V International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2013
S. Idelsohn, M. Papadrakakis and B. Schrefler (Eds)

COUPLING GENERALISED-α METHODS: ANALYSIS, ADAPTIVITY, AND NUMERICS

JOACHIM RANG*

*Institute of Scientific Computing
TU Braunschweig
Hans-Sommer Str. 65, 38106 Braunschweig, Germany
e-mail: j.rang@tu-bs, web page: http://www.wire.tu-bs.de

Key words: Coupled Problems, generalised-α methods

Abstract. In this article we consider the generalised-α methods, make an analysis of the methods and apply them to a coupled model problem. A new adaptive timestep control is presented.

1 INTRODUCTION

Coupled problems appear in different research areas. One common example is the interaction of structure and fluid [DR08], e.g. the numerical simulation of offshore wind turbines, see [MM04], or of biomechanical processes. Coupled problems consist of two or more different physical problems which are in general space and time dependent. The discretisation in space leads to a high dimensional system of ordinary differential equations (ODEs) or differential algebraic equations (DAEs). The computation of the numerical solution needs the simultaneous solution of the strong coupled equations of each problem. But often for each subproblem different discretisation schemes are used. In the case of fluid-structure interaction the fluid is often discretised with Finite Volumes, and the structure with Finite Elements. To build a monolithic solver [RB00] it is often difficult to find a cost free available software system which processes different discretisation methods for different problem classes.

This is one reason to use a modular approach and partitioned methods [RB00, FP80, MW01, PFL95, MS02, MNS06], i.e. the subproblems are solved by different codes which communicate with each other. The communication between the solvers can be realised with the help of the Component Template Library (CTL), i.e. the solvers are transformed into software components and are controlled with an independent central unit. In [RSM09] the CTL is used to solve FSI problems.

In this paper we consider the generalised-α methods, which are introduced for first order ODEs in [JWH00], and for second order ODEs in [CH93]. The generalised-α methods are in general of second order and allow the damping of high frequencies, which can be controlled by certain parameters. An analysis for first order problems can be found in [DP03]. In the case of second order ODEs many papers can be found, which analyse the generalised-α method, for example [EBB02]. It is well known that the generalised-α method for first order problems can be formulated as onestep and as multistep method. In the case of second order methods this statement is only true, if the ODE is linear in the first derivative (see [EBB02]). For both classes of multistep methods second order can be achieved if a further condition is satisfied. Together with the stability conditions (see [EBB02]) a robust and effective class of methods is obtained. If these parameter sets are used for onestep methods theoretically only first order can be reached. But the error constant is very small so that the observed numerical order of convergence is two. Moreover, in our experience the onestep versions obtain better results than the multistep versions.

In this paper we apply the generalised-α methods for first and seond order ODEs on a damped mass system (see [JDP10]). In this paper we show that it is possible to couple the multistep versions of the generalised-α method. The numerical results are a little better than those of onestep versions. But the coupling of onestep methods has the advantage that it allows to easily compute adaptive timestep sizes, which is introduced in this paper, too.

This paper is structured as follows: First we introduce the generalised-α methods for first and second order ODEs. A short analysis about convergency and stability is given. Then we apply both generalised-α methods on the damped mass spring system of [JDP10] and analyse the linear systems.

2 THE GENERALISED-α METHOD FOR 1ST ORDER ODES

In the following we consider the ODE

$$\dot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0. \tag{1}$$

The numerical solution of (1) is determined by the generalised-α method, which is given by the formulas (see [JWH00, DP03])

$$\begin{aligned}
\dot{\mathbf{u}}_{n+\alpha_m} &= \mathbf{f}(t_{n+\alpha_f}, \mathbf{u}_{n+\alpha_f}), && (2)\\
\mathbf{u}_{n+1} &= \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau\gamma(\dot{\mathbf{u}}_{n+1} - \dot{\mathbf{u}}_n), && (3)\\
\dot{\mathbf{u}}_{n+\alpha_m} &= \dot{\mathbf{u}}_n + \alpha_m(\dot{\mathbf{u}}_{n+1} - \dot{\mathbf{u}}_n), && (4)\\
\mathbf{u}_{n+\alpha_f} &= \mathbf{u}_n + \alpha_f(\mathbf{u}_{n+1} - \mathbf{u}_n). && (5)
\end{aligned}$$

It is well known that the generalised-α method can be formulated as onestep and as twostep methods.

