Taylor & Francis
Taylor & Francis Group

Automatika

Journal for Control, Measurement, Electronics, Computing and Communications

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/taut20

Co-simulation perspective on evaluating the simulation with the engine test bench in the loop

Slaven Glumac, Nikola Varga, Fran Raos & Zdenko Kovačić

To cite this article: Slaven Glumac, Nikola Varga, Fran Raos & Zdenko Kovačić (2022) Co-simulation perspective on evaluating the simulation with the engine test bench in the loop, Automatika, 63:2, 275-287, DOI: 10.1080/00051144.2022.2031537

To link to this article: https://doi.org/10.1080/00051144.2022.2031537

Published online: 28 Jan 2022.

AUTOMATIKA
2022, VOL. 63, NO. 2, 275–287
https://doi.org/10.1080/00051144.2022.2031537

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Co-simulation perspective on evaluating the simulation with the engine test bench in the loop

Slaven Glumac[a], Nikola Varga[b], Fran Raos[c] and Zdenko Kovačić[a]

[a]Faculty of Electrical Engineering and Computing, LARICS, University of Zagreb, Zagreb, Croatia; [b]AVL List GmbH, Graz, Austria; [c]AVL-AST d.o.o., Zagreb, Croatia

ABSTRACT

This article shows how a hardware in the loop (HIL) simulation can be formulated in such a way that the quality of the simulation can be assessed by evaluating the total energy residual (TER). The validity of this approach was checked with the standard co-simulation to show that TER can be used to compare non-iterative Jacobi and Gauss-Seidel co-simulation masters. The given example with the known global error shows that the value of TER can correctly determine which of the co-simulation masters is better. In the experimental part of the article, the example of a HIL simulation with an engine test bench in the loop is presented. The experiments show that such a quality assessment approach can be used to determine the speed controller parameters for the engine test bench.

ARTICLE HISTORY
Received 11 February 2021
Accepted 13 January 2022

KEYWORDS
HIL simulation; engine test bench; co-simulation; power residual; controller tuning

1. Introduction

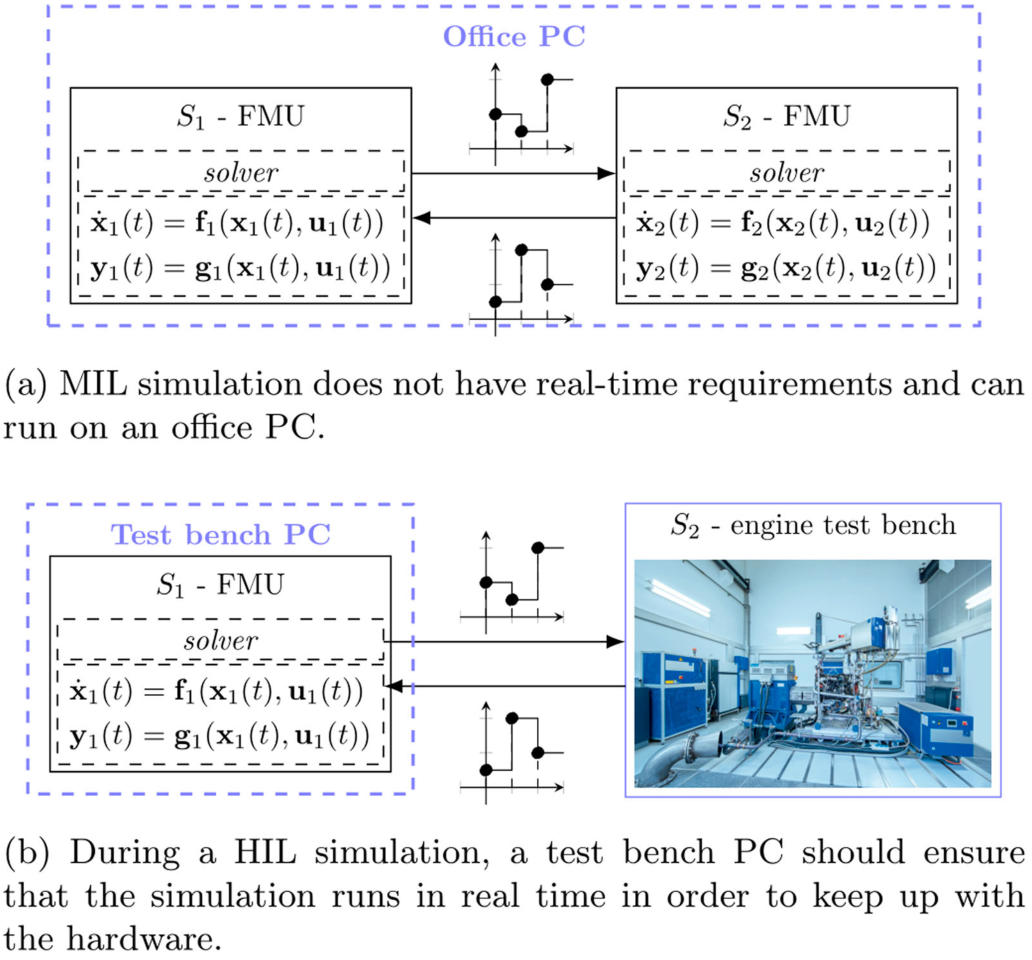
Hardware in the loop (HIL) simulations are used to test and simplify the design of complex systems. Its utility in the automotive area is covered in [1,2]. There are reports of using engine test benches for virtual testing [3,4,5,6]. In such a test, simulation models are connected to the engine test bench in order to form a HIL simulation. However, there is not a lot of work which focuses on the quality of such a co-simulation. The quality of the engine in the loop simulation can be assessed using the data collected from the vehicle [7]. In that case, the underlying model represents an already existing vehicle and the data for its identification is collected through an experiment. However, such data is not available in the vehicle design phase and can be expensive to obtain afterwards. This article presents a quality criterion with which the quality of a HIL simulation can be assessed without reference data. The proposed quality criterion can be used to set the test bench speed controller. Such a criterion can be used to automatically configure the co-simulation similar to [8].

