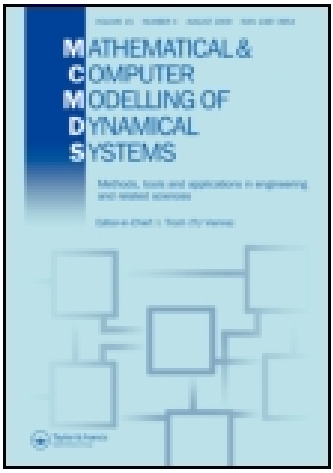


Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences

Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/nmcm20

Two Methods of Simulator Coupling

R. Kübler & W. Schiehlen

Published online: 09 Aug 2010.

To cite this article: R. Kübler & W. Schiehlen (2000) Two Methods of Simulator Coupling, Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences, 6:2, 93-113

To link to this article: http://dx.doi.org/10.1076/1387-3954(200006)6:2;1-M;FT093

Mathematical and Computer Modelling of Dynamical Systems
2000, Vol. 6 No. 2, pp. 93–113

1387-3954/00/0602-0093$15.00

Two Methods of Simulator Coupling

R. KÜBLER* AND W. SCHIEHLEN*

ABSTRACT

Modelling and simulation of complex engineering systems are often relieved by a modular approach in which the global system is decomposed into subsystems. Advantages arise from independent and parallel modelling of subsystems over easy exchange of the resulting modules to the use of different software for each module. However, the modular simulation of the global system by coupling of simulators may result in an unstable integration, if an algebraic loop exists between the subsystems. This numerical phenomenon is analyzed and two methods of simulator coupling which guarantee stability for general systems including algebraic loops are introduced. Numerical results of the modular simulation of a multibody system are presented.

Keywords: Modular simulation, simulator coupling, algebraic loops, nonlinear algebraic equations, multibody systems, mechatronics.

1 INTRODUCTION

Modelling and simulation of complex engineering systems like intelligent automobiles or high precision tool machines require the modelling of many components from different engineering fields, e.g., mechanics, control and electronics. In general, the global system has to be decomposed into subsystems due to the different engineering disciplines to make it treatable efficiently by a team of engineers. Many advantages arise from this approach, once the interfaces of the subsystems have been defined: Subsystems can be modelled independently from each other and in parallel from experts in the corresponding engineering field, the exchange and modification of a subsystem is independent of any other components, different and independent software tools can be used for every engineering discipline, and it is possible to hide the internal dynamics of a subsystem during simulation of the global system.

*Institute B of Mechanics, University of Stuttgart, Pfaffenwaldring 9, 70550 Stuttgart, Germany. E-mail: {rk,wos}@mechb.uni-stuttgart.de

To carry out the overall simulation, all subsystems have to be coupled to obtain the global system behavior. The coupling of the subsystems may be achieved on three different levels of the model description, as illustrated in Figure 1 for mechatronic systems.

In the physical model description the engineering system is represented by a model with physical parameters, such as masses or geometric dimensions in the case of multibody systems. The mathematical model description is an abstract representation of the engineering system by mathematical equations, e.g., the equations of motion of multibody systems. The results of simulation of the mathematical model, for instance positions and velocities of bodies, are considered as the behavioral model description.

Standardization of model description and coupling of models on all three description levels are lively research subjects. Standardization of mechatronic systems on the physical model description level including the standardization of multibody system data within STEP is under development (Dürr et al. [4], Dürr and Schiehlen [5]). Standardization on mathematical model description may be achieved with the hardware description language VHDL–AMS [8]. The textual system description language DSL is based on the nonlinear state space representation and has been proposed by Junker et al. [9]. The coupling on behavioral description level has successfully been tested by coupling of different simulators, see for example Wünsche et al. [16].

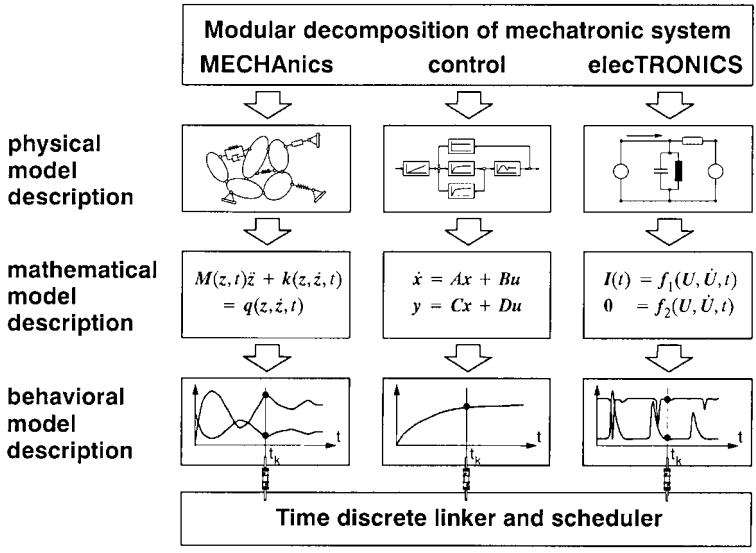


Fig. 1. Modular modelling of mechatronic systems.

Coupling of models in behavioral model description is also referred to as *modular simulation* or *simulator coupling*, respectively. Simulation of the global system is realized by a time discrete linker and scheduler which combines the inputs and outputs of the corresponding subsystems and establishes communication between the subsystems to discrete time instants (Fig. 1). The main advantage of coupling on the behavioral description level is the possibility to use integrated software packages for each subsystem, where preprocessor, solver and postprocessor are linked together resulting in a very user–friendly solution.

This paper is devoted to the theoretical aspects of simulator coupling for more general systems including algebraic loops. In Section 2 the modular description of dynamic systems on the mathematical model description level is introduced which is the basis of modular simulation. Each subsystem is set up by the general state–space formulation, the global system is then formed by interconnections between the inputs and the outputs of the subsystems. In Section 3 time discretization of the subsystems including input and output variables is described. On this basis the analytical analysis of zero–stability of the modular numerical integration is treated. It will be shown that convergence is only guaranteed if algebraic loops do not exist between the subsystems. Section 4 is devoted to two methods of simulator coupling. On the one hand, it will be shown that stability of the modular simulation is always guaranteed by iterative simulator coupling schemes. On the other hand, elimination of algebraic loops by introduction of filters will be presented. In Section 5 the theoretical results will be illustrated by an example from multibody system dynamics. Numerical results of this example will also be used to compare the efficiency of both methods of simulator coupling.