2.1 The formulation as onestep method and its analysis

First we manipulate the formulas (2)–(5) to obtain a non-linear system consisting of two decoupled equations. To abbreviate we define $\mathbf{f}_{n+\alpha_f} := \mathbf{f}(t_{n+\alpha_f}, \mathbf{u}_{n+\alpha_f})$. A simple calculation gives us

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\left(1 - \frac{\gamma}{\alpha_m}\right)\dot{\mathbf{u}}_n + \frac{\tau\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f} \tag{6}$$

$$\dot{\mathbf{u}}_{n+1} = \frac{1}{\tau\gamma}\left(\mathbf{u}_{n+1} - \mathbf{u}_n - \tau(1-\gamma)\dot{\mathbf{u}}_n\right), \tag{7}$$

if $\alpha_m \neq 0$. We call the scheme (6), (7) the *onestep generalised-α method*. The starting value value $\dot{\mathbf{u}}_0$ can be computed from the ODE (1). Next we want to determine the order of consistency. For this the numerical solution $\mathbf{u}_{n+1}$ can be expanded in a Taylor series as follows

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{\tau^2\gamma\alpha_f}{\alpha_m}\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3).$$

For consistency of order 2 we get the condition $\dfrac{\gamma\alpha_f}{\alpha_m} = \dfrac{1}{2}$. Since $\mathbf{u}_{n+1}$ depends on $\dot{\mathbf{u}}_n$ (see equation (6)) we use equation (7) for expanding $\dot{\mathbf{u}}_{n+1}$ in a Taylor series and get

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \frac{\tau\alpha_f}{\alpha_m}\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^2),$$

i. e. $\dot{\mathbf{u}}_{n+1}$ is of order 1 if $\dfrac{\alpha_f}{\alpha_m} = 1$. Summarising our results we have consistency of order 2 if $\alpha_m = \alpha_f$ and $\gamma = 1/2$. The generalised-α method is zero-stable if $\alpha_m > 1/2$. In other words our method is convergent if $\alpha_m > 1/2$.

2.2 Formulation as multistep method and its analysis

The generalised-α method can be formulated as a two-step method as follows

$$\mathbf{u}_{n+1} = \frac{2\alpha_m - 1}{\alpha_m}\mathbf{u}_n - \frac{\alpha_m - 1}{\alpha_m}\mathbf{u}_{n-1} + \frac{\tau(1-\gamma)}{\alpha_m}\mathbf{f}_{n-1+\alpha_f} + \frac{\tau\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f}. \tag{8}$$

For $\alpha_m = 3/2$, $\alpha_f = 1$, and $\gamma = 1$ we obtain the backward difference formula (BDF) from Gear (see [HW96]). Next we expand $\mathbf{u}_{n+1}$ in a Taylor expansion and compare it with the exact solution. Then we have

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{\tau^2}{2}\frac{2\alpha_f - \alpha_m + 2\gamma - 1}{\alpha_m}\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3).$$

Comparing the Taylor expansions for $\mathbf{u}(t_{n+1})$ and $\mathbf{u}_{n+1}$ leads to the condition for accuracy of order 2

$$\gamma = \frac{1}{2} - \alpha_f + \alpha_m, \tag{9}$$

which is already known from [JWH00, EBB02, CH93]). The generalised-α method in form (8) is convergent of order 2 if $\alpha_m > 1/2$ and condition (9) holds. For stability reasons often the setting

$$\alpha_f = \gamma = \frac{1}{1+\rho_\infty}, \qquad \alpha_m = \frac{3-\rho_\infty}{2(1+\rho_\infty)}. \tag{10}$$

is used (see [JWH00, DP03]). Note that the condition (9) is automatically satisfied. For $\rho_\infty = 0$ we get the BDF-2 method.

3 THE GENERALISED-α METHOD FOR SECOND ORDER ODES

3.1 Formulation as onestep method

In the following we consider the second order ODE

$$\ddot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}, \dot{\mathbf{u}}), \quad \mathbf{u}(0) = \mathbf{u}_0, \dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0. \tag{11}$$

The generalised-α method can be written as

$$\mathbf{u}_{n+\alpha_f} = \alpha_f \mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \tag{12}$$

$$\dot{\mathbf{u}}_{n+\alpha_f} = \alpha_f \dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n, \tag{13}$$

$$\ddot{\mathbf{u}}_{n+\alpha_m} = \alpha_m \ddot{\mathbf{u}}_{n+1} + (1-\alpha_m)\ddot{\mathbf{u}}_n, \tag{14}$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau^2\left[\left(\frac{1}{2}-\beta\right)\ddot{\mathbf{u}}_n + \beta\ddot{\mathbf{u}}_{n+1}\right] \tag{15}$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \tau\left[(1-\gamma)\ddot{\mathbf{u}}_n + \gamma\ddot{\mathbf{u}}_{n+1}\right] \tag{16}$$

$$\ddot{\mathbf{u}}_{n+\alpha_m} = \mathbf{f}(t_{n+\alpha_f}, \alpha_f\mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \alpha_f\dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n), \tag{17}$$

where $t_{n+\alpha_f} = t_n + \tau\alpha_f$. To abbreviate we write

$$\mathbf{f}_{n+\alpha_f} := \mathbf{f}(t_{n+\alpha_f}, \alpha_f\mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \alpha_f\dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n).$$

