Proceedings of the 7th GACM Colloquium on Computational Mechanics
for Young Scientists from Academia and Industry
October 11-13, 2017 in Stuttgart, Germany

Error estimation approach for controlling the macro step-size for explicit co-simulation methods

Tobias Meyer[1]*, Jan Kraft[1], Pu Li[1], Daixing Lu[1] and Bernhard Schweizer[1]

Micro Abstract
An approach for controlling the macro-step size in connection with explicit co-simulation methods is suggested. The method is tailored for applied-force coupling techniques. Each macro-time step is carried out with two different explicit co-simulation methods. By comparing the variables of both results, an error estimator for the local error can be constructed. A step-size controller for the macro-step size can be implemented. Examples are presented demonstrating the applicability and accuracy.

[1]Institute of Applied Dynamics, Technische Universität Darmstadt, Darmstadt, Germany
***Corresponding author**: meyer@ad.tu-darmstadt.de

Introduction

The general idea of co-simulation is to separate a system of differential-algebraic equations into several subsystems. The subsystems are integrated in parallel and can be computed more efficiently than the overall system, since the numerical effort for time integration grows with the number of state variables. The subsystems can be computed with tailored subsystem solvers so that the efficiency may be further improved. Co-simulation methods are distinguished with respect to the coupling technique. Applying a constraint coupling approach, the subsystems are coupled by algebraic constraint equations. Using an applied force-coupling approach, the subsystems are coupled by constitutive laws [1]. This manuscript concentrates on applied-force coupling techniques.

A multibody system, described by the equations of motion $\mathbf{B}(t,\mathbf{z})\,\dot{\mathbf{z}} = \mathbf{F}(t,\mathbf{z})$, is considered here, where the vector $\mathbf{z}$ contains the position variables $\mathbf{q}$ (displacements and rotation parameters), the velocity variables $\mathbf{v}$, and the *Lagrange* multipliers $\boldsymbol{\lambda}$. Applying a co-simulation approach, a macro-time grid $T_0,\ldots,T_N,\ldots,T_{\text{end}}$ has to be introduced. The overall system is partitioned into several subsystems. Using an applied-force coupling approach, the vector $\mathbf{z}$ is decomposed into the vectors $\left[\mathbf{z}^1,\ldots,\mathbf{z}^r\right]^T$, where $\mathbf{z}^s = \left[\mathbf{q}^s,\mathbf{v}^s,\boldsymbol{\lambda}^s\right]^T$ contains the variables of subsystem s. We assume that the matrix $\mathbf{B}(t,\mathbf{z})$ is block diagonal. More precisely, there is one block for each subsystem, which is independent of the variables of the other subsystems, i.e. $\mathbf{B}(t,\mathbf{z}) = \text{blockdiag}\left(\mathbf{B}^1\left(t,\mathbf{z}^1\right),\ldots,\mathbf{B}^r(t,\mathbf{z}^r)\right)$. With $\mathbf{F}(t,\mathbf{z}) = \left[\mathbf{F}^1(t,\mathbf{z}),\ldots,\mathbf{F}^r(t,\mathbf{z})\right]^T$, the equations of motion of subsystem s read as $\mathbf{B}^s(t,\mathbf{z}^s)\,\dot{\mathbf{z}}^s = \mathbf{F}^s(t,\mathbf{z})$ for $s = 1,\ldots,r$. A vector of coupling variables $\mathbf{u}(t,\mathbf{z})$ is introduced, which are defined in such a way that the right hand side of each subsystem only depends on the state variables of this subsystem and the coupling variables, i.e. it can be written as $\mathbf{F}^s(t,\mathbf{z}) = \tilde{\mathbf{F}}^s(t,\mathbf{z}^s,\mathbf{u}(t,\mathbf{z}))$. Interpolating the coupling variables between two consecutive macro-time points T_N and T_{N+1} by a vector of polynomials $\mathbf{p}(t)$, the equations of motion of the subsystems can be expressed as $\mathbf{B}^s(t,\mathbf{z}^s)\,\dot{\mathbf{z}}^s = \tilde{\mathbf{F}}^s(t,\mathbf{z}^s,\mathbf{p}(t))$ for $s = 1,\ldots,r$. The subsystems are coupled by the coupling conditions $\mathbf{g}^{\text{co}}(T_N,\mathbf{z}_N,\mathbf{p}_N) := \mathbf{p}_N - \mathbf{u}(T_N,\mathbf{z}_N) = \mathbf{0}$. The coupling conditions are not taken into account between the macro-time points. From T_N to T_{N+1} the subsystems are integrated independently with individual subsystem solvers

and individual micro-step sizes. At the macro-time points, information (coupling variables) is interchanged between the subsystems. The coupling variables at the macro-time points are denoted by $\mathbf{u}_N = \mathbf{u}(T_N, \mathbf{z}_N)$, where $\mathbf{z}_N$ terms the overall vector collecting the numerical solutions of all subsystems at the macro-time point T_N.

