we select and mark two special points from each Figure. We obtain the first marker by selecting all points at the highest area and select the one with the smallest condition error. This marker is then fill in the upper halve, because it is on top of the other solutions. For the second one, we turn around this process and select the solutions with lowest condition errors first and then these with highest area. This marker will be halve filled in the right. We have two integrators types and two order conditions, hence we repeat that process twice in the left and twice in the plot 2b. That procedure leads to the eight methods, four MFS_{EE} and four MFS_{FB}, with the optimal step factors given in the Table 1.

Step factor dependency of optimized MFS_{EE} methods

Let us inspect the dependence of the stability areas on the step factor L in the Figure 3 for the order conditions MFS_{EE}. The markers correspond to the left plot 2a. In the left plot

3a we consider the parameters obtained from exact integration and in the right plot 3b the parameters from numerical integration with the explicit Euler with step factor $L = 2$. Each color corresponds to a step factor from in set $[1, 2, 4, 10, 100]$. The factors 1, 2 and 4 are useful for a fast and efficient method, whereas the factor $L = 100$ resembles nearly exact integration. In the left plot 3a we see an increasing stability triangle area from $L = 1$ to $L = 100$. This observation is in concordance with the convergence to the exact solution. It also shows, that we require step factors above $L = 10$ to obtain a sufficiently large stability area. From that point of view, these methods require ways too many fast scale evaluations and get computational too expensive.

The right plot 3b looks quite different. The intersections C_s^0 , defined by II.5.17a increase only from $L = 1$ to $L = 2$, but decrease for larger factors. Hence we see more or less a convergence to small stability area. This behavior stems from the optimization with $L = 2$. We take the numerical error, and in this case also the instability, of the fast scale integrator into account and minimize it. If we now increase the numbers of steps, we change the errors in every stage, and the complete method gets unstable.

Please note that we obtained a larger area compared to the reference method with both approaches. But these experiments merely indicate, that even an unstable fast scale integrator can lead to a MFS, and MIS, method with a sufficiently large stability area and low number of fast scale evaluations.

Step factor dependency of optimized MFS_{FB} methods

Now we consider the forward-backward Euler method and the corresponding MFS order conditions MFS_{FB}. Hence we select the four halve filled markers in the right plot 2b. The corresponding step factors \hat{L} are given in the last two columns of Table 1. Again we plot the contours of the stability areas from the exact and the numerical integration in the left and right Figure 4 respectively. Let us concentrate on the left plot 4a first.

This time the contour lines have many wiggles for small step factors $L < 10$. Hence, we hardly see the increasing values for C_s^0 with respect to L . This result is more visible in Figure 5. For many steps the node C_s^0 , i.e. the stability boundary at $C_a = 0$, gets larger, but the area remains bounded by the intersection on the line $C_s = 6C_a$. Furthermore we also see the shift between the exact and the numerically optimized parameters. Whereas the fast exact integration might lead to larger nodes C_s^6 , it also requires the larger step factors. In contrast the optimized parameters with the numerical integrator lead to similar nodes at smaller factors.

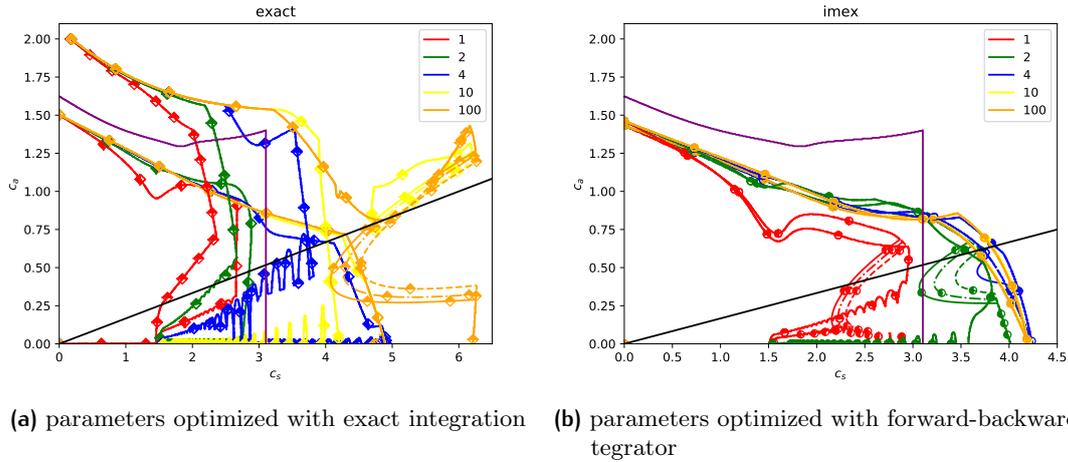


Figure 4: Stability areas of a MFS_{FB} method with the step sequences after optimization with exact and numerical integration respectively. The colors correspond to the step factors L , whereas the markers correspond to the markers in the plot 2b. We see, that the optimization strategy with the exact integrator leads to methods with large stability areas for many fast steps. In contrast to the explicit Euler experiment, the optimization with the numerical integrator and fixed step factor led to better methods than the use of the exact integrator.

In contrast to the explicit Euler, we see the straight lines for every step factor. This implies, that these methods are stable and they damp some waves inside the stability area. The other line styles, corresponding to contours slightly above one, are seldom visible. Hence these bounds are quite sharp. The only exception is the case with 100 steps, where the contour lines for slightly larger stability function get visible. That indicates a slower increasing stability function outside the stable area compared to the other step factors.

A deeper look also shows a difference between the selected step sequences by comparing the top filled with the left filled markers. Whereas the left ones are always below $c_a = 1.5$, and therefore below the violet contour lines, the top filled markers are above the reference contour lines and reach even $c_a = 2.0$. These top filled markers correspond to the solutions with best (largest) stability triangle area, but often with a larger error in the order conditions. Hence we can assume, that the optimization with respect to the area also increases the stability along the line $c_s = 0$. But on the other hand we might expect larger error constants. In the context of the discretized advection equation, the larger stability with respect to c_a means that we can, for slow winds, also reduce the step size restriction with respect to the slower transport.

This Figure also reveals a draw back of the optimization strategy, or in more details, of the stability triangle definition using the two points. The orange line with the top filled

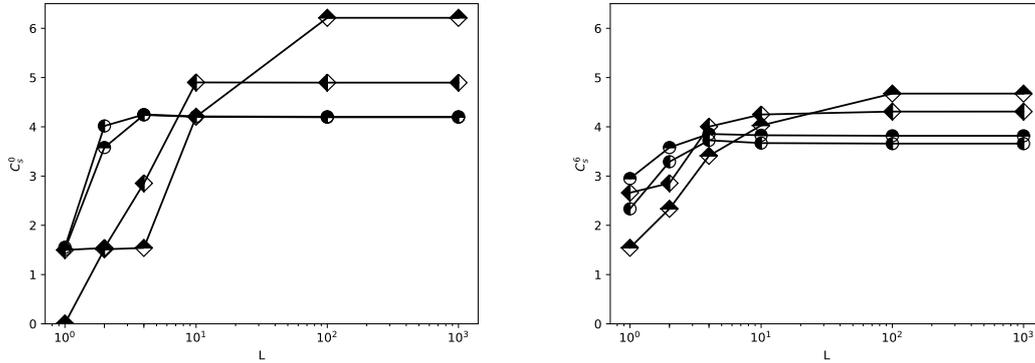


Figure 5: The position C_s^0 and C_s^6 using the four different optimal step sequences for MFS_{FB} with respect to the step factor L .

markers would have the intersection points $C_s^6 \approx 6.2$ and $C_s^0 \approx 4.8$. But the stability triangle ends near the bend at $c_s \approx 4$.

Let us consider the right plot in Figure 4, where we optimized with the numerical integrator with step factor $L = 2$. This time, we also see a convergence with respect to L , which is in contrast to the experiment with the explicit Euler before. But once again, we get hit by the wiggles near $c_a = 0$, hence the stability triangle for the top filled markers would be smaller than the reference triangle. But the contours with left filled markers, representing a smaller condition error, lead to a similar area and we can hope for slightly lower errors with the corresponding method and a step factor $L \geq 2$.

