Engineering with Computers (2020) 36:1579–1587
https://doi.org/10.1007/s00366-019-00783-4

ORIGINAL ARTICLE

Energy conservation and coupling error reduction in non-iterative co-simulations

Severin Sadjina[1] · Eilif Pedersen[2]

Received: 3 August 2016 / Accepted: 17 May 2019 / Published online: 30 May 2019

Abstract
When simulators are energetically coupled in a co-simulation, coupling errors alter the total energy of the full system. This distorts system dynamics, lowers the quality of the results, and can lead to instability. Using power bonds to realize simulator coupling, the *Energy-Conservation-based Co-Simulation* method (ECCO) (Sadjina et al. Eng Comput 33(3):607–620. https://doi.org/10.1007/s00366-016-0492-8, 2017) exploits these concepts to define non-iterative global error estimation and adaptive step size control relying on coupling variable data alone. Following similar argumentation, the *Nearly Energy Preserving Coupling Element* (NEPCE) (Benedikt et al. Nepce—a nearly energy preserving coupling element for weak-coupled problems and co-simulation. In: V International conference on computational methods for coupled problems in science and engineering, coupled problems, pp 1021–1032. International Center for Numerical Methods in Engineering, 2013) uses corrections to the simulator inputs to approximately ensure energy conservation. Here, we discuss a modification to NEPCE for when direct feed-through is present in one of the coupled simulators. We further demonstrate how accuracy and efficiency in non-iterative co-simulations are substantially enhanced when combining NEPCE with ECCO's adaptive step size controller. A quarter car model with linear and nonlinear damping characteristics serves as a co-simulation benchmark, and we observe reductions of the coupling errors of up to 98% utilizing the concepts discussed here.

Keywords Co-simulation · Simulator coupling · Error reduction · Power bond · Energy conservation

CR Subject Classification I.6.8

1 Introduction

Co-simulation allows for the independent and parallel modeling and simulation of complex systems including multiple physical and engineering domains, the efficient use of expert knowledge, tailored software tools, and suited solver methods, and the protection of intellectual property within models. These benefits make this kind of simulator coupling an attractive choice, especially from an industrial perspective. But the fact that coupled subsystems are solved independently of each other between discrete communication time points also emphasizes accuracy and stability issues [9, 13, 14].

The flow and the conservation of energy between coupled simulators can be conveniently studied when using *power bonds* to realize the couplings. A power bond is a direct energetic connection between subsystems defined by inputs and outputs whose product gives a physical power: force and velocity, electric current and voltage, pressure and flow rate, to name a few. Since subsystems in a co-simulation advance in time independently of each other, energy transactions between them are inherently inaccurate. *Energy residuals* emerge as a consequence and directly affect the total energy of the overall coupled system. Consequently, system dynamics are distorted and co-simulation accuracy and stability are challenged.

These concepts are exploited in the *Energy-Conservation-based Co-Simulation* method [11] (ECCO). Since energy residuals are a direct expression of coupling errors, they are a versatile tool to assess the quality of co-simulations. Based

✉ Severin Sadjina
Severin.Sadjina@sintef.no

[1] SINTEF Ålesund, 6009 Ålesund, Norway

[2] Department of Marine Technology, Norwegian University of Science and Technology, 7491 Trondheim, Norway

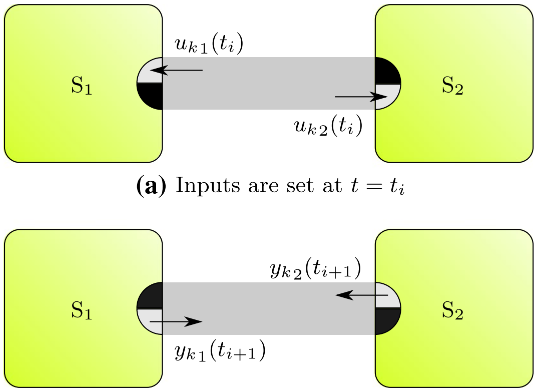
on such error estimators, ECCO defines an adaptive control of the co-simulation step size, and displays significant improvements in the accuracy and efficiency of non-iterative co-simulations. Similar arguments are used in the *Nearly Energy Preserving Coupling Element* [3] (NEPCE) to introduce corrections to the flow of (generalized) power between simulators to minimize coupling errors.

Here, we have a closer look at NEPCE and its energy-conserving properties. We further propose a modification to include the presence of direct feed-through, enhancing its performance. NEPCE's efficiency is based on the assumption that the coupling variables are slowly varying functions of time. This assumption can be violated if the co-simulation (macro) time step is not chosen carefully. We demonstrate how this issue is efficiently handled by ECCO's energy-conservation-based adaptive step size control to substantially improve accuracy and efficiency. Since the resulting framework is non-iterative, it is computationally inexpensive and well suited for industrial applications, where commercial software solutions often prohibit iterative approaches.

This paper is organized as follows: in Sect. 2, we start with a brief recapitulation of the flow and conservation of energy in co-simulations using power bonds. Next, we study NEPCE's non-iterative corrections to the simulator inputs in Sect. 3 and show how they should be modified in the presence of direct feed-through. Section 4 discusses how these corrections can be combined with ECCO's adaptive step size control, and a quarter car model is then used in Sect. 5 to demonstrate the performance of the proposed method and its influence on co-simulation accuracy and efficiency. Finally, we give a conclusion in Sect. 6.

