

# TWO LEVEL ALGORITHMS FOR PARTITIONED FLUID-STRUCTURE INTERACTION COMPUTATIONS

Hester Bijl\*, Alexander H. van Zuijlen<sup>†</sup> and Steven Bosscher

Delft University of Technology, Faculty of Aerospace Engineering  
Kluyverweg 1, 2629 HS Delft, The Netherlands

\* e-mail: [h.bijl@tudelft.nl](mailto:h.bijl@tudelft.nl)

<sup>†</sup> e-mail: [a.h.vanzuijlen@tudelft.nl](mailto:a.h.vanzuijlen@tudelft.nl)

**Key words:** Fluid-structure interaction, domain decomposition, partitioned coupling, multilevel techniques.

**Abstract.** *In this paper we use the multigrid algorithm - commonly used to improve the efficiency of the flow solver - to improve the efficiency of partitioned fluid-structure interaction iterations. Coupling not only the structure with the fine flow mesh, but also with the coarse flow mesh (often present due to the multigrid scheme) leads to a significant efficiency improvement. As solution of the flow equations typically takes much longer than the structure solve, and as multigrid is not standard in structure solvers, we do not coarsen the structure or the interface. As a result, the two level method can be easily implemented into existing solvers.*

*Two types of two level algorithms were implemented: 1) Coarse grid correction of the partitioning error and 2) Coarse grid prediction or full multigrid to generate a better initial guess. The resulting schemes are combined with a fourth-order Runge-Kutta implicit time integration scheme. For the linear, one-dimensional piston problem with compressible flow the superior stability, accuracy and efficiency of the two level algorithms is shown. The parameters of the piston problem were chosen such that both a weak and a strong interaction case were obtained.*

*Even the strong interaction case, with a flexible structure, could be solved with our new two level partitioned scheme with just one iteration on the fine grid. This is a major accomplishment as most weakly coupled methods fail in this case. Of the two algorithms the coarse grid prediction or full multigrid method was found to perform best. The resulting efficiency gain for our one-dimensional problem is around a factor of ten for the coarse to intermediate time steps at which the high order time integration methods should be run. For two- and three-dimensional problems the efficiency gain is expected to be even larger.*

## 1 INTRODUCTION

At present, computational power has increased to such an extent that coupled simulations are possible. Coupled simulations require the integration of flow and structural solvers, either by creating a new solver from scratch or by using existing solvers. A single (monolithic) solver is ideal but requires the CFD and CSM communities to develop a new single code, consuming a lot of time and effort before the end-result can be applied to real-world problems. Creating a solver based on the coupling of two existing codes, has drawbacks with respect to computational efficiency, but can benefit from the efforts already invested by the CFD and CSM communities. We have chosen for partitioned coupling, focusing our research to obtain coupling techniques with superior accuracy and efficiency.

In partitioned coupling the flow and structure equations are integrated separately in time. Interaction effects reduce to boundary conditions imposed at the interface staggered, or asynchronously, in time, see for example [4, 11]. The result is a block Jacobi or block Gauss-Seidel type of iteration over the flow and structure domain. So-called weakly coupled methods only require one flow and one structure solve per time step, whereas strongly coupled methods use multiple iterations per time step to converge to the fully coupled solution.

In general the high computational efficiency of weakly coupled methods is counteracted by stability problems that might occur and result in an undesirable time step restriction. This is especially problematic for problems with strong interactions where the flow and structure equally contribute to the interaction [9].

In this paper a two level technique is presented to improve the efficiency and stability of partitioned fluid-structure interactions. Coarse grid or more general multilevel techniques have been used in domain decomposition problems for parallel computing in CFD [7, 12]. Their application to a partitioned fluid-structure interaction problem is, to our best knowledge, new. Our aim is to improve the stability and accuracy of the first partitioned iteration such that more iterations are unnecessary. The result would then be major efficiency improvements of a superior weakly coupled method. The ultimate challenge for this method is solution of a fluid-structure problem with strong interaction.

Our approach is to combine a fourth-order accurate integration in time using Runge Kutta schemes with an accurate prediction strategy [1, 2, 13], with coarse grid acceleration of the partitioned iteration. For this method, next to the fine flow mesh, a coarse flow mesh is used. This coarse mesh is often present in the flow solver due to the for flows popular multigrid solution strategy. This coarse flow mesh is also coupled to the structure. As solution of the flow equations typically takes much longer than the structure solve, as the flow mesh at the fluid-structure interface typically is much finer than the structure mesh and as multigrid is not standard in structure solvers, we do not coarsen the structure or the interface. Therefore, the two level method can be easily implemented in existing solvers. In this paper both coarse grid correction of the partitioning error as well as coarse grid prediction of the initial guess (also called full multigrid) are investigated.

In this paper we restrict ourselves to a simplified case without a moving mesh, the classical 1D linear piston problem [10] for two reasons. To begin with, for this standard fluid-structure interaction case all new methods can be compared easily without the need for mesh moving algorithms that can significantly influence the temporal accuracy. Secondly, if we are able to obtain an efficiency gain for this 1D problem, we know that the gain for 2D and 3D problems will be even larger. For example for a coarsening factor of 2, the coarse mesh in 1D still has 1/2 cells of the fine mesh, while in 2D this is 1/4 and in 3D 1/8.

For our simple test problem, first, the governing flow and structure equations and the spatial and temporal discretization are given. Thereafter the partitioned method and the two level algorithms are discussed. Finally, results are shown applying the scheme to the piston test problem for two settings: 1) a weak interaction between the flow and a relatively stiff structure and 2) a strong interaction between the flow and a flexible structure.

## 2 LINEAR PISTON PROBLEM

The test case considered is the one-dimensional piston problem (Fig. 1), which is a common test case for fluid-structure interaction simulations [10].

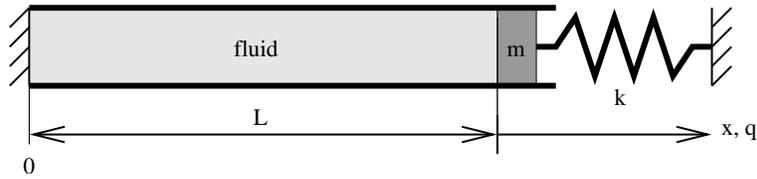


Figure 1: The one-dimensional piston problem.

### 2.1 Governing equations

The flow is one-dimensional, isentropic, and inviscid. Conservation for a deformable control volume  $K(t)$  yields

$$\frac{d}{dt} \int_{K(t)} U dx + \int_{\partial K(t)} [F(U) - \kappa U] \cdot n ds = 0, \quad (1)$$

wherein  $U = \begin{pmatrix} \rho \\ \rho u \end{pmatrix}$  the fluid state vector,  $\partial K(t)$  is the boundary of  $K$ ,  $F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \end{pmatrix}$  is the flux vector,  $\kappa$  is the velocity of the boundary  $\partial K$  and  $n$  is the unit normal vector perpendicular to  $\partial K$  pointing outwards. The governing equation (1) is given in the Arbitrary Lagrange Euler (ALE) formulation to cope with the moving domain [3]. The pressure is given by the isentropic relation  $p/\rho^\gamma$  is constant. The structural system is modelled as a single degree of freedom mass-spring system, with a mass  $m$  and a spring

stiffness  $k$ . The system is governed by the undamped one degree of freedom model

$$m\ddot{q} + kq = f_{\text{ext}}, \quad (2)$$

wherein  $\ddot{q}$  denotes the structural acceleration  $\frac{d^2}{dt^2}q$  and  $f_{\text{ext}}$  is the external force acting on the piston.