2 MODULAR DESCRIPTION OF DYNAMIC SYSTEMS

The basis of any numerical analysis is the mathematical model description. For each subsystem i the state and output equations represent the nonlinear state–space formulation

$$\dot{x}^i = f^i(x^i, u^i, t), \qquad x^i(t_0) = x_0^i, \tag{1a}$$

$$y^i = g^i(x^i, u^i, t), \qquad i = I, II, \ldots, N \tag{1b}$$

where $x^i \in \mathbb{R}^{n_x^i}$, $u^i \in \mathbb{R}^{n_u^i}$ and $y^i \in \mathbb{R}^{n_y^i}$ denote the state, input and output vector and $t \in \mathbb{R}$ denotes time. The subsystems are characterized by roman numbers running from $i = I$ to $i = N$ where N is the total number of subsystems.

The global system is formed by interconnections between the subsystems introduced by coupling equations which define the input vector of each subsystem as a function of the output vectors of the remaining subsystems:

$$u^i = L^i y = \begin{bmatrix} L^{i,I} \dots & L^{i,i-1} & 0 & L^{i,i+1} \dots & L^{i,N} \end{bmatrix} \begin{bmatrix} y^I \\ \vdots \\ y^{i-1} \\ y^i \\ y^{i+1} \\ \vdots \\ y^N \end{bmatrix}, \tag{1c}$$

with the elements of the incidence matrix $L^{i,j} \in \mathbb{R}^{n_u^i \times n_y^j}$ being zero or one.

Since the modelling of subsystems (1a), (1b) is independent of the global system structure (1c), this approach supports well interchangeability and reusability of system modules.

3 MODULAR NUMERICAL INTEGRATION

Methods of numerical integration of the state equations of each subsystem and the concept of zero–stability are introduced. Processing of unknown input variables in the state and output equations by extrapolation is discussed and the time discrete description of subsystems is derived. On this basis zero–stability of the coupled integration is investigated.

3.1 Integration methods

Integration methods are numerical methods for solving the initial value problem

$$\dot{x} = f(x, t), \qquad x(t_0) = x_0. \tag{4}$$

The most common integration methods are one–step (Runge–Kutta) methods and linear multistep methods, which can be written as

$$\sum_{j=0}^{n} \alpha_j x_{i+1-j} = hF(x_{i+1}, t_{i+1}, \dots, x_{i+1-n}, t_{i+1-n}, h), \qquad n \geq 1, \tag{5}$$

describing explicit as well as implicit n–step methods with step size h. The constant coefficients α_j and the vector function F are determined by the integration method chosen.

In general, the state equations (1a) are explicitly dependent on the inputs which have to be taken into account in the integration method. Assuming the inputs u_{i-j}, $j = 0, 1, \ldots, p_E - 1$ are known, polynomial extrapolation can be used for the unknown inputs $u(t_i + rh) =: u_{i+r}$, $r \in [0; 1]$:

$$\tilde{u}_{i+r} = \sum_{j=0}^{p_E-1} \gamma_j(r) u_{i-j}, \quad p_E \geq 1. \tag{4}$$

Extrapolation (4) is of order $p_E - 1$ and the constant coefficients $\gamma_j(r)$ can be calculated efficiently by Neville's algorithm, (Stoer and Bulirsch [15]).

Integration method (3) and extrapolation of the inputs (4) can be combined to

$$x_{i+1} = \varphi\big(x_{i+1}, t_{i+1}, \ldots, x_{i+1-n}, t_{i+1-n}, u_i, \ldots, u_{i+1-p_E}, h\big) \tag{5}$$

which summarizes the method used for integrating the state equations (1a).

An important property of any method to ensure convergence is *zero–stability* which means that the discrete system (5) has to be stable, if the step size goes to zero. For $h \to 0$, formula (5) results in the autonomous linear discrete system

$$\sum_{j=0}^{n} \alpha_j x_{i+1-j} = 0 \tag{6}$$

with the characteristic equation

$$\sum_{j=0}^{n} \alpha_j z^{i+1-j} = 0. \tag{7}$$

Consequently, zero–stability can be defined in terms of the eigenvalues of (7) (Hairer et al. [7]):

Definition 1 (Zero–stability of integration) *Method (5) is called zero–stable, if the eigenvalues of the characteristic equation (7) are located within or on the unit circle and the eigenvalues on the unit circle are simple.*

For example, equations (6) result in $x_{i+1} - x_i = 0$ for one–step methods and the corresponding eigenvalue of the characteristic equation (7) is $z = 1$. Consequently, any one–step method is zero–stable.

3.2 Time discrete description of subsystems

Numerical solution of the state space formulation (1) involves numerical integration of the state equations (1a) and evaluation of the output equations (1b). Figure 2 shows one global integration step from time t_k to t_{k+1} with global step size H. The state equation is integrated by a method (5) with local step size $h = H/m$ in m steps resulting in the state vector x_{k+1}. The factor m is often referred to as *multi–rate factor*. Thereafter, the output equations are evaluated at time t_{k+1} resulting in the output vector y_{k+1}.

For $t > t_k$ extrapolation (4) is applied to calculate the unknown inputs as follows:

$$\tilde{u}_{k+r} = \tilde{u}(t_k + rH) = \sum_{j=0}^{p_E-1} \gamma_j(r) u_{k-j}, \quad r \in [0; 1]. \tag{8}$$

3.3 Coupled integration

The parallel integration of N coupled subsystems is considered. The basis of the following investigation is the time discrete description of each subsystem i as described in Figure 2:

$$x^i_{k+1} = \Phi^i(\varphi^i, m^i, \tilde{u}^i), \tag{9a}$$

$$y^i_{k+1} = g^i(x^i_{k+1}, \tilde{u}^i_{k+1}, t_{k+1}), \qquad i = I, II, \ldots, N. \tag{9b}$$

The inputs of each subsystem are calculated by the algebraic coupling equations (1c) which can be evaluated without approximation error at each global time t_{k+1}:

$$u^i_{k+1} = L^i y_{k+1}. \tag{9c}$$

The coupled integration (9) for $N = 2$ subsystems is illustrated in Figure 3.