This topic has been researched for power HIL simulations. In [9,10,11,12] methods for evaluating hardware interfaces have been assessed. A dynamometer with its control system presents a hardware interface between the engine and a simulation on the engine test bench. However, these methods evaluate the hardware interface using test simulations or their frequency responses. In a general nonlinear simulation, it is difficult to give guarantees based on such an analysis.

In this article, a hardware in the loop (HIL) simulation is a network of co-simulation slaves. The hardware co-simulation slave is an engine with the dynamometer, and other slaves are co-simulation Functional Mock-up Units (FMUs) according to Functional Mock-up Interface (FMI) standard [13]. The use of the FMI standard enables the HIL simulation to run models that have been exported from a variety of modelling tools [14]. In this article, a co-simulation consisting of co-simulation FMUs is referred to as a model in the loop (MIL) simulation (Figure 1a). It is assumed that such a co-simulation does not have real-time requirements and could run on a simple office PC. A HIL simulation (Figure 1b) introduces hardware into a co-simulation loop and requires all slaves to run in real time. MIL and HIL simulations are part of model based development processes [15,16,17,18].

This article shows how an existing quality criterion for the co-simulation evaluation [19] can be applied to a HIL simulation. This criterion is based on the calculation of power residual. It is applicable to complex systems that connect two subsystems with a power bond (Figure 2). Each bond in a bond graph consists of two power variables that represent flow and effort. Effort and flow are generalizations of voltage and current, force and translational speed, or torque and rotational speed, depending on the physical domain of a bond graph [20]. The method used in this article can be used on any system that can be modelled as a bond graph. This makes the method generally applicable to a variety of systems, including actuated systems such as

CONTACT Slaven Glumac slaven.glumac@fer.hr Faculty of Electrical Engineering and Computing, LARICS, University of Zagreb, Zagreb, Croatia