#### *Coupling between fluid and structure*

The coupling between fluid and structure is realized through the external force in (2) and posing boundary conditions on the fluid. The external force is the pressure force originating from the pressure difference between the ambient pressure  $p_0$  and the pressure in the fluid domain at the interface

$$f_{\text{ext}} = p(x = L + q) - p_0. \quad (3)$$

For the flow we prescribe two boundary conditions: one for the solid wall at  $x = 0$  and one for the interface with the piston at  $x = L + q$

$$u(x = 0) = 0, \quad u(x = L + q) = \dot{q}. \quad (4)$$

#### *Linearization*

The governing equations are linearized for small perturbations (denoted by  $'$ ) from an equilibrium state  $\bar{U}_0 = \begin{pmatrix} \bar{p}_0 \\ 0 \end{pmatrix}$ ,  $\bar{q}_0 = 0$ ,  $\bar{\kappa}_0 = 0$ . Satisfying the Geometric Conservation Law [5] implies that

$$\frac{d}{dt} \int_{\bar{K}_0 + \bar{K}'} \bar{U}_0 d\bar{x} - \int_{\partial(\bar{K}_0 + \bar{K}')} \bar{\kappa}' \bar{U}_0 \cdot n d\bar{s} = 0, \quad (5)$$

which is used to eliminate the ALE-terms in the linearized system and so we obtain

$$\frac{d}{dt} \int_{\bar{K}_0} \bar{U}' d\bar{x} + \int_{\partial\bar{K}_0} \bar{J}_0 \bar{U}' \cdot n d\bar{s} = 0, \quad (6)$$

with the Jacobian  $\bar{J}_0 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ . In (6) we non-dimensionalized the variables by using length  $L$ , speed-of-sound  $c_0$  and density  $\rho_0$ , all based on the values for the equilibrium solution:

$$\bar{t} = \frac{c_0}{L}t, \quad \bar{x} = \frac{x}{L}, \quad \bar{\rho} = \frac{\rho}{\rho_0}, \quad \bar{\rho}u = \frac{\rho u}{\rho_0 c_0}. \quad (7)$$

The structural equation yields

$$\bar{m}\ddot{\bar{q}}' + \bar{k}\bar{q}' = \bar{\rho}', \quad (8)$$

wherein  $\bar{m} = m/(\rho_0 L)$  the mass ratio between structural mass and fluid mass and  $\bar{k} = kL/(\rho_0 c_0^2)$ , the ratio between the structural spring and the fluid stiffness. In the linear, non-dimensional form, the boundary conditions yield

$$\bar{u}'(\bar{x} = 0) = 0, \quad \bar{u}'(\bar{x} = 1) = \dot{\bar{q}}'. \quad (9)$$

## 2.2 Space discretization

The flow equations (6) are discretized using a cell-centered Finite Volume (FV) method on a uniform mesh of  $N$  cells. We use a standard second order central scheme [6] without additional stabilization. In this way all numerical dissipation comes from the time integration scheme and possibly the partitioning algorithm. In the fluid domain we introduce two ghost cells  $K_0$  and  $K_{N+1}$  to cope with the boundary conditions in (9). The boundary condition at the wall is a symmetry condition. The boundary condition on the side of the moving piston is a perturbation of the same condition.

The structural equation (8) is transformed to a system of two ordinary differential equations

$$\frac{d}{dt}\bar{Q}' + \begin{pmatrix} 0 & \bar{\omega}^2 \\ -1 & 0 \end{pmatrix} \bar{Q}' = \begin{pmatrix} \frac{\bar{p}'_{N+1/2}}{\bar{m}} \\ 0 \end{pmatrix}, \quad (10)$$

wherein  $\bar{Q}' = \begin{pmatrix} \dot{q}' \\ q' \end{pmatrix}$ , the structural state vector and  $\bar{\omega} = \sqrt{k/\bar{m}}$  the structural natural frequency. For the external force on the structure we need  $\bar{p}'_{N+1/2}$ , which is obtained from a linear interpolation between the last fluid cells  $K_N$  and the ghost cell  $K_{N+1}$ .

When we denote all discrete state variables in a single vector  $w$ , the semi-discrete system is obtained

$$\dot{w} = Aw = \begin{pmatrix} A_s & A_{sf} \\ A_{fs} & A_f \end{pmatrix} \begin{pmatrix} w_s \\ w_f \end{pmatrix}, \quad (11)$$

wherein  $w_s = \bar{Q}'$  the structural state vector,  $w_f = \bar{U}'$  the fluid state vector. The matrix  $A_f$  is a  $2N \times 2N$  band-matrix which contains the discretization of the fluid domain and  $A_s$  is a  $2 \times 2$  matrix as in (10). The coupling matrices  $A_{fs}$  ( $2N \times 2$ ) and  $A_{sf}$  ( $2 \times 2N$ ) will generally only have a relatively small amount of non-zero entries, since the coupling only takes place at the interface between the structure and flow domain.

## 2.3 Time integration

For the time integration we only consider A-stable, implicit time integration methods. For these methods the time step is limited by accuracy considerations rather than stability, which is particularly important for engineering problems. We use the first order implicit Euler time integration method to explain the partitioning and coarse grid correction algorithm in the next two sections, and to show some properties of the resulting scheme. Final results are presented applying the coarse grid correction algorithm to a fourth order Additive Runge-Kutta (ARK) scheme [8].

### *Implicit Euler*

The well-known first order implicit Euler scheme applied to Eq. (11) is:

$$\left( I - \Delta t \begin{pmatrix} A_s & A_{sf} \\ A_{fs} & A_f \end{pmatrix} \right) \begin{pmatrix} w_s \\ w_f \end{pmatrix}^{n+1} = \begin{pmatrix} w_s \\ w_f \end{pmatrix}^n. \quad (12)$$

### Additive Runge-Kutta

ARK schemes are multi-stage methods, for which the solution at  $t^{n+1}$  can be made of arbitrary high order accuracy by a clever combination of the stage values, while ensuring A-stability. For every stage  $k$  we solve

$$w^{(k)} = w^n + \Delta t \sum_{i=1}^k a_{ki} A w^{(i)}, \quad k = 1 \dots s, \quad (13)$$

where  $w^{(i)}$  is the solution at stage  $i$ . After computing  $s$  stages, the solution at the next time level is

$$w^{n+1} = w^n + \Delta t \sum_{i=1}^s b_i A w^{(i)}. \quad (14)$$

In this paper we consider a fourth order ARK method, which combines an ESDIRK (Explicit first stage, Single diagonal coefficient, Diagonally Implicit Runge-Kutta) and ERK (Explicit Runge-Kutta) scheme. The coefficients  $a_{ki}$  and  $b_i$  are usually presented in a Butcher tableau (here represented with  $s = 5$ ):

$c_1$	0	0	0	0	0	$c_1$	0	0	0	0	0
$c_2$	$a_{21}$	$a_{22}$	0	0	0	$c_2$	$\hat{a}_{21}$	0	0	0	0
$c_3$	$a_{31}$	$a_{32}$	$a_{33}$	0	0	$c_3$	$\hat{a}_{31}$	$\hat{a}_{32}$	0	0	0
$c_4$	$a_{41}$	$a_{42}$	$a_{43}$	$a_{44}$	0	$c_4$	$\hat{a}_{41}$	$\hat{a}_{42}$	$\hat{a}_{43}$	0	0
$c_5$	$a_{51}$	$a_{52}$	$a_{53}$	$a_{54}$	$a_{55}$	$c_5$	$\hat{a}_{51}$	$\hat{a}_{52}$	$\hat{a}_{53}$	$\hat{a}_{54}$	0
	$b_1$	$b_2$	$b_3$	$b_4$	$b_5$		$b_1$	$b_2$	$b_3$	$b_4$	$b_5$
	ESDIRK						ERK				

The time level of the stage is  $t^{(i)} = t^n + c_i \Delta t$  and the coefficients  $c_i$  simply follow from  $c_i = \sum_j a_{ij}$ . For the ARK methods the  $c_i$  and  $b_i$  coefficients are identical for the ESDIRK and ERK schemes. The Butcher tableaux for the fourth order ARK-method used in this paper can be found in [8]. In this paper a new coarse grid correction scheme is presented and combined with the fourth-order ARK time integration method.

