

Article

# On Multirate GARK Schemes with Adaptive Micro Step Sizes for Fluid–Structure Interaction: Order Conditions and Preservation of the Geometric Conservation Law

Sascha Bremicker-Trübelhorn and Sigrun Ortleb \*

Department of Mathematics and Natural Sciences, University of Kassel, Heinrich-Plett-Str. 40, 34132 Kassel, Germany; struebel@mathematik.uni-kassel.de

\* Correspondence: ortleb@mathematik.uni-kassel.de; Tel.: +49-561-804-4316

Academic Editors: Hossein Zare-Behtash and Kiran Ramesh

Received: 15 December 2016; Accepted: 7 February 2017; Published: 14 February 2017

**Abstract:** The application of partitioned schemes to fluid–structure interaction (FSI) allows the use of already developed solvers specifically designed for the efficient solution of the corresponding subproblems. In this work, we propose and describe a loosely coupled partitioned scheme based on the recently introduced generalized-structure additively partitioned Runge-Kutta (GARK) framework. The resulting scheme combines implicit-explicit (IMEX) and multirate approaches while coupling of the subproblems is realized both on the level of the discrete time steps and at the level of interior Runge-Kutta stages. Specifically, we allow for varying micro step sizes for the fluid subproblem and therefore extend the multirate GARK framework based on constant micro steps. Furthermore, we derive the order conditions for this extension allowing for coupled time integration schemes of up to third order and discuss specific choices of the Runge-Kutta coefficients complying with the geometric conservation law. Finally, numerical experiments are carried out for uniform flow on a moving grid as well as the classical FSI test case of a moving piston.

**Keywords:** multirate schemes; IMEX coupling; time adaptivity; fluid–structure interaction

## 1. Introduction

Coupled systems as in the context of fluid–structure interaction (FSI) often consist of subsystems with significantly different time scales. These subsystems may also deviate considerably in terms of their stiffness. In this context, using the same explicit time integration scheme with synchronous time steps for all subsystems might enforce very small step sizes which correspond to the small time scale components or those responsible for stiffness. On the other hand, a purely implicit method yields a large (possibly non-linear) coupled system of equations to be solved per time step. Therefore, it is reasonable to allow each subsystem of the coupled problem to advance with its preassigned time integration scheme which is adapted to its stiffness and time scales. In addition, it is sensible to evaluate slow components less often than fast ones.

In the context of fluid–structure interaction, several forms of mixed time integration approaches have been suggested. For example in [1], an implicit-explicit (IMEX) approach based on partitioned Runge-Kutta schemes allows for explicit coupling of the two physical fields, whereas the same implicit scheme for fluid and structure equations is used. A similar IMEX procedure is used in [2]. In [3,4], a multirate scheme referred to as subcycling is used to advance the fluid with small explicit time steps while an implicit scheme is used to discretize the structure equations using large time steps. Hereby, different forms of the subcycling procedure with respect to the treatment of the pressure forces at the fluid-structure interface and the sequencing of fluid and structure solves are investigated in terms of

energy conservation. These algorithms fall into the category of loosely coupled partitioned methods. Generally, numerical approaches to the simulation of FSI phenomena can be classified into monolithic and partitioned solvers. Thereby, monolithic solvers are based on the direct discretization of the fully coupled problem which may then be linearized, e.g., by Newton's method as in [5]. In partitioned approaches, suitable subsolvers, often already available, are used for each subsystem and suitable coupling conditions are devised, e.g., in [6,7]. While in the latter works, strong coupling is enforced using Newton's method or fixed point iteration, respectively, a high order combined interface boundary treatment is devised in [8] in order to enhance the stability of loosely coupled schemes.

Similar to [1,2], we investigate the application of partitioned Runge-Kutta methods to fluid-structure interaction. Hence, this work fits into the category of loosely coupled partitioned schemes as no iterative methods are used for the coupling procedure itself. Nonetheless, as for the IMEX-Runge-Kutta approach in [1,2], the subsolvers are coupled at the level of interior Runge-Kutta stages in addition to the time levels corresponding to the chosen time step sizes. Furthermore, we combine the IMEX approach with the concept of multirate schemes which allows for different time step sizes of the subsolvers, corresponding to the subcycling strategy in [3,4].

The classical multirate idea for systems of ordinary differential equations dates back to the late fifties and was first studied by Rice [9]. Early multirate approaches were based on Runge-Kutta schemes [10,11], backward differentiation formulas [12–14], extrapolation schemes [15] or Rosenbrock-Wanner methods [16,17]. A drawback of these aforementioned early schemes is the fact that the coupling between slow and fast components of the system is done by interpolating and extrapolating state variables. This complicates the implementation into existing simulation packages [18]. Kværnø and Rentrop overcame this shortcoming by developing a concise theory of explicit multirate Runge-Kutta schemes and obtained promising results for electrical network simulations [19]. Deriving information by interpolation and extrapolation was circumvented by using the internal Runge-Kutta stages for coupling the integration schemes for the subsystems. In [18], a generalization to stiff problems was proposed. In that work, explicit Runge-Kutta schemes integrating the (nonstiff) fast system component are combined with linearly implicit Rosenbrock-Wanner schemes integrating the (stiff) slow system component.

IMEX partitioning is often used to term-split the corresponding partial differential equation, e.g., in the context of advection-diffusion-reaction equations, advection terms are classically discretized explicitly while viscous terms or stiff reaction is discretized by an implicit scheme [20–22]. IMEX-RK schemes have also been used in domain-based splittings, e.g., in [23] or based on eigenvalue estimates of the space discrete operators as in [24].

Very recently, Sandu and Günther [25] constructed a generalized-structure approach to additively partitioned Runge-Kutta (GARK) methods. This approach comes with a high level of flexibility and allows for different Runge-Kutta stage values as arguments of different components of the problem's additively partitioned right-hand side. In addition, it combines different Runge-Kutta schemes with inter-component coupling based on the Runge-Kutta stages. In particular, it is possible to combine both the IMEX and the multirate approach. The GARK framework gives the intuition necessary to define new classes of additive schemes and eases the construction of additive schemes which preserve the order of the base ones. Günther and Sandu made use of the GARK framework to derive a multirate scheme solely based on implicit and explicit Runge-Kutta schemes [26]. Günther and Sandu's multirate GARK scheme carries out just one so-called macro step of a "large" step size  $H$  within the slow component per iteration while at the same time carrying out several so-called micro steps of a "small" step size  $h < H$  within the fast component per iteration, until both components "meet up" again. Hence, several equidistant micro time steps are carried out for one macro time step. The GARK formalism includes a variety of well-known schemes for partitioned problems, like the classical implicit-explicit (IMEX) Runge-Kutta schemes [20,23], as well as former multirate methods as in [19,21]. Furthermore, Günther, Hachtel and Sandu showed the feasibility of the multirate GARK

approach for both implicit-implicit and implicit-explicit pairing of basic schemes [27] in the context of thermal-electrical coupling.

Concerning fluid–structure interaction, the time scales of the fluid subsystem are typically much shorter than the ones of the structure subsystem which makes the multirate GARK framework an attractive candidate. In contrast to the original application from electrical circuit simulation [18,19] and the test examples from thermal-electrical coupling [27], the fluid is expected to be stiff in many applications, which is the reason why implicit schemes often need to be applied to the fast fluid part. In case of (partially) explicit time integration of the fluid equations, e.g., in advection-dominated flow, the time steps may be requested to change gradually as a consequence of changes of the characteristic speeds. In case of implicit time integration, the fast fluid step sizes (the micro steps) may have to vary during the structure step (the macro step) due to the convergence behavior of the nonlinear solver or due to error control.

Therefore, in this work, we seek to overcome the major restriction of keeping the micro step sizes constant, by allowing varying micro step sizes that may be adapted according to error or stability requirements. The multirate GARK framework is thus extended towards the adaptive choice of micro time steps within one macro time step. In this work, the corresponding order conditions are derived. Furthermore, we investigate the choice of coupling coefficients which allows the combined method to preserve the order of the single schemes within the partitioned approach. We then derive specific multirate GARK schemes applicable to mechanical fluid–structure interaction, give details regarding compliance with the geometric conservation law and show numerical results for the classical fluid–structure interaction problem of a moving piston.

This paper is structured as follows. Section 2 reviews the concept of partitioned Runge-Kutta schemes for partitioned ordinary differential equations including the recently introduced GARK approach. In Section 3, the concept of adaptive micro step sizes is introduced in order to obtain flexible multirate GARK (MGARK) schemes for coupled problems. We then derive the corresponding order conditions for this extension of the multirate GARK scheme given in [26]. Thereafter, we formulate the adaptive micro step MGARK scheme for mechanical fluid-structure interaction and discuss the resulting restrictions to the specific structure of the base methods as well as to the coupling coefficients. Our approach to time adaptivity based on embedded error estimation is then presented in Section 3.4. Numerical experiments are presented in Section 4. Here, compliance with the geometric conservation law is demonstrated for uniform flow on an artificially moving grid, the viability of adaptive micro steps is shown by an investigation of the step size statistics and finally, we investigate the efficiency of higher order time integration for small tolerances by a comparison of first, second and third order schemes in terms of error vs. CPU time.

## 2. Partitioning Systems of Ordinary Differential Equations

We may generally distinguish two kinds of partitioned systems of ordinary differential equations. The complete system under consideration may be composed of  $D \in \mathbb{N}$  different subsystems, each subsystem containing a specific subset of the given equations. The system is then partitioned by component. Else, if the right hand side of the ODE contains different terms of very different character as in the context of conservation laws containing convective and diffusive fluxes, usually the right hand side of the ODE is additively partitioned as in the following definition.

**Definition 1.** Let  $[0, T] \subseteq \mathbb{R}$  and  $f^{\{k\}}: \mathbb{R}^m \rightarrow \mathbb{R}^m$  for  $k = 1, \dots, D$ . A system of (autonomous) ordinary differential equations of the form

$$u'(t) = \sum_{k=1}^D f^{\{k\}}(u(t)), \quad t \in [0, T], \quad (1)$$

for  $\mathbf{u} \in C^1([0, T], \mathbb{R}^m)$  is called an additively partitioned system of ordinary differential equations. Adding initial conditions

$$\mathbf{u}(0) = \mathbf{u}_0 \in \mathbb{R}^m, \quad (2)$$

we obtain an additively partitioned initial value problem.

Every component-partitioned system may obviously be rewritten as an additively partitioned system of the same dimension inserting zeros into suitable positions of the right hand side terms. Therefore, we will restrict the discussion to the more general case of additively partitioned systems at this point.

For the numerical solution of additively partitioned initial value problems, Kennedy and Carpenter [22] constructed specific Runge-Kutta schemes called additive Runge-Kutta schemes (ARK). The ARK schemes apply different Runge-Kutta methods  $(\mathbf{A}^{\{k\}}, \mathbf{b}^{\{k\}})$ ,  $k = 1, \dots, D$ , to the terms on the right hand side of the ODE in the following form:

$$\mathbf{u}_{n,i} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{j=1}^s a_{i,j}^{\{k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,j}), \quad i = 1, \dots, s, \quad (3)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{i=1}^s b_i^{\{k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,i}), \quad (4)$$

with the corresponding Butcher table

$$\begin{array}{cccc} \mathbf{A}^{\{1\}} & \mathbf{A}^{\{2\}} & \dots & \mathbf{A}^{\{D\}} \\ \hline \mathbf{b}^{\{1\}\top} & \mathbf{b}^{\{2\}\top} & \dots & \mathbf{b}^{\{D\}\top} \end{array},$$

in which the individual Runge-Kutta schemes  $(\mathbf{A}^{\{k\}}, \mathbf{b}^{\{k\}})$  have the same number of stages  $s$ . For the special case of  $D = 2$ , an equivalent class for the solution of component-partitioned systems is given by Hairer et al. ([28], [II.15]). Sandu and Günther [25,26] extended the ARK schemes to the so-called generalized-structure additively partitioned Runge-Kutta (GARK) schemes. The most general form of a GARK scheme is given by the following definition.

**Definition 2.** A GARK scheme for the numerical solution of a  $D$ -times partitioned initial value problem (see Definition 1) with  $d \leq D$  GARK stages and step size  $h$  is given by

$$\mathbf{u}_{n,i}^{\{q\}} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{j=1}^{s^{\{k\}}} a_{i,j}^{\{q,k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,j}^{\{J(k)\}}), \quad i = 1, \dots, s^{\{q\}}, \quad q = 1, \dots, d \quad (5)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{i=1}^{s^{\{k\}}} b_i^{\{k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,i}^{\{J(k)\}}), \quad (6)$$

where  $J: \{1, \dots, D\} \rightarrow \{1, \dots, d\}$  is a map which determines which of the stage values  $\mathbf{u}_{n,i}^{\{J(k)\}}$  is assigned to the  $k$ -th right hand side term. We denote this scheme by  $\text{GARK}(D, d, J)$ .

The corresponding generalized Butcher table reads

$$\begin{array}{cccc} \mathbf{A}^{\{1,1\}} & \mathbf{A}^{\{1,2\}} & \dots & \mathbf{A}^{\{1,D\}} \\ \mathbf{A}^{\{2,1\}} & \mathbf{A}^{\{2,2\}} & \dots & \mathbf{A}^{\{2,D\}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{\{d,1\}} & \mathbf{A}^{\{d,2\}} & \dots & \mathbf{A}^{\{d,D\}} \\ \hline \mathbf{b}^{\{1\}\top} & \mathbf{b}^{\{2\}\top} & \dots & \mathbf{b}^{\{D\}\top} \end{array}.$$

In [25], Sandu and Günther have shown that GARK schemes are equivalent to ARK schemes in the sense that, on the one hand, every  $\text{ARK}(D)$  scheme can be written as  $\text{GARK}(D, 1, J)$  scheme with  $J$  given by the map  $\{1, \dots, D\} \ni x \mapsto 1$ , and on the other hand, every  $\text{GARK}(D, d, J)$  scheme can be written as an (artificially enlarged)  $\text{ARK}(D)$  scheme.

As an example illustrating the generality of the GARK formulation, we may consider the classical fractional-step- $\theta$  scheme introduced in [29] which is widely used for time integration of the Navier-Stokes equations. Given an additively split ordinary differential equation

$$u'(t) = f^{\{1\}}(u(t)) + f^{\{2\}}(u(t)),$$

the fractional-step- $\theta$  method reads

$$\begin{aligned} \frac{u_{n+\theta} - u_n}{\theta h} &= f^{\{1\}}(u_{n+\theta}) + f^{\{2\}}(u_n), \\ \frac{u_{n+1-\theta} - u_{n+\theta}}{(1-2\theta)h} &= f^{\{1\}}(u_{n+\theta}) + f^{\{2\}}(u_{n+1-\theta}), \\ \frac{u_{n+1} - u_{n+1-\theta}}{\theta h} &= f^{\{1\}}(u_{n+1}) + f^{\{2\}}(u_{n+1-\theta}). \end{aligned}$$

Setting  $U_{n,1} = u_n$ ,  $U_{n,2} = u_{n+\theta}$ ,  $U_{n,3} = u_{n+1-\theta}$ ,  $U_{n,4} = u_{n+1}$ , we may rewrite this scheme as a four-stage ARK scheme as defined in Equations (3) and (4) using the Butcher table

$$\begin{array}{c|cc} A^{\{1\}} & A^{\{2\}} & \\ \hline b^{\{1\}\top} & b^{\{2\}\top} & \end{array} = \begin{array}{cccccc} & 0 & & 0 & & \\ 0 & \theta & & \theta & 0 & \\ 0 & 1-\theta & 0 & \theta & 0 & 1-2\theta \\ 0 & 1-\theta & 0 & \theta & \theta & 0 & 1-\theta & 0 \\ \hline 0 & 1-\theta & 0 & \theta & \theta & 0 & 1-\theta & 0 \end{array}.$$

There are several ways to write the fractional-step- $\theta$  method as a GARK scheme. A trivial way of achieving that is to set  $d = 1$ ,  $A^{\{1,1\}} = A^{\{1\}}$ ,  $A^{\{1,2\}} = A^{\{2\}}$  and  $J(1) = J(2) = 1$ , which results in a  $\text{GARK}(2, 1, J)$  scheme with the generalized Butcher table

$$\begin{array}{c|c} A^{\{1,1\}} & A^{\{1,2\}} \\ \hline b^{\{1\}\top} & b^{\{2\}\top} \end{array}.$$

Another possible representation of the fractional-step- $\theta$  method within the GARK framework that makes use of different GARK stages is given by setting  $U_{n,1}^{\{1\}} = u_{n+\theta}$ ,  $U_{n,2}^{\{1\}} = u_{n+1}$ ,  $U_{n,1}^{\{2\}} = u_n$ ,  $U_{n,2}^{\{2\}} = u_{n+1-\theta}$ . With these GARK stage values the scheme can be written as a  $\text{GARK}(2, 2, \text{id})$  scheme with the corresponding generalized Butcher table

$$\begin{array}{c|cc} A^{\{1,1\}} & A^{\{1,2\}} & \\ \hline A^{\{2,1\}} & A^{\{2,2\}} & \\ \hline b^{\{1\}\top} & b^{\{2\}\top} & \end{array} = \begin{array}{cccc} \theta & 0 & \theta & 0 \\ 1-\theta & \theta & \theta & 1-\theta \\ 0 & 0 & 0 & 0 \\ \hline 1-\theta & 0 & \theta & 1-2\theta \\ \hline 1-\theta & \theta & \theta & 1-\theta \end{array}.$$

This formulation gets rid of the zero columns of the above ARK/trivial GARK formulation of the scheme.