2 Energy conservation in co-simulations

Most commonly, co-simulations are realized by letting the simulators advance in time in parallel and independently of each other, and then synchronizing coupling data at discrete communication time points. This weak coupling approach is easily implemented and relatively efficient on paper: it is universally applicable for industrial applications (which usually prohibit iterative schemes) and the parallelization potential holds the promise of substantial simulation speed-ups. Its major weaknesses, however, are accuracy and stability. Input quantities are generally unknown during the time integrations inside the simulators. They must, therefore, be approximated, and are often simply held constant. A sufficiently small macro-time step has to be chosen to keep the coupling errors which result from this scheme contained.[1]



(a) Inputs are set at $t = t_i$

(b) Outputs are retrieved at $t = t_{i+1}$

Fig. 1 Two coupled simulators exchange energy through a power bond in a co-simulation [11]

2.1 Power and energy residuals

The use of power bonds from bond graph theory [4, 10] allows to reframe these issues in terms of energy conservation considerations [11]. A power bond k is defined by a pair of power variables—a flow f_k and an effort e_k—whose product $P_k = e_k f_k$ gives a physical power. Powers and energies, the universal currencies of physical systems, are directly accessible in co-simulations when using power bonds.

As an example, consider the flow of energy between two simulators S$_1$ and S$_2$ that are coupled via a power bond k, see Fig. 1. From the point of view of S$_1$, energy is transferred to S$_2$ at a rate:

$$P_{k_1}(t) = \tilde{u}_{k_1}(t) y_{k_1}(t), \tag{1a}$$

where $y_{k_1}(t)$ is the output and $\tilde{u}_{k_1}(t) \approx u_{k_1}(t)$ is an approximation of the generally unknown value $u_{k_1}(t)$. If, instead, we consider the energy transfer from the other simulator's perspective, we conclude that:

$$P_{k_2}(t) = \tilde{u}_{k_2}(t) y_{k_2}(t). \tag{1b}$$

This is problematic because it fundamentally violates the conservation of energy,

$$-(P_{k_1} + P_{k_2}) \neq 0, \tag{2}$$

because, generally, $\tilde{u}_{k_1}(t) \neq u_{k_1}(t)$ and $\tilde{u}_{k_2}(t) \neq u_{k_2}(t)$. Hence, a *residual energy* is incorrectly created due to the independent time integrations of the simulators during the macro-time step $t_i \to t_{i+1} = t_i + \Delta t_i$, [11]

$$\delta E_k(t_{i+1}) \equiv \int_{t_i}^{t_{i+1}} \delta P_k(t)\,\mathrm{d}t, \tag{3a}$$

where

$$\delta P_k \equiv -(P_{k_1} + P_{k_2}) \tag{3b}$$

[1] For a comprehensive overview over numerous co-simulation approaches and methodologies see, for example, Ref. [8].

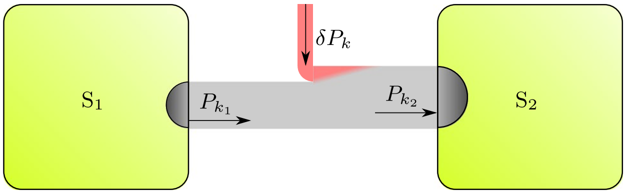


Fig. 2 Total system dynamics are distorted by a residual power δP_k between two energetically coupled simulators due to the independent time integrations [11]

is the *residual power* for the power bond k, see Fig. 2 for an illustration. At each macro-time step, the residual energy δE_k is directly added to the total energy of the overall coupled system [11]. As a consequence, system dynamics are distorted and the quality of the co-simulation reduced. The power $P_{k_{12}}$ transmitted from S_1 to S_2 can be obtained from the simulator outputs as:

$$P_{k_{12}}(t) = \sigma_{12} y_{k_1}(t) y_{k_2}(t), \tag{4}$$

where the sign $\sigma_{12} \equiv (\mathsf{L}_{12} - \mathsf{L}_{21})/2$ is determined by the corresponding elements of the connection graph matrix L, which we will shortly introduce in Sect. 2.2.

Luckily, inaccurate energy transactions provide us with a versatile error estimator because the corresponding residual energies are a direct expression of the co-simulation coupling errors and the violation of energy conservation. This is exploited by the ECCO algorithm to define an adaptive macro-step size controller: For input extrapolation of order m, the residual energy scales quadratic with the step size [11], $\delta E_k = \mathcal{O}(\Delta t^{m+2})$. Consequently, the conservation of energy can be approximately satisfied by controlling the macro-step size, optimizing the quality and efficiency of co-simulations.

2.2 Local errors in the coupling variables

Considering the time evolution of the internal states $\mathbf{x} = \{x_1, x_2\}$ of the coupled simulators between the discrete communication time points t_i and t_{i+1},

$$\dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \tilde{\mathbf{u}}(t)\big), \quad t \in (t_i, t_{i+1}], \tag{5a}$$

simulator coupling can be expressed as:

$$\mathbf{y}(t_{i+1}) = \mathbf{g}\big(\mathbf{x}(t_{i+1}), \tilde{\mathbf{u}}(t_{i+1})\big), \tag{5b}$$

$$\mathbf{u}(t_{i+1}) = \mathsf{L}\mathbf{y}(t_{i+1}), \tag{5c}$$

where L is a connection graph matrix that relates outputs $\mathbf{y}$ and inputs $\mathbf{u}$ at communication time points. In non-iterative co-simulations, the inputs are generally unknown and have to be approximated during the time integrations inside the simulators, $\tilde{\mathbf{u}}(t) \approx \mathbf{u}(t)$. Most commonly, they are simply held constant such that $\tilde{\mathbf{u}}(t) = \mathbf{u}(t_i)$ for $t \in (t_i, t_{i+1}]$.