### 3 PARTITIONING ALGORITHM

We apply a block Gauss-Seidel or multiplicative Schwartz type of partitioning to the discrete system of equations. First the partitioning algorithm is discussed for the implicit Euler time integration method. Then the partitioning algorithm for the ARK-method is shown.

#### Implicit Euler

For the implicit Euler method, Eq.(12), the first iteration, starting the integration with the structure, gives:

$$(I - \Delta t A_s) w_s^{n+1} = w_s^n + \Delta t A_{sf} w_f^*. \quad (15)$$

wherein the coupling term  $A_{sf}w_f^*$  needs to be predicted. Next the fluid is integrated

$$(I - \Delta t A_f)w_f^{n+1} = w_f^n + \Delta t A_{fs}w_s^{n+1}. \quad (16)$$

As predictor in Eq. (15) we use  $w_f^* = w_f^n$ . In matrix notation this is:

$$\left( I - \Delta t \begin{pmatrix} A_s & 0 \\ A_{fs} & A_f \end{pmatrix} \right) \begin{pmatrix} w_s \\ w_f \end{pmatrix}^{n+1} = \begin{pmatrix} I & \Delta t A_{sf} \\ 0 & I \end{pmatrix} \begin{pmatrix} w_s \\ w_f \end{pmatrix}^n. \quad (17)$$

In optional subsequent iterations, this prediction is replaced by the best possible guess, i.e. the result from the previous iteration,  $w_f^* = w_f^{n+1}$ .

#### Additive Runge-Kutta

For the ARK-method, Eq. (13), the first partitioned Gauss-Seidel iteration, starting the integration with the structure, gives:

$$(I - a_{kk}\Delta t A_s)w_s^{(k)} = E_s^{(k)} + a_{kk}\Delta t F_{sf}^*. \quad (18)$$

wherein the coupling term  $F_{sf}^* = A_{sf}w_f^*$  needs to be predicted. Next the fluid is integrated

$$(I - a_{kk}\Delta t A_f)w_f^{(k)} = E_f^{(k)} + a_{kk}\Delta t A_{fs}w_s^{(k)}. \quad (19)$$

In (18) and (19) the explicitly known contributions are

$$E_s^{(k)} = w_s^n + \Delta t \sum_{i=1}^{k-1} a_{ki}(F_s^{(i)} + F_{sf}^{(i)}), \quad (20)$$

$$E_f^{(k)} = w_f^n + \Delta t \sum_{i=1}^{k-1} a_{ki}(F_f^{(i)} + F_{fs}^{(i)}), \quad (21)$$

with the fluxes defined by  $F_s^{(i)} = A_s w_s^{(i)}$ ,  $F_{sf}^{(i)} = A_{sf} w_f^{(i)}$ ,  $F_f^{(i)} = A_f w_f^{(i)}$  and  $F_{fs}^{(i)} = A_{fs} w_s^{(i)}$ . Optional next iterations work in the same way as described for the implicit Euler method.

The predictor for  $F_{sf}^* = A_{sf}w_f^*$  for the fourth-order ARK method is chosen as in [13]:

$$F_{sf}^* = \sum_{i=1}^{k-1} \frac{\hat{a}_{ki} - a_{ki}}{a_{kk}} F_{sf}^{(i)}, \quad (22)$$

which is a predictor for the coupling flux at stage  $k = 2 \dots s$ . In the present case, the flux predictor is equal to a state predictor for  $w_f$ , since  $A_{sf}$  is constant. Substitution of the predictor (22) into (18) shows that in fact we solve

$$(I - a_{kk}\Delta t A_s)w_s^{(k)} = w_s^n + \Delta t \sum_{i=1}^{k-1} a_{ki}F_s^{(i)} + \Delta t \sum_{j=1}^{k-1} \hat{a}_{kj}F_{sf}^{(j)}. \quad (23)$$

In (23) the coupling fluxes are treated in a consistent, explicit way in order to retain the higher order of the scheme. Due to the combined implicit/explicit nature of the scheme, it is referred to as IMEX.

## 4 TWO LEVEL ALGORITHMS

Solution of the monolithic system of equations, Eq. (12), on the fine grid is written as:

$$\left( I - \Delta t \begin{pmatrix} A_s & A_{sf} \\ A_{fs} & A_f \end{pmatrix}_h \right) \begin{pmatrix} w_s \\ w_f \end{pmatrix}_h^{n+1} - \begin{pmatrix} w_s \\ w_f \end{pmatrix}_h^n = 0. \quad (24)$$

In our partitioned algorithm Eq.(17), where we ignore the implicit coupling between the flow and structure in the computation of the structure, we actually solve a different system. This leads to a partitioning error. In this section we propose two two level algorithms: a coarse grid correction, and a coarse grid prediction or full multigrid algorithm. For the coarse grid correction algorithm, the solution from one partitioned block Gauss Seidel iteration computed on a fine grid is corrected with a defect computed on a coarse grid. For the coarse grid prediction algorithm, a prediction for the initial guess of the block Gauss-Seidel iteration is made on a coarse grid.

### 4.1 Coarse grid correction algorithm

The basic idea of the coarse grid correction algorithm is to correct the solution obtained after one partitioned iteration with a coarse grid approximation of the partitioning error. The method, here explained for implicit Euler time integration, works as follows: to begin with, partitioned iteration of Eq. (24), starting with the structure and using the previous time level as initial guess is:

$$\left( I - \Delta t \begin{pmatrix} A_s & 0 \\ A_{fs} & A_f \end{pmatrix}_h \right) \begin{pmatrix} \tilde{w}_s \\ \tilde{w}_f \end{pmatrix}_h^{n+1} - \begin{pmatrix} I & \Delta t A_{sf} \\ 0 & I \end{pmatrix}_h \begin{pmatrix} w_s \\ w_f \end{pmatrix}_h^n = 0. \quad (25)$$

Substituting the partitioned solution  $\tilde{w}$  into the original equation (24) we find the residual:

$$\begin{pmatrix} r_s \\ r_f \end{pmatrix}_h = -\Delta t \begin{pmatrix} A_{sf}(w_f^n - w_f^{n+1}) \\ 0 \end{pmatrix}_h \quad (26)$$

Not surprisingly the residual in the flow is zero, due to the fact that we have chosen a block Gauss-Seidel type iteration, taking into account the latest structure solution in the flow domain. For a Jacobi type iteration the residual would be non-zero in both flow and structure. Combining Eq. (24), (25) and (26) we obtain the following equation for the error or defect  $\epsilon_h = w_h - \tilde{w}_h$ :

$$\left( I - \Delta t \begin{pmatrix} A_s & A_{sf} \\ A_{fs} & A_f \end{pmatrix}_h \right) \begin{pmatrix} \epsilon_s \\ \epsilon_f \end{pmatrix}_h^{n+1} = \begin{pmatrix} r_s \\ r_f \end{pmatrix}_h, \quad \text{or} \quad A_h \epsilon_h = r_h. \quad (27)$$