In section 1, two different co-simulation approaches are explained. In section 2, the equations of motion of the multibody subsystems are described in more detail. In section 3, a co-simulation approach for controlling the macro-step size for coupled multibody systems is introduced.

1 Two Explicit Co-Simulation Approaches

In this section, two different methods for approximating the coupling variables are explained. The two methods are illustrated in Figure 1. The polynomial degree is denoted by k. The approximation polynomials for the macro-time step from T_N to T_{N+1} are indicated by the subscript $N+1$. In order to distinguish between the approximation polynomials of the two different methods, the approximation polynomials are extended by appropriate superscripts (ext, int).

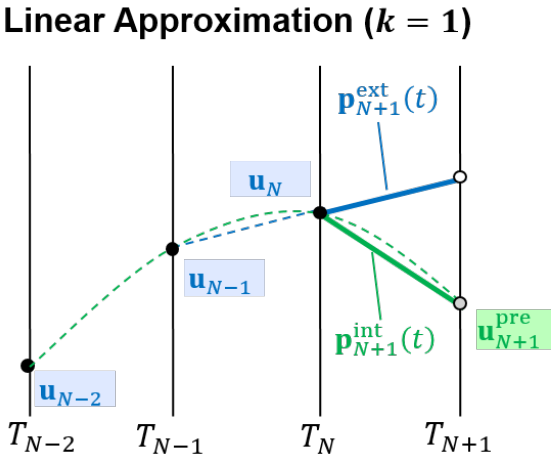

Figure 1. Approximation Polynomials

1.1 Method 1

The first approach makes use of a classical extrapolation technique [4]. The approximation polynomials are denoted by $\mathbf{p}_{N+1}^{\text{ext}}$. If the co-simulation is carried out with polynomials of degree k, the vector $\mathbf{p}_{N+1}^{\text{ext}}(t)$ extrapolates $\mathbf{u}(t, \mathbf{z}(t))$ in the interval $[T_N, T_{N+1}]$ by the $k+1$ sampling points $\mathbf{p}_{N+1}^{\text{ext}}(T_{N-k}) = \mathbf{u}_{N-k}, \ldots, \mathbf{p}_{N+1}^{\text{ext}}(T_N) = \mathbf{u}_N$.

1.2 Method 2

Using the second method, the approximation polynomials are termed by $\mathbf{p}_{N+1}^{\text{int}}$. At first, a predictor vector $\mathbf{u}_{N+1}^{\text{pre}}$ extrapolating $\mathbf{u}_{N+1}$ by the $k+2$ supporting points $\mathbf{u}_{N-k-1}, \ldots, \mathbf{u}_N$ is computed with the help of the *Neville-Aitken* scheme. Then, the vector $\mathbf{p}_{N+1}^{\text{int}}(t)$ is constructed, which interpolates $\mathbf{u}(t, \mathbf{z}(t))$ from T_N to T_{N+1} by the $k+1$ sampling points $\mathbf{p}_{N+1}^{\text{int}}(T_{N-k+1}) = \mathbf{u}_{N-k+1}, \ldots, \mathbf{p}_{N+1}^{\text{int}}(T_N) = \mathbf{u}_N, \mathbf{p}_{N+1}^{\text{int}}(T_{N+1}) = \mathbf{u}_{N+1}^{\text{pre}}$.

1.3 Summary

The idea is to execute each macro-time step twice. Namely, once with the polynomials $\mathbf{p}_{N+1}^{\text{ext}}(t)$ and secondly with the polynomials $\mathbf{p}_{N+1}^{\text{int}}(t)$. The user chooses one of the two results for the co-simulation. The result of the other integration only serves for the error estimation. Since the integrations are executed in parallel, there is only little extra computation time necessary. It can be shown that both methods have the same convergence order. Computing the error constants of both methods, the local error can be estimated with the help of the *Milne*-device approach [3].
.