Accuracy analysis - linear problem

Now we select the MFS methods with the best stability areas for step factors below 10, i.e. the top filled circle for MFS_{EE} and the left filled circle for MFS_{FB} . We apply these methods on the space discretized linear advection problem II.5.1 with a fixed mesh and initial value. Due to the linearity we compute the exact solution on this mesh at the end time using the matrix exponential function [1]. This solution serves as the reference for the time error. We fix the step sequences and vary the macro step size h and the step factors L . This problem is the base for the stability analysis. Hence we check with these experiments not only the numerical error in maximum norm, but also the stability. The left and right Figure 6 depict the results using the explicit Euler and the forward-backward Euler as fast scale integrators respectively. For a reference we also apply the starting method MIS4 and the reference method RK3 with the same fast scale integrator. From the left Figure we immediately recognize two regions, methods in the top part, and the lower errors. The

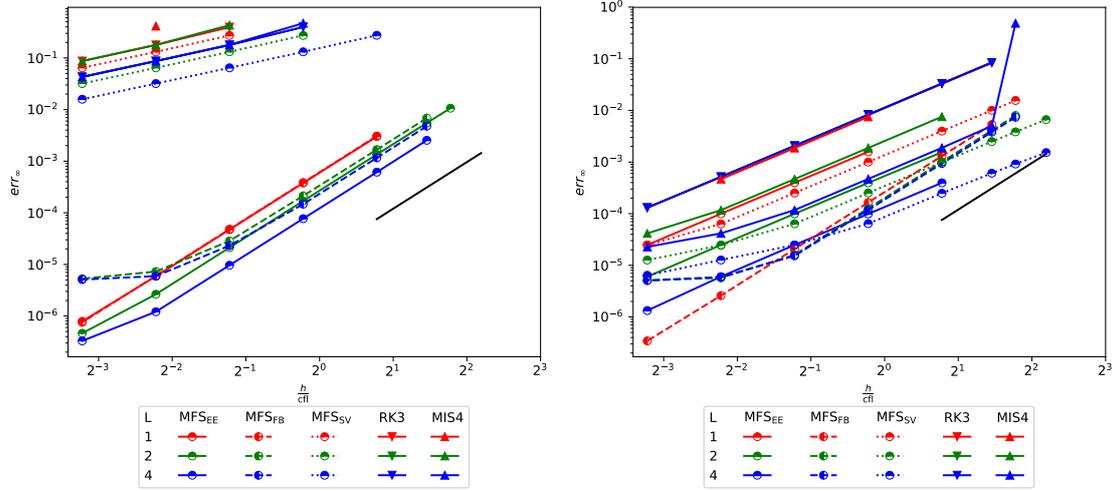


Figure 6: Compare the maximum error for linear acoustics at t_e of MFS_{EE} and MFS_{FB} methods by using the fast scale integrators explicit Euler (left) and forward-backward Euler (right). The colors correspond to the step factor L and the markers represent the methods. As a helper for the eye we add the third order line in black.

methods in the top, with the higher errors, are the MIS4 and RK3. These methods not only have a high error, they also do not behave like order three. But at least, they are stable. In contrast we see the MFS methods far below the aforementioned ones, leading to at least 100 times lower maximum errors. Furthermore these methods behave as order three, independent of the step factor L , as predicted by the order conditions.

The right Figure looks a bit different. This time both non-MFS methods, also have lower errors and behave like order two. A deeper look also reveals the expected order three for the MFS_{FB} method, and as the order conditions suggest, independent of the step factor L . But, in contrast to the expected selection, we see a lower bound on the error. This observation stems from the numerically solved order conditions. But up to this bound, the method behaves like order three.

But we also see, that the MFS method MFS_{EE} , when used with the forward-backward Euler as fast scale integrator, reaches only order two, too. Remember that the MFS_{FB} conditions consisted of three sub sets, namely the classical, the MIS and the MFS conditions. Furthermore the derivation of the MFS conditions led to parasitic derivatives of the form $\partial_2 g_1^X \langle g_Y \langle F \rangle \rangle$. These additional conditions arose due to the splitting of the forward-backward Euler method and are not fulfilled by the MFS_{EE} method. Hence these methods have a step factor dependent order when the fast scale integrator is a first order splitting method.

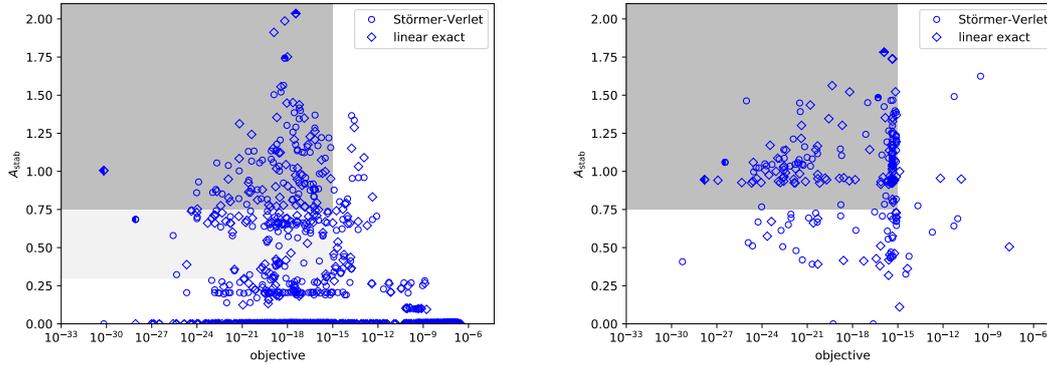


Figure 7: The area with respect to the order condition objective (II.6.2) with the initial method (left) and after optimization (right). This time we obtain 126 useful step ratios. In analogy to the first order methods the circle represents the numerical integrator and the diamond the exact solution.

III.1.2 Second order fast scale integrator

The construction of MFS_{SV} methods follows basically the same strategy as for the first order methods. We only change the initial method from MIS4 to MIS3A [52], because three stages suffice in this case. After that we fix all step size ratios and compute a preliminary coefficient set.

Optimization with second order integrator

Then we compute the stability areas (II.5.18) using the Störmer-Verlet method and the exact solution, see left Figure 7. Again many initial solutions lead to a vanishing stability triangle. But this time we have already 126 useful optimization candidates using the numerical and the exact integration. Furthermore already the initial methods lead to seemingly large areas, even larger than the reference method RK3. After optimization we obtain the areas and objectives in the right Figure 7. This time the optimization is not as effective as with the first order integrators. Furthermore in most times the optimization routine left the constraints and led to too large errors in the order conditions. And sadly we got even with the exact integration smaller stability areas as prior optimization. This behavior is a result of the two stage optimization. The first optimization stage starts from the initial parameters of the left Figure 7, but often leaves the constraint set, the second step mainly remains in the constraint set, but loses the better stability area. Hence we

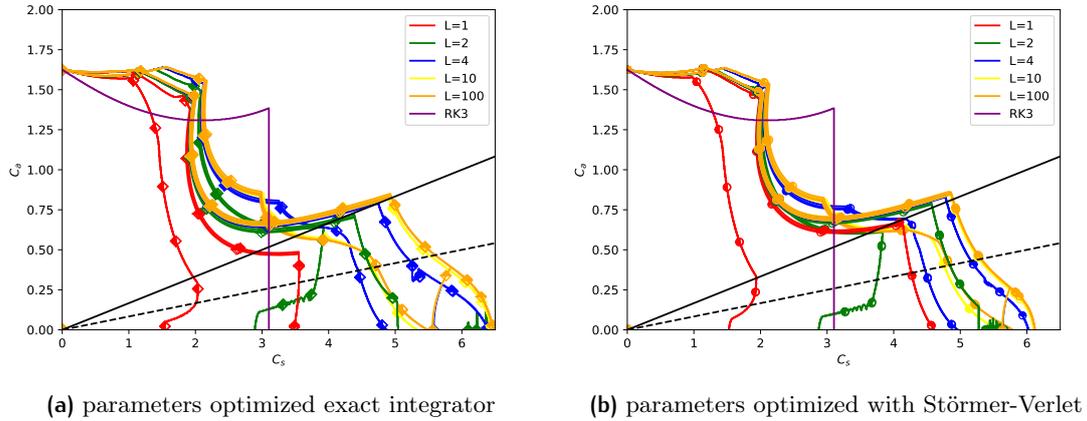


Figure 8: Stability areas for the MFS_{SV} methods in Table 2.

Φ	MFS_{SV}
exact	(1,1,3) \blacklozenge (1,1,7) \blacklozenge
numerical	(3,3,10) \bullet (1,1,5) \bullet

Table 2: Step factors for the optimal methods derived from the left Figure 7. We list the method parameters in the Tables I.9, I.10, I.11 and I.12 in the appendix.

neglect the optimization results and choose the methods obtained from the initial method, see Table 2.

Optimized stability areas

In Figure 8 we depict the stability area contour lines for the four methods in Table 2. The left Figure shows the contours with the best parameters for the exact stability areas, whereas the contours in the right Figure correspond to the numerical integrator. At the first sight both Figures are very similar. In particular the the left filled markers are both on nearly the same lines. That behavior is expected because the ratios are nearly the same and the coefficients too. Furthermore we obtain already with small step factors $L = 2$ and $L = 4$ very similar stability triangle sizes. Hence this time we are already fine with initial coefficients and can choose the coefficients, which lead to the largest stability triangle for exact integration. This result is in contrast to the first order integrators and a clear consequence of the smaller number of order conditions.