In a classic multigrid fashion we now approximate the error on a coarse grid by solving the coarse grid variant of Eq. (27):

$$A_H \epsilon_H = R r_h, \quad (28)$$

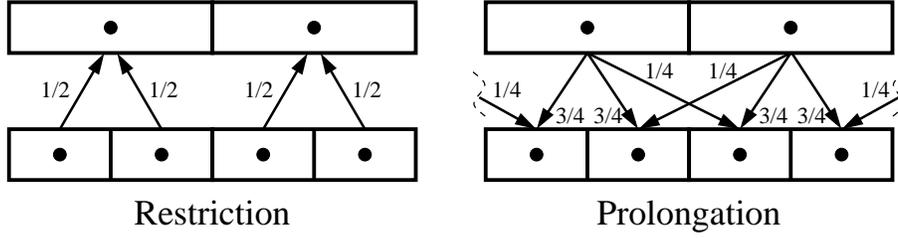


Figure 2: Restriction by summation and prolongation by linear interpolation

where  $A_H = RA_hP$  is the matrix for the coarse grid,  $R$  is a restriction operator, and  $P$  is a prolongation operator. Since the coarse mesh has fewer cells than the fine grid, solution of the coarse grid equation (28) is relatively cheap. The corrected solution then becomes:

$$\hat{w}_h = \tilde{w}_h + P\epsilon_H = \tilde{w}_h + P(A_H)^{-1}Rr_h. \quad (29)$$

In order to facilitate partitioned solution of the flow and structure part by separate solvers the restriction operator  $R$  and the prolongation operator  $P$  are chosen to be:

$$R = \begin{pmatrix} R_s & 0 \\ 0 & R_f \end{pmatrix}, \quad P = \begin{pmatrix} P_s & 0 \\ 0 & P_f \end{pmatrix} \quad (30)$$

Consequently, the coarse grid correction equation (28) becomes:

$$\left( I - \Delta t \begin{pmatrix} R_s A_s P_s & R_s A_s P_f \\ R_f A_f P_s & R_f A_f P_f \end{pmatrix} \right) \begin{pmatrix} \epsilon_s \\ \epsilon_f \end{pmatrix}_H^{n+1} = \begin{pmatrix} R_s r_s \\ R_f r_f \end{pmatrix}. \quad (31)$$

Along the same lines as the original equation for the solution on the fine grid, the coarse grid correction equation (31) is solved in a partitioned way using block Gauss-Seidel iterations. The predictor for the error in the flow  $\epsilon_f^*$  on the coarse grid is here chosen to be zero.

Multigrid is a relatively common algorithm in flow solvers, but not in structure solvers. Therefore we have in the present paper chosen only to coarsen the flow mesh. As a consequence, implementation of the two level algorithm, does not imply major changes to the existing flow and structure solver. Since solution of the, in general non-linear, flow equations is much more demanding than solution of the structure equations, we do not expect this to have a large influence on the computational time. The resulting restriction and prolongation operators for the structure domain are simply the identity matrix:  $R_s = P_s = I$ . The restriction operator in the fluid domain is a standard summation operator and prolongation is performed through linear interpolation. The restriction and prolongation operators are graphically represented in Fig. 4.1. In the figure it is assumed that the coarse grid to fine grid cell ratio is one half and that the meshes are uniform.

For partitioned solution using the fourth order ARK method derivation of the coarse grid correction equation follows the same lines. One important difference is that the coarse

grid correction is now applied to the stage solutions, that is, to Eqs. (15) and (16). This means that, in Eq. (29),  $\hat{w}_h$  is now a stage solution  $\tilde{w}_h^{(k)}$ , and in Eqs. (27) and (28) a different residual has to be used. This ARK residual is equal to

$$\begin{pmatrix} r_s \\ r_f \end{pmatrix}_h^{(k)} = \Delta t \begin{pmatrix} A_{sf} \tilde{w}_f^{(k)} \\ 0 \end{pmatrix}_h - \Delta t \sum_{i=1}^{k-1} (\hat{a}_{ki} - a_{ki}) \begin{pmatrix} A_{sf} w_f^{(i)} \\ 0 \end{pmatrix}_h \quad (32)$$

wherein  $\tilde{w}^{(k)}$  are stage solutions from the ARK scheme.

## 4.2 Coarse grid prediction algorithm

In the coarse grid prediction algorithm the explicit prediction or initial guess for the first block Gauss-Seidel iteration is corrected using a coarse grid solve. This contrasts with the coarse grid correction algorithm, which works on the solution obtained after a block Gauss-Seidel iteration. In order to maintain design order for this method for the ARK scheme we start with a high order initial guess, resulting from explicit time integration with the explicit Runge-Kutta method. For our initial analysis for the implicit Euler time integration scheme, starting with the trivial guess, the solution at the previous time level  $n$  is enough. Starting with explicit time integration of the entire system we obtain:

$$\begin{pmatrix} \tilde{w}_s \\ \tilde{w}_f \end{pmatrix}_h^{n+1} - \begin{pmatrix} I + \Delta t A_f & \Delta t A_{sf} \\ \Delta t A_{fs} & I + \Delta t A_s \end{pmatrix}_h \begin{pmatrix} w_s \\ w_f \end{pmatrix}_h^n = 0. \quad (33)$$

Substituting  $\tilde{w}$  into the original monolithic implicitly discretised equation (24) we find the following residual:

$$\begin{pmatrix} r_s \\ r_f \end{pmatrix}_h = -\Delta t \begin{pmatrix} A_s & A_{sf} \\ A_{fs} & A_f \end{pmatrix} \begin{pmatrix} (w_f^n - w_f^{n+1}) \\ (w_s^n - w_s^{n+1}) \end{pmatrix}_h \quad (34)$$

Along the same lines as for the coarse grid correction algorithm Eq. (29) the equation for the corrected prediction is obtained:

$$w_h^* = \tilde{w}_h + P\epsilon_H = \tilde{w}_h + P(A_H)^{-1} Rr_h. \quad (35)$$

Finally, this corrected prediction is used in the first block Gauss-Seidel iteration to obtain the improved solution:

$$\begin{aligned} & \left( I - \Delta t \begin{pmatrix} A_s & 0 \\ A_{fs} & A_f \end{pmatrix}_h \right) \begin{pmatrix} \tilde{w}_s \\ \tilde{w}_f \end{pmatrix}_h^{n+1} - \\ & \begin{pmatrix} 0 & \Delta t A_{sf} \\ 0 & 0 \end{pmatrix}_h \begin{pmatrix} w_s^* \\ w_f^* \end{pmatrix}_h - \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}_h \begin{pmatrix} w_s \\ w_f \end{pmatrix}_h^n = 0. \end{aligned} \quad (36)$$

The coarse grid prediction algorithm can be applied to the stage solutions of ARK methods in a similar fashion as described above for the coarse grid correction algorithm. The first

method starts each stage with the trivial guess, the solution of the previous time step. For the second method explicit integration of Eq.(33) is replaced with the Explicit Runge Kutta scheme from the ARK pairs discussed in Section 2.3. The final solution is obtained solving Eq.(24) with a single Gauss-Seidel type iteration using  $w_h^*$  as the initial guess.

## 5 RESULTS AND DISCUSSION

In this section results applying the coarse grid correction and the coarse grid prediction scheme to the 1D linear piston problem are shown. First, properties of the two level algorithms are investigated for the implicit Euler scheme. Then, stability, accuracy and efficiency of the two level algorithm in combination with the fourth order Runge Kutta time integration scheme is investigated. The settings of the piston problem are chosen such that we have two cases: 1) a weak interaction between the flow and a relatively stiff structure and 2) a strong interaction between the flow and a relatively light and flexible structure.