First we determine the order of consistency and use equations (14) and (17) for manipulating (15). We obtain

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau^2\left[\left(\frac{1}{2}-\frac{\beta}{\alpha_m}\right)\ddot{\mathbf{u}}_n + \frac{\beta}{\alpha_m}\mathbf{f}_{n+\alpha_f}\right], \tag{18}$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \tau\left[\left(1-\frac{\gamma}{\alpha_m}\right)\ddot{\mathbf{u}}_n + \frac{\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f}\right], \tag{19}$$

$$\ddot{\mathbf{u}}_{n+1} = \frac{1}{\alpha_m}\left[\ddot{\mathbf{u}}_{n+\alpha_m} - (1-\alpha_m)\ddot{\mathbf{u}}_n\right] = \frac{1}{\alpha_m}\left[\mathbf{f}_{n+\alpha_f} - (1-\alpha_m)\ddot{\mathbf{u}}_n\right]. \tag{20}$$

Next we expand these three expression into Taylor expansions and get

$$\begin{aligned}
\mathbf{u}_{n+1} &= \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{1}{2}\tau^2\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3),\\
\dot{\mathbf{u}}_{n+1} &= \dot{\mathbf{u}}_n + \tau\ddot{\mathbf{u}}_n + \gamma\frac{\alpha_f}{\alpha_m}\tau^2\dddot{\mathbf{u}}_n + \mathcal{O}(\tau^3),\\
\ddot{\mathbf{u}}_{n+1} &= \ddot{\mathbf{u}}_n + \tau\frac{\alpha_f}{\alpha_m}\dddot{\mathbf{u}}_n + \mathcal{O}(\tau^2).
\end{aligned}$$

It follows that the method is of order 2 if

$$\frac{\alpha_f}{\alpha_m} = 1 \quad \text{and} \quad \gamma\frac{\alpha_f}{\alpha_m} = \frac{1}{2}.$$

This is the same result as in the previous section.

3.2 Formulation as multistep method

As in the previous section the generalised-α method can be written as a multistep method if the ODE (11) is linear in $\dot{\mathbf{u}}$. Therefore we consider the problem as in [EBB02]

$$M\ddot{\mathbf{u}} + C\dot{\mathbf{u}} + \mathbf{S}(\mathbf{u}) = \mathbf{F}(t). \tag{21}$$

Then equation (17) reads as

$$M\ddot{\mathbf{u}}_{n+\alpha_m} = \mathbf{F}(t_{n+\alpha_f}) - S(\alpha_f\mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n) - C(\alpha_f\dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n). \tag{22}$$

The generalised-α method can be formulated as a multistep method with the help of (15), (16), and (22). These formulas are evaluated at time t_n, t_{n+1}, and t_{n+2} (see for example [EBB02]). Then we get

$$\sum_{j=0}^{3}[M\alpha_j + \tau C\gamma_j]\mathbf{u}_{n+j} + \tau^2\sum_{j=0}^{2}\delta_j[\mathbf{S}_{n+j+\alpha_f} - \mathbf{F}(t_{n+j+\alpha_f})] = 0, \tag{23}$$

where

$$\begin{aligned}
&\alpha_0 = 1-\alpha_m, \quad \alpha_1 = 3\alpha_m - 2, \quad \alpha_2 = 1-3\alpha_m, \quad \alpha_3 = \alpha_m,\\
&\gamma_0 = (1-\alpha_f)(\gamma-1), \quad \gamma_1 = 1-2\alpha_f-2\gamma+3\gamma\alpha_f, \gamma_2 = \alpha_f+\gamma-3\gamma\alpha_f, \quad \gamma_3 = \alpha_f\gamma,\\
&\delta_0 = \frac{1}{2}+\beta-\gamma, \quad \delta_1 = \frac{1}{2}-2\beta+\gamma, \quad \delta_2 = \beta,
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{F}_{n+j-\alpha_f} &= \mathbf{F}(\alpha_f t_{n+j+1} + (1-\alpha_f)t_{n+j}) = \mathbf{F}(t_{n+j} + \alpha_f\tau)\\
\mathbf{S}_{n+j+\alpha_f} &= \alpha_f\mathbf{S}(\mathbf{u}_{n+j+1}) + (1-\alpha_f)\mathbf{S}(\mathbf{u}_{n+j}).
\end{aligned}$$