GARK schemes which assign to each term of the right hand side of Equation (1) a specific stage value are particularly convenient, as the general definition further simplifies.

**Definition 3.** GARK schemes with separate stage value for each right hand side term of a  $D$ -times partitioned initial value problem (see Definition 1) result from the choice  $d = D$  and  $J = \text{id}$  within the general GARK scheme. They are given by

$$\mathbf{u}_{n,i}^{\{q\}} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{j=1}^{s^{\{k\}}} a_{i,j}^{\{q,k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,j}^{\{k\}}), \quad i = 1, \dots, s^{\{q\}}, \quad q = 1, \dots, D \quad (7)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h \sum_{k=1}^D \sum_{i=1}^{s^{\{k\}}} b_i^{\{k\}} \mathbf{f}^{\{k\}}(\mathbf{u}_{n,i}^{\{k\}}). \quad (8)$$

We denote this scheme by  $\text{GARK}(D) = \text{GARK}(D, D, \text{id})$ .

### 3. Multirate GARK (MGARK) Schemes

For splittings into two time scales, i.e., for  $D = 2$ , which is a reasonable starting point for problems in the context of fluid–structure interactions, we now define the multirate generalized-structure additively partitioned Runge-Kutta (MGARK) schemes as follows.

**Definition 4** (MGARK scheme). An MGARK scheme using  $N \in \mathbb{N}$  micro steps of step size  $h_\lambda \leq H$  ( $\lambda = 1, \dots, N$ ) for fast components and a macro step of step size  $H$  for slow components is given by

$$\mathbf{u}_{n,i}^{\{s\}} = \mathbf{u}_n + \sum_{\lambda=1}^N h_\lambda \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{s,f,\lambda\}} \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\lambda\}}) + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{s,s\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,j}^{\{s\}}), \quad i = 1, \dots, s^{\{s\}}, \quad (9)$$

$$\begin{aligned} \mathbf{u}_{n,i}^{\{f,\lambda\}} &= \mathbf{u}_n + \sum_{\ell=1}^{\lambda-1} h_\ell \sum_{j=1}^{s^{\{f\}}} b_j^{\{f\}} \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\ell\}}) + h_\lambda \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{f,f\}} \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\lambda\}}) \\ &\quad + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{f,s,\lambda\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,j}^{\{s\}}), \quad i = 1, \dots, s^{\{f\}}, \quad \lambda = 1, \dots, N, \end{aligned} \quad (10)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \sum_{\lambda=1}^N h_\lambda \sum_{i=1}^{s^{\{f\}}} b_i^{\{f\}} \mathbf{f}^{\{f\}}(\mathbf{u}_{n,i}^{\{f,\lambda\}}) + H \sum_{i=1}^{s^{\{s\}}} b_i^{\{s\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,i}^{\{s\}}). \quad (11)$$

Thus, one step of an MGARK scheme includes one macro step of step size  $H$  as well as all of the  $N$  micro steps having potentially different step sizes  $h_\lambda$ . In order to apply the order theory for GARK schemes as in Theorem 1 to MGARK schemes, we would like to reformulate the MGARK method as a GARK scheme using separate stage values for fast and slow components as in Definition 3. As the classical form of GARK schemes depends on constant micro step sizes within one macro step, the step size of the micro steps will be denoted by  $h_\lambda = m_\lambda H$ ,  $\lambda = 1, \dots, N$  using the coefficients  $m_\lambda \in (0, 1]$ . Substituting these step sizes into Equations (9)–(11) and considering the coefficients  $m_\lambda$  as modifications to the Runge-Kutta coefficients  $\mathbf{A}^{\{f,f\}}$ ,  $\mathbf{A}^{\{s,f,\lambda\}}$  and  $\mathbf{b}^{\{f\}}$ , we obtain

$$\mathbf{u}_{n,i}^{\{s\}} = \mathbf{u}_n + H \sum_{\lambda=1}^N \sum_{j=1}^{s^{\{f\}}} \left( m_\lambda a_{i,j}^{\{s,f,\lambda\}} \right) \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\lambda\}}) + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{s,s\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,j}^{\{s\}}), \quad i = 1, \dots, s^{\{s\}} \quad (12)$$

$$\begin{aligned} \mathbf{u}_{n,i}^{\{f,\lambda\}} &= \mathbf{u}_n + H \sum_{\ell=1}^{\lambda-1} \sum_{j=1}^{s^{\{f\}}} \left( m_\ell b_j^{\{f\}} \right) \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\ell\}}) + H \sum_{j=1}^{s^{\{f\}}} \left( m_\lambda a_{i,j}^{\{f,f\}} \right) \mathbf{f}^{\{f\}}(\mathbf{u}_{n,j}^{\{f,\lambda\}}) \\ &\quad + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{f,s,\lambda\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,j}^{\{s\}}), \quad i = 1, \dots, s^{\{f\}}, \quad \lambda = 1, \dots, N, \end{aligned} \quad (13)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + H \sum_{\lambda=1}^N \sum_{i=1}^{s^{\{f\}}} \left( m_\lambda b_i^{\{f\}} \right) \mathbf{f}^{\{f\}}(\mathbf{u}_{n,i}^{\{f,\lambda\}}) + H \sum_{i=1}^{s^{\{s\}}} b_i^{\{s\}} \mathbf{f}^{\{s\}}(\mathbf{u}_{n,i}^{\{s\}}). \quad (14)$$

From Equations (12)–(14) it is in fact obvious that the MGARK scheme is equivalent to a GARK(2) scheme with the corresponding generalized Butcher table

$$\begin{array}{c|c} \underline{A}^{\{f,f\}} & \underline{A}^{\{f,s\}} \\ \hline \underline{A}^{\{s,f\}} & \underline{A}^{\{s,s\}} \\ \hline \underline{b}^{\{f\}\top} & \underline{b}^{\{s\}\top} \end{array} = \begin{array}{ccc|c} m_1 A^{\{f,f\}} & & & A^{\{f,s,1\}} \\ m_1 \mathbf{1} b^{\{f\}\top} & m_2 A^{\{f,f\}} & & A^{\{f,s,2\}} \\ \vdots & \vdots & \ddots & \vdots \\ m_1 \mathbf{1} b^{\{f\}\top} & m_2 \mathbf{1} b^{\{f\}\top} & \dots & m_N A^{\{f,f\}} \\ \hline m_1 A^{\{s,f,1\}} & m_2 A^{\{s,f,2\}} & \dots & m_N A^{\{s,f,N\}} \\ m_1 b^{\{f\}\top} & m_2 b^{\{f\}\top} & \dots & m_N b^{\{f\}\top} \end{array} \begin{array}{c} A^{\{f,s,1\}} \\ A^{\{f,s,2\}} \\ \vdots \\ A^{\{f,s,N\}} \\ A^{\{s,s\}} \\ b^{\{s\}\top} \end{array} \quad (15)$$

containing the coefficients  $m_\lambda$ .

### 3.1. Order Conditions for MGARK Schemes with Variable Micro Steps

The order conditions of GARK schemes have been explicitly derived by Sandu and Günther up to schemes of 4th order, see [25]. The derivation is based on the application of the theory of  $N$ -trees by Araújo et al. [30] on the representation of the GARK scheme in Equations (5) and (6). In order to obtain simple formulations for these order conditions, we denote by “ $\cdot$ ” the usual scalar product of two vectors as well as the usual matrix vector product while the symbol “ $*$ ” denotes a component-wise multiplication of two vectors.

**Theorem 1** (GARK order conditions; Sandu and Günther ([25], Theorem 2.6)). *For  $i, j \in \{1, \dots, D\}$ , we define  $\mathbf{c}^{\{J(i),j\}} := A^{\{J(i),j\}} \cdot \mathbf{1}$ . We have the following order conditions for the GARK( $D, d, J$ ) scheme as in Definition 2, where  $i, j, k \in \{1, \dots, D\}$ :*

*First order*

$$\mathbf{b}^{\{i\}\top} \cdot \mathbf{1} = 1 \quad (16)$$

*Second order*

$$\mathbf{b}^{\{i\}\top} \cdot \mathbf{c}^{\{J(i),j\}} = \frac{1}{2}, \quad (17)$$

*Third order*

$$\mathbf{b}^{\{i\}\top} \cdot \left( \mathbf{c}^{\{J(i),j\}} * \mathbf{c}^{\{J(i),k\}} \right) = \frac{1}{3}, \quad (18)$$

$$\mathbf{b}^{\{i\}\top} \cdot A^{\{J(i),j\}} \cdot \mathbf{c}^{\{J(j),k\}} = \frac{1}{6}, \quad (19)$$

*Fourth order*

$$\mathbf{b}^{\{i\}\top} \cdot \left( \mathbf{c}^{\{J(i),j\}} * \mathbf{c}^{\{J(i),k\}} * \mathbf{c}^{\{J(i),l\}} \right) = \frac{1}{4}, \quad (20)$$

$$\left( \mathbf{b}^{\{i\}} * \mathbf{c}^{\{J(i),j\}} \right)^\top \cdot A^{\{J(i),k\}} \cdot \mathbf{c}^{\{J(k),l\}} = \frac{1}{8}, \quad (21)$$

$$\mathbf{b}^{\{i\}\top} \cdot A^{\{J(i),j\}} \cdot \left( \mathbf{c}^{\{J(j),k\}} * \mathbf{c}^{\{J(j),l\}} \right) = \frac{1}{12}, \quad (22)$$

$$\mathbf{b}^{\{i\}\top} \cdot A^{\{J(i),j\}} \cdot A^{\{J(k),l\}} \cdot \mathbf{c}^{\{J(k),l\}} = \frac{1}{24}. \quad (23)$$

Now, the application of the GARK order conditions to the Butcher table given in Equation (15) yields the following assertion.



**Theorem 2** (MGARK order conditions). We define  $\mathbf{c}^{\{f\}} := \mathbf{A}^{\{f,f\}} \cdot \mathbf{1}$ ,  $\mathbf{c}^{\{s\}} := \mathbf{A}^{\{s,s\}} \cdot \mathbf{1}$ ,  $\mathbf{c}^{\{f,s,\lambda\}} := \mathbf{A}^{\{f,s,\lambda\}} \cdot \mathbf{1}$ ,  $\mathbf{c}^{\{s,f,\lambda\}} := \mathbf{A}^{\{s,f,\lambda\}} \cdot \mathbf{1}$ . The following conditions are sufficient to obtain the desired order of an MGARK scheme.

*First order*

$$\mathbf{b}^{\{f\}\top} \cdot \mathbf{1} = 1, \quad (24)$$

$$\mathbf{b}^{\{s\}\top} \cdot \mathbf{1} = 1, \quad (25)$$

$$\sum_{\lambda=1}^N m_{\lambda} = 1 \quad (26)$$

*Second order*

$$\mathbf{b}^{\{f\}\top} \cdot \mathbf{c}^{\{f\}} = \frac{1}{2}, \quad (27)$$

$$\mathbf{b}^{\{s\}\top} \cdot \mathbf{c}^{\{s\}} = \frac{1}{2}, \quad (28)$$

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{f,s,\lambda\}} = \frac{1}{2}, \quad (29)$$

$$\mathbf{b}^{\{s\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{s,f,\lambda\}} = \frac{1}{2} \quad (30)$$

*Third order*

$$\mathbf{b}^{\{f\}\top} \cdot \text{diag}(\mathbf{c}^{\{f\}}) \cdot \mathbf{c}^{\{f\}} = \frac{1}{3}, \quad (31)$$

$$\mathbf{b}^{\{s\}\top} \cdot \text{diag}(\mathbf{c}^{\{s\}}) \cdot \mathbf{c}^{\{s\}} = \frac{1}{3}, \quad (32)$$

$$\mathbf{b}^{\{f\}\top} \cdot \mathbf{A}^{\{f,f\}} \cdot \mathbf{c}^{\{f\}} = \frac{1}{6}, \quad (33)$$

$$\mathbf{b}^{\{s\}\top} \cdot \mathbf{A}^{\{s,s\}} \cdot \mathbf{c}^{\{s\}} = \frac{1}{6}, \quad (34)$$

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{f,s,\lambda\}} \cdot \mathbf{c}^{\{s\}} = \frac{1}{6}, \quad (35)$$

$$\mathbf{b}^{\{s\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{s,f,\lambda\}} \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} \right) = \frac{1}{6} \quad (36)$$

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \text{diag}(\mathbf{c}^{\{f,s,\lambda\}}) \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} \right) = \frac{1}{3}, \quad (37)$$

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \text{diag}(\mathbf{c}^{\{f,s,\lambda\}}) \cdot \mathbf{c}^{\{f,s,\lambda\}} = \frac{1}{3}, \quad (38)$$

$$\mathbf{b}^{\{s\}\top} \cdot \text{diag} \left( \sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{s,f,\lambda\}} \right) \cdot \mathbf{c}^{\{s\}} = \frac{1}{3}, \quad (39)$$

$$\mathbf{b}^{\{s\}\top} \cdot \text{diag} \left( \sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{s,f,\lambda\}} \right) \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{s,f,\lambda\}} = \frac{1}{3}, \quad (40)$$

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{c}^{\{f,s,\ell\}} + m_{\lambda} \mathbf{A}^{\{f,f\}} \cdot \mathbf{c}^{\{f,s,\lambda\}} \right) = \frac{1}{6}, \quad (41)$$



and

$$\mathbf{b}^{\{f\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{f,s,\lambda\}} \cdot \sum_{\ell=1}^N m_{\ell} \mathbf{c}^{\{s,f,\ell\}} = \frac{1}{6}, \quad (42)$$

$$\mathbf{b}^{\{s\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{s,s\}} \cdot \mathbf{c}^{\{s,f,\lambda\}} = \frac{1}{6}, \quad (43)$$

$$\mathbf{b}^{\{s\}\top} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{s,f,\lambda\}} \cdot \mathbf{c}^{\{f,s,\lambda\}} = \frac{1}{6}. \quad (44)$$

**Remark 1.** In [26], order conditions for multirate GARK schemes using constant micro step sizes within the macro step are presented. These order conditions require that all micro steps taken in the same macro step have the same step size. Of course, micro step sizes may change after the macro step is completed. The conditions given in Theorem 2 are thus more general because they allow for varying micro step sizes also within the macro step. In the special case of constant micro step sizes per macro step we recover the order conditions in [26].

**Proof of Theorem 2.** We apply the order conditions of Theorem 1 to the Butcher table of the MGARK scheme given in Equation (15). Herein, the conditions have to be applied to the underlined quantities. The assertion is proven if each order condition of Theorem 1 holds due to an equivalent order condition or due to a subset of order conditions formulated in Theorem 2. First, we remark when requiring the conditions (24), (27), (31) and (33) we assume in particular that the corresponding fast base method ( $\mathbf{A}^{\{f,f\}}, \mathbf{b}^{\{f\}\top}$ ) is at least of the same order as the coupled scheme.

- First order conditions: If Equations (24)–(26) are fulfilled, we directly obtain

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}\top} \cdot \mathbf{1} = 1, \quad \mathbf{b}^{\{s\}\top} \cdot \mathbf{1} = 1,$$

due to the third condition  $\sum_{\lambda=1}^N m_{\lambda} = 1$ . Hence, the Equation (16) are fulfilled for  $i \in \{f, s\}$ .

- Second order conditions: Equation (17) for  $(i, j) = (f, f)$  can be rewritten as

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}\top} \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}\top} \right) = \frac{1}{2} \iff 1 = \sum_{\lambda=1}^N \left( \sum_{\ell=1}^{\lambda-1} 2m_{\lambda} m_{\ell} + m_{\lambda}^2 \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^2,$$

using Equations (24) and (27). The corresponding condition on the right hand side is then fulfilled due to Equation (26). Equation (17) for  $(i, j) = (f, s)$  and for  $(i, j) = (s, f)$  is directly equivalent to Equations (29) and (30), respectively, while  $(i, j) = (s, s)$  leads to Equation (28).

- Third order conditions (Part 1): Equation (18) for  $(i, j, k) = (f, f, f)$  rewrites as

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}\top} \cdot \left( \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}\top} \right) * \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}\top} \right) \right) = \frac{1}{3}.$$

Using Equations (24), (27) and (31) this is equivalent to

$$1 = \sum_{\lambda=1}^N \left( m_{\lambda}^3 + 3m_{\lambda} \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \right)^2 + 3m_{\lambda}^2 \sum_{\ell=1}^{\lambda-1} m_{\ell} \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^3.$$

As before, this condition is valid due to Equation (26). For  $(i, j, k) = (f, s, f)$  and  $(i, j, k) = (f, f, s)$ , Equation (18) yields

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}\top} \cdot \left( \mathbf{c}^{\{f,s,\lambda\}} * \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}\top} \right) \right) = \frac{1}{3}.$$

Due to the equality  $\mathbf{a} * \mathbf{a} = \text{diag}(\mathbf{a}) \cdot \mathbf{a}$  for an arbitrary vector  $\mathbf{a}$ , this condition is equivalent to Equation (37). For  $(i, j, k) = (f, s, s)$ , Equation (18) directly rewrites as Equation (38). The two choices  $(i, j, k) = (s, f, s)$  and  $(i, j, k) = (s, s, f)$  for Equation (18) lead to the Equation (39). The remaining choices of  $(i, j, k)$  in Equation (18) finally yield the condition (40) for  $(i, j, k) = (s, f, f)$  and the condition (32) for  $(i, j, k) = (s, s, s)$ .