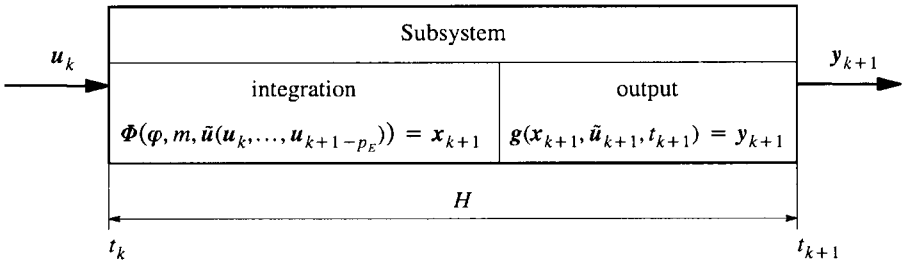


Fig. 2. Time discrete description of subsystems.

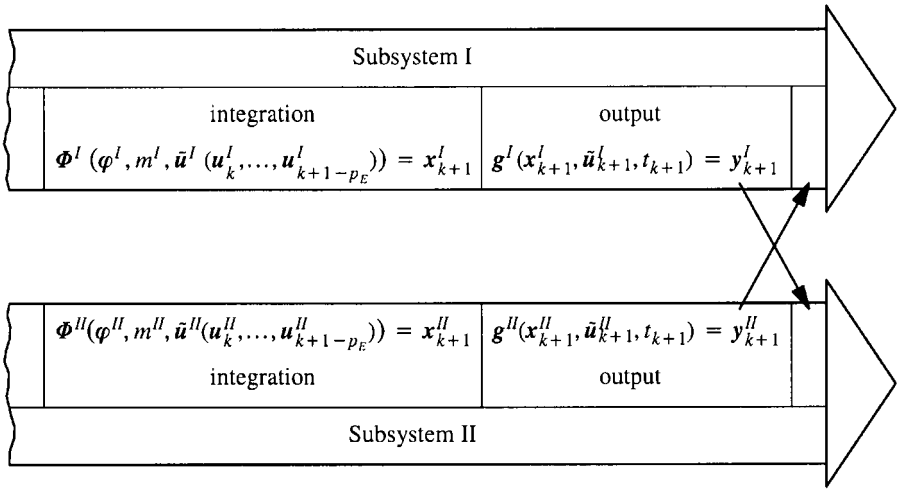


Fig. 3. Coupled integration of two subsystems.

As demonstrated in Section 3.1, the integration method is independent of the inputs for vanishing step sizes. For constant multi–rate factor m^i the equations (9a) result in

$$x^i_{k+1} = \lim_{H \to 0} \Phi^i(\varphi^i, m^i, \tilde{u}^i) = \Phi^i(\varphi^i(h^i \to 0), m^i). \tag{10}$$

This system corresponds to the linear system (6) which is stable, if zero–stable integration methods are used. As mentioned before, for one–step methods equations (10) result in

$$x^i_{k+1} = x^i_k = x^i_0 = const. \tag{11}$$

The output equations (9c) result for vanishing step size in

$$y^i_{k+1} = \lim_{H \to 0} g^i(x^i_{k+1}, \tilde{u}^i_{k+1}, t_{k+1}) = g^i(x^i_{k+1}, u^i_k, t_{k+1}) \tag{12}$$

whereas the coupling equations (9c) are independent of the step size.

Then, the definition of zero–stability of the coupled integration reads as follows:

Definition 2 (Zero–stability of coupled integration) *The coupled integration (9) is zero–stable, if the discrete coupled system*

$$x_{k+1}^i = \Phi^i(\varphi^i(h^i \to 0), m^i), \tag{13a}$$
$$y_{k+1}^i = g^i(x_{k+1}^i, u_k^i, t_{k+1}), \tag{13b}$$
$$u_k^i = L^i y_k, \qquad i = I, II, \ldots, N \tag{13c}$$

is stable.

System (13) represents a nonlinear discrete system, if any of the output equations is nonlinear.

For further analysis of zero–stability the following assumptions are made: one–step methods are used for numerical integration and the output equations are time–invariant and linearly dependent on the inputs. Under these assumptions the output equations (1b) can be rewritten in the form

$$y^i = \bar{g}^i(x^i) + D^i(x^i)u^i. \tag{14}$$

In addition, the state vector is constant for vanishing step size in the case of one–step methods due to equations (11), resulting in the discretized output equations (12)

$$y_{k+1}^i = \bar{g}^i + D^i u_k^i, \quad \bar{g}^i, D^i = const. \tag{15}$$

Using the coupling equations (13c), the outputs of the global system (13) can be written as

$$y_{k+1} = \bar{g} + \underbrace{\begin{bmatrix} 0 & D^I L^{I,II} & \cdots & D^I L^{I,N} \\ D^{II} L^{II,I} & 0 & \cdots & D^{II} L^{II,N} \\ \vdots & \vdots & \ddots & \vdots \\ D^N L^{N,I} & D^N L^{N,II} & \cdots & 0 \end{bmatrix}}_{D} y_k. \tag{16}$$

Necessary for stability of the linear discrete system (16) is $\rho(D) \leq 1$, with $\rho(D)$ being the spectral radius of D. It can be observed from (16) that for $N = 2$ subsystems, $\rho(D) = 0$, if $D^I = 0$ or $D^{II} = 0$. This is the case if one of the subsystems has no *feed–through*, i.e., the outputs are not explicitly dependent on the inputs. If both subsystems are feed–through systems, a so called *algebraic loop* is created. Consequently, zero–stability of the coupled integration of two subsystems is guaranteed, if an algebraic loop between the subsystems does not exist.