Let us in the following have a closer look at the local coupling errors which stem from these approximations and the independent time integrations in the subsystems between communication time points. For the case of coupling via power bonds, these errors are conveniently represented as power and energy errors and directly related to the conservation of energy throughout the entire coupled system. Using energies and powers as error metrics instead of non-energetic quantities has two major advantages: (i) they offer a more holistic and intuitive approach by considering the flow of energy between subsystems directly. (ii) They avoid that some simulator's contributions to the global error are given too much weight. If, for example, one simulator outputs a force and another a position, the numerical values of the force output will typically be much larger than those of the position output.[2] The same will then generally be true for the numerical values of the corresponding errors, skewing the actual simulators' contributions to the global co-simulation error. The use of energy and power errors solves this issue in an elegant fashion.

In the next section, we will discuss how we can minimize local coupling errors. The subsystem states are inaccessible in a typical co-simulation setting and can not be directly altered. Instead, corrections to the inputs can be derived such that the residual energies between simulators are minimized and energy conservation is approximately satisfied. These corrections ideally cancel the local errors in the inputs which are given by

$$\begin{aligned} \Delta\mathbf{u}(t) &= \tilde{\mathbf{u}}(t) - \mathbf{u}_0(t) \\ &= \tilde{\mathbf{u}}(t) - \mathsf{L}\mathbf{y}_0(t) \\ &= \tilde{\mathbf{u}}(t) - \mathsf{L}\big(\mathbf{y}(t) - \Delta\mathbf{y}(t)\big), \end{aligned} \tag{6a}$$

where $\mathbf{u}_0(t)$ is the exact solution and we used that $\mathbf{u}_0(t) = \mathsf{L}\mathbf{y}_0(t)$ for any time t. The errors in the outputs evaluate to[3]

$$\begin{aligned} \Delta\mathbf{y}(t) &= \mathbf{y}(t) - \mathbf{y}_0(t) \\ &= \mathbf{g}\big(\mathbf{x}(t), \tilde{\mathbf{u}}(t)\big) - \mathbf{g}\big(\mathbf{x}_0(t), \mathbf{u}_0(t)\big) \\ &= \mathsf{J}_{\mathbf{g}}(\mathbf{u})\Delta\mathbf{u}(t) + \mathsf{J}_{\mathbf{g}}(\mathbf{x})\Delta\mathbf{x}(t) \\ &\quad + \mathcal{O}(\Delta t^{m+2}), \end{aligned} \tag{6b}$$

where $\mathsf{J}_{\mathbf{g}_{ij}}(\mathbf{u}) = \partial g_i / \partial u_j$ is the interface Jacobian and $\mathsf{J}_{\mathbf{g}_{ij}}(\mathbf{x}) = \partial g_i / \partial x_j$. While the error contributions from the state vector are

$$\Delta\mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}_0(t) = \mathcal{O}(\Delta t^{m+2}), \tag{6c}$$

[2] Note that a good error estimator should take this into account, regardless of whether or not energies and powers are used directly.

[3] Compare to, for example, Ref. [1].

the input errors appear to order $\Delta\mathbf{u} = \mathcal{O}(\Delta t^{m+1})$. Consequently, if one of the simulators S_k has direct feed-through, the output errors are also of order $\mathcal{O}(\Delta t^{m+1})$ because then $\mathsf{J}_{\mathbf{g}_{kk}}(\mathbf{u}) \neq 0$. Using Eq. (6b) in Eq. (6a) and rewriting thus gives:

$$\Delta\mathbf{u}(t) = \left(1 - \mathsf{LJ}\right)^{-1}\left(\tilde{\mathbf{u}}(t) - \mathsf{L}\mathbf{y}(t)\right) + \mathcal{O}(\Delta t^{m+2}), \tag{7}$$

where we set $\mathsf{J} \equiv \mathsf{J}_{\mathbf{g}}(\mathbf{u})$ for brevity.[4]

3 Non-iterative energy-conservation-based corrections

Let us now take the idea of energy conservation in co-simulations a step further by directly modifying the coupling variables such that energy transactions between simulators are described more accurately. This concept was first introduced by NEPCE [3]. In this section, we will explore it in detail and generalize it to include the presence of direct feed-through. In Sect. 4, we then discuss how the energy-conservation-based corrections studied here can be combined with ECCO's non-iterative adaptive step size controller, and Sect. 5 demonstrates the substantial improvements in accuracy and efficiency thus obtained using a quarter car co-simulation benchmark model.