- Third order conditions (Part 2): Equation (19) for  $(i, j, k) = (f, f, f)$  can be rewritten as

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}} \cdot \left[ m_1 \mathbf{1} \cdot \mathbf{b}^{\{f\}} \cdot \dots, m_{\lambda-1} \mathbf{1} \cdot \mathbf{b}^{\{f\}} \cdot \dots, m_{\lambda} \mathbf{A}^{\{f,f\}}, \mathbf{0}, \dots, \mathbf{0} \right] \cdot \left[ \sum_{\ell=1}^{v-1} m_{\ell} \mathbf{1} + m_v \mathbf{c}^{\{f\}} \right]_{v=1, \dots, N} = \frac{1}{6}.$$

Using Equations (24), (27) and (33) this is equivalent to

$$1 = \sum_{\lambda=1}^N \left( m_{\lambda}^3 + 3m_{\lambda} \sum_{\ell=1}^{\lambda-1} m_{\ell}^2 + 6m_{\lambda} \sum_{\ell=1}^{\lambda-1} \sum_{\mu=1}^{\ell-1} m_{\ell} m_{\mu} + 3m_{\lambda}^2 \sum_{\ell=1}^{\lambda-1} m_{\ell} \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^3,$$

i.e., validity follows from Equation (26). For  $(i, j, k) = (f, f, s)$ , Equation (19) rewrites as

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{b}^{\{f\}} \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} \mathbf{b}^{\{f\}} \cdot \mathbf{c}^{\{f,s,\ell\}} + m_{\lambda} \mathbf{A}^{\{f,f\}} \cdot \mathbf{c}^{\{f,s,\lambda\}} \right) = \frac{1}{6}.$$

Considering Equation (24), this is equivalent to Equation (41). Furthermore, for  $(i, j, k) = (f, s, f)$ ,  $(i, j, k) = (f, s, s)$ ,  $(i, j, k) = (s, f, f)$ ,  $(i, j, k) = (s, f, s)$ ,  $(i, j, k) = (s, s, f)$  and  $(i, j, k) = (s, s, s)$ , Equation (19) rewrites as Equations (42), (35), (36), (44), (43) and (34), respectively.  $\square$

The order conditions as in Theorem 2 include the assumptions that the fast base method  $(\mathbf{A}^{\{f,f\}}, \mathbf{b}^{\{f\}})$  is at least of the same order as the coupled scheme and that the sum of micro step sizes is equal to the size of the macro step. This assumption is incorporated into the Equations (24), (27), (31), (33), respectively, depending on the considered order, as well as the Equation (26). Thus, the MGARK order conditions of Theorem 2 are sufficient, but not necessary. In fact, this degree of freedom within the MGARK scheme can already be observed in the method formulation given in Equations (12)–(14) as well as on the left side of the right hand side Butcher table in Equation (15).

However, in this work, we will only consider MGARK schemes which fulfill the above assumptions. In this case, the conditions in Theorem 2 become necessary and the order of the complete MGARK scheme is restricted by the minimum order of the respective base methods. In order to achieve this maximum order, the coupling Equations (29) and (30), (35) and (36), (37)–(44) have to be fulfilled. This can be achieved by a suitable choice of the coupling coefficients  $\underline{\mathbf{A}}^{\{f,s\}}$  and  $\underline{\mathbf{A}}^{\{s,f\}}$ .

In addition, we would prefer internally consistent MGARK schemes as they already fulfill a significant number of the order conditions in Theorem 2 which leads to a considerable simplification. According to [25], internally consistent GARK schemes are defined as follows.

**Definition 5.** A GARK( $D, d, J$ ) scheme given by Definition 2 is internally consistent, if

$$\mathbf{A}^{\{q,1\}} \cdot \mathbf{1} = \mathbf{A}^{\{q,2\}} \cdot \mathbf{1} = \dots = \mathbf{A}^{\{q,D\}} \cdot \mathbf{1} =: \mathbf{c}^{\{q\}}$$

holds for all  $q = 1, \dots, d$ .

According to the notation of MGARK schemes used in Definition 4 and specified by the generalized Butcher table in Equation (15), the above definition of internal consistency takes the form  $\underline{\mathbf{A}}^{\{f,f\}} \cdot \mathbf{1} = \underline{\mathbf{A}}^{\{f,s\}} \cdot \mathbf{1} = \underline{\mathbf{c}}^{\{f\}}$  and  $\underline{\mathbf{A}}^{\{s,f\}} \cdot \mathbf{1} = \underline{\mathbf{A}}^{\{s,s\}} \cdot \mathbf{1} = \underline{\mathbf{c}}^{\{s\}}$  in case of an MGARK scheme. For internally consistent MGARK schemes, the order conditions in Theorem 2 simplify to

**Theorem 3** (Order conditions for internally consistent MGARK schemes). *An internally consistent MGARK scheme as specified by Definitions 4 and 5 is at least of first order if Equations (24)–(26) hold. The scheme is at least of second order, if in addition to the first order conditions the Equations (27) and (28) hold. The scheme is at least of third order, if in addition to the second order conditions the third order conditions for the individual schemes, i.e., Equations (31)–(34) as well as the coupling Equations (35) and (36) are fulfilled.*

**Proof.** Let  $\mathbf{c}^{\{f,s,\lambda\}} := \mathbf{A}^{\{f,s,\lambda\}} \cdot \mathbf{1}$ ,  $\mathbf{c}^{\{s,f,\lambda\}} := \mathbf{A}^{\{s,f,\lambda\}} \cdot \mathbf{1}$ . For an internally consistent MGARK scheme, Definition 5 and the subsequent remark yield

$$\begin{aligned}\underline{\mathbf{A}}^{\{f,f\}} \cdot \mathbf{1} &= \underline{\mathbf{A}}^{\{f,s\}} \cdot \mathbf{1} = \underline{\mathbf{c}}^{\{f\}}, \\ \underline{\mathbf{A}}^{\{s,f\}} \cdot \mathbf{1} &= \underline{\mathbf{A}}^{\{s,s\}} \cdot \mathbf{1} = \underline{\mathbf{c}}^{\{s\}}.\end{aligned}$$

These two above conditions are equivalent to

$$\sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} = \mathbf{c}^{\{f,s,\lambda\}} \quad \text{for } \lambda = 1, \dots, N, \quad (45)$$

$$\sum_{\lambda=1}^N m_{\lambda} \mathbf{c}^{\{s,f,\lambda\}} = \mathbf{c}^{\{s\}}. \quad (46)$$

We will show that the sufficient conditions stated in Theorem 2 are fulfilled. For first order, nothing has to be shown as the above first order conditions are precisely the ones in Theorem 2.

- Second order conditions: Only the validity of Equations (29) and (30) needs to be proven. The Equation (29), combined with Equations (24), (27) and (45), reduces to

$$\sum_{\lambda=1}^N \left( \sum_{\ell=1}^{\lambda-1} 2m_{\lambda} m_{\ell} + m_{\lambda}^2 \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^2 = 1,$$

which is fulfilled due to Equation (26). Validity of the Equation (30) directly follows from Equations (28) and (46).

- Third order conditions (Part 1): The validity of Equations (37)–(40) needs to be shown. Both of the Equations (37) and (38), in combination with Equation (45), reduce to

$$\mathbf{b}^{\{f\}T} \cdot \sum_{\lambda=1}^N m_{\lambda} \operatorname{diag} \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} \right) \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} \right) = \frac{1}{3}.$$

Due to Equations (24), (27) and (31), the above equations further reduce to

$$\sum_{\lambda=1}^N \left( m_{\lambda}^3 + 3m_{\lambda} \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \right)^2 + 3m_{\lambda}^2 \sum_{\ell=1}^{\lambda-1} m_{\ell} \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^3 = 1,$$

which is fulfilled due to Equation (26). Furthermore, the validity of the Equations (39) and (40) immediately follows from Equations (32) and (46).

- Third order conditions (Part 2): Finally, the validity of Equations (41)–(44) remains to be shown. Using Equations (24), (27), (33) and (45), Equation (41) reduces to

$$\sum_{\lambda=1}^N \left( m_{\lambda}^3 + 3m_{\lambda} \sum_{\ell=1}^{\lambda-1} m_{\ell}^2 + 6m_{\lambda} \sum_{\ell=1}^{\lambda-1} \sum_{\mu=1}^{\ell-1} m_{\ell} m_{\mu} + 3m_{\lambda}^2 \sum_{\ell=1}^{\lambda-1} m_{\ell} \right) = \left( \sum_{\lambda=1}^N m_{\lambda} \right)^3 = 1$$

and is thus fulfilled due to Equation (26). The validity of the Equation (43) follows from Equations (34) and (46). Considering Equations (45) and (46), Equations (42) and (44) are equivalent to Equations (35) and (36), respectively.  $\square$

The only coupling Equation (26) of the general MGARK scheme is thus required also for internally consistent MGARK schemes. However, the two second order coupling conditions are automatically fulfilled for internally consistent MGARK schemes without requiring additional conditions. The large number of third order coupling conditions of the general MGARK scheme given in Theorem 2 now reduces to only the two coupling Equations (35) and (36).

Concerning the concrete choice of an MGARK scheme in practice, we notice that the computational effort needs to be low enough in order to not exceed the cost of solving the coupled system with constant micro step sizes per macro step. Coupling the calculation of the slow components to all micro steps seems impractical from this point of view. The reasonable partial decoupling of the macro step calculation from the micro step computations has already been suggested by Günther et al. [18,26]. Their approach consists in coupling the macro step only to the first micro step. Formally this is realized by setting  $A^{\{s,f,\lambda\}} = \mathbf{0}$  for  $\lambda > 1$ .

### 3.2. Specific Coupling Conditions Retaining the Order of the Base Schemes

In the following, we consider specific choices of base methods  $(A^{\{f,f\}}, b^{\{f\}})$  and  $(A^{\{s,s\}}, b^{\{s\}})$  and the corresponding choice of coupling conditions which retain the order of the base methods. Herein, we always assume that the Equation (26) holds, i.e., we have  $\sum_{\lambda=1}^N h_{\lambda} = H$ . We consider the following cases which are ordered from general to specific ones.

- The base methods both have  $s$  stages and the macro step may be coupled to all of the  $N$  micro steps.
- The base methods both have  $s$  stages and the macro step is only coupled to the first of the  $N$  micro steps.
- The base methods may be either explicit first stage, singly diagonal implicit Runge-Kutta (ESDIRK) or explicit methods having 4 stages and the macro step is only coupled to the first of the  $N$  micro steps. With  $c^{\{f\}} = A^{\{f,f\}} \cdot \mathbf{1}$ ,  $c^{\{s\}} = A^{\{s,s\}} \cdot \mathbf{1}$  we additionally assume  $c_2^{\{f\}} \neq c_3^{\{f\}}$ ,  $c_2^{\{s\}} \neq c_3^{\{s\}}$ ,  $b_4^{\{f\}} \neq 0$ ,  $b_4^{\{s\}} \neq 0$ .

From Theorem 2 we immediately obtain

**Corollary 1** (First order coupling). *For all of the cases (a)–(c), the MGARK scheme is at least of first order if this is true for the base methods—irrespective of the concrete choice of coupling.*

**Corollary 2** (Second order coupling). *If the coupling is realized as*

*Choice A*

$$A^{\{f,s,\lambda\}} = m_{\lambda} A^{\{f,f\}} + \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathcal{I} \quad (47)$$

and

$$A^{\{s,f,\lambda\}} = A^{\{s,s\}}, \quad \text{in case (a),} \quad (48)$$

or

$$A^{\{s,f,1\}} = \frac{1}{m_1} A^{\{s,s\}}, \quad A^{\{s,f,\lambda\}} = \mathbf{0}, \quad \lambda > 1, \quad \text{in cases (b) and (c),} \quad (49)$$

or

*Choice B*

$$a_{i,j}^{\{f,s,1\}} = \begin{cases} m_1 c_i^{\{f\}}, & \text{if } j = \min\{i-1, s^{\{s\}}\}, \\ 0, & \text{otherwise,} \end{cases} \quad (50)$$

$$a_{i,j}^{\{f,s,\lambda\}} = \begin{cases} \sum_{\ell=1}^{\lambda-1} m_{\ell} + m_{\lambda} c_i^{\{f\}}, & \text{if } j = \min\{i, s^{\{s\}}\}, \\ 0, & \text{otherwise,} \end{cases}, \quad \lambda > 1, \quad (51)$$

and

$$a_{i,j}^{\{s,f,1\}} = \begin{cases} \frac{1}{m_1} c_i^{\{s\}} & \text{if } j = \min\{i-1, s^{\{f\}}\} \\ 0 & \text{otherwise} \end{cases}, \quad A^{\{s,f,\lambda\}} = \mathbf{0}, \lambda > 1 \quad (52)$$

then the MGARK scheme is at least of second order, if this is true for the base methods.

**Proof.** With the Equation (47), the Equation (45) is fulfilled and with Equation (48) or (49), respectively, also Equation (46) holds, which is equivalent to the internal consistency of the MGARK scheme. Equations (50)–(52) obviously lead to an internally consistent MGARK scheme as well. According to Theorem 3, internally consistent MGARK schemes do not have to fulfill additional coupling conditions in order to achieve second order.  $\square$

**Corollary 3** (Third order coupling). *If in case (c) the coupling is realized as*

$$A^{\{f,s,\lambda\}} = \begin{bmatrix} \alpha_1^{\{\lambda\}} & 0 & 0 & 0 \\ \alpha_2^{\{\lambda\}} & 0 & 0 & 0 \\ 0 & \alpha_3^{\{\lambda\}} & 0 & 0 \\ 0 & \alpha_4^{\{\lambda\}} & \alpha_5^{\{\lambda\}} & 0 \end{bmatrix}, \quad \alpha_1^{\{\lambda\}} = \sum_{\ell=1}^{\lambda-1} m_\ell, \quad \alpha_2^{\{\lambda\}} = \alpha_1^{\{\lambda\}} + m_\lambda c_2^{\{f\}},$$

$$\alpha_3^{\{\lambda\}} = \alpha_1^{\{\lambda\}} + m_\lambda c_3^{\{f\}}, \quad \tilde{\alpha}_4 = \frac{1}{6} - \mathbf{b}^{\{f\}\top} \cdot \sum_{\ell=1}^{N-1} m_\ell A^{\{f,s,\ell\}} \cdot \mathbf{c}^{\{s\}}, \quad (53)$$

$$\alpha_4^{\{\lambda\}} = \begin{cases} 0, & \lambda < N, \\ \frac{\frac{1}{m_\lambda} \tilde{\alpha}_4 - b_3^{\{f\}} \alpha_3^{\{\lambda\}} c_2^{\{s\}} - b_4^{\{f\}} (\alpha_1^{\{\lambda\}} + m_\lambda c_4^{\{f\}}) c_3^{\{s\}}}{b_4^{\{f\}} (c_2^{\{s\}} - c_3^{\{s\}})}, & \lambda = N, \end{cases}$$

$$\alpha_5^{\{\lambda\}} = \alpha_1^{\{\lambda\}} + m_\lambda c_4^{\{f\}} - \alpha_4^{\{\lambda\}}$$

and

$$A^{\{s,f,1\}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 0 \\ 0 & \beta_3 & \beta_4 & 0 \end{bmatrix}, \quad \beta_1 = \frac{1}{m_1} c_2^{\{s\}}, \quad \beta_2 = \frac{1}{m_1} c_3^{\{s\}},$$

$$\beta_3 = \frac{\frac{1}{6m_1^2} - b_3^{\{s\}} \beta_2 c_2^{\{f\}} - \frac{1}{m_1} b_4^{\{s\}} c_4^{\{s\}} c_3^{\{f\}}}{b_4^{\{s\}} (c_2^{\{f\}} - c_3^{\{f\}})}, \quad \beta_4 = \frac{1}{m_1} c_4^{\{s\}} - \beta_3, \quad (54)$$

$$A^{\{s,f,\lambda\}} = \mathbf{0}, \lambda > 1, \quad (55)$$

the MGARK scheme is at least of third order, if this is true for the base methods.