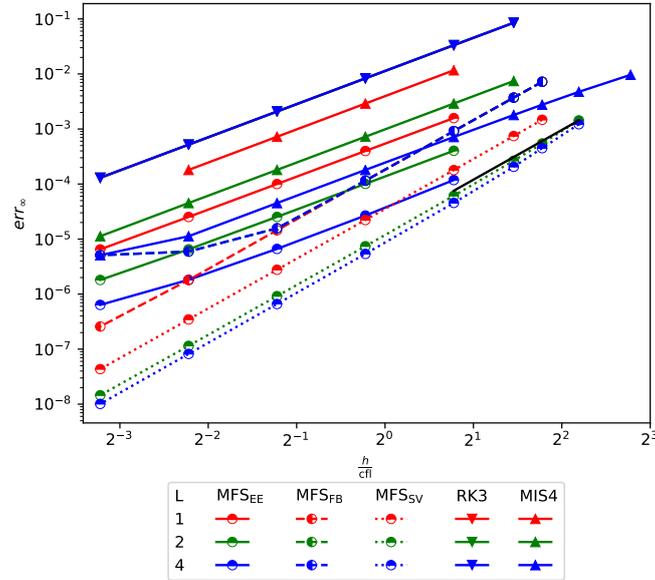


Figure 9: The errors using Störmer-Verlet as fast scale integrator for all MFS methods. The line styles with circled markers distinguish between the MFS methods.

Accuracy analysis - linear problem

Again we apply the MFS_{SV} method on the linear advection problem and depict the errors with the Störmer-Verlet as fast scale integrator in Figure 9. The advantage in accuracy compared to both reference methods and the other MFS methods is clearly visible.

III.1.3 Conclusions

This section discussed the construction of MFS methods of order three with different fast scale integrators. As already the derivation of the order conditions showed, the first order fast scale integrators lead to many more conditions, compared to the second order fast scale integrator. Our aim are order three methods with a large stability triangle. The stability function is already an optimization problem, which finds the maximum amplification along a halve circle. This increases the complexity and numerical effort further.

This increase in number of parameters, and order condition complexity, for the MFS_{EE} and MFS_{FB} conditions required a multistage numerical optimization procedure. Due to the local nature of optimization algorithms, we required good starting parameters. The different parameter sets, namely rational parameters for the step factors and real

parameters for the method coefficients, form a mixed optimization problem. Instead of a mixed integer-real optimization we fixed the rational parameters and solved around 10.000 independent optimization problems for every feasible step factor combination.

Nevertheless finding third order methods with the first order integrator remained hard. But we found an improved method with larger stability area for the step factor $L = 2$ and the step factors in Table 1.

Using a second order fast scale integrator led to the set of order conditions MFS_{SV} , with leads less order conditions compared to the first order integrator. Hence we used a method with less stages and therefore got a smaller parameter set. In turn we were able to optimize the parameters with a one stage procedure and obtained the optimal step factors in Table 2. Again the methods have a larger stability area compared to the reference method.

After the stability analysis we also analyzed the accuracy on the (discretized) linear advection equation. In that regard, all newly constructed methods are more accurate by magnitudes.

III.2 BUBBLES AND EXTERNALLY DRIVEN FLOWS

The previous section discussed the advantages and disadvantages of the MFS methods with the linear advection problem. Now we to proceed to two nonlinear problems. Hence from every MFS method the parameter set with the largest stability area and neglect the markers for them.

III.2.1 The force driven flow

Let us consider the benchmark problem from Durran and Blossey [10]. The PDE

$$\partial_t \mathbf{u} + \mathbf{u} \partial_x \mathbf{u} + w \partial_z \mathbf{u} + \partial_z \psi - \frac{\mathbf{u}_0(z) - \bar{\mathbf{u}}(z, t)}{\tau} = -\partial_x P \quad (2.1a)$$

$$\partial_t w + \mathbf{u} \partial_x w + w \partial_z w - \partial_x \psi - \mathbf{b} = -\partial_z P \quad (2.1b)$$

$$\partial_t \mathbf{b} + \mathbf{u} \partial_x \mathbf{b} + w \partial_z \mathbf{b} = N^2 w \quad (2.1c)$$

$$\partial_t P + \mathbf{u} \partial_x P + w \partial_z P = -c_s^2 (\partial_x \mathbf{u} + \partial_z w) \quad (2.1d)$$

Parameter	Value	Parameter	Value
ω	[1/s] 0.005	Δx	[m] 250
L_x	[km] 10.0	Δz	[m] 50
L_z	[km] 2.5	t_d	[s] 3000
Ψ_0	$[\frac{m^2}{s}]$ 80.0	K	[1/s] 4.69×10^{-4}
width	[km] 300		
height	[km] 10		
N	[1/s] 0.02		
c_s	$[\frac{m}{s}]$ 350		

Table 3: Physical (left) and numerical (right) parameters for the corrected non-hydrostatic test case. The fourth hyper diffusion coefficient K has numerical character because it serves a stabilization term.

describes a stratified shear flow, accelerated by a non-divergent forcing term

$$\psi(t, x, z) = \Psi_0 \frac{\pi x}{L_x} \sin(\omega t) \exp \left[- \left(\frac{\pi x}{L_x} \right)^2 - \left(\frac{\pi z}{L_z} \right)^2 \right].$$

The domain is a 300 km wide and 10 km height rectangle. At the bottom and top boundary, i.e. $z = -5$ km and $z = 5$ km we assume a rigid wall. The left and right boundaries are periodic. We discretize the PDE with finite differences on a staggered mesh with the stencils given by Durran and Blossey [10]. The remaining physical and numerical parameters are listed in Table 3. The experiment is non hydro static because the time scale of the forcing $\nabla \times \Psi$ is in the same range as the scale of the gravity N^2 . We will keep the space discretization fixed and analyze the convergence with respect to the buoyancy errors at the end time t_d . As a reference solution we solve the space discretized system with the explicit RK4 and a step size $h = 5 \times 10^{-3}$ s. The top Figure 10 depicts the horizontal velocity contours at the end time and the gray scaled contour lines of the forcing term ψ . These results are qualitatively comparable to the solutions in [50].

Error in buoyancy

We apply the reference methods RK3 and MIS4 and the new MFS methods MFS_{EE} , MFS_{FB} and MFS_{SV} with all three fast scale integrators on this benchmark problem. The left and right Figure 11 depict the errors for all combinations.

Due to the vast number of combination we obtain a quite full graph with many details and therefore we split the diagram in two and use a common legend. The top legend explains the markers, which correspond to the outer methods. With the two quadrangle markers, namely square and diamond, distinguish the MFS methods which base on first

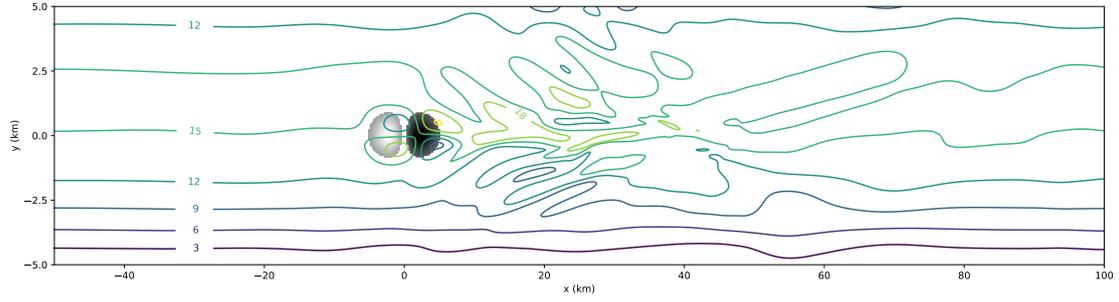


Figure 10: Contour lines of the horizontal reference velocities computed with RK4 and $h = 5 \times 10^{-3}$ s.

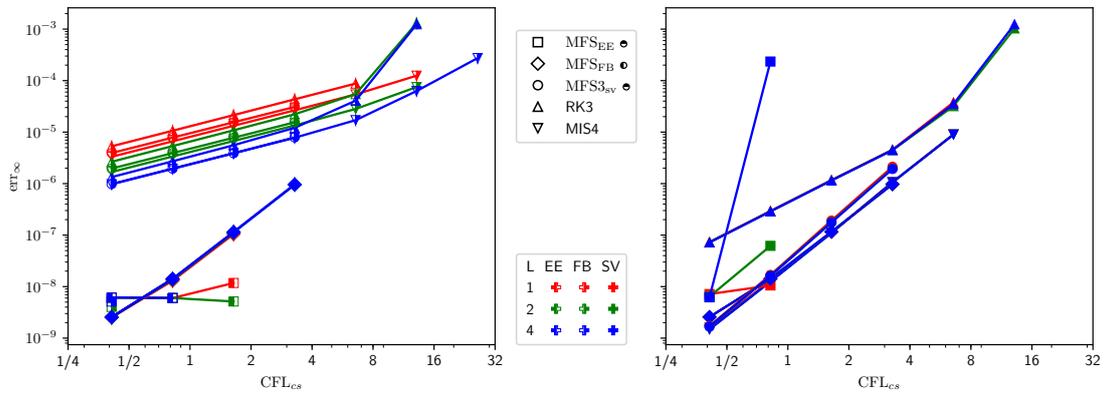


Figure 11: The maximal buoyancy error for the non-hydrostatic driven flow experiment. Again the colors represent the step factors, whereas the markers correspond to the outer methods. We distinct the fast scale solvers explicit Euler, forward-backward Euler and Störmer-verlet by the marker fill styles left, right and full respectively. These fill styles can be remembered as first order explicit, first order (semi)-implicit and second order mixed.