This fact is also true for an arbitrary number of subsystems. If any interconnections in the global system form a closed loop of subsystems all of which are feed–through systems, then an algebraic loop exists and explicit determination of all inputs is not possible. However, if there are no algebraic loops, all outputs can be determined explicitly as a function of the states:

$$y_k^i = h^i(x_k). \tag{17}$$

Using this relation and considering the constant state vector for one–step methods, system (16) can be written as

$$y_{k+1} = \bar{g} + Dh = const. \tag{18}$$

Obviously, system (18) is stable, leading to the statement that the coupled integration (9) is zero–stable if algebraic loops do not exist between the subsystems.

An example of an algebraic loop between two subsystems is illustrated in Figure 4. In both subsystems the output is explicitly dependent on the input, leading to an algebraic relation between the two subsystems. If $d^I = 0$ or $d^{II} = 0$, the algebraic loop is eliminated.

If the coupled integration is zero–stable, the question arises how large the global step size may be chosen such that the coupled integration is still numerically stable. For non–modular integration the critical step size is only dependent on the continuous system (2) and the integration method (3) (Hairer et al. [7]). In addition, the critical step size is dependent on the multi–rate factor, the order of extrapolation and the integration method of each subsystem in the case of coupled integration, see for example Gomm [6] and Kübler [10].

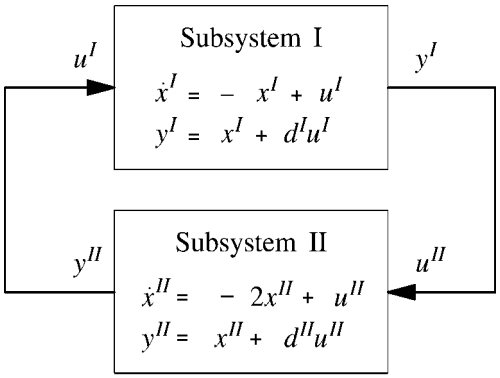


Fig. 4. Algebraic loop between two subsystems.

For practical applications the use of step size control often increases numerical efficiency. The choice of step size is based on prediction of the numerical integration error which is well–known in non–modular integration (Hairer et al. [7]). For the coupled integration the control of local and global step size is also an important issue but so far there is no general theory for error prediction and step size control.

4 METHODS OF SIMULATOR COUPLING

The analysis in Section 3 showed that zero–stability is only guaranteed if algebraic loops do not exist between the subsystems. Otherwise, instability of the modular simulation may occur due to the dynamics of the extrapolation of the unknown inputs in the output equations. This leads to two methods to obtain zero–stability in the general case if simulators are coupled for modular simulation: On the one hand, the unknown inputs may be solved for in each global time step by iterative methods, on the other hand, algebraic loops between subsystems may be eliminated by introducing additional filters in the mathematical model.

4.1 Iterative methods

Two variants of iterative simulator coupling are presented: *iteration of output equations* and *iteration of global integration step*. Both variants lead to a system of nonlinear algebraic equations for each global time step which has to be solved iteratively, in general.

4.1.1 Iteration of output equations

Integration of each subsystem is carried out according to Section 3 with extrapolation of the inputs, followed by the coupled solution of the output equations of all subsystems.

For an arbitrary number of simulators, each subsystem i can be written as

$$x^i_{k+1} = \Phi^i(\varphi^i, m^i, \tilde{u}^i), \tag{19a}$$

$$y^i_{k+1} = g^i(x^i_{k+1}, u^i_{k+1}, t_{k+1}), \qquad i = I, II, \ldots, N. \tag{19b}$$

In contrast to equations (9b) no extrapolation is needed for the input vector u^i_{k+1} for the evaluation of the output equations.

The inputs of the output equations can be eliminated using the coupling equations $u^i_{k+1} = L^i y_{k+1}$, leading to the system of nonlinear algebraic equations for the outputs of the global system

$$y_{k+1} = \Psi_o(y_{k+1}) \tag{20}$$

which has to be solved for each global time step.

In Figure 5 iteration of output equations is illustrated for $N = 2$ subsystems.

Iteration of global integration step

In contrast to iteration of output equations, the new inputs u^i_{k+1} are not only used for evaluation of the output equations, but also for interpolation of the inputs in the integration procedure, marked by $\hat{u}$ instead of $\tilde{u}$:

$$\hat{u}_{k+r} = \sum_{j=0}^{p_E} \bar{\gamma}_j(r) u_{k+1-j}. \tag{21}$$

Since the new inputs u^i_{k+1} are used in addition to the inputs used for extrapolation (8), the order of polynomial interpolation (21) is p_E. The coefficients $\bar{\gamma}_j(r)$ again can be calculated by Neville's algorithm.

Consequently, each subsystem i can be written as

$$x^i_{k+1} = \Phi^i(\varphi^i, m^i, \hat{u}^i), \tag{22a}$$

$$y^i_{k+1} = g^i(x^i_{k+1}, u^i_{k+1}, t_{k+1}), \qquad i = I, II, \ldots, N. \tag{22b}$$

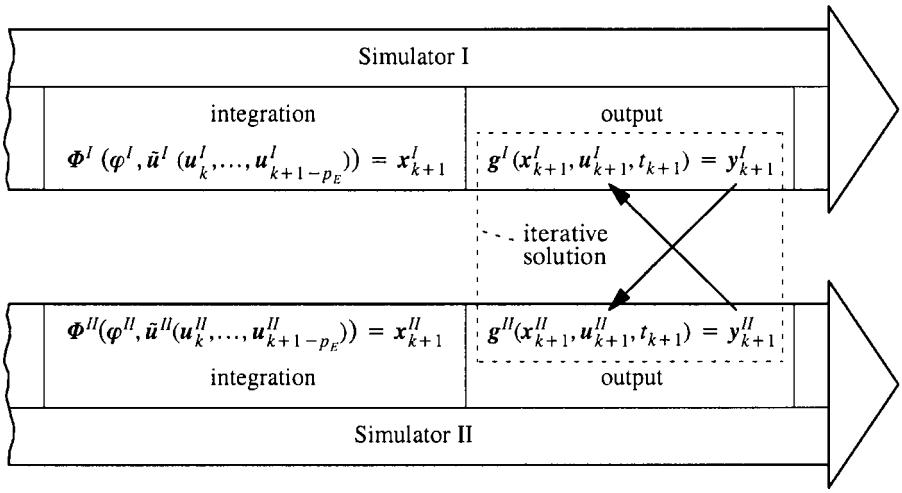


Fig. 5. Simulator coupling with iteration of output equations of two subsystems.