The method has consistency order 2 if $\gamma = \frac{1}{2} + \alpha_m - \alpha_f$. It is zero-stable and convergent if $\alpha_m \geq 1/2$, $\alpha_f \leq 1/2$ and $\gamma \leq 1/2$ (see [EBB02]). For stability reasons often the setting

$$\beta = \frac{(1 + \alpha_m - \alpha_f)^2}{4}, \alpha_f = \frac{1}{1 + \rho_\infty}, \alpha_m = \frac{2 - \rho_\infty}{1 + \rho_\infty}$$

is used (see [CH93, EBB02, JDP10]).

4 Coupling of generalised-α methods

In the following we consider the model problem

$$\ddot{u} + 2\xi\omega\dot{u} + \omega^2 u = 0, \quad u(0) = u_0, \dot{u}(0) = v_0,$$

where ξ is a given damping factor and ω a given frequency.

4.1 The onestep methods

As in [JDP10] we introduce a partitioning of the problem as follows

$$(1 - \alpha)\ddot{u}^s + \omega^2 u^s = -\alpha\ddot{u}^f - 2\xi\omega\dot{u}^f, \tag{24}$$

where the left-hand side of (24) is integrated with the generalised-α for second order problems and the the right-hand side of (24) is integrated with the generalised-α for first order problems. In the case of onestep formulation we have

$$\begin{aligned}(1 - \alpha)\ddot{u}^s_{n+\alpha^s_m} + \omega^2 u^s_{n+\alpha^s_f} &= -\alpha\ddot{u}^f_{n+\alpha^f_m} - \xi\omega\dot{u}^f_{n+\alpha^f_f} \\ \dot{u}^I_{n+1} = \dot{u}^I_{n+1} =: \dot{u}^I_{n+1}, &\quad \dot{u}^I_n = \dot{u}^I_n =: \dot{u}^I_n.\end{aligned}$$

Quantities $\dot{u}^I_{n+1}$ and $\dot{u}^I_n$ represent the interface velocities at time t_{n+1} and t_n (see [JDP10]). We insert the formulas for $\ddot{u}^s_{n+\alpha^s_m}$, $\ddot{u}^f_{n+\alpha^f_m}$, and $\dot{u}^f_{n+\alpha^f_f}$ and get

$$\begin{aligned}(1 - \alpha)\left[\alpha^s_m\ddot{u}^s_{n+1} + (1 - \alpha^s_m)\ddot{u}^s_n\right] + \omega^2\left[\alpha^s_f u^s_{n+1} + (1 - \alpha^s_f)u^s_n\right] \\ = -\alpha\left[\alpha^f_m\ddot{u}^f_{n+1} + (1 - \alpha^f_m)\ddot{u}^f_n\right] - 2\xi\omega\left[\alpha^f_f\dot{u}^I_{n+1} + (1 - \alpha^f_f)\dot{u}^I_n\right].\end{aligned}$$

Moreover we have

$$\begin{aligned}u^s_{n+1} &= u^s_n + \tau\dot{u}^I_n + \tau^2\left[\left(\frac{1}{2} - \beta^s\right)\ddot{u}^s_n + \beta_s\ddot{u}^s_{n+1}\right], \\ \dot{u}^I_{n+1} &= \dot{u}^I_n + \tau\left[\gamma^f\ddot{u}^f_{n+1} + (1 - \gamma^f)\ddot{u}^f_n\right], \\ \dot{u}^I_{n+1} &= \dot{u}^I_n + \tau\left[\gamma^s\ddot{u}^s_{n+1} + (1 - \gamma^s)\ddot{u}^s_n\right].\end{aligned}$$