order integrators, namely MFS_{EE} and MFS_{FB} . For the MFS_{SV} , which bases on the second order integrator Störmer-Verlet, we use the circle. For the references methods RK3 and MIS4 we triangles with top and bottom pointing corners respectively. The second legend at the bottom describes the fill styles and colors. We use the fill style to describe the fast scale integrator. A halve-left filled marker corresponds to the explicit Euler method, the halve-right filled marker to the forward-backward Euler and a completely filled marker represents the second order Störmer-Verlet method. This fill style resembles the time dependency and (halved) order within a fast step. The explicit method is of order one and uses only the old value, which is left in time, the semi-implicit forward backward Euler is also of order one and has an implicit part. In contrast the Störmer-Verlet method

is of order two and uses therefore a full marker. In the left Figure we show only the errors of the first order methods and in the left one only the second order method. But both Figures use the same error axis and the same scale for the CFL_{CS} axis. Hence a seemingly full marker in the left Figure does not show a second order integrator, but the two first order methods lead to the same errors.

The colors red, green and blue in turn represent the different step factors L analogous to the previous sections. Hence lines with the same circle or quadrangle marker and same fill style should be parallel, because the order of MFS methods is independent of the step factor.

Let us first inspect the left Figure. On the first sight we see all three colors with the halve-filled triangles in the left center above all blue lines. Hence the reference methods using first order integrators are very inaccurate with small step factors. The slope of these lines is near one, hence the methods reduce to order one. A deeper look in this area also reveals the halve filled circles, which represent to the MFS_{SV} methods with three stages. These lines also have a slope of one. This clearly shows the necessity of the additional order conditions in the Tables II.2 and II.3. Below these lines we see the halve filled quadrangles and the diamonds, which look filled. This shows us that both first order fast integrators lead to nearly the same errors in case of MFS_{FB} , independent of the step scale factor L . But we see also a drawback of the numerical solution process described earlier. The halve filled squares lead to an step size independent error for a too small step size. This indicates a poor solution of the order conditions. Furthermore we see also the effect of the optimization with a fixed step factor. We find left most halve filled square near $cfl_{cs} \approx 4$ colored in green. Even with more steps, the method gets unstable. The very same is true for the right filled square, it has only one point near $cfl_{cs} \approx \frac{1}{2}$.

In contrast the diamonds exhibit order three up to an absolute error near 1.0×10^{-9} , also independent of the step factor L as predicted by the order conditions. And at the same time the accuracy and stability is less effected by the step factor then the MFS_{EE} method. This results shows more a lack in the optimization, because the MFS_{EE} order conditions are a subset of the MFS_{FB} order conditions. Also note the markers at larger CFL numbers for the reference methods. Hence in terms of accuracy the MFS methods are better by several magnitudes and lead to a predictable order. But in terms of stability it is the other way around.

The right Figure uses the same global methods as the left one, but only the second order fast scale integrator Störmer-Verlet. Hence we see only the fully filled markers. This time we have two groups of triangles. The triangles with top corner are always, and for every color, above the other lines. Hence the reference method RK3 is again worse

in terms of accuracy, but reaches order two instead of order one before. Furthermore the lines with triangle marker with bottom corner, which represent the MIS4 method, have slope three, indicating the expected order three, independent of the step factor. The same applies to the MFS_{SV} and MFS_{FB} method. In contrast again, the MFS_{EE} method is only stable for a very small step size.

Conclusions

The nonlinear benchmark example showed the effect of the fast scale integrator on the accuracy of complete method. The new MFS schemes behave as order three for all fast scale integrators and independent in the step factor. This is on contrast to the reference methods RK3 and MIS4, which reduce the order with the first order fast scale integrators for smaller step sizes and too small step factors. Despite the good behavior in the linear model problem and the slightly larger stability areas, the MFS_{EE} methods are unstable in a wide range of macro step sizes.

Using the second order fast scale integrator leads to order three methods in all cases. Furthermore the MFS are again more accurate than the reference method RK3, but require smaller time steps due to the stability constraints.

III.2.2 The cold bubble down burst experiment

In 1993 the group Straka et al. [43] introduced the so called *cold bubble down blast* experiment. In this benchmark we solve the (inviscid) compressible Euler equations

$$\partial_t \rho = -\partial_x(\rho u) - \partial_z(\rho w) \quad (2.2a)$$

$$\partial_t(\rho u) + u\partial_x(\rho u) + w\partial_z(\rho u) = -\partial_x P \quad (2.2b)$$

$$\partial_t(\rho w) + u\partial_x(\rho w) + w\partial_z(\rho w) = -\partial_z P - \rho g \quad (2.2c)$$

$$\partial_t(\rho \theta) + u\partial_x(\rho \theta) + w\partial_z(\rho \theta) = 0 \quad (2.2d)$$

in terms of the potential temperature $\theta = T \left(\frac{P}{P_0} \right)^{-\frac{R}{C_p}}$. The physical parameters ideal gas constant $R = 287 \frac{\text{J}}{\text{kg K}}$ and the specific heat at constant pressure $C_p = 1004 \frac{\text{J}}{\text{kg K}}$ correspond to the ideal gas law [28]. The reference pressure $P_0 = 10 \times 10^5 \text{ Pa}$ corresponds nearly to the standard pressure near ground. Note that we neglected the diffusion, hence the system constitutes a pure hyperbolic PDE without any damping. A PDE requires at least a domain. We choose a 18.4 km wide and 10 km height rectangle, see Figure 12.



Figure 12: Domain and boundary for the cold bubble down blast benchmark. The orange line represents a rigid boundary, i.e. $\mathbf{u} = \mathbf{w} = 0$ with zero normal pressure gradient. The blue lines represent the periodic boundary.

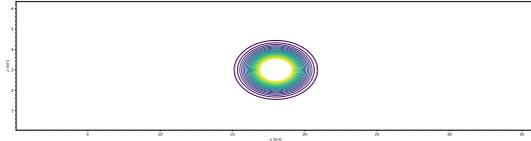


Figure 13: Initial potential temperature for the cold bubble down blast benchmark. We placed a 15 K colder elliptic bubble, centered at (3 km, 18 km).

Initially we place a 15 K colder bubble in the center of the domain. After $t_e = 900$ s we obtain a nearly symmetric solution, see the lowest Figure in 14. These solutions only show the lower halve contours. More importantly we have very similar solutions as [43].

Stability of MFS methods

Whereas the previous section analyzed the accuracy of the numerical methods, this time we concentrate in the stability. Hence we solve the benchmark problem up to the end time $t_d = 900$ s and check if the method remains stable. The three Tables 4 summarize the required numbers of macro steps without divergence damping. Hence we require much more steps, then Knoth and Wensch [26] in their experiments for RK3. Each (sub) table corresponds to the fast scale integrator, whereas the columns represent the corresponding outer methods with varying step factors $L \in \{1, 2, 4\}$. The first table uses the explicit Euler as fast scale integrator. This time the explicit Euler method leads to a stable complete method using the order conditions MFS_{EE} . But this comes clearly from the optimization procedure. But it is interesting to see, that we need five times more steps for RK3, which is better then the stability analysis predicted. Furthermore we see that we obtained the smallest number of macro steps for the step factor $L = 2$, which stems clearly from the optimization with this step factor. Furthermore the MFS_{FB} is also more stable than the MIS4 and RK3 method and, more importantly seems to be even more stable than MFS_{EE} . Again the ratios between MIS4 and MFS_{FB} are unexpected high. In this sense we also see a small benefit of the MFS_{SV} conditions II.4. These methods also requires more macro steps then MFS_{EE} and MFS_{FB} , but less then MIS4 and RK3.