Using equations (22a) in (22b) leads to

$$y_{k+1}^i = g^i(x_{k+1}^i, u_{k+1}^i, t_{k+1}) = g^i(\Phi^i(\varphi^i, m^i, \hat{u}^i), u_{k+1}^i, t_{k+1}) \tag{23}$$

which can be interpreted as a function of the new inputs u_{k+1}^i only. Using the coupling equations $u_{k+1}^i = L^i y_{k+1}$ leads to the system of nonlinear algebraic equations

$$y_{k+1} = \Psi_s(y_{k+1}) \tag{24}$$

which is of the same structure as (20).

In Figure 6 iteration of global integration step is illustrated for $N = 2$ subsystems.

Analysis of zero–stability of iterative methods

Since the integration is independent of the inputs for vanishing step sizes and the output equations are the same for the two variants of iterative simulator coupling, both iterative simulator coupling schemes lead to the same system if the global step size goes to zero:

$$x_{k+1}^i = \Phi^i(\varphi^i(h^i \to 0), m^i), \tag{25a}$$

$$y_{k+1}^i = g^i(x_{k+1}^i, u_{k+1}^i, t_{k+1}), \tag{25b}$$

$$u_k^i = L^i y_k, \qquad i = I, II, \ldots, N. \tag{25c}$$

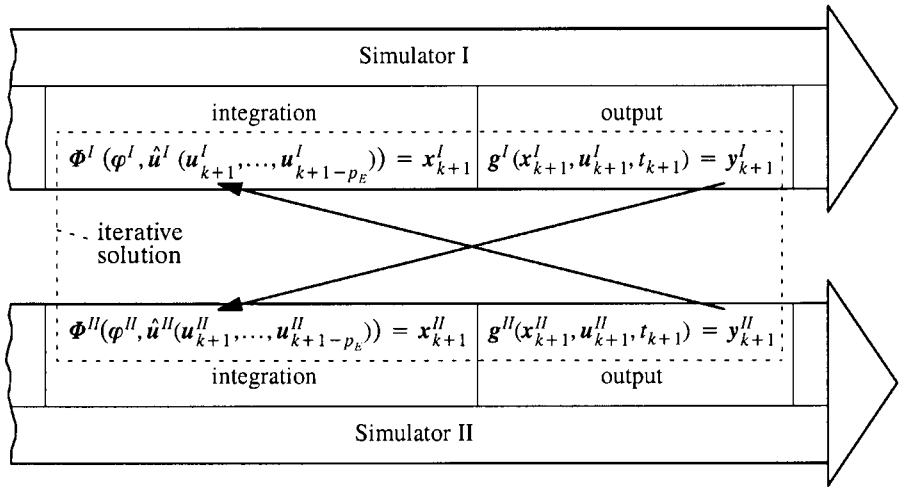


Fig. 6. Simulator coupling with iteration of global integration step of two subsystems.

System (25a) is stable, if zero–stable integrators are used, equations (25b) and (25c) are purely algebraic. Consequently, the only condition of stability of the global system (25) is the use of zero–stable integrators, leading to the statement that simulator coupling with iteration of output equations or iteration of global integration step is zero–stable, if zero–stable integrators are used.

Iterative solution of nonlinear algebraic equations

The system of nonlinear algebraic equations (20) or (24) has to be solved for each global time step at time t_{k+1} by an iterative method, in general, see Dennis and Schnabel [3], Ortega and Rheinboldt [12]. *Iteration of output equations* requires multiple evaluation of the output equations of all subsystems after each global time step with different inputs. In application of *iteration of global integration step*, the global integration step of each subsystem has to be repeated with different inputs until convergence is reached. In both cases each subsystem has to be evaluated without knowledge of the internal structure allowing access to each subsystem only by means of the input and output terminals since arbitrary simulation tools are coupled. Furthermore, no gradient information is available in general.

The Gauß–Jacobi and Gauß–Seidel method do not require gradient information but local convergence is not guaranteed by these methods. Newton's method is locally convergent, however, the Jacobian is not available in modular simulation, in general. Consequently, a quasi–Newton method which uses an approximation to the Jacobian has to be used. One of the most reliable quasi–Newton methods is Broyden's method, see Broyden [2], Dennis and Schnabel [3], which uses a secant approximation to the Jacobian and is also locally convergent.

4.2 Filter method

In Section 3 it was demonstrated that all subsystems of an algebraic loop are feed–through systems, i.e., the outputs are explicitly dependent on the inputs in each subsystem. Consequently, if one subsystem

$$\dot{x}^J = f^J(x^J, u^J, t), \tag{26a}$$

$$y^J = g^J(x^J, u^J, t), \qquad J \in [I; N] \tag{26b}$$

involved in the algebraic loop is modified, such that the outputs are no longer explicitly dependent on the inputs, the algebraic loop is eliminated. The modified mathematical model can be set up using the outputs of the original model (26) as the inputs of the filter

$$\dot{x}^F = f^F(x^F, u^F, t), \tag{27a}$$

$$y^F = g^F(x^F, t) \tag{27b}$$

whose output equations are not explicitly dependent on the inputs. Consequently, the modified mathematical model is found from equations (26), (27) and the coupling equations $u^F = y^J$:

$$\begin{bmatrix} \dot{x}^J \\ \dot{x}^F \end{bmatrix} = \begin{bmatrix} f^J(x^J, u^J, t) \\ f^F(x^F, g^J(x^J, u^J, t), t) \end{bmatrix}, \tag{28}$$

$$y^F = g^F(x^F, t). \tag{29}$$

Introducing the new state vector

$$\bar{x}^J = \begin{bmatrix} x^J \\ x^F \end{bmatrix}, \tag{30}$$

the state and output equations can be written as

$$\dot{\bar{x}}^J = \bar{f}^J(\bar{x}^J, u^J, t), \tag{31}$$

$$\bar{y}^J = \bar{g}^J(\bar{x}^J, t). \tag{32}$$

The modified mathematical model has no feed–through, since the output equations are not explicitly dependent on the inputs. Consequently, the algebraic loop is eliminated.