Since the left- and right-hand side of (24) are taken at different times we set

$$F^s_{n+\alpha^s_f} = (1-\alpha)\ddot{u}^s_{n+\alpha^s_m} + \omega^2 u^s_{n+\alpha^s_f},$$
$$F^f_{n+\alpha^f_f} = -\alpha\ddot{u}^f_{n+\alpha^f_m} - 2\xi\omega\dot{u}^I_{n+\alpha^f_f},$$

where $F^s_{n+\alpha^s_f} = \alpha^s_f F_{n+1} + (1-\alpha^s_f)F_n$ and $F^f_{n+\alpha^f_f} = \alpha^f_f F_{n+1} + (1-\alpha^f_f)F_n$. It then follows

$$\alpha^s_f F_{n+1} + (1-\alpha^s_f)F_n = (1-\alpha)\ddot{u}^s_{n+\alpha^s_m} + \omega^2 u^s_{n+\alpha^s_f},$$
$$\alpha^f_f F_{n+1} + (1-\alpha^f_f)F_n = -\alpha\ddot{u}^f_{n+\alpha^f_m} - 2\xi\omega\dot{u}^I_{n+\alpha^f_f}.$$

Then our problem reads as

$$\begin{aligned}
(1-\alpha)\left[\alpha^s_m\ddot{u}^s_{n+1} + (1-\alpha^s_m)\ddot{u}^s_n\right] \\
+\omega^2\left[\alpha^s_f u^s_{n+1} + (1-\alpha^s_f)u^s_n\right] &= \alpha^s_f F_{n+1} + (1-\alpha^s_f)F_n, \\
\alpha^f_f F_{n+1} + (1-\alpha^f_f)F_n &= -\alpha\left[\alpha^f_m\ddot{u}^f_{n+1} + (1-\alpha^f_m)\ddot{u}^f_n\right] \\
&\quad - 2\xi\omega\left[\alpha^f_f\dot{u}^I_{n+1} + (1-\alpha^f_f)\dot{u}^I_n\right] \\
u^s_{n+1} &= u^s_n + \tau\dot{u}^I_n + \tau^2\left[\left(\frac{1}{2}-\beta^s\right)\ddot{u}^s_n + \beta_s\ddot{u}^s_{n+1}\right], \\
\dot{u}^I_{n+1} &= \dot{u}^I_n + \tau\left[\gamma^f\ddot{u}^f_{n+1} + (1-\gamma^f)\ddot{u}^f_n\right], \\
\dot{u}^I_{n+1} &= \dot{u}^I_n + \tau\left[\gamma^s\ddot{u}^s_{n+1} + (1-\gamma^s)\ddot{u}^s_n\right].
\end{aligned}$$

Finally we arrive at the problem

$$\mathbf{v}_{n+1} = A_1^{-1}A_2\mathbf{v}_n \tag{25}$$

with

$$A_1 = \begin{pmatrix}
\omega^2\alpha^s_f & 0 & \frac{(1-\alpha)\alpha^s_m}{\tau^2} & 0 & -\frac{\alpha^s_f}{\tau^2} \\
0 & 2\frac{\xi\omega}{\tau}\alpha^f_f & 0 & \frac{\alpha\alpha^f_m}{\tau^2} & \frac{\alpha^f_f}{\tau^2} \\
1 & 0 & -\beta^s & 0 & 0 \\
0 & \frac{1}{\tau} & 0 & -\frac{\gamma^f}{\tau} & 0 \\
0 & \frac{1}{\tau} & -\frac{\gamma^s}{\tau} & 0 & 0
\end{pmatrix},$$

$$A_2 = \begin{pmatrix}
-\omega^2(1-\alpha^s_f) & 0 & -\frac{(1-\alpha)(1-\alpha^s_m)}{\tau^2} & 0 & \frac{1-\alpha^s_f}{\tau^2} \\
0 & -2\frac{\xi\omega}{\tau}(1-\alpha^f_f) & 0 & -\frac{\alpha(1-\alpha^f_m)}{\tau^2} & -\frac{1-\alpha^f_f}{\tau^2} \\
1 & 1 & \frac{1}{2}-\beta^s & 0 & 0 \\
0 & \frac{1}{\tau} & 0 & \frac{1-\gamma^f}{\tau} & 0 \\
0 & \frac{1}{\tau} & \frac{1-\gamma^s}{\tau} & 0 & 0
\end{pmatrix},$$

$$\mathbf{v}_n = (u^s_n, \tau\dot{u}^I_n, \tau^2\ddot{u}^s_n, \tau^2\ddot{u}^f_n, \tau^2 F_n)^\top.$$

Using a computer algebra package we can compute u_{n+1} and expand it into a Taylor series. Then the local error reads $\delta_\tau = \mathcal{O}(\tau^2)$. We test the implementation on two different settings, which are considered in [JDP10], too. In setting 1 we set $\omega = 1$, $\xi = 0.001$, and $\alpha = 0.5$, in setting 2 we have $\omega = 1$, $\xi = 0.01$, and $\alpha = 0.8$. The parameters ρ_f and ρ_s are taken as variables and we test the above method with each combination of ρ_f and ρ_s. The numerical results are presented in Figure 1. We see that the coupled method produces