2 Coupling Multibody Systems with an Applied-Force Coupling Approach

Assuming that each subsystem is described by a multibody system, the equations of motion of subsystem s $(s = 1, \ldots, r)$ read as

$$\begin{aligned} \dot{\mathbf{q}}^s &= \mathbf{K}^s(t, \mathbf{q}^s)\, \mathbf{v}^s, \\ \mathbf{M}^s(t, \mathbf{q}^s)\, \dot{\mathbf{v}}^s &= \mathbf{f}^s(t, \mathbf{q}^s, \mathbf{v}^s, \boldsymbol{\lambda}^s, \mathbf{p}(t)), \\ \mathbf{0} &= \mathbf{g}^s(t, \mathbf{q}^s, \mathbf{v}^s, \boldsymbol{\lambda}^s), \end{aligned} \tag{1}$$

where $\mathbf{p}(t)$ denotes a vector of polynomials approximating the coupling variables $\mathbf{u}(t, \mathbf{q}(t), \mathbf{v}(t))$ between two consecutive macro-time points. The coupling variables are assumed to be independent of the *Lagrange* multipliers $\boldsymbol{\lambda}$, since we make use of an applied-force coupling approach. At the macro-time point T_N, the subsystems are coupled by $\mathbf{p}(T_N) - \mathbf{u}(T_N, \mathbf{q}(T_N), \mathbf{v}(T_N)) = 0$. The matrix $\mathbf{K}^s(t, \mathbf{q}^s)$ describes the relationship between the velocity variables and the derivatives of the position variables. The matrix $\mathbf{M}^s(t, \mathbf{q}^s)$ denotes the mass matrix of subsystem s. The inner (not necessarily holonomic) constraints of subsystem s are denoted by the vector $\mathbf{g}^s(t, \mathbf{q}^s, \mathbf{v}^s, \boldsymbol{\lambda}^s)$. The vector $\mathbf{f}^s(t, \mathbf{q}^s, \mathbf{v}^s, \boldsymbol{\lambda}^s, \mathbf{p}(t))$ contains the forces and torques in subsystem s. We assume that the accelerations can be represented by

$$\ddot{\mathbf{q}}^s = \mathbf{a}^s(t, \mathbf{q}^s, \mathbf{v}^s, \mathbf{p}(t)).$$

The accelerations of all subsystems are collected in the vectors

$$\ddot{\mathbf{q}} = \mathbf{a}(t, \mathbf{q}, \mathbf{v}, \mathbf{p}(t)).$$

3 Co-Simulation Approach with Variable Macro-Step Size

3.1 Error Analysis

For the error analysis, it is assumed that the subsystems are solved exactly between two consecutive macro-time points T_N and T_{N+1}. Hence, the error analysis only deals with the error generated by the co-simulation, i.e. by the approximation of the coupling variables. The numerical solutions at the macro-time point T_N of all subsystems are collected in the vectors $\mathbf{q}_N$, $\mathbf{v}_N$, $\boldsymbol{\lambda}_N$. The coupling variables can be expressed by $\mathbf{u}_N = \mathbf{u}(T_N, \mathbf{q}_N, \mathbf{v}_N)$. We consider the macro-time step from T_N to T_{N+1}. In the following considerations, $\mathbf{q}(t)$, $\mathbf{v}(t)$, $\boldsymbol{\lambda}(t)$ denote the exact solutions with respect to the initial conditions $\mathbf{q}(T_N) = \mathbf{q}_N$ and $\mathbf{v}(T_N) = \mathbf{v}_N$. We assume that the numerical solutions $\mathbf{q}_{N-k-1}, \ldots, \mathbf{q}_{N-1}$ and $\mathbf{v}_{N-k-1}, \ldots, \mathbf{v}_{N-1}$ agree with this trajectory with order $\mathcal{O}\big(H^{k+2}\big)$ at the macro-time points $T_{N-k-1}, \ldots, T_{N-1}$, where $H = T_{N+1} - T_N$ denotes the current macro-step size. Furthermore, the constants

$$C_{N+1}^i := \int_{T_N}^{T_{N+1}} \int_{T_N}^{\tau} L_{N+1}^i(t)\, \mathrm{d}t \mathrm{d}\tau \qquad (i \in \{k, k+1\}) \tag{2}$$