The described approach is closely related to the Baumgarte stabilization [1] and the force coupling approach in multibody system dynamics described by Lückel et al. [11]. The main advantage of the application of filters is, that no iterations have to be performed in modular simulation. On the other hand this approach suffers from the following disadvantages:

- The original problem is not solved, since the filter changes its dynamics.
- The parameters of the filter have to be chosen such that the dynamics of the resulting system is close to the dynamics of the original system. This choice is highly dependent on the problem to be solved.
- If the parameters are chosen such that the filter introduces only small changes in the dynamics of the original system, high frequencies are also introduced, which force the numerical integration to use substantially smaller time steps.

5 EXAMPLE

The analytical procedure is illustrated by a double pendulum modelled as a multibody system, Schiehlen [13,14]. After description and decomposition of the physical model into two subsystems, the mathematical model description for each subsystem can be set up independently. For simulator coupling without

iteration the condition of zero–stability is checked analytically and verified in numerical experiments. Finally, efficiency of the presented methods of simulator coupling is compared using this example.

5.1 Physical model description and decomposition

The double pendulum is specified by Figure 7. The physical system is decomposed in two subsystems with the corresponding physical input and output quantities as illustrated in Figure 8.

5.2 Modular mathematical description

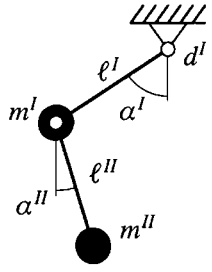
Both subsystems are treated independently, taking into consideration their corresponding inputs and outputs. Thereafter, the coupling equations are introduced to form the global mathematical model.

Subsystem I
The input vector of subsystem I is the applied force $u^I = \begin{bmatrix} u^{I1} & u^{I2} \end{bmatrix}^T$ in the inertial frame $\{e_1, e_2\}$. The equation of motion with the generalized coordinate α^I reads

$$m^I l^{I^2} \ddot{\alpha}^I = u^{I1} l^I \cos\alpha^I + (u^{I2} - m^I g) l^I \sin\alpha^I - d^I \dot{\alpha}^I. \tag{33}$$

The output vector is given by the absolute acceleration of the body in the inertial frame

$$\begin{aligned} y^I &= \begin{bmatrix} y^{I1} \\ y^{I2} \end{bmatrix} \\ &= \begin{bmatrix} \cos\alpha^I \\ \sin\alpha^I \end{bmatrix} \frac{1}{m^I} \left(u^{I1} \cos\alpha^I + (u^{I2} - m^I g) \sin\alpha^I \right) + \begin{bmatrix} -\sin\alpha^I \\ \cos\alpha^I \end{bmatrix} l^I \dot{\alpha}^{I^2}. \end{aligned} \tag{34}$$



mass	$m^I = 1,\ m^{II} = \beta$
length	$\ell^I = \ell^{II} = 1$
damping	$d^I = 0.5$
initial condition	$\alpha^I(0) = -1,\ \alpha^{II}(0) = 0.3$

Fig. 7. Double pendulum.

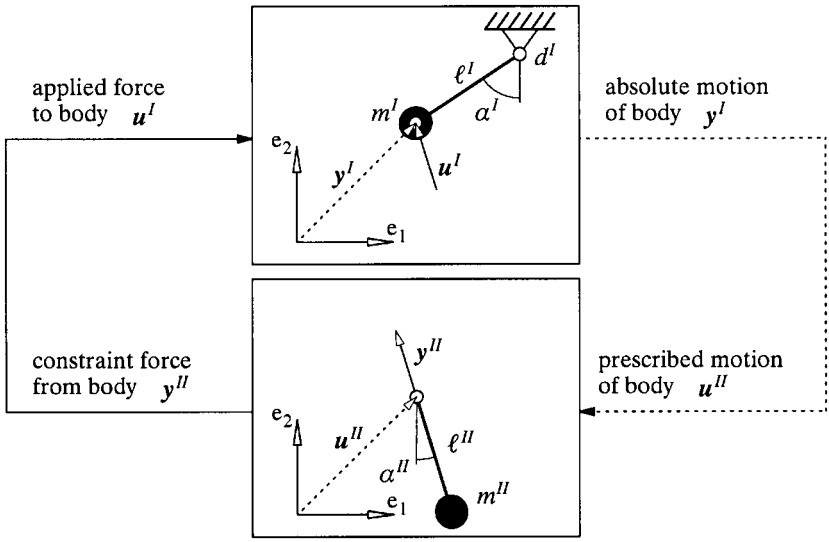


Fig. 8. Modular double pendulum.

Subsystem II

The input vector of subsystem II is the absolute acceleration of the moving hinge $u^{II} = \begin{bmatrix} u^{II1} & u^{II2} \end{bmatrix}^T$ of the pendulum in the inertial frame. The equation of motion with the generalized coordinate α^{II} is given by

$$m^{II} l^{II^2} \dot{\alpha}^{II} + m^{II} l^{II} u^{II1} \cos\alpha^{II} + m^{II} l^{II} u^{II2} \sin\alpha^{II} = -m^{II} g l^{II} \sin\alpha^{II}. \tag{35}$$

The output vector of subsystem II is given by the constraint force in the inertial frame

$$\begin{aligned} y^{II} &= \begin{bmatrix} y^{II1} \\ y^{II2} \end{bmatrix} \\ &= \begin{bmatrix} -\sin\alpha^{II} \\ \cos\alpha^{II} \end{bmatrix} m^{II} \left(u^{II1} \sin\alpha^{II} - u^{II2} \cos\alpha^{II} - l^{II} \dot{\alpha}^{II^2} - g \cos\alpha^{II} \right). \end{aligned} \tag{36}$$

Mathematical coupling

The coupling equations are given according to equation (1c) by

$$\begin{bmatrix} u^I \\ u^{II} \end{bmatrix} = \begin{bmatrix} 0 & E \\ E & 0 \end{bmatrix} \begin{bmatrix} y^I \\ y^{II} \end{bmatrix} \tag{37}$$

with E being the 2×2–identity matrix.