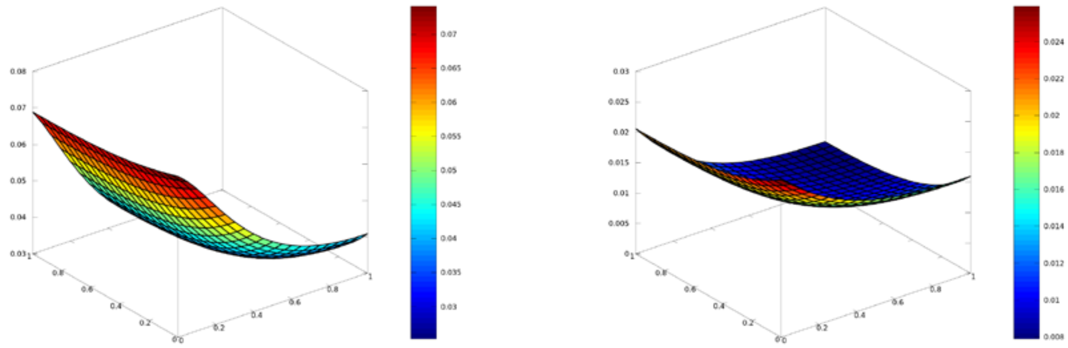

Figure 1: Numerical errors of the coupled method: setting 1 (left) and setting 2 (right)

accurate results. It can be observed that the monolythic approach does not always give the best results. The best result is obtained with setting $\rho_f = \rho_s = 1$.

4.2 The multistep methods

In the next step we couple the multistep versions. First we have

$$\mathbf{u}_{n+1} = \frac{2\alpha_m - 1}{\alpha_m}\mathbf{u}_n - \frac{\alpha_m - 1}{\alpha_m}\mathbf{u}_{n-1} + \frac{\tau(1-\gamma)}{\alpha_m}\mathbf{f}_{n-1+\alpha_f} + \frac{\tau\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f}.$$

for the first order problem and

$$\sum_{j=0}^{3}[M\alpha_j + \tau C\gamma_j]\mathbf{u}_{n+j} + \tau^2\sum_{j=0}^{2}\delta_j[\mathbf{S}_{n+j+\alpha_f} - \mathbf{F}(t_{n+j+\alpha_f})] = 0.$$

for the second order problem. For coupling u_{n+1} and $\dot{u}_{n+1}$ we use the difference quotient

$$11u_{n+3} - 18u_{n+2} + 9u_{n+1} - 2u_n = 6\tau\dot{u}_{n+3}.$$

Finally we have a coupled system of three equations

$$(1-\alpha)\sum_{j=0}^{3}\alpha_j^s\mathbf{u}_{n+j}^s+\tau^2\sum_{j=0}^{2}\delta_j[\omega^2\mathbf{u}_{n+j+\alpha_f}-\mathbf{F}(t_{n+j+\alpha_f})]=0$$

$$\alpha_m^f\dot{\mathbf{u}}_{n+3}^f-(2\alpha_m^f-1)\dot{\mathbf{u}}_{n+2}^f+(\alpha_m^f-1)\dot{\mathbf{u}}_{n+1}^f$$
$$+\tau(1-\gamma^f)(2\xi\omega\dot{\mathbf{u}}_{n+1+\alpha_f}^f+\mathbf{f}_{n+1+\alpha_f})+\tau\gamma(2\xi\omega\dot{\mathbf{u}}_{n+2+\alpha_f}^f+\mathbf{f}_{n+2+\alpha_f})=0$$

$$6\tau\dot{\mathbf{u}}_{n+3}-11u_{n+3}+18u_{n+2}-9u_{n+1}+2u_n=0.$$

Using a computer algebra package we can compute u_{n+1} and expand it into a Taylor series. Then the local error reads $\delta_\tau=\mathcal{O}(\tau^2)$. As for the onestep methods the coupled method is applied on two different settings. Again, in setting 1 we set $\omega=1$, $\xi=0.001$, and $\alpha=0.5$, and in setting 2 we have $\omega=1$, $\xi=0.01$, and $\alpha=0.8$. Parameters ρ_f and ρ_s are taken as variables and we test the above method with each combination of ρ_f and ρ_s. The numerical results are presented in Figure 2. We see that the coupled