**Proof.** First we show that the above coupling strategy leads to an internally consistent MGARK scheme, i.e., Equations (45)–(46) are fulfilled. Equation (45) is fulfilled due to

$$\mathbf{c}^{\{f,s,\lambda\}} = [\alpha_1^{\{\lambda\}}, \alpha_2^{\{\lambda\}}, \alpha_3^{\{\lambda\}}, \alpha_4^{\{\lambda\}} + \alpha_5^{\{\lambda\}}]^\top = \sum_{\ell=1}^{\lambda-1} m_\ell \mathbf{1} + m_\lambda [0, c_2^{\{f\}}, c_3^{\{f\}}, c_4^{\{f\}}]^\top = \sum_{\ell=1}^{\lambda-1} m_\ell \mathbf{1} + m_\lambda \mathbf{c}^{\{f\}},$$

and Equation (46) holds due to

$$\sum_{\lambda=1}^N m_\lambda \mathbf{c}^{\{s,f,\lambda\}} = m_1 [0, \beta_1, \beta_2, \beta_3 + \beta_4]^\top = [0, c_2^{\{s\}}, c_3^{\{s\}}, c_4^{\{s\}}]^\top = \mathbf{c}^{\{s\}}.$$

Hence, the validity of Equations (35) and (36) remains to be shown. We have

$$\begin{aligned}
 \mathbf{b}^{\{f\}T} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{f,s,\lambda\}} \cdot \mathbf{c}^{\{s\}} &= \sum_{\lambda=1}^N m_{\lambda} (b_3^{\{f\}} \alpha_3^{\{\lambda\}} c_2^{\{s\}} + b_4^{\{f\}} \alpha_4^{\{\lambda\}} c_2^{\{s\}} + b_4^{\{f\}} \alpha_5^{\{\lambda\}} c_3^{\{s\}}) \\
 &= \sum_{\lambda=1}^N m_{\lambda} \left( (b_3^{\{f\}} c_2^{\{s\}} + b_4^{\{f\}} c_3^{\{s\}}) \sum_{\ell=1}^{\lambda-1} m_{\ell} + m_{\lambda} b_3^{\{f\}} c_3^{\{f\}} c_2^{\{s\}} + b_4^{\{f\}} \alpha_4^{\{\lambda\}} (c_2^{\{s\}} - c_3^{\{s\}}) \right. \\
 &\quad \left. + m_{\lambda} b_4^{\{f\}} c_4^{\{f\}} c_3^{\{s\}} \right) \\
 &= \sum_{\lambda=1}^N m_{\lambda} \left( (b_3^{\{f\}} c_2^{\{s\}} + b_4^{\{f\}} c_3^{\{s\}}) \sum_{\ell=1}^{\lambda-1} m_{\ell} + m_{\lambda} b_3^{\{f\}} c_3^{\{f\}} c_2^{\{s\}} + m_{\lambda} b_4^{\{f\}} c_4^{\{f\}} c_3^{\{s\}} \right) \\
 &\quad + \tilde{\alpha}_4 - m_N \left( (b_3^{\{f\}} c_2^{\{s\}} + b_4^{\{f\}} c_3^{\{s\}}) \sum_{\ell=1}^{N-1} m_{\ell} + m_N b_3^{\{f\}} c_3^{\{f\}} c_2^{\{s\}} + m_N b_4^{\{f\}} c_4^{\{f\}} c_3^{\{s\}} \right) = \frac{1}{6},
 \end{aligned}$$

which proves Equation (35) due to the definition of  $\tilde{\alpha}_4$ . As the slow component is only coupled to the first micro step of the fast component, we have

$$\begin{aligned}
 \mathbf{b}^{\{s\}T} \cdot \sum_{\lambda=1}^N m_{\lambda} \mathbf{A}^{\{s,f,\lambda\}} \cdot \left( \sum_{\ell=1}^{\lambda-1} m_{\ell} \mathbf{1} + m_{\lambda} \mathbf{c}^{\{f\}} \right) &= m_1^2 \mathbf{b}^{\{s\}T} \cdot \mathbf{A}^{\{s,f,1\}} \cdot \mathbf{c}^{\{f\}} \\
 &= m_1^2 (b_3^{\{s\}} \beta_2 c_2^{\{f\}} + b_4^{\{s\}} \beta_3 c_2^{\{f\}} + b_4^{\{s\}} \beta_4 c_3^{\{f\}}) \\
 &= m_1 (b_3^{\{s\}} c_3^{\{s\}} c_2^{\{f\}} + m_1 b_4^{\{s\}} \beta_3 (c_2^{\{f\}} - c_3^{\{f\}}) + b_4^{\{s\}} c_4^{\{s\}} c_3^{\{f\}}) = \frac{1}{6},
 \end{aligned}$$

proving Equation (36).  $\square$

The advantage of the presented coupling strategies is given by the fact that neither the micro step size  $h_{\lambda}$  (and thus  $m_{\lambda}$ ) nor the final number  $N$  of micro steps per macro step needs to be known a priori. Only the macro step size  $H$  is provided while the micro step sizes may be varied adaptively in the process of taking micro steps. For each micro step, the coupling coefficients will be newly adapted to the micro step size. Switching on the coefficient  $\alpha_4^{\{N\}}$  in the last micro step ensures the coupling condition Equation (35). The number of micro steps is thereby irrelevant, the scheme only has to identify the last micro step. Since the slow component is only coupled to the fast component within the first micro step, the corresponding coupling condition Equation (36) is already fulfilled by the specific choice of the parameter  $\beta_3$  in the first micro step.

### 3.3. Application of MGARK to Coupled Problems of Fluid–Structure Interaction

For mechanical fluid–structure interaction, the fluid domain is time-dependent. In order to deal with such moving fluid domains, the arbitrary Lagrangian-Eulerian(ALE) formulation for the fluid variables  $\mathbf{u}: \Omega \times [0, T] \rightarrow \mathbb{R}^d$  on a time-dependent domain  $\Omega(t) = \bigcup_{i \in \mathcal{A}} \Omega_i(t)$  is used in this work. The ALE formulation is given by

$$\frac{d}{dt} \int_{\Omega_i(t)} \mathbf{u} \, d\mathbf{x} = - \oint_{\partial\Omega_i(t)} \mathcal{F}_{\text{ALE}}(\mathbf{u}, \dot{\mathbf{x}}) \cdot \mathbf{n} \, d\sigma, \quad \mathcal{F}_{\text{ALE}}(\mathbf{u}, \dot{\mathbf{x}}) \cdot \mathbf{n} = \mathcal{F}(\mathbf{u}) \cdot \mathbf{n} - \mathbf{u}(\dot{\mathbf{x}} \cdot \mathbf{n}),$$

see [1,31]. The fluid equations will then be discretized by a finite volume method based on the cell means of the conservative variables and cell volume, i.e.,

$$\bar{\mathbf{u}}_i = \frac{1}{V_i} \int_{\Omega_i} \mathbf{u} \, d\mathbf{x}, \quad V_i = \text{vol}(\Omega_i) = \int_{\Omega_i} 1 \, d\mathbf{x}.$$

Rewriting the discretized right hand side as a function  $\mathbf{R}(\bar{\mathbf{u}}, \mathbf{V}, \dot{\mathbf{x}})$ , which incorporates a numerical flux function, we obtain a system of ordinary differential equations of the form

$$(\mathbf{V}\bar{\mathbf{u}})' = -\mathbf{R}(\bar{\mathbf{u}}, \mathbf{V}, \dot{\mathbf{x}}), \quad \mathbf{V} = [V_i]_{i \in \mathcal{A}}, \quad \bar{\mathbf{u}} = [\bar{u}_i]_{i \in \mathcal{A}}, \quad \mathbf{V}\bar{\mathbf{u}} = [V_i \bar{u}_i]_{i \in \mathcal{A}}. \quad (56)$$

Herein, the fast components are given by the fluid variables, i.e.,  $\mathbf{u}^{\{\text{f}\}} = \mathbf{V}\bar{\mathbf{u}}$  and the slow components will be obtained from the deformation of a structure which influences the fluid domain and thus determines the cell sizes, i.e.,  $\mathbf{V} = \mathbf{V}(\mathbf{u}^{\{\text{s}\}})$ . Thus, Equation (56) may be written as

$$\mathbf{u}^{\{\text{f}\}}' = \mathbf{f}^{\{\text{f}\}}(\mathbf{u}^{\{\text{f}\}}, \mathbf{u}^{\{\text{s}\}}, \dot{\mathbf{x}}), \quad (57)$$

where  $\dot{\mathbf{x}}$  denotes grid velocities which will be determined by the so-called geometric conservation law (GCL). The GCL basically states that constant solutions  $\bar{\mathbf{u}} = \text{const.}$  have to be preserved by the discretization, i.e., constant states are exact solutions of Equation (56). This can be formulated by the condition

$$\bar{\mathbf{u}}\mathbf{V}' = -\mathbf{R}(\bar{\mathbf{u}}, \mathbf{V}, \dot{\mathbf{x}}) = \bar{\mathbf{u}}\bar{\mathbf{R}}(\dot{\mathbf{x}}) \iff \mathbf{V}' = \bar{\mathbf{R}}(\dot{\mathbf{x}}), \quad (58)$$

where  $\bar{\mathbf{R}}$  denotes the discretization of

$$\oint_{\partial\Omega(t)} \dot{\mathbf{x}} \cdot \mathbf{n} \, d\sigma$$

corresponding to  $\mathbf{R}$ , see [31]. The application of a Runge-Kutta scheme  $(\mathbf{A}^{\{\text{f},\text{f}\}}, \mathbf{b}^{\{\text{f}\}})$  with step size  $h$  to the system on the right hand side of Equation (58) yields

$$\mathbf{V}_{n,i} = \mathbf{V}_n + h \sum_{j=1}^{s^{\{\text{f}\}}} a_{i,j}^{\{\text{f},\text{f}\}} \bar{\mathbf{R}}(\dot{\mathbf{x}}_{n,j}), \quad i = 1, \dots, s^{\{\text{f}\}}, \quad (59)$$

$$\mathbf{V}_{n+1} = \mathbf{V}_n + h \sum_{i=1}^{s^{\{\text{f}\}}} b_i^{\{\text{f}\}} \bar{\mathbf{R}}(\dot{\mathbf{x}}_{n,i}). \quad (60)$$

In case of an explicit method with  $a_{i+1,i}^{\{\text{f},\text{f}\}} \neq 0$  for  $i = 1, \dots, s^{\{\text{f}\}} - 1$ , Equation (59) provides a successive determination of grid velocities  $\dot{\mathbf{x}}_{n,1}, \dots, \dot{\mathbf{x}}_{n,s^{\{\text{f}\}}-1}$ . In addition, if  $b_{s^{\{\text{f}\}}}^{\{\text{f}\}} \neq 0$ , the final grid velocities  $\dot{\mathbf{x}}_{n,s^{\{\text{f}\}}}$  can be calculated from Equation (60). Thus, the grid velocities are solely determined by the cell volumes at internal stage time levels  $t_n \mathbf{1} + hc^{\{\text{f}\}}$  and hence by the corresponding structure states. We have to keep in mind that the determination of  $\dot{\mathbf{x}}_{n,i}$  for  $i = 1, \dots, s^{\{\text{f}\}} - 1$  needs the cell volumes  $V_{n,i+1}$  at the successive Runge-Kutta stage  $i + 1$  following the current stage  $i$  while the velocities  $\dot{\mathbf{x}}_{n,s^{\{\text{f}\}}}$  are determined from the cell volumes  $V_{n+1}$  at the end of the Runge-Kutta step. The precise calculation of these values in the context of an MGARK scheme is given in Algorithm 1 and the corresponding description. As in [31] we may require Equations (59) and (60) to hold face-wise, i.e.,

$$(\mathbf{V}_{n,i} - \mathbf{V}_n)_E = h \sum_{j=1}^{s^{\{\text{f}\}}} a_{i,j}^{\{\text{f},\text{f}\}} (\dot{\mathbf{n}}_{n,j})_E, \quad i = 1, \dots, s^{\{\text{f}\}}, \quad (61)$$

$$(\mathbf{V}_{n+1} - \mathbf{V}_n)_E = h \sum_{i=1}^{s^{\{\text{f}\}}} b_i^{\{\text{f}\}} (\dot{\mathbf{n}}_{n,i})_E, \quad (62)$$

where  $(\dot{\mathbf{n}})_E = (\dot{\mathbf{x}} \cdot \mathbf{n})_E$  is the face integrated grid velocity and  $(\mathbf{V}_{n,i} - \mathbf{V}_n)_E$  and  $(\mathbf{V}_{n+1} - \mathbf{V}_n)_E$  denote the volume change due to the moving face  $E$  at the  $i$ -th Runge-Kutta stage and at time  $t^{n+1}$ , respectively.



In particular, in the situation of one spatial dimension, we obtain a relation to the grid point positions  $x_k$ ,

$$x_k^{n,i} - x_k^n = h \sum_{j=1}^{s\{f\}} a_{i,j}^{\{f,f\}} \dot{x}_k^{n,j}, \quad i = 1, \dots, s\{f\}, \quad (63)$$

$$x_k^{n+1} - x_k^n = h \sum_{i=1}^{s\{f\}} b_i^{\{f\}} \dot{x}_k^{n,i}. \quad (64)$$

In the context of fluid–structure interaction, we rewrite the MGARK method in the more convenient component partitioned form. In addition, we explicitly use the intermediate values  $\mathbf{u}_n^{\{f,\lambda\}}$  of the fast components at the intermediate micro step times and introduce the stage derivatives  $\mathbf{k}_{n,i}^{\{f,\lambda\}}$  and  $\mathbf{k}_{n,i}^{\{s\}}$  in order to avoid unnecessary function evaluations.

The MGARK scheme with  $N \in \mathbb{N}$  micro steps of step sizes  $h_\lambda \leq H$  ( $\lambda = 1, \dots, N$ ) for the fast components and a macro step of step size  $H$  for the slow component is then given by

$$\mathbf{u}_n^{\{f,0\}} = \mathbf{u}_n^{\{f\}}, \quad (65)$$

$$\mathbf{u}_n^{\{f,\lambda\}} = \mathbf{u}_n^{\{f,\lambda-1\}} + h_\lambda \sum_{i=1}^{s\{f\}} b_i^{\{f\}} \mathbf{k}_{n,i}^{\{f,\lambda\}}, \quad \lambda = 1, \dots, N, \quad (66)$$

$$\mathbf{u}_{n+1}^{\{f\}} = \mathbf{u}_n^{\{f,N\}}, \quad (67)$$

$$\mathbf{u}_{n+1}^{\{s\}} = \mathbf{u}_n^{\{s\}} + H \sum_{i=1}^{s\{s\}} b_i^{\{s\}} \mathbf{k}_{n,i}^{\{s\}}, \quad (68)$$

with stage derivatives

$$\mathbf{k}_{n,i}^{\{s\}} = \mathbf{f}^{\{s\}}(\mathbf{u}_{n,i}^{\{s,f\}}, \mathbf{u}_{n,i}^{\{s,s\}}), \quad i = 1, \dots, s\{s\}, \quad (69)$$

$$\mathbf{k}_{n,i}^{\{f,\lambda\}} = \mathbf{f}^{\{f\}}(\mathbf{u}_{n,i}^{\{f,f,\lambda\}}, \mathbf{u}_{n,i}^{\{f,s,\lambda\}}), \quad i = 1, \dots, s\{f\}, \lambda = 1, \dots, N, \quad (70)$$

based on the stage vectors

$$\mathbf{u}_{n,i}^{\{s,f\}} = \mathbf{u}_n^{\{f\}} + \sum_{\lambda=1}^N h_\lambda \sum_{j=1}^{s\{f\}} a_{i,j}^{\{s,f,\lambda\}} \mathbf{k}_{n,j}^{\{f,\lambda\}}, \quad i = 1, \dots, s\{s\}, \quad (71)$$

$$\mathbf{u}_{n,i}^{\{f,f,\lambda\}} = \mathbf{u}_n^{\{f,\lambda-1\}} + h_\lambda \sum_{j=1}^{s\{f\}} a_{i,j}^{\{f,f\}} \mathbf{k}_{n,j}^{\{f,\lambda\}}, \quad i = 1, \dots, s\{f\}, \lambda = 1, \dots, N, \quad (72)$$

$$\mathbf{u}_{n,i}^{\{f,s,\lambda\}} = \mathbf{u}_n^{\{s\}} + H \sum_{j=1}^{s\{s\}} a_{i,j}^{\{f,s,\lambda\}} \mathbf{k}_{n,j}^{\{s\}}, \quad i = 1, \dots, s\{f\}, \lambda = 1, \dots, N, \quad (73)$$

$$\mathbf{u}_{n,i}^{\{s,s\}} = \mathbf{u}_n^{\{s\}} + H \sum_{j=1}^{s\{s\}} a_{i,j}^{\{s,s\}} \mathbf{k}_{n,j}^{\{s\}}, \quad i = 1, \dots, s\{s\}. \quad (74)$$

In case of fluid–structure interaction problems, where the evaluation of the fluid right hand side depends on the grid velocities, Equation (70) needs to be replaced by

$$\mathbf{k}_{n,i}^{\{f,\lambda\}} = \mathbf{f}^{\{f\}}(\mathbf{u}_{n,i}^{\{f,f,\lambda\}}, \mathbf{u}_{n,i}^{\{f,s,\lambda\}}, \dot{\mathbf{x}}_{n,i}^{\{\lambda\}}), \quad \lambda = 1, \dots, N, \quad (75)$$

where the grid velocities  $\dot{\mathbf{x}}_{n,i}^{\{\lambda\}}$  need to be computed in accordance with the GCL.

We note that the MGARK scheme in the above form given in Equations (65)–(75) may be still be fully coupled, i.e., in order to compute the stage derivatives  $k_{n,i}^{\{f,\lambda\}}$  and  $k_{n,i}^{\{s\}}$  the iterative solution of the above coupled equations may be necessary.