5.3 Modular simulation

Analysis of zero–stability

The output equations (34), (36) are linearly dependent on the inputs and are therefore of the form (14) with

$$D^I(x^I) = \frac{1}{m^I}\begin{bmatrix} \cos^2\alpha^I & \sin\alpha^I\cos\alpha^I \\ \sin\alpha^I\cos\alpha^I & \sin^2\alpha^I \end{bmatrix}, \tag{38a}$$

$$D^{II}(x^{II}) = m^{II}\begin{bmatrix} -\sin^2\alpha^{II} & \sin\alpha^{II}\cos\alpha^{II} \\ \sin\alpha^{II}\cos\alpha^{II} & -\cos^2\alpha^{II} \end{bmatrix}. \tag{38b}$$

Following the outline of Section 3.3, zero–stability of simulator coupling without iteration is investigated with the use of one–step methods, resulting in the linear discrete system (16) with the matrix

$$D(_0) = \begin{bmatrix} 0 & D^I(x_0^I) \\ D^{II}(x_0^{II}) & 0 \end{bmatrix}. \tag{39}$$

The spectral radius can be calculated analytically resulting in

$$\rho(D(x_0)) = \underbrace{\left|\sin(\alpha_0^I - \alpha_0^{II})\right|}_{\leq 1}\underbrace{\sqrt{\frac{m^{II}}{m^I}}}_{=:\beta}. \tag{40}$$

Consequently, if $m^I > m^{II}$ ($\beta < 1$), the simulator coupling without iteration is always zero–stable. If $m^I \leq m^{II}$ ($\beta \geq 1$), zero–stability is dependent on the initial states.

Numerical results

To compare the methods of simulator coupling presented in section 4, both subsystems are simulated using Euler's method with constant step size and extrapolation of the inputs of order zero ($p_E = 1$). Both multi–rate factors are set to one ($m^I = m^{II} = 1$), resulting in the same local and global step size. For this choice, simulator coupling with iteration of the outputs and with iteration of the global step are identical and are referred to as *simulator coupling with iteration* in the following. For the iterative solution Broyden's method is used.

To stabilize the modular simulation by the filter method, a PT_2–system with the roots located on the lines at a 45 degree angle in the left complex half–plane is used to filter the output of subsystem II.

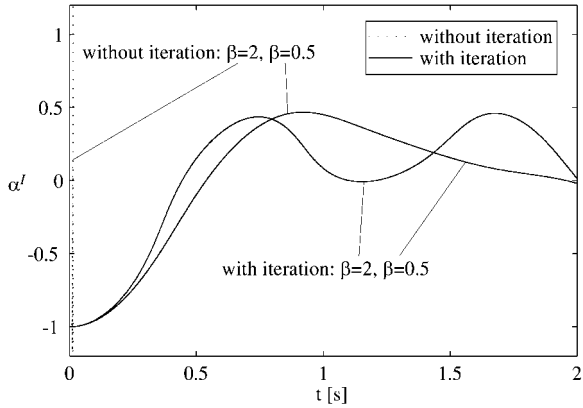


Fig. 9. Stability of simulator coupling with and without iteration.

To verify the analysis of zero–stability, Figure 9 shows the results of simulator coupling with and without iteration for mass ratio $\beta = 2$ and $\beta = 0.5$. Simulator coupling with iteration is stable for both mass ratios, whereas simulator coupling without iteration is unstable for mass ratio $\beta = 2$ as predicted by the analytical analysis. This result remains unchanged, if the step size is reduced to arbitrarily small values.

To compare the accuracy of the different methods with same global step size $H = 0.1$ ms, the mean of the global relative integration error is plotted over the mass ratio β, Figure 10. The relative error tolerance of Broyden's method is set to 10^{-8} for the iterative method. For the filter method the remaining filter parameter is selected as "stiff" as possible for each mass ratio β, such that integration is still numerically stable, e.g., for $\beta = 4$ the absolute value of the eigen-

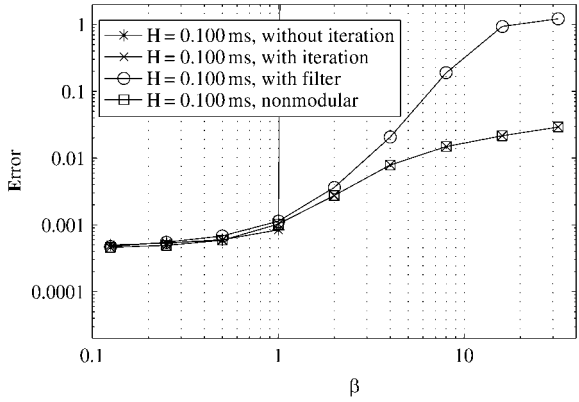


Fig. 10. Error of different methods of simulator coupling with same step size.