In the second Table we use the forward-backward Euler method as fast scale integrator. This time the RK3 methods is clearly better then in the previous experiment. Furthermore

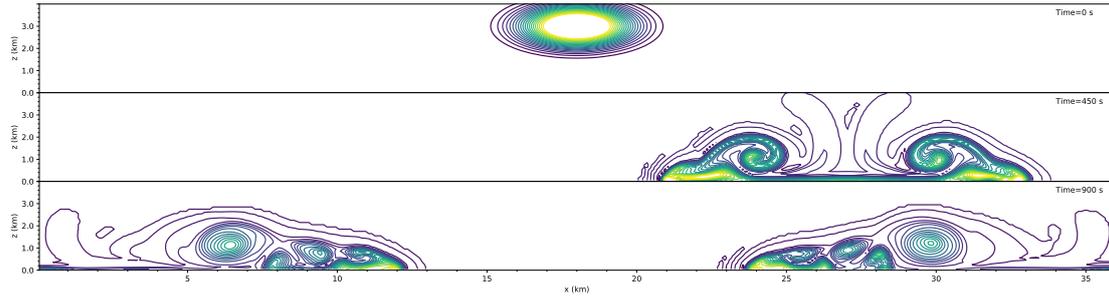


Figure 14: The solutions of the cold bubble down blast experiment computed with RK3, $L=1$ and the forward-backward Euler as fast scale integrator. From top to bottom we see contours of the initial values, and the solutions at $t = 450$ s and $t = 900$ s.

the MFS_{EE} gets worse than before. Although the MFS_{FB} , which order conditions belongs to this fast scale integrator, benefits from higher step factors, the benefit seems to stop at $L = 4$. It outperforms MIS4 by a factor of nearly two for $L = 2$ and $L = 4$. The benefit compared to RK3 is negligible. Again the MFS_{SV} is better than MFS_{EE} and MFS_{FB} . Furthermore it is interesting to see that this method also outperforms MIS4 by a factor of two and uses only three stages instead of four. But the method uses more step within one stage, hence it is twice efficient. Compare to the previous Table, we clearly see the benefit of the stable fast scale integrator.

The last Table uses the second order integrator Störmer-Verlet. This time the MFS_{SV} method performs best for all step factors L and MFS_{EE} performs worst. We see the largest benefit for the smallest step factor $L = 1$ in particular compared to MIS4. With larger factors the benefit decreases. We also see a way smaller dependency on the step factor for all methods. Only the MIS4 method requires three times more macro steps for $L = 1$ compare to $L = 2$. But for the higher factor, the required macro steps remains stable.

Conclusions

Let us summarize the stability results. On the one hand we see the benefit of the stable and higher order fast scale integrator. This is true for the pure MIS methods and the MFS methods. But we also see that one can obtain useful MFS methods using the first order methods, where the MIS methods perform worse. The results for the explicit Euler are worst, even with the MFS conditions. This might be related to the unstable integrator, but the stability analysis on the linear problem looked more promising. Hence it is not completely clear if the drawback stems from the low accuracy, or the missing stability

explicit Euler					
L	MFS _{EE} ●	MFS _{FB} ●	MFS _{SV} ●	MIS4	RK3
1	1,933	1,860	5,507	11,199	10,200
2	1,532	1,398	4,172	4,533	6,240
4	2,016	1,153	3,393	4,533	4,728
forward-backward Euler					
L	MFS _{EE} ●	MFS _{FB} ●	MFS _{SV} ●	MIS4	RK3
1	3,062	2,759	1,000	3,385	1,331
2	2,845	1,139	829	1,992	1,258
4	2,805	1,045	807	1,992	1,241
Störmer-Verlet					
L	MFS _{EE} ●	MFS _{FB} ●	MFS _{SV} ●	MIS4	RK3
1	2,372	1,082	953	2,656	1,282
2	2,431	1,065	819	847	1,249
4	2,537	1,060	713	847	1,237

Table 4: Number of macro steps required for stable integration of the cold bubble down blast problem without diffusion and without divergence damping.

properties. If the instability would dominate the stability of the complete method, we would expect sufficiently lower stable number of steps using the forward-backward Euler. But this is not the case. Also the additional order conditions make this conclusion a little bit more vague, because switching to the stable first order method also require more order conditions, as discussed earlier.

IV

CONCLUSIONS AND OUTLOOK

IV.1 CONCLUSIONS

The construction of suitable multirate methods for multi scale problems is an ongoing challenge. One method class are the multirate infinitesimal steps methods, which are based on the exact resolution of the shortest time scale. In practice one has to resolve this scale numerically too, hence the order conditions of MIS are only true for an infinite (or many) steps. We extended these methods by a class of scalable fast scale integrator, where we scale the number of steps. This extensions leads to the multirate finite step methods. In particular we constructed the order conditions such, that the order is independent in the (local) step size scaling. These order conditions, and in particular the additional MFS conditions for the factor independence, are the theoretical basis for the method construction.

The derivation of the order conditions revealed nearly no structure in the method coefficients. Furthermore these order conditions also depend on the partitioning structure of the fast scale integrator. Using a first order partitioned method nearly doubles the number of conditions, which reduces the positive effect of increased stability. Due to the higher number of conditions, one has to increase the number of stages, which in turn reduces the overall efficiency. In this regard we also showed the benefit of the second order fast scale integrator. This method not only requires less order conditions, but the increasing accuracy seems to increase the possible stability area even with a lower number of stages.

We also analyzed the stability of this new method class. The basis for the stability analysis are the advection equation for linear acoustics in an infinite domain. We derived the parameterized linear ODE (II.5.8), with the two parameters C_s and C_a , which represent the CFL number with respect to the speed of sound and background velocity, respectively. We apply the MFS method on this linear ODE and derive the corresponding stability functions (II.5.13c). It turns out, that these functions are the maxima of norms of a matrix function in terms of the two parameters C_a and C_s . Hence already the stability function is an optimization problem and computational expensive. The methods

are stable in a subset of the (C_a, C_s) space, where the stability functions are bounded from above by one. These subsets have an arbitrary structure. For a feasible optimization we defined the stability triangle within this stable subset.

A good MFS method then requires parameters with a large stability triangle area, and must fulfill the order conditions at the same time. Furthermore the method parameters are the union of two sets, namely the real method coefficients and the rational step ratios. We see, that this optimization process has a complicated cost function and nonlinear constraints. Numerical optimization algorithms depend highly on the initial values, hence we implemented a multistage optimization process and fixed the rational parameters. This approach led to many independent optimization problems, in our case around 10.000 problems. Many sub problems did converge to a solution, which solves the order conditions and had a sufficiently large stability area. From the remaining solutions, we selected the best methods according to two different criteria and analyzed their accuracy and sensitivity with respect to the step factor. In the linear case, these newly constructed methods had a sufficiently large stability area and are by magnitudes more accurate than reference method without the additional (MFS) order conditions.

The analysis on the nonlinear benchmark problems also showed the main benefit of these methods. Despite their slightly slower stable macro step size, they lead to magnitudes lower errors within the stable macro step sizes. As expected, this behavior depends highly on the underlying fast scale integrator. Whereas the use of the second order fast scale integrator led to a large improvement, with a smaller sensitivity in the micro step size factor L , first order integrators are more sensitive to the micro step size scaling factor.

IV.2 OUTLOOK

The main result are the order conditions, such that the order is independent in the step scaling factor L . To achieve that, we introduce a step ratio between the stages. This approach renders the coefficient optimization with respect to the stability triangle as a mixed optimization problem with real and rational parameters. At the current stage we fixed the ratios and optimized the real parameters with fixed integers. Hence we do not know if there are better ratios, we did not try yet and a mixed integer-real optimization method could reveal better methods with larger stability areas.

Also the optimization with fixed ratios depends heavily on the initial parameters. In this work we started from a good MIS method and hope to keep the good stability properties. But the additional MFS conditions are such hard that even the optimization

with an exact integrator loses the good stability areas of the initial method. Hence we can expect even better parameters with other optimization strategies. In particular a good heuristic for some coefficients could improve the stability optimization drastically.

We can expect another improvement by fixing the step factor and derive the corresponding order conditions. This can be done with the already derived conditions by collecting the derivatives and summing up their coefficients. The advantage of the MFS over the GARK methods is the known component partitioning for partitioned fast scale integrators. In this case we lose independence, but should gain additional degrees of freedom which could help increasing the stability areas.

BIBLIOGRAPHY

- [1] Awad H Al-Mohy and Nicholas J Higham. A new scaling and squaring algorithm for the matrix exponential. SIAM Journal on Matrix Analysis and Applications, 31(3): 970–989, 2009.
- [2] David G Andrews. An introduction to atmospheric physics. Cambridge University Press, 2010.
- [3] Josef Apfelbeck, T Krimly, and M Huigen. Managemententscheidungen im pflanzenbau in abhängigkeit von klima-/wettereinflüssen. Schriften der Gesellschaft für Wirtschafts-und Sozialwissenschaften des Landbaus eV Band 45, page 433, 2010.
- [4] A. L. Araujo, A. Murua, and J. M. Sanz-Serna. Symplectic methods based on decompositions. SIAM Journal on Numerical Analysis, 34(5):1926–1947, 1997. ISSN 00361429.
- [5] Uri M. Ascher, Steven J. Ruuth, and Raymond J. Spiteri. Implicit-explicit runge-kutta methods for time-dependent partial differential equations. Applied Numerical Mathematics, 25(2):151 – 167, 1997. ISSN 0168-9274. doi: [https://doi.org/10.1016/S0168-9274\(97\)00056-1](https://doi.org/10.1016/S0168-9274(97)00056-1). Special Issue on Time Integration.
- [6] Germund G. Dahlquist. A special stability problem for linear multistep methods. BIT Numerical Mathematics, 3(1):27–43, Mar 1963. ISSN 1572-9125. doi: 10.1007/BF01963532.
- [7] Maplesoft, division of Waterloo Maple Inc. Maple, 2017.
- [8] Günther Doms and M Baldauf. A description of the nonhydrostatic regional cosmo model. part i: Dynamics and numerics. Deutscher Wetterdienst, Offenbach, 2011.
- [9] Dale R. Durran. Numerical methods for wave equations in geophysical fluid dynamics /. Springer,, New York ; , Berlin ; , Heidelberg [u.a.] :, 1999. ISBN 0387983767.
- [10] Dale R Durran and Peter N Blossey. Implicit–explicit multistep methods for fast-wave–slow-wave problems. Monthly Weather Review, 140(4):1307–1325, 2012.