The ratio of the structure mass and the mass of the fluid trapped in the tube is  $\bar{m}$ , while  $\bar{k}$  is a characteristic number for the ratio of the spring force and the pressure force acting on the piston. For the first, standard, test case,  $\bar{m}$  and  $\bar{k}$  were scaled in such a way that the effect of the fluid and structure on each other are roughly of the same order of magnitude:  $\bar{m} = 2.0$  and  $\bar{k} = 1.429$ . This gives a period of the coupled problem equal to  $P = 6.1916$ . For the second test case with a more flexible structure  $\bar{m} = 0.2$  and  $\bar{k} = 1.429$ . So the spring stiffness of the structure is a factor 100 smaller than that of the flow. This results in a period of the coupled problem of  $P = 3.2763$ . All results, except those for flexible structures presented in Section 5.4, concern the first standard test case. For both investigations, we have used the same simulation setup. The fluid is spatially discretized using a central second order scheme. Time steps are given relative to the periods of the coupled problem  $P$ .

### 5.1 Properties of two level algorithms

First the properties of the coarse grid correction and the coarse grid prediction schemes are investigated for the implicit Euler time integration scheme. For this first investigation, on the coarse level always monolithic solution of the FSI equations is performed. The two level schemes are compared to the sub-iterated block Gauss-Seidel partitioning algorithm. For these results obtained with the implicit Euler scheme we used 64 finite volume cells and a time step equal to  $P/5$ .

#### Monolithic coarse grid correction

In Figure 3 the fine grid partitioning error in the density of the fluid after 1 time step. The partitioning error is computed with respect to monolithic solution on the fine grid with the same time step. In Figure 3(a) the block Gauss-Seidel schemes with one and two fine grid partitioned iterations, GS1 and GS2, are compared with the coarse grid

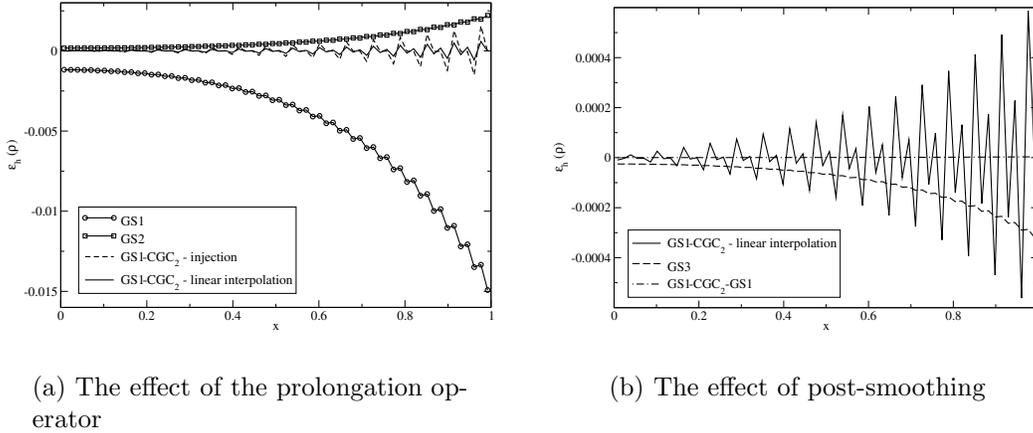


Figure 3: Fine grid partitioning error in the fluid density after one implicit Euler time step for several partitioning schemes, time step =  $P/5$ , 64 finite volume cells

correction algorithm, GS1-CGC<sub>2</sub>, with the injection and the linear prolongation operator. Here the subscript <sub>2</sub> stands for a coarsening factor of 2. All methods start with one Gauss-Seidel step. The methods with only one Gauss-Seidel step GS1 shows to have the largest partitioning error in the density. The error itself has two distinct modes. One mode with a small wave number and a large amplitude and superimposed to it a mode with a large wave number and a smaller amplitude. An additional fine grid Gauss-Seidel iteration (GS2) reduces the amplitude of both modes. The coarse grid correction effectively eliminates the mode with the small wave number, while high wave number modes, which cannot be presented on the coarse mesh remain virtually unchanged. Linear interpolation performs better than injection. Hence, injection is not considered for future simulations. The coarse grid correction scheme with one fine grid iteration has a smaller error than the GS2 scheme which needs two fine grid iterations. In Figure 3(b) we zoom in to find that the error of GS3 is smaller than that of the coarse grid correction scheme GS1-CGC<sub>2</sub>. However, one additional fine grid Gauss-Seidel iteration, resulting in GS1-CGC<sub>2</sub>-GS1, resulted in the smallest error, at less work than GS3 which requires three fine grid solves. Apparently, a post-smoothing step after the coarse grid correction is beneficial.

### Level of coarsening

Next the level of coarsening is varied from 2 to 4 and 8, which means that the coarse grid contains 32, 16 and 8 finite volume cells, respectively. In Figure 4 the fine grid partitioning error in the density is presented for the GS1-CGC and GS1-CGC-GS1 algorithms. Here the subscript after CGC indicates the coarsening factor, i.e CGC<sub>2</sub> is a coarse grid correction scheme with a coarse mesh with 1/2 of the cells of the fine mesh. Fig. 4(a) shows that for all GS1-CGC methods a mode with a lower wave number is retained in the fine grid error when the coarsening ratio is increased. The maximum fine grid error,

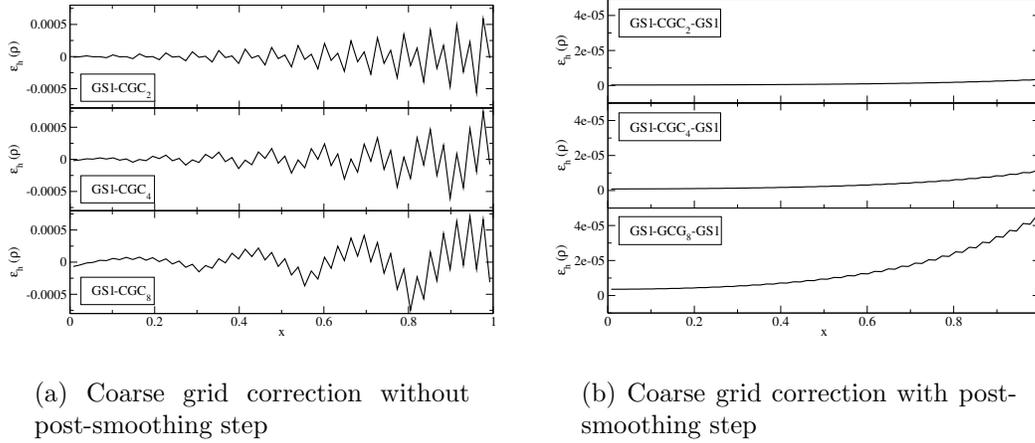


Figure 4: Fine grid partitioning error in the fluid density after one implicit Euler time step with different coarsening levels 2, 4 and 8

however, is approximately the same for all coarsening ratios. After an additional fine grid solve (GS1-CGC-GS1), the high wave number modes are effectively removed from the fine grid error, see 4(b). However, the error in the low wave number mode increases as the coarsening level increases. For the one-dimensional case the amount of work to solve the coarse grid system may still have a significant influence on the total amount of work for an entire simulation. The influence of resolving the coarse grid system on the total computing time decreases when two- and three-dimensional problems are considered. Therefore, it is expected that a more aggressive coarsening is not desired for multi-dimensional problems, since the decreased accuracy with higher coarsening levels is not compensated by the reduced computational effort.