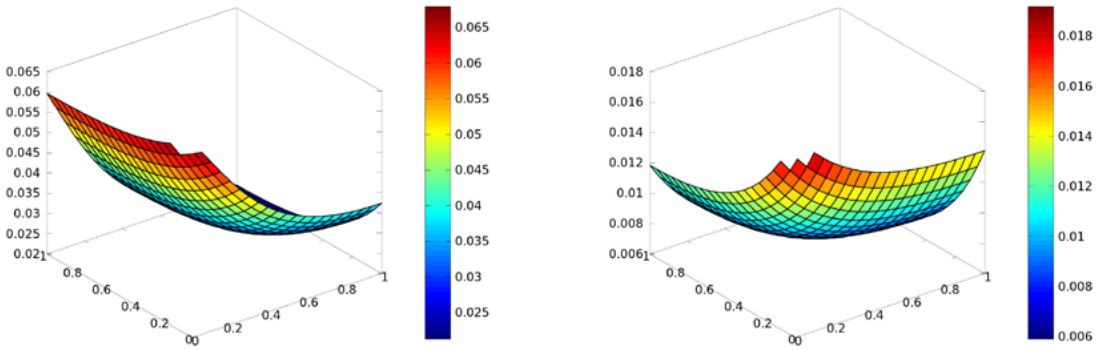

Figure 2: Numerical errors of the coupled method: setting 1 (left) and setting 2 (right)

multistep methods produce more accurate results than the previous approach with the onestep methods. It can be observed that the monolythic approach does not always give the best results. The best result is obtained with setting $\rho_f=\rho_s=1$.

5 Adaptivity

In [Ran13] an adaptive timestep control for the generalised-α method is introduced for onestep and for multistep versions. In this article only adaptivity for onestep methods is considered. With the help of the backward Euler method a second solution can be computed. Then the next timestep size τ_{n+1} is proposed to be

$$\tau_{n+1}=\rho\frac{\tau_n^2}{\tau_{n-1}}\left(\frac{TOL\cdot r_n}{r_{n+1}^2}\right)^{1/p}, \tag{26}$$

where $\rho \in (0,1]$ is a safety factor, $TOL > 0$ is a given tolerance, and $r_{n+1} := \|\mathbf{u}_{n+1} - \hat{\mathbf{u}}_{n+1}\|$. For details about the numerical error and the implementation of automatic steplength control, we refer to [HW96, Lan01]. For our coupled problem (25) the algorithm reads as follows:

- First compute the solution of (25). Then we have u^s_{n+1} and $\dot{u}^I_{n+1}$
- Compute $\ddot{u}_{n+1}$ by evaluating the model problem.
- Compute a second solution with the backward Euler method, i. e. $u^s_{n+1} = u^s_n + \tau \dot{u}^I_{n+1}$ and $\dot{u}^I_{n+1} = \dot{u}^I_n + \tau \ddot{u}_{n+1}$.
- Compute the numerical error r_{n+1} and approximate the new timestep length τ_{n+1} with (26).
- If the numerical error is smaller than the given tolerance the timestep is accepted otherwise it is rejected and has to recomputed with the new timestep length τ_{n+1} .

As before we test our numerical method on the problem with settings 1 and 2. We choose different tolerances and compute the numerical errors with respect to the computing time (see Figure 3). It can be observed that our approach produces stable numerical results.

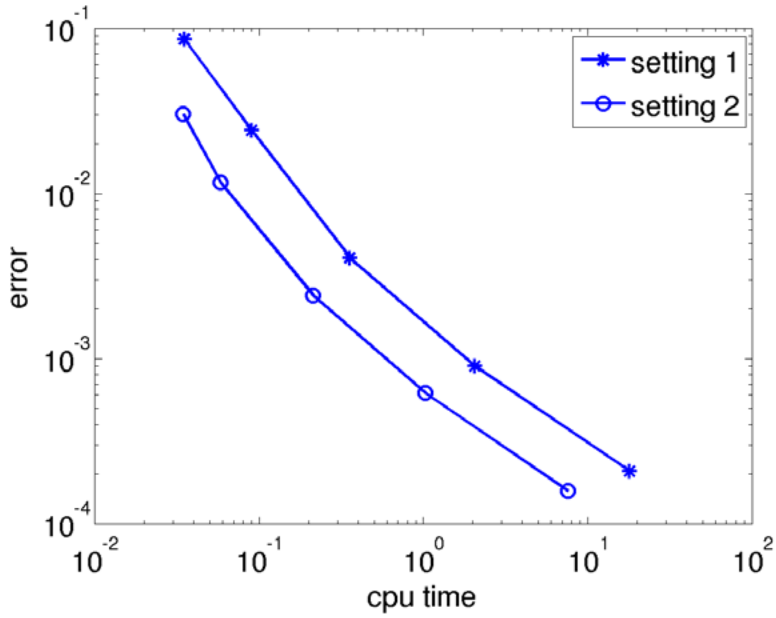

Figure 3: Numerical errors of the adaptive method

6 Summary and Outlook

In this article we introduced a new coupling scheme of generalised-α methods, which use multistep formulation. As in the case of onestep methods order 2 is theoretically not reached. In the second part of the paper we developed an adaptive timestep control for the partitioned approach, which gives good numerical results.