- [11] Charles William Gear and DR Wells. Multirate linear multistep methods. BIT Numerical Mathematics, 24(4):484–502, 1984. doi: <https://doi.org/10.1007/BF01934907>.
- [12] Adrian E Gill. Atmosphere. Ocean dynamics, 30:662, 1982.
- [13] Pierre-Arnaud Godlewski, Edwige Raviart. Numerical approximation of hyperbolic systems of conservation laws, 1996.
- [14] Knut Großmann et al. THERMO-ENERGETIC DESIGN OF MACHINE TOOLS. Springer, 2016.
- [15] Michael Günther and Peter Rentrop. Multirate row methods and latency of electric circuits. Applied Numerical Mathematics, 13(1-3):83–102, 1993.
- [16] Michael Günther and Adrian Sandu. Multirate generalized additive runge kutta methods. Numerische Mathematik, 133(3):497–524, 2016.
- [17] Ernst Hairer. Order conditions for numerical methods for partitioned ordinary differential equations. Numerische Mathematik, 36(4):431–445, 1981.
- [18] Ernst Hairer, Christian Lubich, and Gerhard Wanner. Geometric numerical integration illustrated by the Störmer-Verlet method. Acta Numerica, 12:399–450, may 2003. doi: 10.1017/s0962492902000144.
- [19] Ernst Hairer, Christian Lubich, and Gerhard Wanner. Geometric numerical integration : structure-preserving algorithms for ordinary differential equations /. Springer,, Berlin ; , Heidelberg [u.a.] ; , 2. ed. edition, 2006. ISBN 3540306633.
- [20] Ernst Hairer, S.P. Nørsett, and G. Wanner. Solving ordinary differential equations. 1, Nonstiff problems /. Springer,, Berlin ; , Heidelberg [u.a.] ; , 2., rev. ed., 1. softcover printing edition, 2009. ISBN 9783642051630.
- [21] Peter Henrici. Discrete variable methods in ordinary differential equations /. Wiley,, New York, NY [u.a.] ; , c 1962. ISBN 0471372242.
- [22] W Hundsdorfer, B Koren, JG Verwer, et al. A positive finite-difference advection scheme. Journal of computational physics, 117(1):35–46, 1995.
- [23] Stefan Jebens, Oswald Knöth, and Rüdiger Weiner. Explicit two-step peer methods for the compressible euler equations. Monthly Weather Review, 137(7):2380–2392, 2009.

- [24] Christopher A. Kennedy and Mark H. Carpenter. Additive runge–kutta schemes for convection–diffusion–reaction equations. Applied Numerical Mathematics, 44(1):139 – 181, 2003. ISSN 0168-9274. doi: [https://doi.org/10.1016/S0168-9274\(02\)00138-1](https://doi.org/10.1016/S0168-9274(02)00138-1).
- [25] JT Kiehl, JJ Hack, GB Bonan, BA Boville, DL Williamson, and PJ Rasch. The national center for atmospheric research community climate model: Ccm3. Journal of Climate, 11(6):1131–1149, 1998.
- [26] Oswald Knoth and Joerg Wensch. Generalized split-explicit runge–kutta methods for the compressible euler equations. Monthly Weather Review, 142(5):2067–2081, 2014. doi: 10.1175/MWR-D-13-00068.1.
- [27] Oswald Knoth and Ralf Wolke. Implicit-explicit runge-kutta methods for computing atmospheric reactive flows. Applied Numerical Mathematics, 28(2):327 – 341, 1998. ISSN 0168-9274. doi: [http://dx.doi.org/10.1016/S0168-9274\(98\)00051-8](http://dx.doi.org/10.1016/S0168-9274(98)00051-8).
- [28] Dilip Kondepudi. Introduction to modern thermodynamics. Wiley, Chichester, 2008. ISBN 9780470015995.
- [29] Karen Kuhn. Stability and applications of higher-order multirate Rosenbrock and Peer methods. PhD thesis, Darmstadt, 2014.
- [30] Randall J. LeVeque. Finite volume methods for hyperbolic problems. Cambridge Univ. Press, Cambridge [u.a.], repr. edition, 2005. ISBN 0521009243.
- [31] John Marshall, R Alan Plumb, et al. Atmosphere, ocean, and climate dynamics. Elsevier Academic Press, 2008.
- [32] Fedor Mesinger. Forward-backward scheme, and its use in a limited area model. Contrib. Atmos. Phys, 50(1977):200–210, 1977.
- [33] Andreas Naumann and Jörg Wensch. Multirate finite step methods with varying step sizes. PAMM, 17(1):851–852, 2017. doi: 10.1002/pamm.201710393.
- [34] Andreas Naumann and Jörg Wensch. Multirate finite step methods. Numerical Algorithms, 2019.
- [35] Andreas Naumann, Norman Lang, Marian Partzsch, Michael Beitelschmidt, Peter Benner, Axel Voigt, and Jörg Wensch. Computation of thermo-elastic deformations on machine tools a study of numerical methods. Production Engineering, 10(3): 253–263, Jun 2016. ISSN 1863-7353. doi: 10.1007/s11740-016-0674-7.

- [36] Andreas Naumann, Daniel Ruprecht, and Joerg Wensch. Toward transient finite element simulation of thermal deformation of machine tools in real-time. Computational Mechanics, Jan 2018. ISSN 1432-0924. doi: 10.1007/s00466-018-1540-6.
- [37] Helmut Podhaisky, Rüdiger Weiner, and Bernhard A Schmitt. Rosenbrock-type ‘peer’two-step methods. Applied numerical mathematics, 53(2-4):409–420, 2005.
- [38] Adrian Sandu and Michael Günther. A generalized-structure approach to additive runge–kutta methods. SIAM Journal on Numerical Analysis, 53(1):17–42, 2015.
- [39] Valeriu Savcenco, W Hundsdorfer, and JG Verwer. A multirate time stepping strategy for stiff ordinary differential equations. BIT Numerical Mathematics, 47(1):137–155, 2007.
- [40] William C. Skamarock and Joseph B. Klemp. The stability of time-split numerical methods for the hydrostatic and the nonhydrostatic elastic equations. Monthly Weather Review, 120(9):2109–2127, 1992. doi: 10.1175/1520-0493(1992)120<2109:TSOTSN>2.0.CO;2.
- [41] William C Skamarock, Joseph B Klemp, Jimy Dudhia, David O Gill, Dale M Barker, Wei Wang, and Jordan G Powers. A description of the advanced research wrf version 2. Technical report, DTIC Document, 2005.
- [42] Stig Skelboe and Per Ulfkjaer Andersen. Stability properties of backward euler multirate formulas. SIAM journal on scientific and statistical computing, 10(5):1000–1009, 1989.
- [43] J. M. Straka, Robert B. Wilhelmson, Louis J. Wicker, John R. Anderson, and Kelvin K. Droegemeier. Numerical solutions of a non-linear density current: A benchmark solution and comparisons. International Journal for Numerical Methods in Fluids, 17(1):1–22, 1993. doi: 10.1002/flid.1650170103.
- [44] K. Strehmel and R. Weiner. Numerik gewöhnlicher Differentialgleichungen. Teubner Studienbücher. B.G. Teubner, 1995. ISBN 9783519020974.
- [45] Geoffrey Vallis. Geophysical fluid dynamics: Whence, whither and why? Proceedings. Mathematical, physical, and engineering sciences / the Royal Society, 472:20160140, 08 2016. doi: 10.1098/rspa.2016.0140.
- [46] Charles F Van Loan. The ubiquitous kronecker product. Journal of computational and applied mathematics, 123(1):85–100, 2000.