### Monolithic coarse grid prediction

The results for the coarse grid correction algorithms already demonstrated that the partitioning error in the low wave number mode is effectively reduced by a monolithic coarse grid computation. In order to eliminate the high wave number mode from the error, an expensive, additional fine grid or post-smoothing iteration was required. In order to avoid a second fine grid iteration, but still have a smoothing of the error at the end of a time step, the coarse grid prediction algorithm can be used. As a first comparison, the error in the trivial guess,  $w^0$ , is shown together with the error after the coarse grid prediction based on the trivial guess (CGP) with a coarsening factor of 2 in Fig. 5(a). The coarse grid prediction truly provides a better initial guess than the trivial guess. When we zoom in to the error after coarse grid prediction in Fig. 5(b), we find that a high wave number is still present. This can be effectively eliminated with one post-smoothing Gauss-Seidel iteration, scheme CGP-GS1. The error of this CGP-GS1 scheme is even lower than that of

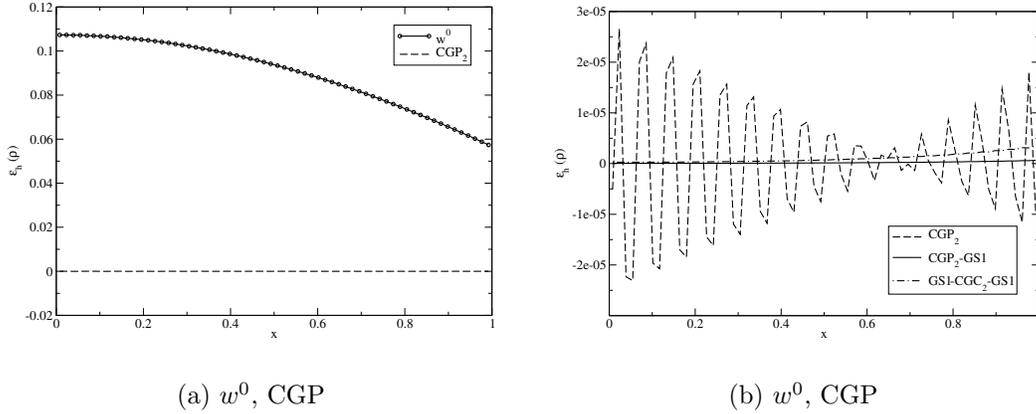


Figure 5: Fine grid partitioning error in the fluid density after one implicit Euler time step for CGP, CGP-GS, and GS-CGP-GS

the best coarse grid correction scheme GS1-CGP-GS1, which requires an additional fine grid solve. Note that the most efficient coarse grid prediction scheme CGP-GS1 results in a much lower partitioning error in the density than the original GS1 scheme, shown in Fig. 3. With coarse grid correction the maximum density error reduces from around 0.01 to  $1e-06$ , four orders of magnitude.

## 5.2 Two level ARK schemes

In this section, first the overall effect of the two level ARK algorithm on the solution, stability and the accuracy is investigated. Then it is demonstrated that the new scheme has a smaller error for a given time step, and that the two level schemes result in a higher accuracy for a fixed amount of work. Again, the coarse grid prediction scheme is found to outperform the other scheme by far.

### 5.2.1 Stability

First we investigate the stability of the new scheme. The system is integrated in time using ARK4-based schemes with a large time step equal to  $\Delta t = P/5$ . We used 256 finite volume cells and a coarsening factor of 2. The most interesting property to look at for a stability investigation is the total energy in the coupled system, which is equal to the sum of the potential and kinetic energy in the fluid and the structure. The system energy after 10 coupled periods is compared to the initial system energy. The results are plotted in Fig. 6. The figure shows that at this large time step, a loss of energy is visible even for the monolithic implicit fourth-order RK scheme. As all results were obtained using the non-dissipative second-order central space discretisation the energy loss for the monolithic solution originates solely from the time integration scheme. The IMEX ARK scheme with one Gauss-Seidel partitioned iteration, however, shows a rapid

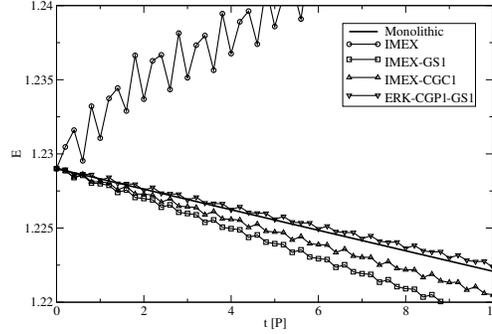


Figure 6: System energy  $E$  compared to initial system energy  $E_0$ , ARK4, 256 finite volume cells,  $\Delta t = \frac{P}{5}$

increase in system energy, which is consistent with what one expects of (partially) explicit time integration schemes, especially at large time steps. The IMEX scheme with one additional fine grid iteration, i.e. two iterations, (IMEX-GS1) is stable, but dissipates a significant amount of energy. The new schemes with coarse grid correction and coarse grid prediction, clearly are superior to the IMEX scheme. However, the coarse grid correction scheme (IMEX-CGC1) displayed - using just one Gauss-Seidel iteration on the coarse grid - does not bring down the system energy to the level of the monolithic scheme. Note that performing an additional iteration on the fine grid is significantly worse than adding a much cheaper coarse grid solve. The coarse grid prediction scheme starting with explicit RK time integration (ERK-CGP1-GS1) with one coarse grid and one fine grid Gauss-Seidel iteration performs best, since its error is closest to that of the monolithic scheme, so has the smallest partitioning error.

### 5.2.2 Accuracy

Next, we investigate the accuracy of the new scheme after 10 coupled periods. The system is integrated in time with each scheme using time steps  $\Delta t$  ranging from  $P/5$  to  $P/1280$ . In space 256 finite volume cells were used. Interesting measures for accuracy are again the system energy, but now also the frequency error that results in a phase shift in the solution is studied. The frequency error follows from the error in speed of the structure  $dq/dt$ . Since we study the solution after exactly 10 periods, we know that  $dq/dt$  should be zero there. Frequency errors lead to a shift in the solution, a much larger contribution than amplitude errors make here. We are still only interested in the partitioning error, so both error measures are relative to the monolithic solution at the same time step. The error is absolute, so the sign of the frequency change is not measured. The results are plotted in Fig. 7. The error for the monolithic solution, i.e. compared to the exact solution is plotted for reference. This error consists of the temporal error, dominant for the larger time steps, and a spatial error dominant for the smaller time steps. For this case with 256 finite volume cells, the system energy the temporal error is dominant until  $\Delta t/P = 10^{-2.5}$ . Thereafter, the spatial error, approximately equal to  $10^{-11}$ , is dominant.

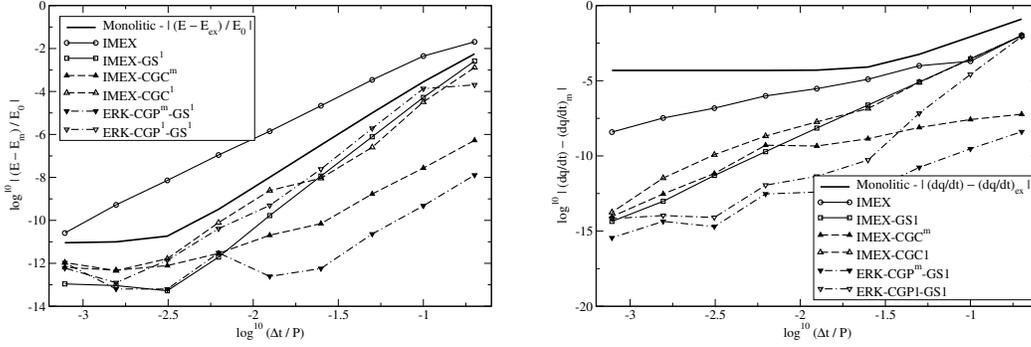


Figure 7: Partitioning error convergence for energy and frequency after 10 periods (compared to monolithic case), ARK4, 256 finite volume cells, coarsening factor=2. Monolithic total error (compared to the exact solution) included for reference.