As can be seen from Eqs. (1) and (3), a residual energy

$$\delta E_k(t_{i+1}) = -\int_{t_i}^{t_{i+1}} \tilde{\mathbf{u}}_k(t)\cdot\mathbf{y}_k(t)\,\mathrm{d}t \tag{8}$$

is accumulated during the time step $t_i \rightarrow t_{i+1}$ for a power bond k connecting the inputs $\tilde{\mathbf{u}}_k = \{\tilde{u}_{k_1}, \tilde{u}_{k_2}\}$ and the outputs $\mathbf{y}_k = \{y_{k_1}, y_{k_2}\}$. The concept behind NEPCE is to find corrections $\delta\mathbf{u}_k = \{\delta u_{k_1}, \delta u_{k_2}\}$ to the inputs at communication time instant $t = t_i$ with the aim of reducing the residual energy by a factor of $(1-\alpha)$, such that:

$$(\alpha - 1)\delta E_k(t_{i+1}) = \int_{t_i}^{t_{i+1}} \left(\tilde{\mathbf{u}}_k(t) + \delta\mathbf{u}_k(t)\right)\cdot\mathbf{y}_k(t)\,\mathrm{d}t \tag{9}$$

with the tuning factor $\alpha \in [0,1]$. Ideally, $\alpha = 1$ if the corrections accurately track the errors in the inputs, $\delta\mathbf{u}_k(t) = -\Delta\mathbf{u}_k(t)$. While this can not be realized in general for non-iterative co-simulations, however, corrections should be of the same order as the errors in the input (6a), $\delta\mathbf{u}_k = \mathcal{O}(\Delta t^{m+1})$, to mitigate their effects.

3.1 NEPCE corrections

First, let us discuss the case where none of the simulators have direct feed-through. This corresponds to NEPCE as introduced in Ref. [3]. Then, the errors in the inputs (7) are simply

$$\Delta\mathbf{u}(t) = \tilde{\mathbf{u}}(t) - \mathsf{L}\mathbf{y}(t) + \mathcal{O}(\Delta t^{m+2}), \tag{10}$$

and Eq. (9) suffices. Choosing

$$\delta\mathbf{u}(t) = -\Delta\mathbf{u}(t) \approx \mathsf{L}\mathbf{y}(t) - \tilde{\mathbf{u}}(t)$$

would make the residual energy vanish and the coupling quantities exact to order $\mathcal{O}(\Delta t^{m+1})$. As already mentioned, this is not possible for non-iterative co-simulations because $\mathbf{y}(t)$ is unknown a priori for $t = (t_i, t_{i+1}]$. Instead, we realize the correction in terms of previous coupling data, [3]

$$\delta\mathbf{u}(t) \approx \frac{\alpha}{\Delta t_i}\int_{t_{i-1}}^{t_i} \left(\mathsf{L}\mathbf{y}(\tau) - \tilde{\mathbf{u}}(\tau)\right)\mathrm{d}\tau, \tag{11}$$

for $t \in (t_i, t_{i+1}]$, assuming that the coupling variables and the errors are slowly varying on the scale of the time step Δt.

Note that this is a reasonable assumption in theory: in a co-simulation the macro-time step should be chosen such that the dynamics of the system can be sufficiently well resolved in time. A violation of this assumption is equivalent to the macro-time step simply being too large for the problem at hand. In Sect. 4, we will take a big step towards ensuring that this crucial assumption holds by combining the energy-conservation-based input corrections discussed in the present section with the energy-conservation-based adaptive step size controller ECCO.

3.2 Corrections with direct feed-through

As discussed previously, in the presence of direct feed-through, the errors in the inputs are given by Eq. (7), and Eq. (11) should be modified to

$$\delta\mathbf{u}(t) \approx \frac{\alpha}{\Delta t_i}\left(1 - \mathsf{LJ}\right)^{-1}\int_{t_{i-1}}^{t_i} \left(\mathsf{L}\mathbf{y}(\tau) - \tilde{\mathbf{u}}(\tau)\right)\mathrm{d}\tau \tag{12}$$

to include all coupling errors of order $\mathcal{O}(\Delta t^{m+1})$. It is important to point out that Eq. (12) requires the knowledge of the generally time-dependent interface Jacobian $\mathsf{J}_{ij} = \partial g_i/\partial u_j$. In practical applications, it will likely not be available and the unmodified NEPCE form (11) should be chosen. While disregarding the output error contribution in Eqs. (6), it is still an improvement over the uncorrected co-simulation in the presence of direct feed-through.

Finally, note that we can safely disregard the case where both simulators exhibit direct feed-through, because it amounts to an algebraic loop which indicates that the

[4] Note that the contribution from $\mathsf{J}_{\mathbf{g}}(\mathbf{x})$ in Eq. (6b) is of order $\mathcal{O}(\Delta t^{m+2})$ due to Eq. (6c) and is thus not shown.

particular system reticulation is not suitable for non-iterative co-simulation and ill-chosen.

4 Energy-conservation-based adaptive step size control

The previous section discussed NEPCE and a modification to it in the presence of direct feed-through. The approach to energy-conservation-based corrections to the inputs in non-iterative co-simulations relies on the assumption that the coupling variables are slowly varying functions of time on the scale of the macro-time step. When this assumption does not hold, the corrections become increasingly ineffective and can even lead to instability by exciting relatively fast dynamics in the subsystems [3]. In other words, the smaller the chosen macro-time step the more efficient and beneficial the input corrections become. The *Energy-Conservation-based Co-Simulation* method (ECCO) provides a framework that allows us to adaptively choose a macro-step size which (given some tolerances) approximately ensures energy conservation in non-iterative co-simulations. This concept and its performance have recently been studied [11], and we shall in the following combine it with the energy-conservation-based input corrections from the previous section to define a non-iterative co-simulation framework yielding high accuracy and efficiency without the use of any simulator-internal data.