- [47] Loup Verlet. Computer "experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. Phys. Rev., 159:98–103, Jul 1967. doi: 10.1103/PhysRev.159.98.
- [48] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical programming, 106(1):25–57, 2006.
- [49] Rüdiger Weiner, Katja Biermann, Bernhard A Schmitt, and Helmut Podhaisky. Explicit two-step peer methods. Computers & Mathematics with Applications, 55(4):609–619, 2008.
- [50] Hilary Weller, Sarah-Jane Lock, and Nigel Wood. Runge–kutta imex schemes for the horizontally explicit/vertically implicit (hevi) solution of wave equations. Journal of Computational Physics, 252:365 – 381, 2013. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2013.06.025>.
- [51] Neil C. Wells, Neil Wells. The atmosphere and ocean a physical introduction. Wiley-Blackwell, Chichester, West Sussex [u.a.], 3. ed. edition, 2012. ISBN 0470694688.
- [52] Jörg Wensch, Oswald Knöth, and Alexander Galant. Multirate infinitesimal step methods for atmospheric flow simulation. BIT Numerical Mathematics, 49(2):449–473, 2009.
- [53] Gerald Beresford Whitham. Linear and nonlinear waves, volume 42. John Wiley & Sons, 2011.
- [54] Louis J Wicker and William C Skamarock. A time-splitting scheme for the elastic equations incorporating second-order runge-kutta time differencing. Monthly Weather Review, 126(7):1992–1999, 1998.

Appendices

.1 ORDER CONDITIONS FOR STÖRMER-VERLET

The Störmer-Verlet method [47, 19] leads to the fast scale function (II.2.3c). This function switches between odd and even steps. Due to the switching the relations for the intermediate expressions (II.3.15) are quite lengthy. Nevertheless we derive them in details by considering the stage i and stack the results one over the other afterwards.

All expressions involve the partial derivative of the fast scale function with respect to the second argument at $\mathbf{y}_n = (\mathbf{y}^p, \mathbf{y}^q)$. Hence we first denote them for the odd and even steps, i.e.

$$\begin{aligned}\partial_2 g_1^{\text{SV}} \langle F \rangle &= \begin{bmatrix} g_p^p & 0 \\ g_p^q & 0 \end{bmatrix} \cdot \begin{bmatrix} F^p \\ F^q \end{bmatrix} \\ \partial_2 g_2^{\text{SV}} \langle F \rangle &= \begin{bmatrix} 0 & g_p^p \\ 0 & g_p^q \end{bmatrix} \cdot \begin{bmatrix} F^p \\ F^q \end{bmatrix}\end{aligned}$$

,and conclude

$$\partial_2 g_1^{\text{SV}} \langle F \rangle + \partial_2 g_2^{\text{SV}} \langle F \rangle = \mathbf{g}_y \langle F \rangle .$$

We assumed an even number of total fast steps in every stage, hence the sum of all odd and all even steps is

$$\sum_{j=1}^{\mathfrak{L}_{ii}} \partial_2 g_j^{\text{SV}} \langle F \rangle = \frac{\mathfrak{L}_{ii}}{2} \mathbf{g}_y \langle F \rangle .$$

After stacking all stages together, we end up with equation (II.3.36a).

The equations (II.3.36) are the sums of derivatives of \mathbf{g}_j . so we have to split the steps in an odd and an even set. We assume an even number of steps in every stage, such that $\mathfrak{L}_{ii} = 2M_{ii}$. Let us start with the double sum. First we split the outer index in the odd and even set, i.e.

$$\sum_{l=1}^{\mathfrak{L}_{ii}} \sum_{j=1}^{l-1} \partial_2 g_j^{\text{SV}} \langle F \rangle = \sum_{l=1}^{M_{ii}} S_{2l} + S_{2l-1} .$$

We simplify the inner sums, i.e.

$$\begin{aligned}
 S_{2l} &= \sum_{j=1}^{2l-1} \partial_2 g_j^{\text{SV}} \langle F \rangle \\
 &= \sum_{j=1}^{2l} \partial_2 g_j^{\text{SV}} \langle F \rangle - \partial_2 g_{2l}^{\text{SV}} \langle F \rangle \\
 &= l (\partial_2 g_1^{\text{SV}} \langle F \rangle + \partial_2 g_2^{\text{SV}} \langle F \rangle) - \partial_2 g_2^{\text{SV}} \langle F \rangle \\
 S_{2l-1} &= \sum_{j=1}^{2l-2} \partial_2 g_j^{\text{SV}} \langle F \rangle = (l-1) (\partial_2 g_1^{\text{SV}} \langle F \rangle + \partial_2 g_2^{\text{SV}} \langle F \rangle) ,
 \end{aligned}$$

and then sum them up to M_{ii} , i.e.

$$\sum_{l=1}^{M_{ii}} S_{2l} + S_{2l-1} = M_{ii}^2 g_y \langle F \rangle - M_{ii} \partial_2 g_2^{\text{SV}} \langle F \rangle .$$

After stacking the stages one over the other, we obtain (II.3.36b). The next sum is quite straight forward. We consider the definition for stage i and split the sum in odd and even summands, i.e.

$$\begin{aligned}
 \sum_{l=1}^{2M_{ii}} (l-1) \partial_2 g_l^{\text{SV}} \langle F \rangle &= \sum_{l=1}^{M_{ii}} (2l-2) \partial_2 g_{2l-1}^{\text{SV}} \langle F \rangle + (2l-1) \partial_2 g_{2l}^{\text{SV}} \langle F \rangle \\
 &= M_{ii} (M_{ii} - 1) \partial_2 g_1^{\text{SV}} \langle F \rangle + M_{ii}^2 \partial_2 g_2^{\text{SV}} \langle F \rangle
 \end{aligned}$$

and collect the quadratic factor M_{ii} to end up with

$$= M_{ii}^2 g_y \langle F \rangle - M_{ii} \partial_2 g_1^{\text{SV}} \langle F \rangle .$$

Again after stacking the last equations for every i , we end up with (II.3.36c). We simplify the sum (II.3.15d)

$$\sum_{l=1}^{2M_{ii}} \partial_2 g_l^{\text{SV}} \langle \partial_2 g_l^{\text{SV}} \langle F \rangle \rangle = M_{ii} (\partial_2 g_1^{\text{SV}} \langle \partial_2 g_1^{\text{SV}} \langle F \rangle \rangle + \partial_2 g_2^{\text{SV}} \langle \partial_2 g_2^{\text{SV}} \langle F \rangle \rangle)$$

by using the splitting in odd and even steps. The first order expressions for $\mathfrak{G}_1^{\hat{1}}$ in equation (II.3.22c) contain the difference between (II.3.36b) and (II.3.36c). All other expressions depend on the elementary differential $g_y \langle g_y \langle F \rangle \rangle$, which is definitely part of the exact

solution. Hence we improve the readability expressing the elementary differentials in (II.3.38a) and (II.3.38b) in terms of $\mathbf{g}_y \langle \mathbf{g}_y \langle F \rangle \rangle$, $\partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle$, $\partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_2^{\text{SV}} \langle F \rangle \rangle$ and $\partial_2 \mathbf{g}_2^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle$. We obtain the relation by comparing the coefficients. In details we start from

$$\begin{aligned} \alpha_1 \partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle + \alpha_2 \partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_2^{\text{SV}} \langle F \rangle \rangle + \alpha_3 \partial_2 \mathbf{g}_2^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle + \alpha_4 \partial_2 \mathbf{g}_2^{\text{SV}} \langle \partial_2 \mathbf{g}_2^{\text{SV}} \langle F \rangle \rangle \\ = \beta_1 \mathbf{g}_y \langle \mathbf{g}_y \langle F \rangle \rangle + \beta_2 \partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle + \beta_3 \partial_2 \mathbf{g}_1^{\text{SV}} \langle \partial_2 \mathbf{g}_2^{\text{SV}} \langle F \rangle \rangle + \alpha_4 \partial_2 \mathbf{g}_2^{\text{SV}} \langle \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle \rangle \end{aligned}$$

and use the relation $\mathbf{g}_y \langle F \rangle = \partial_2 \mathbf{g}_1^{\text{SV}} \langle F \rangle + \partial_2 \mathbf{g}_2^{\text{SV}} \langle F \rangle$ for the coefficient β_1 to obtain

$$\alpha_1 = \beta_1 + \beta_2$$

$$\alpha_2 = \beta_1 + \beta_3$$

$$\alpha_3 = \beta_1 + \beta_4$$

$$\alpha_4 = \beta_1$$

In matrix notation we have simply

$$\alpha = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \beta.$$

A

METHOD PARAMETERS

This section lists the method parameters for all MFS methods. The matrices are all lower diagonal, hence we neglect the zeros in the left upper part.