Hence, in the following, we consider an efficient implementation of the MGARK scheme given in Equations (65)–(75) for the numerical solution of coupled fluid-structure interaction. As base methods, we choose an explicit method for the fast components, i.e., the fluid variables, as well as a diagonally implicit Runge-Kutta (DIRK) scheme for the slow components, i.e., the structure variables. For efficiency reasons, we couple the macro step only to the first micro step. In this way, the stage derivatives  $k_{n,i}^{\{s\}}$  of the slow component are calculated together with the corresponding stage derivatives of the fast components  $k_{n,i}^{\{f,1\}}$  in the first micro step. In the subsequent micro steps ( $\lambda > 1$ ), only the remaining stage derivatives  $k_{n,i}^{\{f,\lambda\}}$  of the fast components need to be calculated.

---

**Algorithm 1** MGARK algorithm for coupled problems with moving grids respecting the GCL.

---

```

1: procedure MGARK( $u_n^{\{f\}}, u_n^{\{s\}}, H, N, h$ )
2:    $u_n^{\{f,0\}} \leftarrow u_n^{\{f\}}$ 
3:   for  $\lambda \leftarrow 1$  to  $N$  do loop over micro steps
4:     for  $i \leftarrow 1$  to  $\max(s^{\{f\}}, s^{\{s\}})$  do loop over Runge-Kutta stages
5:       if  $\lambda = 1$  then
6:         if  $i \leq s^{\{s\}}$  then calculation of MGARK stage derivatives of the structure
7:            $U_{n,i}^{\{s,f\}} \leftarrow u_n^{\{f\}} + h_1 \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{s,f,1\}} k_{n,j}^{\{f,1\}}$ 
8:            $U_{n,i}^{\{s,s\}} \leftarrow u_n^{\{s\}} + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{s,s\}} k_{n,j}^{\{s\}}$ 
9:            $k_{n,i}^{\{s\}} \leftarrow f^{\{s\}}(U_{n,i}^{\{s,f\}}, U_{n,i}^{\{s,s\}})$ 
10:        end if
11:        if  $i = s^{\{s\}}$  then
12:           $u_{n+1}^{\{s\}} \leftarrow u_n^{\{s\}} + H \sum_{j=1}^{s^{\{s\}}} b_j^{\{s\}} k_{n,j}^{\{s\}}$  macro update of the structure
13:        end if
14:      end if
15:      if  $i \leq s^{\{f\}}$  then calculation of MGARK stage derivatives of the fluid
16:         $U_{n,i}^{\{f,f,\lambda\}} \leftarrow u_n^{\{f,\lambda-1\}} + h_\lambda \sum_{j=1}^{s^{\{f\}}} a_{i,j}^{\{f,f\}} k_{n,j}^{\{f,\lambda\}}$ 
17:         $U_{n,i}^{\{f,s,\lambda\}} \leftarrow u_n^{\{s\}} + H \sum_{j=1}^{s^{\{s\}}} a_{i,j}^{\{f,s,\lambda\}} k_{n,j}^{\{s\}}$ 
18:         $\dot{x}_{n,i}^{\{\lambda\}} \leftarrow \text{MESHVELOCITY}$  GCL-compliant grid velocities
19:         $k_{n,i}^{\{f,\lambda\}} \leftarrow f^{\{f\}}(U_{n,i}^{\{f,f,\lambda\}}, U_{n,i}^{\{f,s,\lambda\}}, \dot{x}_{n,i}^{\{\lambda\}})$ 
20:      end if
21:      if  $i = s^{\{f\}}$  then
22:         $u_n^{\{f,\lambda\}} \leftarrow u_n^{\{f,\lambda-1\}} + h_\lambda \sum_{j=1}^{s^{\{f\}}} b_j^{\{f\}} k_{n,j}^{\{f,\lambda\}}$  micro update of the fluid
23:      end if
24:    end for
25:  end for
26:   $u_{n+1}^{\{f\}} \leftarrow u_n^{\{f,N\}}$  macro update of the fluid
27:  return  $u_{n+1}^{\{f\}}, u_{n+1}^{\{s\}}$ 
28: end procedure

```

---

**Algorithm 2** Computation of GCL-compliant mesh velocity.

---

```

1: procedure MESHVELOCITY
2:   if  $i < s^{\{f\}}$  then not the last stage within current micro step
3:      $\tilde{\mathbf{u}}_{n,i}^{\{f,s,\lambda\}} \leftarrow \mathbf{u}_n^{\{s\}} + H \sum_{j=1}^{s^{\{s\}}} a_{i+1,j}^{\{f,s,\lambda\}} \mathbf{k}_{n,j}^{\{s\}}$ 
4:      $\dot{\mathbf{x}}_{n,i}^{\{\lambda\}} \leftarrow \text{Solve } \frac{1}{h_\lambda} (V(\tilde{\mathbf{u}}_{n,i}^{\{f,s,\lambda\}}) - V(\mathbf{u}_{n,1}^{\{f,s,\lambda\}})) = \sum_{j=1}^i a_{i+1,j}^{\{f,f\}} \bar{\mathbf{R}}(\dot{\mathbf{x}}_{n,j}^{\{\lambda\}}) \text{ for } \dot{\mathbf{x}}_{n,i}^{\{\lambda\}}$ 
5:   else
6:     if  $\lambda < N$  then last stage within current micro step, not the last micro step within current macro step
7:        $\tilde{\mathbf{u}}_{n,i}^{\{f,s,\lambda\}} \leftarrow \mathbf{u}_n^{\{s\}} + H \sum_{j=1}^{s^{\{s\}}} a_{1,j}^{\{f,s,\lambda+1\}} \mathbf{k}_{n,j}^{\{s\}}$ 
8:     else last stage within current micro step, last micro step within current macro step
9:        $\tilde{\mathbf{u}}_{n,i}^{\{f,s,\lambda\}} \leftarrow \mathbf{u}_{n+1}^{\{s\}}$ 
10:    end if
11:     $\dot{\mathbf{x}}_{n,i}^{\{\lambda\}} \leftarrow \text{Solve } \frac{1}{h_\lambda} (V(\tilde{\mathbf{u}}_{n,i}^{\{f,s,\lambda\}}) - V(\mathbf{u}_{n,1}^{\{f,s,\lambda\}})) = \sum_{j=1}^i b_j^{\{f\}} \bar{\mathbf{R}}(\dot{\mathbf{x}}_{n,j}^{\{\lambda\}}) \text{ for } \dot{\mathbf{x}}_{n,i}^{\{\lambda\}}$ 
12:  end if
13:  return  $\dot{\mathbf{x}}_{n,i}^{\{\lambda\}}$ 
14: end procedure

```

---

Algorithms 1 and 2 summarize the concrete procedure for the implementation of the MGARK scheme for fluid–structure interaction including GCL-compliant grid velocities. The preservation of the GCL is guaranteed by computing the grid velocities in accordance with Equations (59) and (60). In Algorithm 1, line 18, the velocities are determined by the procedure MESHVELOCITY defined in Algorithm 2.

The efficiency and practicability of the scheme formulated in the above Algorithms 1 and 2 requires additional conditions concerning the chosen base methods and the coupling coefficients.

1. The decision to couple the structure macro step solely to the first fluid micro step for efficiency reasons reduces the second condition Equation (46) of internal consistency to

$$m_1 \mathbf{c}^{\{s,f,1\}} = \mathbf{c}^{\{s\}}. \quad (76)$$

- In order to access only the already determined fluid stage derivatives of the first micro step when calling Algorithm 1, line 7 the condition  $a_{i,j}^{\{s,f,1\}} = 0$  for  $i \leq j$  is necessary, in particular, we need to set  $a_{1,j}^{\{s,f,1\}} = 0$  for  $j = 1, \dots, s^{\{f\}}$ . This does only agree with Equation (76) if  $c_1^{\{s\}} = 0$  holds, i.e., if the first stage of the slow base method  $(\mathbf{A}^{\{s,s\}}, \mathbf{b}^{\{s\}})$  is an explicit stage. This requires using an E(S)DIRK scheme instead of an (S)DIRK scheme if internal consistency is to be guaranteed. However, without internal consistency, determining suitable coupling coefficients would require significantly more effort, as can be seen in Theorems 2 and 3.
2. In order to access only the already determined structure stage derivatives when calling Algorithm 1, line 17, the condition  $a_{i,j}^{\{f,s,1\}} = 0$  for  $i < j$  is necessary. For the coefficients  $\mathbf{A}^{\{f,s,\lambda\}}$  with  $\lambda > 1$  no additional conditions have to be fulfilled as at the end of the first micro step, all structure stage derivatives have been calculated and can be accessed.
  3. The construction of suitable structure states for the determination of GCL-compliant mesh velocities according to Algorithm 2 poses additional restrictions to the choice of the  $(f, s)$  coupling parameters. From line 3, we obtain the stronger condition  $a_{i,j}^{\{f,s,1\}} = 0$  for  $1 < i \leq j$ . From line 7, we also obtain a condition for the coupling parameters of the second micro step which takes the form  $a_{1,j}^{\{f,s,2\}} = 0$  for  $j > s^{\{f\}}$ . However, this condition is only relevant in the case  $s^{\{s\}} > s^{\{f\}}$ .
  4. The first condition Equation (45) of internal consistency for  $\lambda = 1$  is given by

$$m_1 \mathbf{c}^{\{f\}} = \mathbf{c}^{\{f,s,1\}},$$

which in combination with the above points leads to the additional condition  $a_{1,1}^{\{f,s,1\}} = 0$ .

5. In Algorithm 2, line 9 the value of  $u_{n+1}^{\{s\}}$  is accessed if  $i = s^{\{f\}}, \lambda = N$ . If  $N > 1$ , this value has already been determined in Algorithm 1, line 12. However, if  $N = 1$  this value is only available if  $s^{\{s\}} \leq s^{\{f\}}$ .
6. For an explicit base method  $(A^{\{f,f\}}, b^{\{f\}})$ , the computation of mesh velocities in Algorithm 2, lines 4 and 11 requires nonzero entries of  $a_{i,i-1}^{\{f,f\}}, i = 2, \dots, s^{\{f\}}$  and  $b_{s^{\{f\}}}^{\{f\}}$ .

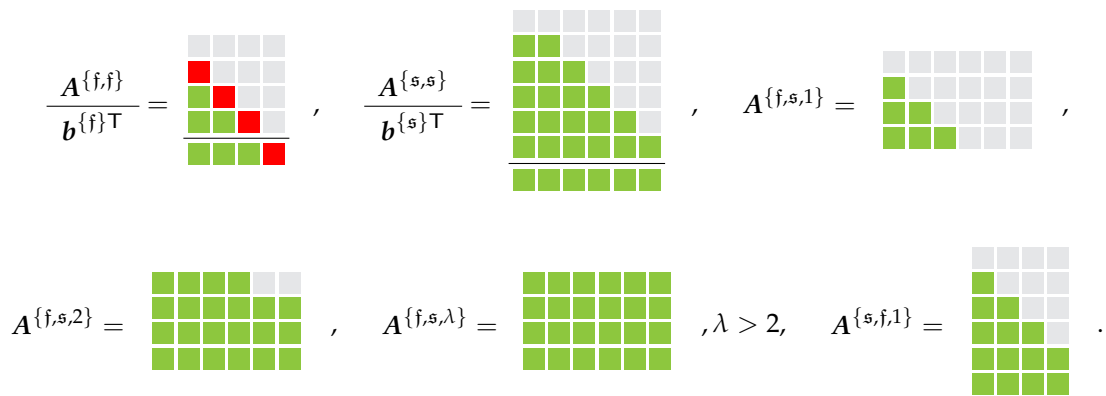
In summary, we may specify the structure of the base methods in the following form, visualized in Figure 1 for  $s^{\{f\}} = 4, s^{\{s\}} = 6$ . Here, the gray-colored blocks (■) denote zero entries while the red ones (■) denote entries required to be non-zero in agreement with our strategy of enforcing the GCL and the green-colored blocks (■) symbolize arbitrary entries.

Looking at the required structures of  $A^{\{f,s,1\}}$  and  $A^{\{s,f,1\}}$  in Figure 1, suitable coupling parameters to achieve overall second order are given by the choice of  $A^{\{f,s,\lambda\}}$  according to Equations (50) and (51) and of  $A^{\{s,f,\lambda\}}$  according to Equation (52), while the choice given by Equations (47) and (49) does not conform to items 1, 3 and 4 given in the above discussion. Concerning third order MGARK schemes, the coupling conditions given by Equations (53)–(55) agree to all of the above requirements.

We further remark that the determination of  $k_{n,i}^{\{s\}}$  in Algorithm 1, line 9 may be implicit. Prescribing an (E)SDIRK scheme as slow base method, a potentially nonlinear system of equations has to be solved for  $i > 1$ , given by

$$k_{n,i}^{\{s\}} = f^{\{s\}}(u_{n,i}^{\{s,f\}}, u_n^{\{s\}} + H \sum_{j=1}^i a_{i,j}^{\{s,s\}} k_{n,j}^{\{s\}}), \quad (77)$$

combining lines 8 and 9 in Algorithm 1.



**Figure 1.** Required structure of base methods. Gray-colored blocks denote zero entries, red-colored blocks denote entries required to be non-zero, green-colored blocks symbolize arbitrary entries.

### 3.4. Time Adaptivity

Efficiency for practical applications needs time adaptivity. Hence, the macro and micro step sizes will be adaptively chosen depending on an error estimator based on embedded Runge-Kutta methods. The discrete macro time levels are then given by

$$t_{n+1} = t_n + H_{n+1}, \quad \text{i.e.,} \quad t_n = t_0 + \sum_{i=1}^n H_i, \quad n = 0, 1, 2, \dots$$

and the discrete micro time levels are

$$\tau_{n,\lambda+1} = \tau_{n,\lambda} + h_{n,\lambda+1} \quad \text{i.e.,} \quad \tau_{n,\lambda+1} = \tau_{n,0} + \sum_{\ell=1}^{\lambda+1} h_{n,\ell}, \quad \lambda = 0, \dots, N_n - 1,$$

where  $N_n$  is the number of micro steps within the  $n$ -th macro step and the micro time levels relate to the macro time levels as  $\tau_{n,0} = t_n$  and  $\tau_{n,N_n} = t_{n+1}$ .

Figure 2 illustrates the potential positions of discrete macro and micro time levels. The concrete computation of macro and micro step sizes depends on error control using the parameters  $\underline{\alpha} \in (0, 1)$ ,  $\bar{\alpha} > 1$  and  $\beta \in (0, 1)$  as well as the relative and absolute tolerances  $\text{TOL}_r \in (0, 1)$  and  $\text{TOL}_a > 0$  according to Algorithms 3 and 4.

Since the final number of micro steps  $N_n$  within the  $n$ -th macro step is not known a priori, Algorithm 4 is devised to work without this information. For the present adaptivity strategies, the first macro step size  $H_1$  as well as the first micro step size  $h_{0,1} \leq H_1$  need to be prescribed. All other step sizes are then automatically calculated by the given Algorithms. This time step calculation can then be combined with CFL based time step computation depending on the specific set-up of a numerical test case. The corresponding details will be clarified in Section 4.

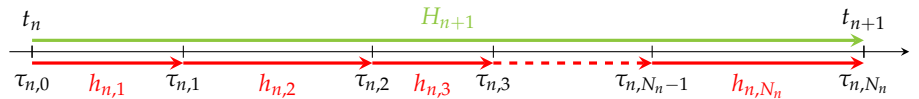


Figure 2. Depiction of macro and micro step sizes including corresponding notation.

---

**Algorithm 3** Adaptive step size determination for macro step  $t_{n-1} \rightarrow t_n$  with step size  $H_n$ .

---

- 1: **procedure** ADAPT\_MACRO\_STEP\_SIZE( $H_n, \mathbf{u}_{n-1}^{\{s\}}, \mathbf{k}_{n-1}^{\{s\}}, \text{TOL}_r, \text{TOL}_a, \underline{\alpha}, \bar{\alpha}, \beta$ )
  - 2:    $\text{TOL} \leftarrow \text{TOL}_r \|\mathbf{u}_{n-1}^{\{s\}}\| + \text{TOL}_a$
  - 3:    $\epsilon \leftarrow H_n \|\sum_{i=1}^{s^{\{s\}}} (b_i^{\{s\}} - \hat{b}_i^{\{s\}}) \mathbf{k}_{n-1,i}^{\{s\}}\|$
  - 4:   **if**  $\epsilon \leq \text{TOL}$  **then**
  - 5:      $\tilde{H} \leftarrow H_n \min\{\bar{\alpha}, \beta(\text{TOL}/\epsilon)^{1/(\hat{p}+1)}\}$
  - 6:      $H_{n+1} \leftarrow \min\{\tilde{H}, T - t_n\}$
  - 7:     Accept solution and employ step size  $H_{n+1}$  in the next macro step  $t_n \rightarrow t_{n+1}$
  - 8:   **else**
  - 9:      $H_n \leftarrow H_n \max\{\underline{\alpha}, \beta(\text{TOL}/\epsilon)^{1/(\hat{p}+1)}\}$
  - 10:   Repeat macro step  $t_{n-1} \rightarrow t_n$  with new step size  $H_n$
  - 11:   **end if**
  - 12: **end procedure**
-

**Algorithm 4** Adaptive step size determination for micro step  $\tau_{n,\lambda-1} \rightarrow \tau_{n,\lambda}$  with step size  $h_{n,\lambda}$ .