We propose the use of an I-controller to determine a new optimal step size

$$\Delta t_{i+1} = \alpha_{\mathrm{s}} \epsilon(t_i)^{-k_{\mathrm{I}}} \Delta t_i \tag{13}$$

as a function of an error indicator ϵ. Here, $k_{\mathrm{I}} = 0.3/(m+2)$ is the integral gain[5], m is the extrapolation order ($m = 0$ for constant extrapolation), and $\alpha_{\mathrm{s}} \in [0.8, 0.9]$ is a safety factor. The scalar error indicator can be defined as [11]:

$$\epsilon(t) \equiv \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\frac{\delta E_k(t)}{r_k\left(E_{0k} + |E_k(t)|\right)}\right)^2}, \tag{14}$$

using the residual energies δE_k and energies $E_k(t_{i+1}) \approx P_{k_{12}}(t_{i+1})\Delta t_i$ transmitted per time step for all N power bonds. Here, the typical energy scale E_{0k} and the relative tolerance r_k are freely configurable parameters which determine the energy resolution for the power bond k. The I-controller (13) aims to find and maintain a balance between accuracy and efficiency by choosing a step size for which $\epsilon \approx 1$: efficiency can be improved if $\epsilon < 1$ by increasing the step size, while accuracy needs to be increased by choosing smaller time steps if $\epsilon > 1$. To avoid rapid oscillations in the step size on one hand, and inefficiently small step sizes on the other, the step size itself and its rate of change are restricted by the parameters Δt_{min} and Δt_{max}, and Θ_{min} and Θ_{max}, respectively. Table 1 lists the full configuration used for the benchmark tests of Sect. 5.

Table 1 Configuration of the adaptive step size controller (13) for the benchmark model in Sect. 5

	Value	Unit
α_{s}	0.8	
Δt_{min}	10	µs
Δt_{max}	10	ms
Θ_{min}	0.2	
Θ_{max}	1.5	
E_0	750	J

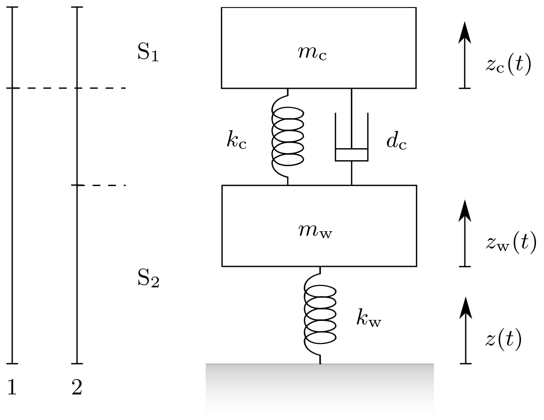


Fig. 3 The quarter car benchmark model is split into the subsystems S_1 and S_2 for co-simulation using the two distinct reticulations 1 and 2

5 Co-simulation benchmark tests

To assess the performance of the methods discussed in Sects. 3 and 4, we employ a quarter car model as described in Ref. [1] and split it into two subsystems connected via a power bond, see Fig. 3. This model can be considered two coupled Dahlquist test equations [7] and is thus well suited as a co-simulation benchmark test case [1, 2, 6, 11, 12]. We further study two different reticulations for the co-simulation

[5] The denominator $m+2$ represents the order of the error, here $\delta E_k = \mathcal{O}(\Delta t^{m+2})$. The corrections discussed in Sect. 3 have the aim of canceling the energy error δE_k to leading order in Δt, and thus one may expect to choose $k_{\mathrm{I}} = 0.3/(m+3)$. Here, we decide against this alteration for two reasons: (i) as mentioned previously, the leading terms in the error can in general not be canceled exactly for the non-iterative case, as expressed by the tuning factor α. (ii) The actual benchmark results discussed in Sect. 5 show little to no dependence on this chance.

Table 2 Parameters for the linear quarter car benchmark model according to Ref. [1]

	Value	Unit
m_c	400	kg
m_w	40	kg
k_c	15,000	N m^{-1}
k_w	150,000	N m^{-1}
d_c	1000	N s m^{-1}
n_d	0.5	

Table 3 Parameter changes to include nonlinear damping forces in the benchmark model according to Ref. [5]

	Value	Unit
d_c	900	N(s/m)$^{1/2}$
n_d	1.5	

and also investigate nonlinear damping characteristics.[6] The corresponding model and the underlying equations are directly adapted from Ref. [11], the parameters are summarized in Table 2 for the linear test case and in Table 3 for the nonlinear case.[7]

We generally carry out the time integrations in the subsimulators using micro-step sizes of $\Delta t_{\mathrm{S}_1} = \Delta t_{\mathrm{S}_2} = \Delta t/256$ with the forward Euler method to focus on the co-simulation coupling errors.[8] As mentioned previously, we use energies and powers as error metrics to assess the quality of the co-simulation results: on one hand, we consider the average error in the power (4) transmitted over the power bond from simulator S_1 to simulator S_2,

$$\Delta P(t_{i+1}) \equiv \frac{1}{T}\sum_{j=0}^{i} |P_{12}(t_{j+1}) - P^0_{12}(t_{j+1})|\Delta t_j, \tag{15a}$$

where $P^0_{12}(t)$ is the exact solution and T is the total simulated time duration. On the other hand, the total accumulated residual energy

$$\Delta E(t_{i+1}) \equiv \sum_{j=0}^{i} \delta P(t_{j+1})\Delta t_j \tag{15b}$$

gives the amount of energy wrongfully added to the full system during the entire simulation time interval $t \in [t_0, t_{i+1}]$ and is thus used as another indicator of co-simulation accuracy.