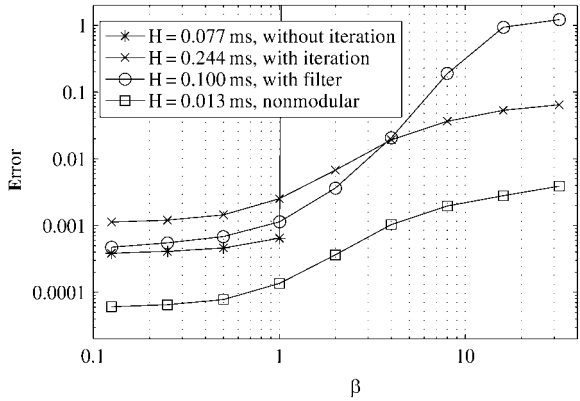


Fig. 11. Error of different methods of simulator coupling with same computation time.

values of the PT_2–system is 1900 s^{-1}. For mass ratio $\beta \leq 1$ the integration error of all methods is approximately the same. According to the analysis of zero–stability, simulator coupling becomes unstable without iteration for mass ratio $\beta > 1$. Simulator coupling with iteration yields almost identical accuracy as non–modular integration of the global system for the whole range of mass ratio β, whereby the error gets larger with increasing mass ratio β due to the higher frequencies of the system, see Figure 9. Simulator coupling with application of filters is also stable for the whole range of mass ratio β. However, for $\beta > 1$ the error gets much larger than the error of simulator coupling with iteration.

To compare the efficiency of the two methods of stabilization, i.e., simulator coupling with iteration or filtering, the global step size for each method is chosen such that the same computation time is achieved. Again, the integration error is plotted over the mass ratio β (Fig. 11). For small mass ratios β, the error of the filter method is smaller than the error of the iteration method. For larger mass ratio $\beta > 4$, the filter method is still stable, but the error is getting larger and larger, such that simulation results are worthless. However, simulator coupling with iteration leads to reliable results for the whole range of mass ratio β. This phenomenon is also illustrated conclusively in Figure 12 for mass ratio $\beta = 32$.

6 CONCLUSIONS

Modular simulation of a system is characterized by decomposition in subsystems and coupling of the independent subsystem simulators. The investigation of zero–stability shows that convergence of the modular simulation is dependent on the structure of the mathematical model of the subsystems: Stability for simulator coupling without iteration is only guaranteed if algebraic loops do not exist

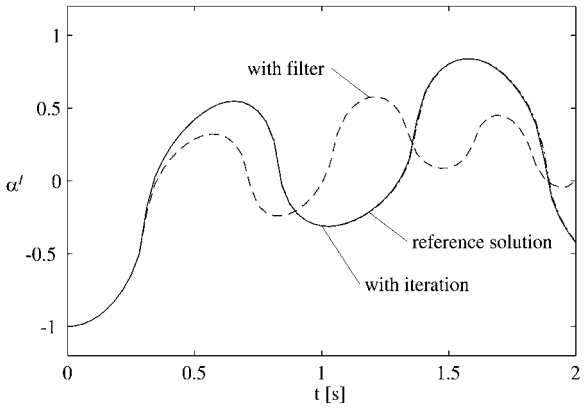


Fig. 12. Simulator coupling with iteration and filtering for mass ratio $\beta = 32$.

between the subsystems. Two methods of simulator coupling for more general systems including algebraic loops are presented: On the one hand, elimination of algebraic loops by application of filters is demonstrated, on the other hand, it is shown that stability is always guaranteed by iterative simulator coupling schemes.

Two variants of the iterative method are introduced: *iteration of output equations* and *iteration of global integration step*. In contrast to the application of filters the iterative simulator coupling schemes open a systematic and accurate way to couple simulation tools.

The analytical analysis is illustrated with numerical experiments for a double pendulum decomposed into two single pendula modelled as multibody systems.

REFERENCES

1. Baumgarte, J.: Stabilization of constraints and integrals of motion in dynamical systems. *Comput. Methods Appl. Mech. Engrg*., 1, 1–16, 1972.
2. Broyden, C.G.: Solving Nonlinear Simultaneous Equations. *Mathematics of Computation*, 19, 577–593, 1965.
3. Dennis, J.E., Schnabel, R.B.: *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1983.
4. Dürr, R., Neerpasch U., Schiehlen W., Witte L.: Mechatronik und STEP, Standardisierung eines neutralen Datenformats in STEP für die Simulation mechatronischer Systeme. *Produktdaten Journal*, 2, 12–19, 1995.
5. Dürr, R., Schiehlen, W.: Exchange of Design Data for Mechatronic Systems. *Booklet of Abstracts, EUROMECH Colloquium 370 'Synthesis of Mechatronic Systems',* 9–11, 1997.
6. Gomm,W.: Stability Analysis of Explicit Multirate Methods. *Mathematics and Computers in Simulation* XXIII, 34–50, 1981.

7. Hairer, E., Nørsett, S.P., Wanner, G.: *Solving Ordinary Differential Equations I. Nonstiff Problems.* Berlin: Springer, 1993.
8. *IEEE standard 1076.1*. Draft. 1998.
9. Junker, F., Hahn, M., Lückel, J.: Systematic Modelling of Mechanical Parts in Mechatronic Systems. *Systems Analysis Modelling Simulation*, 18–19, 619–624b, 1995.
10. Kübler, R.: Numerische Stabilität und Genauigkeit bei modularer Integration mit Anwendung auf die Hardware-in-the-Loop-Simultion. *Z. Angew. Math. Mech*. 78 S2, 569–570, 1998.
11. Lückel, J., Junker, F., Toepper, S.: Block–oriented Modelling of Rigid Multibody Systems with Regard to Subsystem Techniques. In: Schiehlen, W.: *Advanced Multibody Dynamics.* Dordrecht: Kluwer Academic Publishers, 49–66, 1993.
12. Ortega, J.M., Rheinboldt, W.C.: *Iterative Solution of Nonlinear Equations in Several Variables.* New York: Academic Press, 1970.
13. Schiehlen, W.: *Technische Dynamik*. Stuttgart: Teubner, 1986.
14. Schiehlen, W.: Multibody System Dynamics: Roots and Perspectives. *Multibody System Dynamics*, 1, 149–188, 1997.
15. Stoer, J., Bulirsch, R.: *Introduction to Numerical Analysis.* New York: Springer, 1980.
16. Wünsche, S., Clauß, C., Schwarz, P., Winkler, F.: Microsystem Design Using Simulator Coupling. *Proceedings of the European Design and Test Conference*, 113–118, 1997.