---

```

1: procedure ADAPT_MICRO_STEP_SIZE( $h_{n,1}, \dots, h_{n,\lambda}, \mathbf{u}_n^{\{f,\lambda-1\}}, \mathbf{k}_n^{\{f,\lambda-1\}}, \text{TOL}_r, \text{TOL}_a, \underline{\alpha}, \bar{\alpha}, \beta$ )
2:    $\text{TOL} \leftarrow \text{TOL}_r \|\mathbf{u}_n^{\{f,\lambda-1\}}\| + \text{TOL}_a$ 
3:    $\epsilon \leftarrow h_{n,\lambda} \|\sum_{i=1}^{s\{s\}} (b_i^{\{f\}} - \hat{b}_i^{\{f\}}) \mathbf{k}_{n,i}^{\{f,\lambda-1\}}\|$ 
4:   if  $\epsilon \leq \text{TOL}$  then
5:      $\tilde{h} \leftarrow h_{n,\lambda} \min\{\bar{\alpha}, \beta(\text{TOL}/\epsilon)^{1/(\hat{p}+1)}\}$ 
6:      $\bar{h} \leftarrow \sum_{\ell=1}^{\lambda} h_{n,\ell}$ 
7:     if  $\bar{h} < H_{n+1}$  then not the last micro step within current macro step
8:        $h_{n,\lambda+1} \leftarrow \min\{\tilde{h}, H_{n+1} - \bar{h}\}$ 
9:       Accept solution and employ step size  $h_{n,\lambda+1}$  in the next micro step  $\tau_{n,\lambda} \rightarrow \tau_{n,\lambda+1}$ 
10:    else last micro step within current macro step
11:       $h_{n+1,1} \leftarrow \min\{\tilde{h}, H_{n+2}\}$ 
12:      Accept solution and employ step size  $h_{n+1,1}$  in the first micro step  $\tau_{n+1,0} \rightarrow \tau_{n+1,1}$ 
        of the next macro step  $t_{n+1} \rightarrow t_{n+2}$ 
13:    end if
14:  else
15:     $h_{n,\lambda} \leftarrow h_{n,\lambda} \max\{\underline{\alpha}, \beta(\text{TOL}/\epsilon)^{1/(\hat{p}+1)}\}$ 
16:    Repeat micro step  $\tau_{n,\lambda-1} \rightarrow \tau_{n,\lambda}$  with new step size  $h_{n,\lambda}$ 
17:  end if
18: end procedure

```

---

**4. Numerical Experiments**

In the following numerical experiments, the considered MGARK schemes are constructed from base methods of second and third order, respectively. These base methods are given in the form

$$\frac{A}{\begin{matrix} \mathbf{b}^\top \\ \hat{\mathbf{b}}^\top \end{matrix}}$$

where  $(A, \hat{\mathbf{b}})$  denotes the embedded method to the base method  $(A, \mathbf{b})$  such that if  $p$  is the order of the base method and  $\hat{p}$  denotes the order of the embedded method, we have  $0 < \hat{p} < p$ .

**Second order base methods**

A second order MGARK scheme is derived from the explicit Heun method ( $p = 2, \hat{p} = 1$ ) and the implicit trapezoidal rule ( $p = 2, \hat{p} = 1$ ) which is an ESDIRK scheme.

$$\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 1 & 0 & 1/2 & 1/2 \\ \hline 1/2 & 1/2 & 1/2 & 1/2 \\ 0 & 1 & 0 & 1 \end{array}$$

Explicit Heun scheme      Implicit trapezoidal rule

**Third order base methods**

A pair of third order base methods from which we construct the third order MGARK scheme is given by the following four-stage RK schemes of order  $p = 3$  and embedded order  $\hat{p} = 2$  which is

taken from [32]. The corresponding IMEX scheme is denoted by IMEXRKC3c in that work.

*RK explicit 3rd order*

$$\begin{array}{cccc}
 0 & 0 & 0 & 0 \\
 \frac{3375509829940}{4525919076317} & 0 & 0 & 0 \\
 0 & \frac{272778623835}{1039454778728} & 0 & 0 \\
 0 & \frac{673488652607}{2334033219546} & \frac{1660544566939}{2334033219546} & 0 \\
 \hline
 0 & \frac{673488652607}{2334033219546} & \frac{493801219040}{853653026979} & \frac{184814777513}{1389668723319} \\
 \frac{449556814708}{1155810555193} & 0 & \frac{210901428686}{1400818478499} & \frac{480175564215}{1042748212601}
 \end{array} \quad (78)$$

*3rd order DIRK scheme*

$$\begin{array}{cccc}
 0 & 0 & 0 & 0 \\
 0 & \frac{3375509829940}{4525919076317} & 0 & 0 \\
 0 & \frac{-11712383888607531889907}{32694570495602105556248} & \frac{566138307881}{912153721139} & 0 \\
 0 & \frac{673488652607}{2334033219546} & \frac{493801219040}{853653026979} & \frac{184814777513}{1389668723319} \\
 \hline
 0 & \frac{673488652607}{2334033219546} & \frac{493801219040}{853653026979} & \frac{184814777513}{1389668723319} \\
 0 & \frac{366319659506}{1093160237145} & \frac{270096253287}{480244073137} & \frac{104228367309}{1017021570740}
 \end{array} \quad (79)$$

#### 4.1. Constant Flow with Prescribed Grid Movement

First, to check the viability of our implementation concerning the preservation of the GCL, we apply the MGARK approach to constant flow with an artificial grid movement.

Hence, we consider constant solutions  $\mathbf{u}: \Omega \times [0, T] \rightarrow \mathbb{R}^m$  of the initial value problem

$$\partial_t \mathbf{u} + \partial_x f(\mathbf{u}) = \mathbf{0}, \quad \mathbf{u}(x, 0) = \mathbf{u}_0(x).$$

If the initial conditions is given by a constant initial state  $\mathbf{u}(x, 0) = \mathbf{u}_0$ ,  $x \in \Omega$ , the exact solution is constant in time and space,  $\mathbf{u}(x, t) = \mathbf{u}_0$  for  $(x, t) \in \Omega \times [0, T]$ , hence we expect the numerical approximation of  $\mathbf{u}$  to be constant as well. Starting from an initial grid  $\Omega_i(0)$ ,  $i \in \mathcal{A} = \{1, \dots, K\}$ , the method runs on a time-dependent spatial discretization

$$\Omega = [0, 1] = \bigcup_{i \in \mathcal{A}} \Omega_i(t), \quad t \in [0, T],$$

where the grid points are moved according to the rule

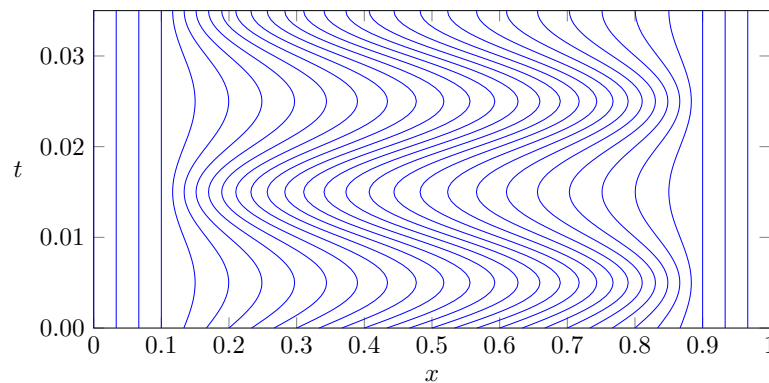
$$x_i(t) = \alpha_i \sin(\omega \pi t) + x_i(0), \quad i = 1, \dots, K + 1, \quad (80)$$

$$\alpha_i = \begin{cases} \frac{\xi_R - \xi_L}{2\pi} \sin\left(\frac{\pi}{\xi_R - \xi_L}(x_i(0) - \xi_L)\right), & x_i(0) \in (\xi_L, \xi_R), \\ 0, & \text{otherwise,} \end{cases} \quad (81)$$

with the parameters  $\omega > 0$  and  $0 \leq \xi_L < \xi_R \leq 1$ . The grid cells are then determined from the grid points according to  $\Omega_i(t) = [x_i(t), x_{i+1}(t)]$ . Hence, in this set-up the grid points with  $x_i(0) \in [0, \xi_L] \cup [\xi_R, 1]$  are fixed while the grid points with  $x_i(0) \in (\xi_L, \xi_R)$  oscillate in form of a sine wave in time with frequency  $\omega$ . The choice of the amplitude  $\alpha_i$  according to Equation (81) guarantees that the grid cells do not overlap, i.e., for all  $t \in [0, T]$ , we have  $x_i(t) < x_{i+1}(t)$ . Figure 3 depicts the



corresponding grid motion for an equidistant initial grid with  $K = 30$  cells, a frequency of  $\omega = 100$  and a confined region of grid movement bounded by  $\xi_L = 0.1$ ,  $\xi_R = 0.9$ .



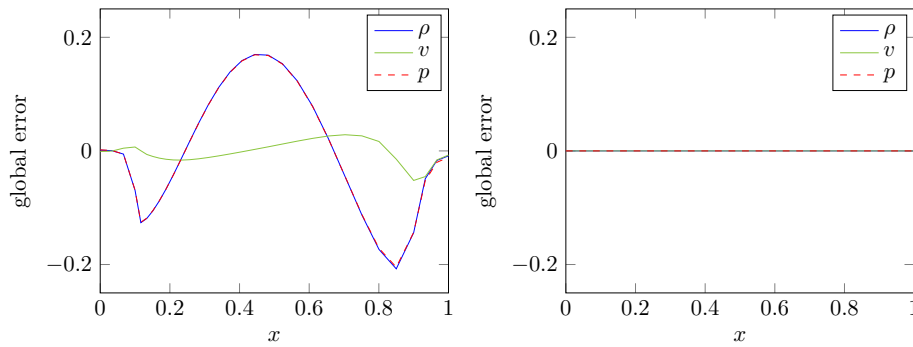
**Figure 3.** Time evolution of an equidistant initial grid under the grid motion given in Equations (80) and (81) until time  $T = 0.035$ .

In order to illustrate the importance of calculating mesh velocities which are consistent with the GCL, Figure 4 depicts the error with respect to density, velocity and pressure of the explicit Euler scheme applied to the fluid equations. More precisely, the right hand side of Figure 4 shows the result for mesh velocities computed according to Equations (59) and (60) while the left hand side depicts the corresponding errors obtained by grid velocities determined from differentiation of Equations (80) and (81) with respect to  $t$ . In the first case the constant initial state is preserved in accordance with the theoretical considerations while the second case does not lead to a numerical method consistent with the GCL.

Now we simulate this test case using the proposed MGARK scheme. In order to obtain a partitioning in fast and slow components, we consider the autonomous coupled system composed of the fluid variables as the fast components and the grid points as well as the time variable as slow components. Therefore, the grid motion given in Equations (80) and (81) is not prescribed exactly but provides a differential equation for the slow components of the coupled system. The slow components are hence evolved according to

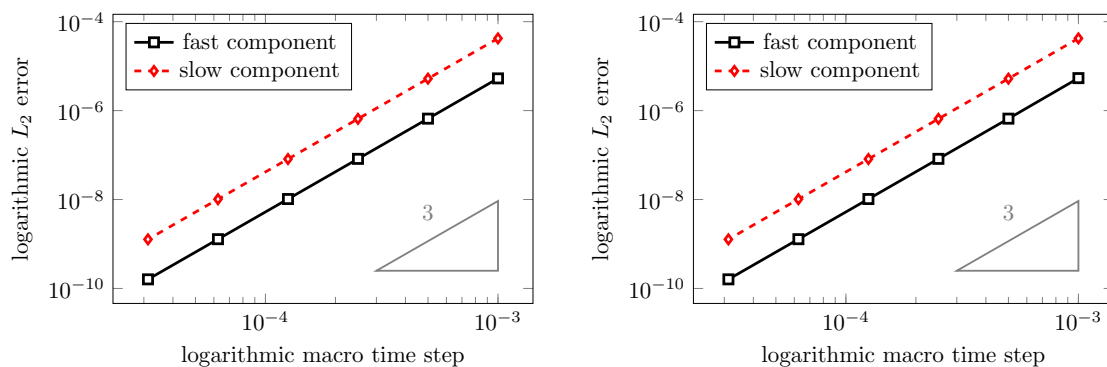
$$x_i'(t) = \alpha_i \omega \pi \cos(\omega \pi t), \quad t \in (0, T],$$

starting from a given initial state  $x_i(0)$ . Due to the ALE formulation Equation (56) of the fluid equations, the fast components  $\mathbf{u}^{\{f\}}$  are given by the products of the cell averages of the conservative fluid variables with the corresponding cell volumes. By the construction just described, the slow components are given by  $\mathbf{u}^{\{s\}} = [t, x_1, \dots, x_{K+1}]^T$  and the coupling between these subsystems now takes the form  $\mathbf{f}^{\{f\}} = \mathbf{f}^{\{f\}}(\mathbf{u}^{\{f\}}, \mathbf{u}^{\{s\}})$  and  $\mathbf{f}^{\{s\}} = \mathbf{f}^{\{s\}}(\mathbf{u}^{\{s\}})$ , which shows that the evolution of slow components is actually independent of the fast ones in this case.



**Figure 4.** Global error with respect to density, velocity and pressure at time  $T = 0.035$  for explicit Euler scheme with step size  $h = 10^{-3}$  applied to constant flow. **Left:** Use of exact grid velocities; **Right:** Computation of grid velocities consistent with the geometric conservation law (GCL).

Figure 5 depicts the experimental order of convergence obtained by two different choices of base methods. The results on the left hand side are obtained by choosing the IMEXRKCB3c explicit/implicit schemes (78) and (79), i.e., the explicit third order scheme for the fast fluid component and the implicit third order scheme for the slow mesh movement component together with coupling conditions as given in Corollary 3. The right hand side shows the results obtained when replacing the explicit IMEXRKCB3c scheme with first order explicit Euler scheme for the fluid equations and setting all coupling coefficients to zero.



**Figure 5.** Experimental order of convergence. **Left:** Base methods IMEXRKCB3c (explicit/implicit) of third order; **Right:** Explicit Euler scheme for fast components, third order implicit base method of IMEXRKCB3c for slow components.

First of all, we observe that the chosen coupling strategies lead to an experimentally obtained convergence of third order. Though at first sight, it is striking that a global convergence of third order is also obtained using a fluid time discretization of only first order, we recall that our GCL-consistent MGARK scheme does not produce any disturbances of the constant fluid state, independent of the order of the fast base method. Hence, the global error is only determined by the slow base method which approximately determines the grid positions. Thus, the slow method solely determines the order of the coupled scheme in this case. However, in general, the accuracy of the coupled scheme reduces to first order if one of the base methods is only of first order.

#### 4.2. The One-Dimensional Piston Problem

The interaction between a piston which is attached to a spring and an inviscid fluid which is contained in a chamber is a classical test case in the context of fluid–structure interaction [33,34]. The coupled piston problem and its one-dimensional set-up are illustrated in Figure 6.

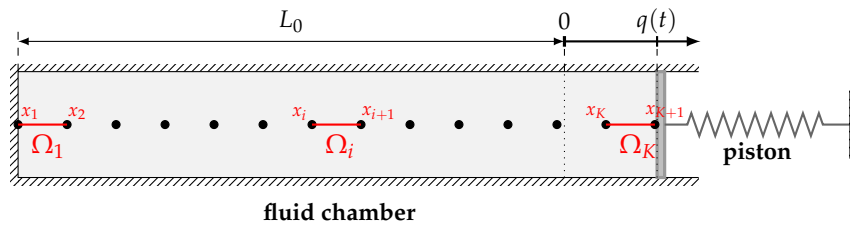


Figure 6. One-dimensional piston problem.

In the mathematical formulation, the fluid is described by the one-dimensional inviscid Euler equations while the displacement of the piston is modelled by an undamped harmonic oscillator. In non-dimensional form we obtain a coupled system of the fluid equations given by

$$\partial_t \mathbf{u} + \partial_x \mathbf{f}(\mathbf{u}) = \mathbf{0}, \quad \mathbf{u}(x, 0) = \mathbf{u}_0(x), \quad (x, t) \in \Omega \times [0, T]$$

with

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho v \\ \rho E \end{bmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} \rho v \\ \rho v^2 + p \\ \rho E v + p v \end{bmatrix}, \quad p = (\gamma - 1)\rho(E - \frac{1}{2}v^2), \quad \gamma = 1.4,$$

where  $\rho$ ,  $v$ ,  $E$  and  $p$  denote density, velocity, specific total energy and pressure of the fluid, respectively, while  $\gamma$  denotes the adiabatic coefficient, and the structure model for the piston displacement  $q(t)$  given by

$$mq'' + kq = A(p_I - p_A), \quad q(0) = q_0, \quad q'(0) = r_0,$$

where the parameters in this non-dimensional formulation correspond to the mass  $m$  and the stiffness  $k$  of the spring as well as the cross-sectional area  $A$  of the piston as well as its initial displacement  $q_0$  and velocity  $r_0$ . The piston displacement determines the length of the time-dependent fluid domain  $\Omega(t)$  according to

$$\Omega(t) = [0, L_0 + q(t)] = \bigcup_{i \in \mathcal{A}} \Omega_i(t), \quad t \in [0, T], \quad \mathcal{A} = \{1, \dots, K\}.$$

This domain is discretized by a constant number of grid cells, cf., Figure 6. Let  $x_i$  denote the grid points such that  $\Omega_i(t) = [x_i(t), x_{i+1}(t)]$ . The influence of the fluid subsystem on the structure subsystem is given by the difference of the fluid pressure  $p_I(t) = p(x_{K+1}, t)$  acting on the piston at the fluid-structure interface, i.e., the right boundary of the fluid domain and the ambient pressure  $p_A$ .