In a future work this approach should be applied on other problems like, for example, FSI problems.

REFERENCES

[CH93] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *J. Appl. Mech.*, 60(2):371–375, 1993.

[DP03] W. Dettmer and D. Perić. An analysis of the time integration algorithms for the finite element solutions of incompressible Navier-Stokes equations based on a stabilised formulation. *Comput. Methods Appl. Mech. Engrg.*, 192(9-10):1177–1226, 2003.

[DR08] D. Dinkler and Joachim Rang, editors. *Wechselwirkung von Struktur und Fluid - Abschlussbericht eines Graduiertenkollegs*. Wolfram Schmidt Buchbinderei & Druckerei, Braunschweig, 2008.

[EBB02] S. Erlicher, L. Bonaventura, and O. S. Bursi. The analysis of the generalized-α method for non-linear dynamic problems. *Comput. Mech.*, 28(2):83–104, 2002.

[FP80] C.A. Felippa and K.C. Park. Staggered transient analysis procedures for coupled mechanical systems: Formulation. *Comput. Methods Appl. Mech. Eng.*, 24:61–111, 1980.

[HW96] E. Hairer and G. Wanner. *Solving ordinary differential equations. II: Stiff and differential-algebraic problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1996.

[JDP10] M.M. Joosten, W.G. Dettmer, and D. Perić. On the temporal stability and accuracy of coupled problems with reference to fluid–structure interaction. *Int. J. Numer. Methods Fluids*, 64(10-12):1363–1378, 2010.

[JWH00] Kenneth E. Jansen, Christian H. Whiting, and Gregory M. Hulbert. A generalized-α method for integrating the filtered Navier-Stokes equations with a stabilized finite element method. *Comput. Methods Appl. Mech. Eng.*, 190(3-4):305–319, 2000.

[Lan01] J. Lang. *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems*, volume 16 of *Lecture Notes in Computational Science and Engineering*. Springer-Verlag, Berlin, 2001.

[MM04] M. Meyer and H. G. Matthies. State-space representation of instationary two-dimensional airfoil aerodynamics. *Journal of Wind Engineering and Industrial Aerodynamics*, 92(3-4):263–274, 2004.

[MNS06] Hermann G. Matthies, Rainer Niekamp, and Jan Steindorf. Algorithms for strong coupling procedures. *Comput. Methods Appl. Mech. Eng.*, 195(17-18):2028–2049, 2006.

[MS02] H. G. Matthies and J. Steindorf. Partitioned but strongly coupled iteration schemes for nonlinear fluid-structure interaction. *Comput. Struct.*, 80:1991–1999, 2002.

[MW01] D. P. Mok and W. A. Wall. Partitioned analysis schemes for the transient interaction of incompressible flows and nonlinear flexible structures. In K. Schweizerhof W. A. Wall, K.-U. Bletzinger, editor, *Trends in Computational Structural Mechanics*, pages 689–698. CIMNE, Barcelona, 2001.

[PFL95] S. Piperno, C. Farhat, and B. Larrouturou. Partitioned procedures for the transient solution of coupled aeroelastic problems. I: Model problem, theory and two-dimensional application. *Comput. Methods Appl. Mech. Eng.*, 124(1-2):79–112, 1995.

[Ran13] J. Rang. Adaptive timestep control for the generalised-α method. In *Adaptive Modelling and Simulation*. CIMNE, Barcelona, 2013. to appear.

[RB00] S. Rugonyi and K.-J. Bathe. On the analysis of fully-coupled fluid flows with structural interactions — a coupling and condensation procedure. *Int. J. of Comp. Civil and Struct. Eng.*, 1:29–41, 2000.

[RSM09] Joachim Rang, Johannes Schöön, and Hermann G. Matthies. Solving FSI problems with high resolution and using a component framework in parallel. In Stefan Hartmann, Andreas Meister, Michael Schäfer, and Stefan Turek, editors, *International Workshop on Fluid-Structure Interaction: Theory, Numerics and Applications*, Kassel, 2009. Kassel Univ. Press.