Office PC
S_1 - FMU
solver
$\dot{\mathbf{x}}_1(t) = \mathbf{f}_1(\mathbf{x}_1(t), \mathbf{u}_1(t))$
$\mathbf{y}_1(t) = \mathbf{g}_1(\mathbf{x}_1(t), \mathbf{u}_1(t))$
S_2 - FMU
solver
$\dot{\mathbf{x}}_2(t) = \mathbf{f}_2(\mathbf{x}_2(t), \mathbf{u}_2(t))$
$\mathbf{y}_2(t) = \mathbf{g}_2(\mathbf{x}_2(t), \mathbf{u}_2(t))$
Test bench PC
S_1 - FMU
solver
$\dot{\mathbf{x}}_1(t) = \mathbf{f}_1(\mathbf{x}_1(t), \mathbf{u}_1(t))$
$\mathbf{y}_1(t) = \mathbf{g}_1(\mathbf{x}_1(t), \mathbf{u}_1(t))$
S_2 - engine test bench

(a) MIL simulation does not have real-time requirements and can run on an office PC.

(b) During a HIL simulation, a test bench PC should ensure that the simulation runs in real time in order to keep up with the hardware.

Figure 1. This article suggests a technique that can be used to assess the co-simulation quality regardless of whether the hardware is in the loop or not.

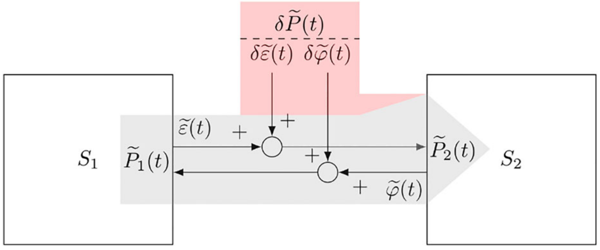


Figure 2. A power bond can be used to measure the consistency of a connection between two co-simulation slaves. The influence of the automation system, its actuators and sensors on power bonds can be modelled with the help of power residuals.

engine test beds. The work in [19] shows that it can be applied to MIL simulations. This article shows that the method can be used to determine which non-iterative co-simulation master is better.

An alternative that can be used to evaluate MIL and HIL simulation can be found in [21]. The advantage of the energy-based evaluation of the co-simulation quality is that errors can easily be aggregated across several power bonds. An energy control approach is used in [22] to create a coupling element that controls the error introduced in a single connection. In [23] the authors suggest that for the presented two-mass oscillator test problem, the energy error is a better indication of the co-simulation quality than the position error. Furthermore, the used criterion enables the monitoring of simulation errors that are aggregated over several power bonds.

There are additional methods for assessing the quality of the co-simulation. The introduction of an engine test bench in the co-simulation loop, however, represents a significant limitation on the number of options. A test bench cannot repeat a communication step. This prevents the use of quality assessment criteria based on the Richardson extrapolation [24,25] or any backwards differentiation techniques [26]. It also prevents the usage of methods that rely on internal solver functions of co-simulation slaves [27,28]. In addition, the introduction of hardware prevents the step size from changing during the simulation. In [19] the quality criterion is used to vary the step size of co-simulation slaves.

In the second section, the total energy residual (TER) is presented and linked to the defect analysis of the system of coupled ordinary differential and algebraic equations. The defect analysis provides a theoretical justification for the co-simulation quality measurement proposed in the article. In the next section, the

TER is used to compare standard co-simulation masters when co-simulating a rotational oscillator with two inertia. The example shows that such an evaluation can show the same results as a global error. Publicly available FMUs are used to give a practical confirmation of the proposed quality criterion. In the fourth section, the structure of the HIL simulation with an engine test bench is presented. The structure is visualized in order to see how to apply the quality measurement to the simulation with an engine test bench in the loop. This shows how the same quality measurement can be used to evaluate MIL and HIL simulations. The fifth section describes an experimental approach for tuning the speed controller for the engine test bench. This approach is used to show how the quality measurement can be applied to set up the simulation with an engine test bench in the loop. The last section contains conclusions and ideas for future work.

2. Co-simulation quality evaluation

This section is dedicated to introducing the power residual and its usage to define the co-simulation quality criterion. It also shows the relationship between this criterion and numerical defect analysis.

Suppose the model is defined as a system of coupled ordinary differential and algebraic equations.