are computed by integrating the *Lagrange*-basis polynomials

$$L_{N+1}^i(t) = \prod_{j=0}^{i-1} \frac{t - T_{N-j}}{T_{N+1} - T_{N-j}} \tag{3}$$

twice. In a more detailed analysis, it can be shown that the local errors of the position variables for the two co-simulation methods are given by

$$\mathbf{q}_{N+1}^{\text{ext}} - \mathbf{q}(T_{N+1}) = C_{N+1}^{k+1}\mathbf{a}_{\mathbf{p},N}\Delta\mathbf{p}_{N+1} + \mathcal{O}\Big(H^{k+4}\Big) \tag{4}$$

$$\mathbf{q}_{N+1}^{\text{int}} - \mathbf{q}(T_{N+1}) = \Big(C_{N+1}^{k+1} - C_{N+1}^{k}\Big)\,\mathbf{a}_{\mathbf{p},N}\Delta\mathbf{p}_{N+1} + \mathcal{O}\Big(H^{k+4}\Big) \tag{5}$$

with $\mathbf{a}_{\mathbf{p},N} := \mathbf{a}_{\mathbf{p}}(T_N, \mathbf{q}_N, \mathbf{v}_N, \mathbf{u}_N)$, which denotes the *Jacobian*-matrix of the accelerations $\mathbf{a}(t, \mathbf{q}, \mathbf{v}, \mathbf{p})$ with respect to $\mathbf{p}$. Further,

$$\Delta\mathbf{p}_{N+1} := \mathbf{p}_{N+1}^{\text{ext}}(T_{N+1}) - \mathbf{p}_{N+1}^{\text{int}}(T_{N+1}) \tag{6}$$

terms the difference between the two approximation polynomials at the macro-time point T_{N+1}. Due to the fact that $C_{N+1}^{k} = \mathcal{O}\big(H^2\big)$, $C_{N+1}^{k+1} = \mathcal{O}\big(H^2\big)$, and $\Delta\mathbf{p}_{N+1} = \mathcal{O}\big(H^{k+1}\big)$, the local errors of the position variables converge with order $\mathcal{O}\big(H^{k+3}\big)$.

3.2 Error Estimation

With the help of the *Milne*-device approach, an error estimator is constructed. The local error of the co-simulation approach presented in subsection 1.1 can be estimated by

$$\hat{\varepsilon}_{N+1}^{\text{ext}} := \frac{C_{N+1}^{k+1}}{C_{N+1}^{k}} \left\|\mathbf{q}_{N+1}^{\text{ext}} - \mathbf{q}_{N+1}^{\text{int}}\right\|. \tag{7}$$

The local error of the co-simulation approach explained in subsection 1.2 may be estimated by

$$\hat{\varepsilon}_{N+1}^{\text{int}} := \left(1 - \frac{C_{N+1}^{k+1}}{C_{N+1}^{k}}\right) \left\|\mathbf{q}_{N+1}^{\text{ext}} - \mathbf{q}_{N+1}^{\text{int}}\right\|. \tag{8}$$

Analogously, an error estimator for the velocity variables can be constructed. It should be pointed out that the local error of the velocity variables converges with order $\mathcal{O}\big(H^{k+2}\big)$.

Conclusions

In order to generate an error estimator for explicit co-simulation approaches, each macro-time step is carried out with two different methods of the same convergence order. Computing the error constants, an error estimator for controlling the macro-step size can be constructed with the help of the *Milne*-device approach. With this error estimator, the macro-step size can easily be controlled by using well-known step-size controllers, for instance, a PI-step size controller [2].

References

[1] C. Gomes, C. Thule, D. Broman, P. G. Larsen, and H. Vangheluwe. Co-simulation: State of the art. *arXiv preprint arXiv:1702.00686*, 2017.

[2] K. Gustafsson, M. Lundh, and G. Söderlind. A PI stepsize control for the numerical solution of ordinary differential equations. *BIT Numerical Mathematics*, 28(2):270–287, 1988.

[3] W. Milne. Numerical integration of ordinary differential equations. *American Mathematical Monthly*, pages 455–460, 1926.

[4] B. Schweizer, P. Li, and D. Lu. Explicit and implicit co-simulation methods: Stability and convergence analysis for different solver coupling approaches. *Journal of Computational and Nonlinear Dynamics*, 10(5):051007, 2015.