Table 4 Linear quarter car benchmark results for reticulation 1 with NEPCE and with NEPCE with direct feed-through modification

Algorithm			Power	Error	
Type	Tuning	Steps	$\frac{\overline{P_{12}}}{\mathrm{W}}$	$\frac{\Delta P}{\mathrm{W}}$	$\frac{\Delta E}{\mathrm{J}}$
Constant		4000	0.4	1.0	6.3
NEPCE	0.95	4000	0.01	0.14	3.20
NEPCE mod.	0.95	4000	0.01	0.11	3.20

Table 5 Linear quarter car benchmark results for reticulation 2 with NEPCE and with NEPCE with direct feed-through modification

Algorithm			Power	Error	
Type	Tuning	Steps	$\frac{\overline{P_{12}}}{10^2\mathrm{W}}$	$\frac{\Delta P}{10^2\mathrm{W}}$	$\frac{\Delta E}{10^2\mathrm{J}}$
Constant		4000	– 1.89	0.10	0.22
NEPCE	0.85	4000	– 1.88	0.04	0.11
NEPCE mod.	0.85	4000	– 1.88	0.03	0.10

5.1 NEPCE

Let us first use the quarter car model to benchmark NEPCE's performance alone. The tuning factor α is chosen such that the errors are minimized while avoiding the excitation of fast oscillations and risking instability. The energy errors can be reduced throughout by 49–86% when using NEPCE compared to the uncorrected cases. The results are summarized in Tables 4 and 5, where the tuning factor, the total number of macro-time steps, and the power transmitted over the power bond P_{12} averaged over the entire simulation duration T are shown. Furthermore, the error in the power $\Delta P(T)$ and the total accumulated residual energy $\Delta E(T)$ are given according to Eq. (15) with respect to the simulation duration T.

The quarter car benchmark model does exhibit direct feed-through (in simulator S_2 for system reticulation 1 and in S_1 in system reticulation 2). We thus expect improved performance when including the modification to NEPCE discussed in Sect. 3.2. Indeed, the average error in the power $\Delta P(T)$ is reduced by about another 17–33% with the modification. Figure 4 exemplifies this enhancement by showing the average error in the transmitted power for system reticulation 2. Note, however, that the direct feed-through modification to NEPCE does not significantly influence the overall accumulated residual energy $\Delta E(T)$.

[6] Reticulation 2 is a less favorable way of splitting the system, putting higher demands on the co-simulation algorithm and requiring smaller macro-step.

[7] Note that in the representation used in Ref. [11] and adapted here, the displacement z_c can either be communicated directly from S_1 to S_2 as a signal (in addition to the power bond coupling between both) or integrated internally by S_2 from its input $\dot{z}_\mathrm{c}$.

[8] Even smaller micro-step sizes affect the benchmark results only marginally.

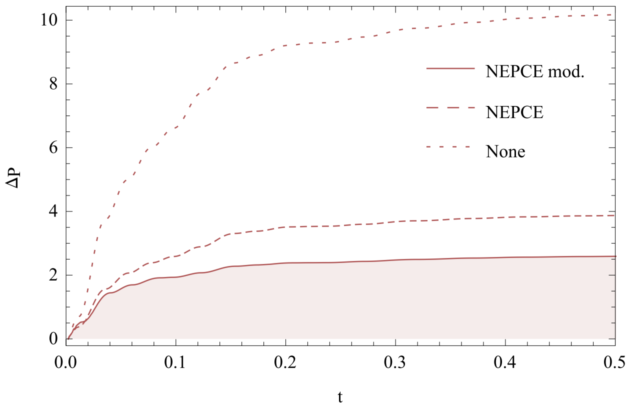


Fig. 4 Average error in the power for the linear quarter car benchmark with reticulation 2 and constant step size: NEPCE with direct feed-through modification (solid), NEPCE alone (dashed), and the uncorrected result (dotted)

Table 6 Linear quarter car benchmark results for reticulation 1 using residual energy-based adaptive step size control with NEPCE and with NEPCE with direct feed-through modification

Algorithm				Power	Error	
Type	Tuning	Tolerance	Steps	$\frac{P_{12}}{\mathrm{W}}$	$\frac{\Delta P}{\mathrm{W}}$	$\frac{\Delta E}{\mathrm{J}}$
Constant			4000	0.4	1.0	6.3
NEPCE	0.95	1.6×10^{-6}	3930	− 0.05	0.08	0.83
NEPCE mod.	0.95	1.6×10^{-6}	4002	− 0.04	0.06	0.81

Table 7 Linear quarter car benchmark results for reticulation 2 using residual energy-based adaptive step size control with NEPCE and with NEPCE with direct feed-through modification

Algorithm				Power	Error	
Type	Tuning	Tolerance	Steps	$\frac{P_{12}}{10^2\mathrm{W}}$	$\frac{\Delta P}{10^2\mathrm{W}}$	$\frac{\Delta E}{10^2\mathrm{J}}$
Constant			4000	− 1.89	0.10	0.22
NEPCE	0.85	1.4×10^{-6}	3921	− 1.872	0.003	0.004
NEPCE mod.	0.85	1.4×10^{-6}	3958	− 1.871	0.002	0.004

5.2 NEPCE combined with ECCO

Let us now demonstrate how the corrections to the inputs are made more efficient by combining them with ECCO's energy-conservation-based adaptive step size control, as proposed in Sect. 4. For this purpose, the I-controller (13) and the scalar error indicator (14) are configured according to the parameters listed in Table 1, and the starting step size is set to $\Delta t_0 = \Delta t_{\min}$. The quarter car system is initially excited with an energy of 750 J which thus suggests a characteristic energy scale for the system, $E_0 = 750\,\mathrm{J}$. The tolerance r is set such that the total number of macro-time steps remains around a constant 4000 steps to keep the computational cost at the same level.