For the frequency error the temporal error is dominant until  $\Delta t/P = 10^{-1.75}$ , whereafter the spatial error, approximately equal to  $10^{-4.5}$  is dominant. The figure shows that all methods show design order. The IMEX method has the largest error in both energy and frequency. For the frequency the partitioning error is always smaller than the temporal error of the monolithic scheme. However, for the system energy, the IMEX partitioning error in the system energy is always larger than the temporal error of the monolithic scheme. In other words, the partitioning error is the dominating source of error for the IMEX scheme. The plot also shows that for any time step, the two level schemes (IMEX-CGC and ERK-CGP-GS1) have a significantly smaller error than the IMEX method and the IMEX method followed by an extra fine grid iteration. Solving the system of the coarse grid just once (IMEX-CGC1 and ERK-CGP1-GS1), or iterating until convergence on the coarse grid (IMEX-CGCm and ERK-CGPm-GS1) here only makes a difference at all time steps of interest. However, solving the system just once on the coarse grid is enough to make the partitioning error an order smaller than the monolithic time integration error, so that the effect of monolithic coarse grid solution will not be noticeable. Note that the coarse grid prediction scheme is more accurate than the coarse grid correction scheme.

### 5.2.3 Efficiency

In order to compare the efficiency, the accuracy of schemes at a fixed amount of work has to be compared. The methods were implemented in Matlab. Since we found the timing routine of Matlab unreliable we decided to use estimates for the amount of work. Solving one implicit stage on the fine grid takes one unit of work. The ARK4 schemes have 5 implicit stages, so, for example, the IMEX scheme takes 10 work units per time step. A solve on the coarse grid takes half the amount of work of a fine grid solve. This is a conservative estimate, as typically solving linear systems takes an amount of time that grows quadratically with the number of unknowns. So in reality a coarse grid solve is probably not this expensive.

The resulting plots of accuracy per unit of work can be found in Fig. 8. The schemes

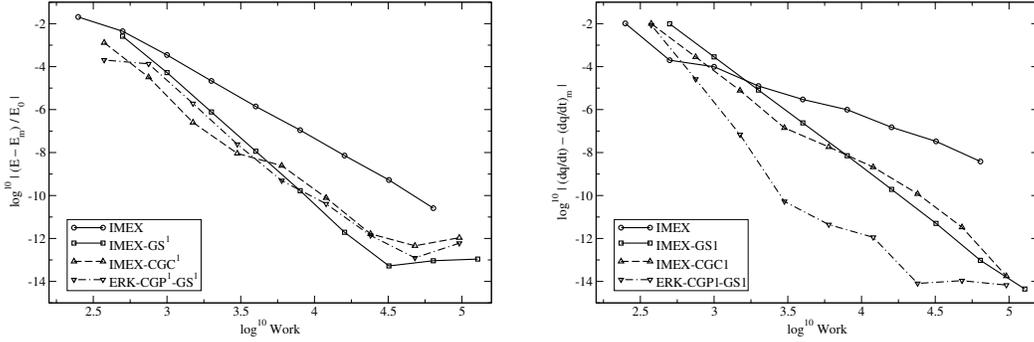


Figure 8: Accuracy of energy and frequency versus units of work after 10 periods relative to monolithic case ARK4, 256 finite volume cells, coarsening factor = 2

involving monolithic solves on the fine or the coarse grid are not in the figure. The figure shows that the schemes using a coarse grid prediction are significantly more efficient than all other methods. For an energy error of  $10^{-3}$  compared to the monolithic solution the efficiency gain is equal to a factor of more than two. For an energy error of  $10^{-8}$  the gain is a factor of four. For a frequency error of  $10^{-3}$  compared to the monolithic solution we also obtain an efficiency increase of a factor of two. This gain is even larger at smaller error tolerances. However, as the spatial error in the frequency is  $10^{-4.5}$  there is no use in obtaining a temporal error much smaller than that.

#### 5.2.4 Stronger coupled problem with light weight structure

Now it is interesting to see how well the resulting weakly coupled partitioned schemes, which uses only one iteration over flow and structure on the fine grid, perform for light weight structures. As explained earlier weak coupling techniques typically fail for these kind of interactions. We consider the second test case, introduced earlier, where the mass is a factor 10 smaller, equal to  $m = 0.2$ , while the spring stiffness is still the same  $k = 1.429$ . The time step is chosen equal to  $P/10$  and we use 256 finite volume cells. Fig. 9 shows the total system energy plotted against time. The figure clearly shows how poor the explicit integration in the IMEX scheme behaves for light weight structures. This was a known shortcoming of the IMEX scheme and is expected for such a weakly coupled partitioned scheme. In order to obtain a sufficiently strongly coupled scheme, subiterations are used. For all methods considered, two subiterations are used, either on the fine grid (IMEX-GS2) or on the coarse grids (IMEX-CGC2 and ERK-CGP2-GS1). The subiterations stabilize the scheme and increase the accuracy of the solution. The results in Fig. 9(b) indicate that approximately the same results can be obtained on a 8 times coarser mesh as on a 2 times coarser mesh. Of course the coarse grid solve on a 8 times coarser mesh is less expensive than a coarse grid solve on a 2 times coarser mesh.

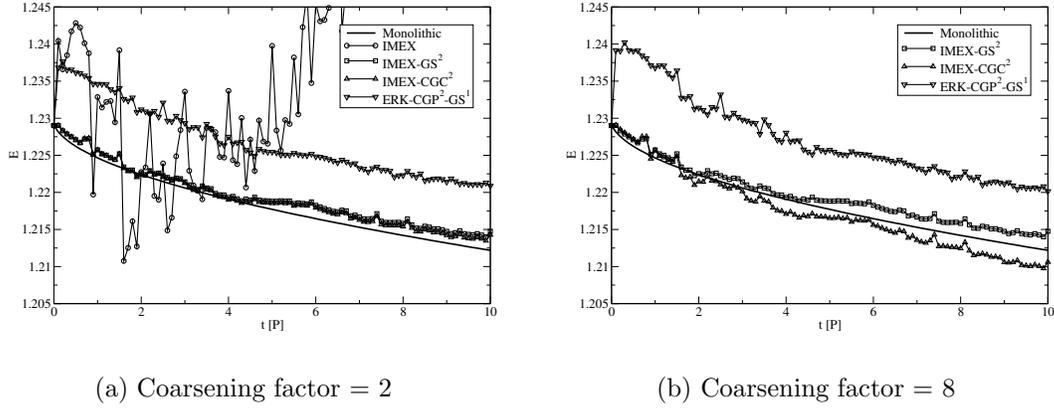


Figure 9: Total system energy after 10 periods,  $\bar{m} = 0.2$  and  $\bar{k} = 1.429$ , ARK4, 256 finite volume cells

This illustrates the potential of the two-level algorithm: when subiterations are necessary to obtain a stronger coupling, the expensive fine grid flow solve is effectively eliminated and replaced by a much smaller, but equally effective coarse grid solve.