The specific setting of the piston parameters  $m$ ,  $k$ ,  $A$  and  $p_A$  and the initial length  $L_0$  of the fluid domain as well as the initially constant fluid variables  $\rho_0$ ,  $v_0$ ,  $p_0$  and the initial piston displacement  $q_0$  and velocity  $r_0$ , which have been used in this test case are summarized in Table 1.

Table 1. Parameters and initial conditions for piston problem.

$m$	$k$	$A$	$p_A$	$L_0$	$\rho_0$	$v_0$	$p_0$	$q_0$	$r_0$
0.6154	0.0781	0.02	1.0	1.0	1.0	0.0	1.0	0.05	0.0

We may now take a closer look at the difference in time scales for the fluid and structure subsystems in this set-up. The characteristic time scale of a fluid is given by the time needed by a pressure wave to traverse the chamber, i.e.,

$$T^{\{f\}}(t) = \frac{L(t)}{c(t)} = (L_0 + q(t)) \sqrt{\frac{1}{\gamma} \frac{\rho(t)}{p(t)}},$$

where  $c$  denotes the sound speed. For the values given in Table 1 we obtain  $T^{\{f\}}(0) \approx 0.887$ .

The characteristic time scale of the structure is determined by the oscillation period, i.e.,

$$T^{\{s\}} = 2\pi\sqrt{\frac{m}{k}},$$

whereby we obtain  $T^{\{s\}} \approx 17.640$  for the values in Table 1.

The one-dimensional Euler equations in ALE formulation are then discretized by a finite volume method which yields

$$(\bar{\mathbf{u}}_i V_i)' = \mathbf{h}_{\text{ALE}}(\bar{\mathbf{u}}_{i-1}, \bar{\mathbf{u}}_i, \dot{x}_i) - \mathbf{h}_{\text{ALE}}(\bar{\mathbf{u}}_i, \bar{\mathbf{u}}_{i+1}, \dot{x}_{i+1}), \quad i = 2, \dots, K-1, \quad (82)$$

for interior grid cells, using a suitable numerical flux function  $\mathbf{h}_{\text{ALE}}$ . At the two boundaries of  $\Omega(t)$  given by the grid points  $x_1$  and  $x_{K+1}$ , boundary conditions are required. As there is no movement of fluid particles through the boundaries of the fluid chamber (no-slip conditions), we demand that at the boundaries, the fluid moves with the same velocity as the corresponding boundary grid point. Suitable boundary conditions may be posed either as strong or as weak boundary conditions. Strong boundary conditions usually modify the numerical solution in a suitable way in order to satisfy the no-slip conditions. Weak boundary conditions usually modify the numerical flux in a suitable way. In comparison to strong boundary conditions, weak boundary conditions often lead to an improved convergence speed regarding iterative solvers and may also lead to an improved robustness, see [35–37]. Concerning weak boundary conditions, both weak prescribed and weak Riemann approaches are common. In this work, we use the weak prescribed approach, i.e., the boundary fluxes are explicitly prescribed according to

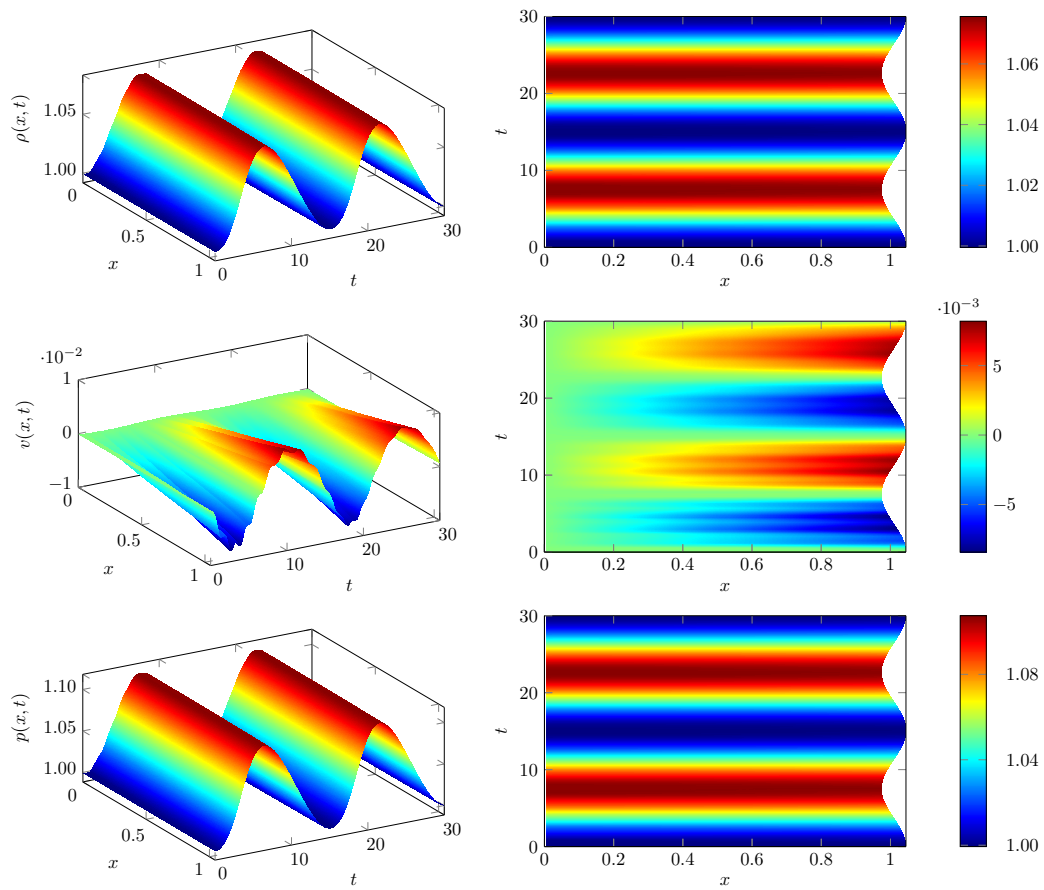
$$(\bar{\mathbf{u}}_1 V_1)' = \mathbf{f}_{\text{fw}}(\bar{\mathbf{u}}_1, \dot{x}_1) - \mathbf{h}_{\text{ALE}}(\bar{\mathbf{u}}_1, \bar{\mathbf{u}}_2, \dot{x}_2), \quad (83)$$

$$(\bar{\mathbf{u}}_K V_K)' = \mathbf{h}_{\text{ALE}}(\bar{\mathbf{u}}_{K-1}, \bar{\mathbf{u}}_K, \dot{x}_K) - \mathbf{f}_{\text{fw}}(\bar{\mathbf{u}}_K, \dot{x}_{K+1}), \quad (84)$$

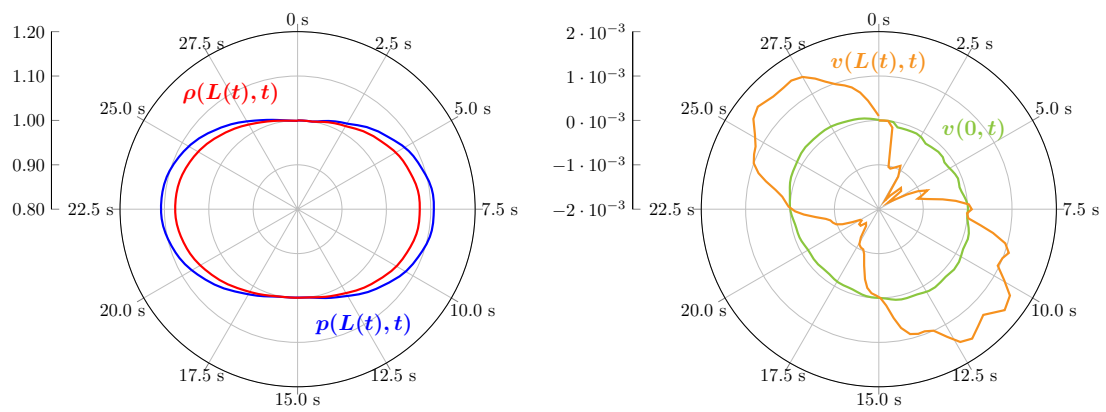
with

$$\mathbf{f}_{\text{fw}}(\bar{\mathbf{u}}, \dot{x}) = \mathbf{f}_{\text{ALE}}(\bar{\mathbf{u}}_{\text{fw}}(\bar{\mathbf{u}}, \dot{x}), \dot{x}), \quad \bar{\mathbf{u}}_{\text{fw}}(\bar{\mathbf{u}}, \dot{x}) = \begin{bmatrix} \bar{\rho}_{\text{fw}} \\ \bar{\rho}_{\text{fw}} \bar{v}_{\text{fw}} \\ \bar{\rho}_{\text{fw}} \bar{E}_{\text{fw}} \end{bmatrix} = \begin{bmatrix} \bar{\rho} \\ \bar{\rho} \dot{x} \\ \frac{\bar{p}}{\gamma-1} + \frac{1}{2} \bar{\rho} \dot{x}^2 \end{bmatrix}.$$

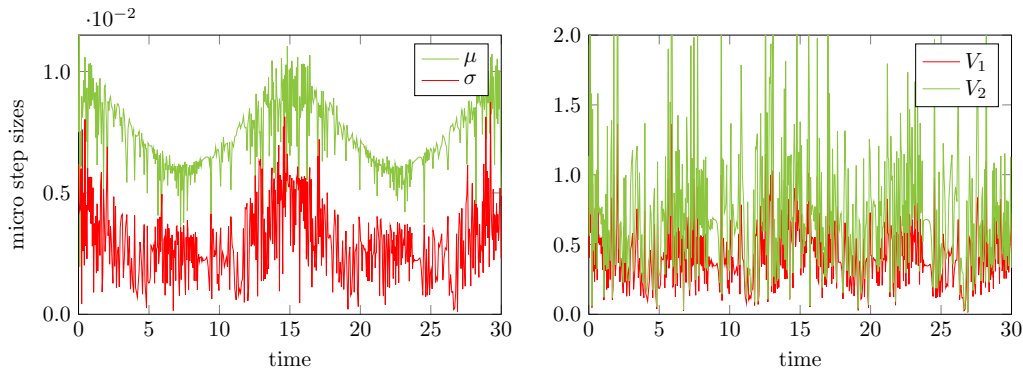
For all of our numerical tests, we initially chose an equidistant discretization of the fluid chamber consisting of 100 finite volume cells and the numerical flux function  $\mathbf{h}_{\text{ALE}}$  is given by the well known Lax-Friedrichs flux. The nonlinear systems of equations within the implicit stages of the RK scheme applied to the structure component (see Equation (77)) are solved using a Newton-Krylov scheme. In order to introduce non-uniformity into this test case, the deformation of the fluid domain by the piston movement is transferred to the individual grid points in a non-uniform manner. In this set-up, the spatial grid is only equidistant at those time levels where the piston achieves its initial maximal displacement. In general, the computational grid is compressed near the piston. The results depicted in Figures 7–9 are obtained by adaptive determination of time step sizes for both macro and micro steps according to the strategies described in Section 3.4, i.e., solely based on embedded error control. The parameters for time adaptivity are set to  $\underline{\alpha} = 0.3$ ,  $\bar{\alpha} = 2$ ,  $\beta = 0.8$  for both micro and macro step sizes, and  $\text{TOL}_r = \text{TOL}_a = 10^{-5}$  for the macro step sizes as well as  $\text{TOL}_r = \text{TOL}_a = 3 \times 10^{-5}$  for the micro step sizes. The simulation is carried out until the final time  $T = 30$ . Figures 7 and 8 show the numerical results for the MGARK scheme based on the explicit Heun scheme for the fluid and the implicit trapezoidal rule for the structure. The coupling procedure is given by Choice B in Corollary 2. In the representations on the right hand side of Figure 7, the piston movement at the right chamber boundary is visible as well.



**Figure 7.** Time evolution of density, velocity and pressure.



**Figure 8.** Time evolution of fluid density, pressure and velocity in polar graph.



**Figure 9.** Statistics of the micro step sizes used by an error-controlled macro and micro step adaptive 2nd order multirate generalized-structure additively partitioned Runge-Kutta (MGARK) approach for solving the piston problem. **Left:** Mean value  $\mu$  and standard deviation  $\sigma$  of micro step sizes; **Right:** Coefficients of variation.

Now we would like to pay more attention to our strategy to allow for adaptive non-constant micro step sizes within one macro step. In this regard, Figure 9 collects the statistics of the computed macro and micro step sizes. For the micro step sizes, the left part of Figure 9 shows the time evolution of the mean value  $\mu$  and the standard deviation  $\sigma$  calculated as

$$\mu(h_n) = \frac{1}{N_n - 1} \sum_{\lambda=1}^{N_n-1} h_{n,\lambda}, \quad \sigma(h_n) = \sqrt{\frac{1}{N_n - 2} \sum_{\lambda=1}^{N_n-1} (h_{n,\lambda} - \mu(h_n))^2}$$

for  $h_n = [h_{n,1}, \dots, h_{n,N_n}]^T$ . Herein, the last micro step size of each macro step usually will be suitably shortened in order to add up to the macro step size, see Algorithm 4, line 8 and is hence not included into the calculation of the statistics. The coefficients of variation  $V_1$  and  $V_2$  relate the standard deviation as well as the maximum deviation to the mean value, i.e.

$$V_1(h_n) = \sigma(h_n) / \mu(h_n), \quad V_2(h_n) = \max_{\lambda=1, \dots, N_n-1} |h_{n,\lambda} - \mu(h_n)| / \mu(h_n).$$

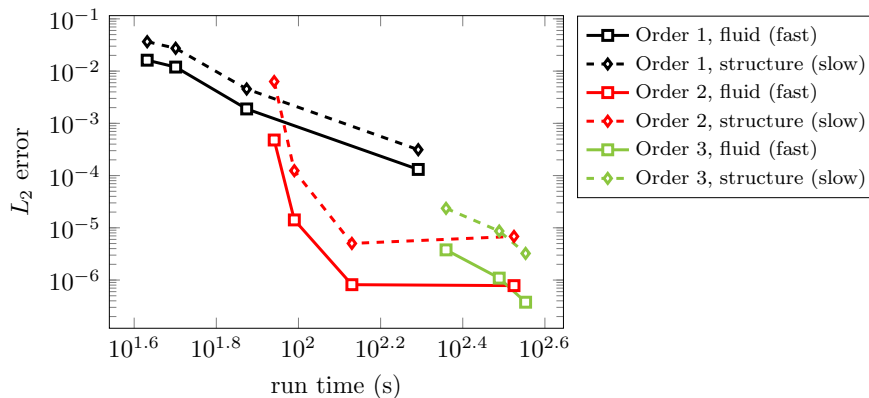
Hence, from these coefficients we obtain a measure of the extend of variability in relation to the mean.

As shown by the results on the right hand side of Figure 9, there are time sections with significant variability of the micro step sizes, i.e., there are sections with  $V_1 > 0.5$  and sections with  $V_2 > 0.5$ . Thus, the application of variable micro step sizes instead of constant ones is justified for this test case. A multirate scheme using constant micro step sizes might have used the smallest micro step size to comply with the given error tolerances which would have led to a larger amount of micro steps per macro step compared to the adaptive choice of micro steps. For the time sections with higher coefficients of variation this means a higher computational effort and thus a lower efficiency compared to the adaptive case.

In Figure 10, an efficiency study regarding numerical error versus CPU time is depicted comparing the described MGARK schemes up to third order. Computations were carried out for the explicit Euler method coupled with the implicit Euler method and coupling coefficients set to zero (order 1), the explicit Heun scheme coupled with implicit trapezoidal scheme and second order coupling coefficients according to Corollary 2, choice B, and the IMEXRKCB3c base methods with third order coupling according to Corollary 3.

A spatial discretization of 100 FV cells and a final time of  $T = 30$  has been used in all test cases. The computations were carried out for different macro step sizes. During each computation, the macro step size was held constant while the micro step sizes were chosen with

respect to the Courant-Friedrichs-Lewy (CFL) condition, with a CFL-number of 0.8, i.e., according to the error-controlled approach described in Section 3.4. In order to compute the numerical error depicted in Figure 10, a reference solution was calculated using the 3rd order explicit IMEXRKCB3c scheme to solve the fluid–structure system as a whole in a monolithic manner with a suitably small time step size and without the use of different step sizes for the subsystems.