Substantial improvements are observed when using NEPCE with ECCO's adaptive step size control: The energy errors in the benchmarks are reduced by 87–92% for system reticulation 1, see Table 6, and by 97–98% for system reticulation 2, see Table 7. This considerable enhancement of the quality of the co-simulation results is also exemplified in Fig. 5.

The situation is further improved by also including the direct feed-through modifications for NEPCE, as shown in Fig. 6. Then, an additional reduction of the average error in the power of 26–36% is achieved compared to the results without the modification. Again, however, the accumulated residual energy is almost unaffected.

In conclusion, the non-iterative energy-conservation-based co-simulation framework presented here (NEPCE with direct feed-through modification combined with ECCO's adaptive step size control) manages to reduce the energy errors by between 87% and 98% in the linear quarter car benchmark at no additional computational cost.

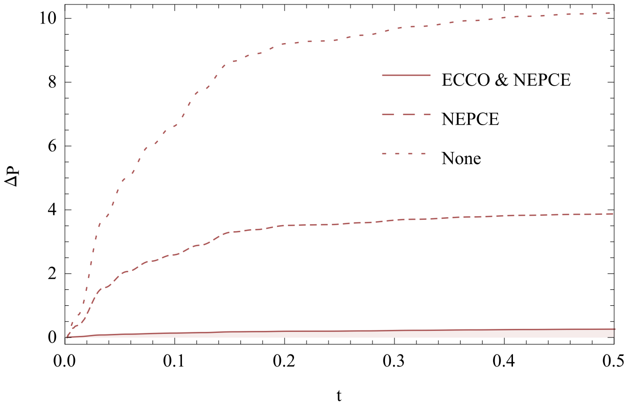


Fig. 5 Average error in the power for the linear quarter car benchmark with reticulation 2: ECCO with NEPCE (solid) against the constant step size results with NEPCE (dashed) and without any corrections (dotted)

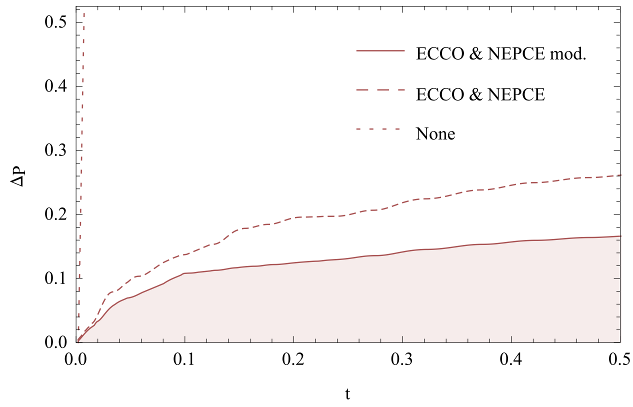


Fig. 6 Average error in the power for the linear quarter car benchmark with reticulation 2: ECCO with NEPCE with direct feed-through modification (solid), ECCO with NEPCE alone (dashed), and uncorrected result with constant step sizes (dotted)

5.3 Nonlinear damping

Finally, let us study the effects of nonlinear damping on the co-simulation errors. To that end, the benchmark model is altered according to Table 3. Note that the total simulation duration is now set to $T = 2$s (a total of 2000 macro-time steps) because the excitations in the system are subdued faster with the more efficient nonlinear damper. In addition, system reticulation 2 is relatively unstable for nonlinear damping, and the macro-step size is thus restricted by $t_{\max} = 2.5$ms for this setup.

The energy-conservation-based corrections to the inputs (as expressed by the tuning factor α) have to be applied less aggressively to avoid rapid oscillations. Yet, using NEPCE alone without modifications yields a reduction in the energy errors of 32–60% when compared to the uncorrected results, as shown in Tables 8 and 9. As was the case for the linear benchmark, significant improvements are obtained by combining NEPCE and ECCO: The energy errors are reduced by 79–91% compared to the uncorrected results obtained with a constant step size. Also including the direct feed-through modifications with NEPCE leads to small additional reductions of 0–19%.

6 Conclusion

The *Energy-Conservation-based Co-Simulation* method [11] (ECCO) provides a generic framework for error estimation and adaptive step size control in non-iterative co-simulations. Using power bonds to realize the simulator coupling, it directly monitors power flows between the subsystems and gives the exact amount of energy wrongfully added to the total energy of the full coupled system during co-simulation (macro) time steps. The resulting so-called *residual energies* are obtain from the coupling variable values alone, and ECCO uses them to propose an optimal macro-time step to minimize energy errors throughout the co-simulation. The *Nearly Energy Preserving Coupling Element* [3] (NEPCE), on the other hand, corrects for coupling errors in non-iterative co-simulations directly to make the flow of energy between subsimulators more accurate.