In Fig. 10(a) the partitioning error for several schemes is shown versus the time step. The IMEX scheme has a partitioning error  $\approx 100\times$  larger than the time discretization

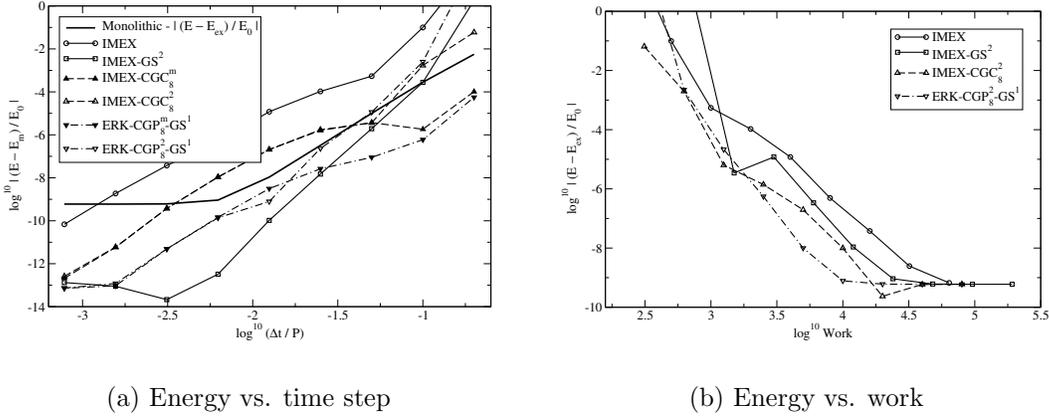


Figure 10: Total system energy after 10 periods,  $\bar{m} = 0.2$  and  $\bar{k} = 1.429$ , ARK4, 256 finite volume cells,  $cf = 8$

error, which is clearly the dominating error in the solution. Two subiterations on the fine mesh (IMEX-GS2) helps to increase the convergence of the partitioning error so that for  $\Delta t < P/10$  the partitioning error is smaller than the time discretization error. The monolithic coarse grid correction and prediction algorithms are particularly effective at large time steps, e.g. reducing the partitioning error by a factor  $10^5$  compared to IMEX for  $\Delta t = P/10$ . The monolithic CGP is  $\approx 100\times$  more accurate compared to monolithic

CGC for time steps  $\Delta t < P/20$ . The monolithic CGC and CGP show the maximum gain in accuracy that can be obtained when the solution on the coarse grid is subiterated, hence for large time steps the two subiterations performed on the coarse grid are not sufficient to obtain the monolithic coarse grid solution. Fig. 10(b) shows the convergence of the time discretization error with the amount of work performed to obtain the solution after 10 periods. The amount of work is expressed as the number of implicit fine-grid systems that need to be solved during the simulation. For the IMEX scheme this is equal to the number of time steps in the simulation times the number of implicit stages (5) for ARK4. For IMEX-GS2 the two subiterations on the fine grid make the algorithm 3 times as expensive as the IMEX scheme, whereas the coarse grid algorithms only require 25% additional work compared to IMEX. The efficiency gains due to the subiterations on an inexpensive coarse grid are clear: apart from a single point at which the partitioning and time discretization errors are nearly identical and counteracting for the IMEX-GS2, the two-level approach requires less computational work than the IMEX and IMEX-GS2 schemes, e.g. for an accuracy of  $10^{-5}$ , the coarse grid methods require between 2.5 and 3 times less work than IMEX and IMEX-GS2.

## 6 CONCLUSIONS

Two new two level partitioned coupling schemes for fluid-structure interaction problems were designed: a coarse grid correction and a coarse grid prediction algorithm. In these schemes not only the fine flow mesh is coupled to the structure, but also the coarse flow mesh (often present due to the common multigrid algorithm) is coupled to the structure. Results for this new scheme for a 1D linear piston problem are compared with results for the original weakly coupled fourth-order in time IMEX scheme and the fourth-order in time monolithic solution.

The study shows that the two level schemes can overcome some of the important problems of the weak coupling of the IMEX scheme and other coupling schemes at a low computational expense. Both the coarse grid correction and the coarse grid prediction scheme are more stable than the IMEX scheme. Also both schemes are significantly more accurate and more efficient than the IMEX scheme.

In addition, it was found that the coarse grid correction or prediction do not have to be computed monolithically. Even a single-step Gauss-Seidel iteration on the coarse grid is enough to significantly improve the accuracy of the simulation. For very strongly coupled problems with light weight structures, the coarse grid prediction scheme is far superior to the others on all accounts that matter: stability, accuracy, and efficiency.

It is expected that the results, here obtained for a one dimensional flow problem, will also be valid for two- and threedimensional problems. For the two- and threedimensional problems the expected efficiency gain is evenmore, as for the same coarsening factor, the number of mesh cells is reduced. For example, a coarsening factor of 1/2 leads in 1D to a coarse mesh with 1/2 the number of cells of the fine mesh, while in 2D the coarse mesh has only 1/4, and in 3D 1/8 the number of fine mesh cells.

**REFERENCES**

- [1] H. Bijl, M.H. Carpenter, V.N. Vatsa, and C.A. Kennedy, Implicit Time integration schemes for the unsteady compressible Navier-Stokes equations: laminar flow, *J. Comput. Phys.*, 179, 1-17, 2002.
- [2] M.H. Carpenter, C.A. Kennedy, H. Bijl, S.A. Viken and V.N. Vatsa, Fourth-order Runge-Kutta schemes for fluid mechanics applications, *J. Sci. Comput.*, vol. 25(1), pp157-194, 2005.
- [3] J. Donea, An arbitrary Lagrangian-Eulerian finite element method for transient fluid-structure interactions, *Comput. Methods Appl. Mech. Engrg.*, 33, 689-723, 1982.
- [4] C.A. Felippa, K.C. Park, and C. Farhat, Partitioned analysis of coupled mechanical systems, *Comput. Methods Appl. Mech. Engrg.*, 190, 3247-3270, 2001.
- [5] H. Guillard, and C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, *Comp. Meth. Appl. Mech. Engrg.*, vol. 190, pp. 1467-1482, 2000.
- [6] C. Hirsch, *Numerical Computation of internal and external flows. Vol. 1: Fundamentals of numerical discretization.* John Wiley & Sons Ltd., 1994.
- [7] C.B. Janssen, and P.A. Weiserfelt, Coarse grid correction scheme for implicit multi-block Euler calculations, *AIAA J.*, 33(1), 1816-1821, 1995.
- [8] C.A. Kennedy and M.H. Carpenter, Additive Runge-Kutta Schemes for Convection-Diffusion-Reaction Equations, *Appl. Num. Math.*, vol. 44(1-2), pp. 139-181, 2002.
- [9] C. Michler, *Efficient numerical methods for fluid-structure interaction*, PhD-thesis, Delft University of Technology, 2005.
- [10] S. Piperno, C. Farhat, and Larrouturou B., Partitioned procedures for the transient solution of coupled aeroelastic problems. Part I: Model problem, theory and two-dimensional application, *Comput. Methods Appl. Mech. Engrg.*, 124, 79-112, 1995.
- [11] S. Piperno, and C. Farhat, Partitioned procedures for the transient solution of coupled aeroelastic problems – Part II: energy transfer analysis and three-dimensional applications, *Comp. Meth. Appl. Mech. Engrg.*, vol. 190, pp. 3147-3170, 2001.
- [12] B. Smith, P. Bjorstad, and W. Gropp, *Domain decomposition*, Cambridge University Press, 1996.
- [13] A.H. van Zuijlen, and H. Bijl, Implicit and explicit higher order time integration schemes for structural dynamics and fluid-structure interaction computations, *Comp. Struct.*, 83 (2-3): 93-105, 2005.