**Figure 10.** Error vs. CPU time for first, second and third order MGARK scheme applied to the moving piston test case.

We remark that the reference solution is obtained on the same spatial grid as the numerical simulations. Therefore, we in fact measure the time integration error of the MGARK variants applied to the semi-discrete system which is generated by finite volume space discretization Equation (82) of first order. As the focus of this work is on time integration, we expect this to be a reasonable measure for the quality of the time integration process. Of course, the full discretization error, which may be measured with respect to a reference solution on a refined spatial grid, will be affected by both spatial discretization method and time integration scheme.

We may observe that the second order coupled time integration is most efficient for moderate to small tolerances in the range of about  $10^{-3}$  to  $10^{-6}$ . For even smaller tolerances, the third order coupled procedure starts to become efficient for this specific test case.

An efficiency study regarding run time and accuracy was carried out for the 2nd order MGARK scheme described above. The results are summarized in Tables 2 and 3 on page 30. Again, we approximated the error using a reference solution obtained by an explicit 3rd order in time monolithic approach using a suitably small constant time step size. Within Table 2, the macro step size is denoted by  $H$  and is held constant,  $N$  denotes the number of micro steps per macro step,  $e^{\{f\}}$  respectively  $e^{\{s\}}$  denote the error of the fast respectively slow system component measured in  $L_2$  norm,  $\bar{N}$  is the average number of micro steps per macro step in case of a micro step size adaptive approach and TOL denotes the relative and absolute tolerances in case of error-controlled adaptivity. The remaining parameters needed by error-controlled adaptivity were always set to  $\underline{\alpha} = 0.3$ ,  $\bar{\alpha} = 2$  and  $\beta = 0.8$  for both micro and macro step sizes.

For the computations with an error-controlled micro step size, an initial step size of  $h_{0,1} = H_1/10$  has been used. A dash indicates that the scheme diverged for the specified settings. The right part of Table 3 was computed using the minimum micro step size given by comparing the error-controlled approach to a CFL-compliant step size computation with a CFL number of 0.9.

We observe that the best error achieved with constant macro and micro step sizes is undercut by the error obtained with both of the fully-adaptive approaches (i.e., adaptive macro step sizes and adaptive micro step sizes, see Table 3) with suitably small tolerances. At the same time the fully-adaptive approaches also perform better regarding run time. In addition to the results shown in Figure 9 these results demonstrate the efficiency of our approach to time adaptive MGARK schemes and show its potential to be successfully applied to more sophisticated real-world problems.



**Table 2.** Comparison of run time and error of different time step adaptation strategies of the 2nd order MGARK scheme applied to the piston problem, Part I. Further explanations are given in the text.

$H$	Constant Micro Step Sizes				CFL-Compliant Micro Step Sizes					Error Controlled Micro Step Sizes				
	$N$	Run Time (s)	$e^{\{f\}}$	$e^{\{s\}}$	CFL Number	$\bar{N}$	Run Time (s)	$e^{\{f\}}$	$e^{\{s\}}$	TOL	$\bar{N}$	Run Time (s)	$e^{\{f\}}$	$e^{\{s\}}$
$5.00 \times 10^{-2}$	1	—	—	—	0.9	9.03	79.2	$4.41 \times 10^{-5}$	$2.75 \times 10^{-4}$	$3.0 \times 10^{-1}$	—	—	—	—
	2	—	—	—	0.5	15.93	138.3	$4.38 \times 10^{-5}$	$2.72 \times 10^{-4}$	$3.0 \times 10^{-2}$	—	—	—	—
	4	—	—	—						$3.0 \times 10^{-3}$	7.45	75.2	$6.43 \times 10^{-4}$	$4.79 \times 10^{-4}$
	8	—	—	—						$3.0 \times 10^{-4}$	7.38	75.8	$1.26 \times 10^{-4}$	$2.55 \times 10^{-4}$
	16	137.1	$4.38 \times 10^{-5}$	$2.72 \times 10^{-4}$						$3.0 \times 10^{-5}$	7.41	75.6	$4.42 \times 10^{-5}$	$2.76 \times 10^{-4}$
$2.50 \times 10^{-2}$	1	—	—	—	0.9	4.84	88.7	$1.43 \times 10^{-5}$	$1.26 \times 10^{-4}$	$3.0 \times 10^{-1}$	—	—	—	—
	2	—	—	—	0.5	8.16	147.2	$1.48 \times 10^{-5}$	$1.28 \times 10^{-4}$	$3.0 \times 10^{-2}$	—	—	—	—
	4	—	—	—						$3.0 \times 10^{-3}$	3.82	76.9	$4.33 \times 10^{-4}$	$2.66 \times 10^{-4}$
	8	139.2	$1.48 \times 10^{-5}$	$1.28 \times 10^{-4}$						$3.0 \times 10^{-4}$	3.82	76.1	$1.31 \times 10^{-4}$	$1.11 \times 10^{-4}$
	16	272.0	$1.49 \times 10^{-5}$	$1.28 \times 10^{-4}$						$3.0 \times 10^{-5}$	3.80	74.9	$1.64 \times 10^{-5}$	$1.21 \times 10^{-4}$
$1.25 \times 10^{-2}$	1	—	—	—	0.9	2.57	96.6	$7.58 \times 10^{-6}$	$6.82 \times 10^{-5}$	$3.0 \times 10^{-1}$	—	—	—	—
	2	—	—	—	0.5	4.28	155.5	$7.67 \times 10^{-6}$	$6.77 \times 10^{-5}$	$3.0 \times 10^{-2}$	—	—	—	—
	4	144.1	$7.67 \times 10^{-6}$	$6.75 \times 10^{-5}$						$3.0 \times 10^{-3}$	2.06	81.4	$1.07 \times 10^{-4}$	$7.18 \times 10^{-5}$
	8	279.8	$7.72 \times 10^{-6}$	$6.75 \times 10^{-5}$						$3.0 \times 10^{-4}$	2.06	83.3	$5.54 \times 10^{-5}$	$7.16 \times 10^{-5}$
	16	552.5	$7.74 \times 10^{-6}$	$6.75 \times 10^{-5}$						$3.0 \times 10^{-5}$	2.08	81.0	$7.87 \times 10^{-6}$	$7.15 \times 10^{-5}$

**Table 3.** Comparison of run time and error of different time step adaptation strategies of the 2nd order MGARK scheme applied to the piston problem, Part II. Further explanations are given in the text.

Error Controlled Macro and Micro Step Sizes						Error Controlled Macro and Micro Step Sizes and CFL-Compliant Micro Step Sizes					
Macro Step TOL	Micro Step TOL	$\bar{N}$	Run Time (s)	$e^{\{f\}}$	$e^{\{s\}}$	Macro Step TOL	Micro Step TOL	$\bar{N}$	Run Time (s)	$e^{\{f\}}$	$e^{\{s\}}$
$1.0 \times 10^{-5}$	$3.0 \times 10^{-4}$	7.94	72.9	$2.06 \times 10^{-4}$	$4.10 \times 10^{-4}$	$1.0 \times 10^{-5}$	$3.0 \times 10^{-4}$	9.90	83.9	$1.67 \times 10^{-4}$	$4.37 \times 10^{-4}$
$1.0 \times 10^{-5}$	$3.0 \times 10^{-5}$	8.14	76.2	$1.68 \times 10^{-4}$	$4.39 \times 10^{-4}$	$1.0 \times 10^{-5}$	$3.0 \times 10^{-5}$	10.23	86.4	$1.67 \times 10^{-4}$	$4.37 \times 10^{-4}$
$1.0 \times 10^{-6}$	$3.0 \times 10^{-4}$	2.62	78.0	$4.47 \times 10^{-5}$	$9.62 \times 10^{-6}$	$1.0 \times 10^{-6}$	$3.0 \times 10^{-4}$	3.67	101.0	$2.87 \times 10^{-6}$	$1.90 \times 10^{-5}$
$1.0 \times 10^{-6}$	$3.0 \times 10^{-5}$	2.69	78.2	$3.47 \times 10^{-6}$	$1.74 \times 10^{-5}$	$1.0 \times 10^{-6}$	$3.0 \times 10^{-5}$	3.67	100.4	$3.13 \times 10^{-6}$	$1.89 \times 10^{-5}$

## 5. Concluding Remarks

In this work, we extended the multirate GARK framework to allow for varying micro step sizes and applied the resulting coupled time integration schemes of first, second and third order to mechanical fluid–structure interaction. The computation of mesh velocities satisfying the discrete geometric conservation law is quite straightforward as long as the coefficients of the fluid base method are correctly accounted for as in Equations (59) and (60) and Algorithm 2. In case of an explicit fluid base method, demanding an internally consistent partitioned scheme requires an explicit first stage of the structure base method as well, e.g., use of an E(S)DIRK scheme. While compliance with the geometric conservation law is demonstrated for uniform flow on an artificially moving grid, we furthermore demonstrated the viability of adaptive micro steps by investigating the step size statistics for the moving piston test case. Efficiency of higher order time integration is investigated by a comparison of first, second and third order schemes in terms of error vs. CPU time where the second order coupled approach seems to be most efficient for moderate tolerances. In a comparison of various step size selection strategies, we observe a better performance of the fully-adaptive approaches choosing macro and micro step sizes based on embedded error control.

The adaptive micro step MGARK scheme could hence be an alternative to lower order coupling approaches also for more complex applications in the context of fluid–structure interaction. For this purpose, a detailed study of the stability properties of the proposed MGARK schemes is required. Corresponding studies for the given schemes as well as for alternative candidates within the MGARK framework are subject to future work.

**Acknowledgments:** Both authors gratefully acknowledge partial support of this study in the period of July 2014 to June 2015 provided by the German Research Foundation, SFB/TR TRR 30.

**Author Contributions:** Sascha Bremicker-Trübelhorn elaborated the theoretical proofs concerning the MGARK order conditions and carried out the numerical experiments in agreement with Sigrun Ortleb. Both authors jointly wrote the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Van Zuijlen, A.H.; Bijl, H. Implicit and Explicit Higher Order Time Integration Schemes for Structural Dynamics and Fluid-structure Interaction Computations. *Comput. Struct.* **2005**, *83*, 93–105.
2. Froehle, B.M.; Persson, P.O. A High-Order Implicit-Explicit Fluid-Structure Interaction Method for Flapping Flight. In Proceedings of the 21st AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences, San Diego, CA, USA, 24–27 June 2013.
3. Piperno, S. Simulation Numérique de Phénomènes D’interaction Fluide-Structure. Ph.D. Thesis, Ecole Nationale des Pont et Chaussées, Champs-sur-Marne, Germany, 1995.
4. Piperno, S. Explicit/implicit fluid/structure staggered procedures with a structural predictor and fluid subcycling for 2D inviscid aeroelastic simulations. *Int. J. Numer. Methods Fluids* **1997**, *25*, 1207–1226.
5. Langer, U.; Yang, H. Robust and efficient monolithic fluid-structure-interaction solvers. *Int. J. Numer. Methods Eng.* **2016**, *108*, 303–325.
6. Ángel Fernández, M.; Moubachir, M. A Newton method using exact jacobians for solving fluid–structure coupling. *Comput. Struct.* **2005**, *83*, 127–142.
7. Küttler, U.; Wall, W.A. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Comput. Mech.* **2008**, *43*, 61–72.
8. Jaiman, R.; Geubelle, P.; Loth, E. Stable and Accurate Loosely-Coupled Scheme for Unsteady Fluid-Structure Interaction. In Proceedings of the 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 8–11 January 2007.
9. Rice, J.R. Split Runge-Kutta Method for Simultaneous Equations. *J. Res. Natl. Bur. Standards* **1960**, *64B*, 151–170.
10. Andrus, J.F. Numerical Solution of Systems of Ordinary Differential Equations Separated into Subsystems. *SIAM J. Numer. Anal.* **1979**, *16*, 605–611.

11. Engstler, C.; Lubich, C. MUR8: A multirate extension of the eighth-order Dormand-Prince method. *Appl. Numer. Math.* **1997**, *25*, 185–192.
12. Gear, C.W.; Wells, D.R. Multirate linear multistep methods. *BIT Numer. Math.* **1984**, *24*, 484–502.
13. Skelboe, S. Stability Properties of Backward Difference Multirate Formulas. *Appl. Numer. Math.* **1989**, *5*, 151–160.
14. Skelboe, S.; Andersen, P.U. Stability Properties of Backward Euler Multirate Formulas. *SIAM J. Sci. Stat. Comput.* **1989**, *10*, 1000–1009.
15. Engstler, C.; Lubich, C. Multirate extrapolation methods for differential equations with different time scales. *Computing* **1997**, *58*, 173–185.
16. Günther, M.; Rentrop, P. Multirate ROW methods and latency of electric circuits. *Appl. Numer. Math.* **1993**, *13*, 83–102.
17. Günther, M.; Rentrop, P. Partitioning and Multirate Strategies in Latent Electric Circuits. In *Mathematical Modelling and Simulation of Electrical Circuits and Semiconductor Devices*; Bank, R., Bulirsch, R., Gajewski, H., Merten, K., Eds.; Birkhäuser: Basel, Switzerland, 1996; pp. 33–60.
18. Günther, M.; Kværnø, A.; Rentrop, P. Multirate Partitioned Runge-Kutta Methods. *BIT Numer. Math.* **2001**, *41*, 504–514.
19. Kværnø, A.; Rentrop, P. Low Order Multirate Runge-Kutta Methods in Electric Circuit Simulation. Available online: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.9.3084&rep=rep1&type=pdf> (accessed on 1 December 2016).
20. Ascher, U.M.; Ruuth, S.J.; Wetton, B.T.R. Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Numer. Anal.* **1995**, *32*, 797–823.
21. Knoth, O.; Wolke, R. Implicit-explicit Runge-Kutta methods for computing atmospheric reactive flow. *Appl. Numer. Math.* **1998**, *28*, 327–341.
22. Kennedy, A.C.; Carpenter, M.H. Additive Runge-Kutta schemes for convection-diffusion-reaction equations. *Appl. Numer. Math.* **2003**, *44*, 139–181.
23. Kanevsky, A.; Carpenter, M.H.; Gottlieb, D.; Hesthaven, J.S. Application of implicit–explicit high order Runge-Kutta methods to discontinuous-Galerkin schemes. *J. Comput. Phys.* **2007**, *225*, 1753–1781.
24. Shoenberger, M.; Svärd, M.; Ham, F.E.; Moin, P. An Adaptive Implicit-explicit Scheme for the DNS and LES of Compressible Flows on Unstructured Grids. *J. Comput. Phys.* **2010**, *229*, 5944–5965.
25. Sandu, A.; Günther, M. A Generalized-Structure Approach to Additive Runge-Kutta Methods. *SIAM J. Numer. Anal.* **2015**, *53*, 17–42.
26. Günther, M.; Sandu, A. Multirate generalized additive Runge Kutta methods. *Numer. Math.* **2016**, *133*, 497–524.
27. Günther, M.; Hachtel, C.; Sandu, A. Multirate GARK Schemes for Multiphysics Problems. In *Scientific Computing in Electrical Engineering—SCEE 2014, Wuppertal, Germany, July 2014*; Bartel, A., Clemens, M., Günther, M., ter Maten, E.J.W., Eds.; Springer: Cham, Switzerland, 2016; pp. 115–121.
28. Hairer, E.; Nørsett, S.P.; Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems*, 2nd ed.; Springer: Berlin/Heidelberg, Germany; New York, NY, USA, 1993.
29. Glowinski, R.; Périaux, J. Numerical Methods for Nonlinear Problem in Fluid Dynamics. In *INRIA Conference on Supercomputing: State-of-the-Art*; Elsevier Science Inc.: New York, NY, USA, 1987; pp. 381–479.
30. Araújo, A.; Murua, A.; Sanz-Serna, J.M. Symplectic Methods Based on Decompositions. *SIAM J. Numer. Anal.* **1997**, *34*, 1926–1947.
31. Mavriplis, D.J.; Yang, Z. Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes. *J. Comput. Phys.* **2006**, *213*, 557–573.
32. Cavaglieri, D.; Bewley, T. Low-storage Implicit/Explicit Runge-Kutta Schemes for the Simulation of Stiff High-dimensional ODE Systems. *J. Comput. Phys.* **2015**, *286*, 172–193.
33. Lefrançois, E.; Boufflet, J.P. An Introduction to Fluid-Structure Interaction: Application to the Piston Problem. *SIAM Rev.* **2010**, *52*, 747–767.
34. Blom, F.J. A monolithic fluid-structure interaction algorithm applied to the piston problem. *Comput. Methods Appl. Mech. Eng.* **1998**, *167*, 369–391.
35. Abbas, Q.; Nordström, J. Weak Versus Strong No-Slip Boundary Conditions for the Navier-Stokes Equations. *Eng. Appl. Comput. Fluid Mech.* **2010**, *4*, 29–38.

36. Eliasson, P.; Eriksson, S.; Nordström, J. The Influence of Weak and Strong Solid Wall Boundary Conditions on the Convergence to Steady-State of the Navier-Stokes Equations. In Proceedings of the 19th AIAA Computational Fluid Dynamics, San Antonio, TX, USA, 22–25 June 2009.
37. Nordström, J.; Eriksson, S.; Eliasson, P. Weak and strong wall boundary procedures and convergence to steady-state of the Navier–Stokes equations. *J. Comput. Phys.* **2012**, *231*, 4867–4884.



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).