In the present paper, we combine both methods to optimize the efficiency and accuracy of non-iterative

Table 8 Nonlinear quarter car benchmark results for reticulation 1

Algorithm				Power	Error	
Type	Tuning	Tolerance	Steps	$\frac{P_{12}}{\mathrm{W}}$	$\frac{\Delta P}{\mathrm{W}}$	$\frac{\Delta E}{\mathrm{J}}$
Constant			2000	0.6	1.3	4.7
NEPCE	0.6		2000	0.1	0.5	2.9
NEPCE mod.	0.6		2000	0.1	0.5	2.9
NEPCE	0.6	4.7×10^{-6}	1991	− 0.1	0.2	1.0
NEPCE mod.	0.6	4.6×10^{-6}	2010	− 0.1	0.2	1.0

Table 9 Nonlinear quarter car benchmark results for reticulation 2

Algorithm				Power	Error	
Type	Tuning	Tolerance	Steps	$\frac{P_{12}}{10^2\mathrm{W}}$	$\frac{\Delta P}{10^2\mathrm{W}}$	$\frac{\Delta E}{10^2\mathrm{J}}$
Constant			2000	− 3.8	0.2	0.4
NEPCE	0.4		2000	− 3.78	0.14	0.30
NEPCE mod.	0.4		2000	− 3.78	0.12	0.30
NEPCE	0.4	2.6×10^{-5}	1986	− 3.77	0.04	0.04
NEPCE mod.	0.4	2.7×10^{-5}	1989	− 3.77	0.03	0.04

co-simulations. NEPCE is based on the assumption that the coupling variables are slowly varying on the scale of the macro-time step. ECCO, on the other hand, provides a systematic approach to fulfill this requirement by adaptively controlling the macro-step size to minimize the violation of energy conservation. We also extend NEPCE to the case where direct feed-through is present. Then, the output errors give contributions to the residual energy which are of the same order as the ones stemming from the input errors. Put differently, additional contributions to the violation of energy conservation should be taken into account when constructing energy-conserving corrections to the coupling variables. This is, however, only possible if the interface Jacobian is known.

The potency of the concepts discussed here is demonstrated by use of a quarter car co-simulation benchmark model. We study two distinct system reticulations, as well as the effects of including nonlinear damping characteristics. In these benchmarks, NEPCE alone generally yields a reduction in the energy errors of 32–86%, depending on how aggressively it can be used before unwanted oscillations are induced. The proposed direct feed-through modification to NEPCE reduces the energy errors by another 0–36%. Also employing ECCO's adaptive step size control leads to substantially higher accuracies in the co-simulation results: energy errors are then reduced by up to 98% when compared to the uncorrected results with constant macro-step sizes.

Acknowledgements This work was funded by the Research Council of Norway (Project No. 225322 MAROFF) and the industrial partners in the ViProMa project consortium (VARD, Rolls-Royce Marine and DNV GL). We are grateful for their financial support. The authors would further like to thank Stian Skjong for fruitful discussions.

References

1. Arnold M, Clauß C, Schierz T (2013) Error analysis and error estimates for co-simulation in fmi for model exchange and co-simulation v2.0. Arch Mech Eng 60(1):75–94. https://doi.org/10.2478/meceng-2013-0005
2. Arnold M, Hante S, Köbis MA (2014) Error analysis for co-simulation with force–displacement coupling. Proc Appl Math Mech 14(1):43–44. https://doi.org/10.1002/pamm.201410014
3. Benedikt M, Watzenig D, Zehetner J, Hofer A (2013) Nepce—a nearly energy preserving coupling element for weak-coupled problems and co-simulation. In: V International conference on computational methods for coupled problems in science and engineering, coupled problems, pp 1021–1032. International Center for Numerical Methods in Engineering
4. Breedveld PC (1984) Physical systems theory in terms of bond graphs, Dissertation. Twente University, Enschede
5. Bußhardt J, Isermann R (1992) Realization of adaptive shock absorbers by estimating physical process coefficients of a vehicle suspension system. Am Control Conf 1992:531–535
6. Clauß C, Arnold M, Schierz T, Bastian J (2012) Master zur simulatorkopplung via fmi. In: Tagungsband der ASIM/GI-Fachgruppen STS und GMMS. Verlag, New York. pp 57–70.
7. Dahlquist G (1956) Convergence and stability in the numerical integration of ordinary differential equations. Math Scand 4:33–53
8. Gomes C, Thule C, Broman D, Larsen PG, Vangheluwe H (2017) Co-simulation: State of the art. arXiv: 1702.00686
9. Kübler R, Schiehlen W (2000) Two methods of simulator coupling. Math Comp Model Dyn 6(2):93–113. https://doi.org/10.1076/1387-3954(200006)6:2;1-M;FT093
10. Paynter HM (1961) Analysis and design of engineering systems: class notes for M.I.T. course 2.751. M.I.T. Press, Boston
11. Sadjina S, Kyllingstad LT, Skjong S, Pedersen E (2017) Energy conservation and power bonds in co-simulations: non-iterative adaptive step size control and error estimation. Eng Comput 33(3):607–620. https://doi.org/10.1007/s00366-016-0492-8
12. Schierz T, Arnold M, Clauß C (2012) Co-simulation with communication step size control in an fmi compatible master algorithm. In: 9th international modelica conference, Linköping electronic conference proceedings, pp 205–214. Linköping University Electronic Press. https://doi.org/10.3384/ecp12076205
13. Schweizer B, Li P, Lu D (2014) Explicit and implicit co-simulation methods: stability and convergence analysis for different solver coupling approaches. J Comput Nonlin Dyn 10(5):051,007. https://doi.org/10.1115/1.4028503
14. Skjong S, Pedersen E (2019) On the numerical stability in dynamical distributed simulations. Math Comput Simul 163:183–203. https://doi.org/10.1016/j.matcom.2019.02.018