A.1 EXPLICIT EULER

	0.0				
	0.0				
α	0.0	1.35696769425831998			
	0.0	1.31327225291199001	0.15436408818894601		
	0.0	0.94002051579741697	2.40650377463183007	-0.22682715992841601	
	0.0				
	0.0				
γ	0.0	-1.50597264576960010			
	0.0	-0.29105924742822997	2.66524779827178993		
	0.0	0.51081132452772005	1.51206007860341995	-0.49061910502352202	
		0.0			
	0.29527044916201600				
β	0.09274719960944830	0.02330752755939680			
	-1.23764131908762587	0.60825426472370259	0.87314944369011205		
	-1.87549345659334765	0.25644691678441212	1.84108741247911345	0.60440783018237298	
D	0.0	0.29527044916201600	0.11605472716884510	0.24376238932618877	0.82644870285255090
\hat{L}	0.0	1	2	1	10

Table 1: Method MFS_{EE}◆

	0.0				
	0.0				
α	0.0	0.50770259025695397			
	0.0	2.55223309427528999	-1.65135794011186010		
	0.0	3.87474984894440011	-3.99414336027856010	1.85975338216983999	
	0.0				
	0.0				
γ	0.0	0.32334505037659700			
	0.0	-0.55635130119042298	-0.62587068636613297		
	0.0	-0.08466733133229019	0.13001064532144399	0.08026651515378780	
		0.0			
		0.76618205998077205			
β	-0.20036432380799540	0.28264534472436298			
	-1.12743602960268197	0.86540890326544295	0.57828277805431405		
	-0.31375318945274439	2.78849404571018544	-2.97297028984825262	0.96898021409557400	
D	0.0	0.76618205998077205	0.08228102091636758	0.31625565171707504	0.47075078050476249
\hat{L}	0.0	3	1	7	2

Table 2: Method MFS_{EE}♦

	0.0				
	0.0				
α	0.0	4.31110086549293037			
	0.0	2.03887934463184983	0.18283213875151999		
	0.0	-3.57019119861879020	0.62377956461540895	3.28845610569607993	
	0.0				
	0.0				
γ	0.0	-3.75518412156469994			
	0.0	-0.54200982473967996	-0.27768290338766999		
	0.0	0.14400670861812601	-1.17417247867254004	0.17365047480579099	
		0.0			
		0.33821700145988498			
β	-0.96644387071343174	1.26764591039591989			
	-0.32316583760744994	0.38387608104649351	0.10544534615568200		
	0.14819976923544154	-0.49493613670258257	-0.26164760195873832	0.86909731048365502	
D	0.0	0.33821700145988498	0.30120203968248815	0.16615558959472559	0.26071334105777566
\hat{L}	0.0	5	4	1	8

Table 3: Method MFS_{EE}•

	0.0				
	0.0				
α	0.0	6.01844137906912025			
	0.0	3.17597064336269996	0.09336837926704129		
	0.0	-6.28804318447577959	0.97378971866153796	2.67750480930958989	
	0.0				
	0.0				
γ	0.0	-4.87354639128365008			
	0.0	-0.90410482493547795	-0.20040253425053900		
	0.0	1.79206300070526003	-1.18359459330093997	0.26666040684843101	
		0.0			
	0.20110677029919699				
β	-2.16115518922650640	2.61980129463846012			
	-0.42811244508738638	0.55029042527596195	0.10414733294970401		
	0.17553602312553196	-0.50196922398821875	-0.16981343878052868	0.75054797533720397	
D	0.0	0.20110677029919699	0.45864610541195372	0.22632531313827958	0.25430133569398850
\hat{L}	0.0	3	3	1	10

Table 4: Method MFS_{EE}•

A.2 FORWARD-BACKWARD EULER

	0.0				
	0.0				
α	0.0	2.20416741941249983			
	0.0	1.00572652778042992	0.01646952103695340		
	0.0	-2.63467977694915989	0.86288667305979805	3.56252042428050997	
	0.0				
	0.0				
γ	0.0	-0.47347843609023499			
	0.0	-0.03775918607238830	-0.63741046760519304		
	0.0	1.91282645226976999	-0.34378232657069602	-1.90780710073839010	

		0.0			
		0.33563055714402601			
β		-0.19135909591037864	0.35027167374789397		
		0.21856435977363553	0.06724424325829589	0.19840693398699299	
		-0.25822536323777184	-0.74577471527137007	0.20754059015047582	1.07600328442156989
D	0.0	0.33563055714402601	0.15891257783751533	0.48421553701892445	0.27954379606290392
\hat{L}	0.0	8	1	10	8

Table 5: Method MFS_{FB} ♦

		0.0			
		0.0			
α	0.0	0.67775513851723901			
	0.0	0.44587916963996699	5.21847269717312034		
	0.0	-2.31815504482795998	8.13924973148414921	-0.00150457635340375	
	0.0				
	0.0				
γ	0.0	-0.38403561650966100			
	0.0	-1.82090162190679994	1.27985638569056004		
	0.0	-0.13713181371771799	1.65592185004063008	-0.78877583252057204	
		0.0			
		0.31325711303701798			
β		0.03876661715627612	0.08153525550394110		
		-0.28399005534444532	2.46599925886028881	-1.88332839239454009	
		-1.190512204344454241	0.75991881319136512	0.91038209549308524	0.19568520526672201
D	0.0	0.31325711303701798	0.12030187266021722	0.29868081112130351	0.67547390960663001
\hat{L}	0.0	7	2	3	8

Table 6: Method MFS_{FB} ♦

	0.0				
	0.0				
α	0.0	1.17444488150848003			
	0.0	2.69878148531170003	1.94605538371321996		
	0.0	1.13184100699080004	4.38816638810659043	-0.40015001962543800	
	0.0				
	0.0				
γ	0.0	-1.27657495230496010			
	0.0	-1.18350663238632992	0.35970717629800703		
	0.0	0.86440691616758403	0.47261604339382501	-0.77177502608269799	
		0.0			
		0.23304407466776300			
β	-0.05189559186747517	0.34939210987684399			
	-0.85760242669516740	1.90021547542302427	-0.86525003268156597		
	-1.08677261700370331	1.34590992990005653	0.11718852033074945	0.18935360450091199	
D	0.0	0.23304407466776300	0.29749651800936883	0.17736301604629090	0.56567943772801466
\hat{L}	0.0	2	7	2	10

Table 7: Method MFS_{FB}•

	0.0				
	0.0				
α	0.0	0.73027817627567204			
	0.0	1.75577347374625004	2.73820353488819013		
	0.0	-1.49594622671587008	6.61583487708920970	-0.08003714588242260	
	0.0				
	0.0				
γ	0.0	-0.52794155976831803			
	0.0	-2.83666854991050021	2.81426142230629983		
	0.0	-0.02084408206216050	1.04058444444464993	-0.68059790500408801	
		0.0			
		0.32950301692732997			
β	0.04692028559841566	0.12703802173876499			
	-0.11432570532779690	2.57849216636630540	-2.20666957928834018		
	-0.70056660468301701	1.19288959448501530	-0.07425488455789164	0.18059006751548701	
D	0.0	0.32950301692732997	0.17395830733718065	0.25749688175016816	0.59865817275959365
\hat{L}	0.0	6	2	3	10

Table 8: Method MFS_{FB}

A.3 STÖRMER-VERLET

	0.0			
	0.0			
α	0.0	-0.03029477970719350		
	0.0	0.87286903313859499	-0.13999909406739400	
	0.0			
	0.0			
γ	0.0	-0.04682174633143820		
	0.0	0.39249640242767397	-0.30530727510842998	
		0.0		
	0.36857320894108703			
β	0.16645240321361443	0.48740994062999599		
	-0.11493004258019834	-0.00068690321206655	0.92774923204120197	
D	0.0	0.36857320894108703	0.65386234384361042	0.81213228624893707
\hat{L}	0.0	1	1	3

Table 9: Method MFS_{SV}

	0.0			
	0.0			
α	0.0	-0.00452160744751265		
	0.0	0.96872582564320897	-0.11318188120983200	
	0.0			
	0.0			
γ	0.0	-0.00651389358391874		
	0.0	0.21516297929229700	-0.16116556194080900	
		0.0		
	0.35366175982409598			
β	0.15594096566386567	0.44982644533336702		
	-0.05551644691924718	-0.24570840380583514	1.04764866016143010	
D	0.0	0.35366175982409598	0.60576741099723264	0.74642380943634778
\hat{L}	0.0	1	1	7

Table 10: Method MFS_{SV}

	0.0			
	0.0			
α	0.0	-0.01982229423679270		
	0.0	0.91181799607547098	-0.16537826893277899	
	0.0			
	0.0			
γ	0.0	-0.03306869176963770		
	0.0	0.38493359646965097	-0.30178074201521898	
		0.0		
	0.37065791406424298			
β	0.19450485163725856	0.42969164749570998		
	-0.08744931713093868	-0.15721149367630635	1.04645018828632996	
D	0.0	0.37065791406424298	0.62419649913296849	0.80178937747908496
\hat{L}	0.0	3	3	10

Table 11: Method MFS_{SV}•

	0.0			
	0.0			
α	0.0	-0.00917893490079230		
	0.0	0.95376284755833596	-0.12390549442314799	
	0.0			
	0.0			
γ	0.0	-0.01349580471993010		
	0.0	0.25315887578639801	-0.19130014656538100	
		0.0		
	0.35713737894325998			
β	0.16006262097610371	0.45358119081342402		
	-0.06752674521576796	-0.20150368779205943	1.02886502881738995	
D	0.0	0.35713737894325998	0.61364381178952776	0.75983459580956259
\hat{L}	0.0	1	1	5

Table 12: Method MFS_{SV}•