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{x_i}}{\mathrm{d}t}(\boldsymbol{t}) &= \boldsymbol{f_i}(\boldsymbol{x_i}(\boldsymbol{t}), \boldsymbol{u_i}(\boldsymbol{t}))\\ \boldsymbol{y_i}(\boldsymbol{t}) &= \boldsymbol{g_i}(\boldsymbol{x_i}(\boldsymbol{t}), \boldsymbol{u_i}(\boldsymbol{t}))\\ \boldsymbol{u_{ij}}(\boldsymbol{t}) &= \boldsymbol{y}_{\imath\jmath}(\boldsymbol{t}), \quad (\imath, \boldsymbol{\jmath}) = \boldsymbol{L}(\boldsymbol{i}, \boldsymbol{j})\\ \boldsymbol{x_i}(0) &= \boldsymbol{x_{i0}} \end{aligned} \tag{1}$$

where

- $i, \imath$ are subsystem indices,
- $\boldsymbol{x}_i$(t), $\mathbf{u}_\mathrm{i}$(t) and $\mathbf{y}_\mathrm{i}$(t) are the state, input and output signals of the i^{th} subsystem, respectively,
- $\boldsymbol{f}_i$ is the state transition function of the i^{th} subsystem,
- $\boldsymbol{g}_i$ is the output function of the i^{th} subsystem,
- $\boldsymbol{x}_{i0}$ is the initial state,
- j is the index of an input port,
- $\jmath$ is the index of an output port,
- and L is the connection function.

The connection function L describes connections between subsystems. It takes the reference of a destination input port and returns the reference to the source output port. The reference to the destination input port is given by the subsystem index i and the input port index j. The reference to the source input port is given by the subsystem index $\imath$ and the output port index $\jmath$. Suppose a numerical solution satisfies the following system.

$$\frac{\mathrm{d}\tilde{\boldsymbol{x}}_{\boldsymbol{i}}}{\mathrm{d}t}(\boldsymbol{t}) = \boldsymbol{f_i}(\tilde{\boldsymbol{x}}_{\boldsymbol{i}}(\boldsymbol{t}), \tilde{\boldsymbol{u}}_{\boldsymbol{i}}(\boldsymbol{t})) + \boldsymbol{\delta}\tilde{\boldsymbol{x}}_{\boldsymbol{i}}(\boldsymbol{t}) \tag{2a}$$

$$\tilde{\boldsymbol{y}}_{\boldsymbol{i}}(\boldsymbol{t}) = \boldsymbol{g_i}(\tilde{\boldsymbol{x}}_{\boldsymbol{i}}(\boldsymbol{t}), \tilde{\boldsymbol{u}}_{\boldsymbol{i}}(\boldsymbol{t})) + \boldsymbol{\delta}\tilde{\boldsymbol{y}}_{\boldsymbol{i}}(\boldsymbol{t}) \tag{2b}$$

$$\tilde{\boldsymbol{u}}_{\boldsymbol{ij}}(\boldsymbol{t}) = \tilde{\boldsymbol{y}}_{\imath\jmath}(\boldsymbol{t}) + \boldsymbol{\delta}\tilde{\boldsymbol{u}}_{\boldsymbol{ij}}(\boldsymbol{t}), \quad (\imath, \boldsymbol{\jmath}) = \boldsymbol{L}(\boldsymbol{i}, \boldsymbol{j}) \tag{2c}$$

$$\tilde{\boldsymbol{x}}_{\boldsymbol{i}}(0) = \boldsymbol{x_{i0}} \tag{2d}$$

for each interval $t_{k-1} < t \leq t_k$ where.

- $\tilde{x}_i(t)$ is the state signal of the i^{th} subsystem found by the numerical solution,
- $\tilde{\boldsymbol{u}}_i(t)$ is the input signal of the i^{th} subsystem found by the numerical solution,
- $\tilde{\boldsymbol{y}}_i(t)$ is the output signal of the i^{th} subsystem found by the numerical solution,
- $\delta\tilde{\boldsymbol{x}}_i(t)$ is the integration defect of the i^{th} subsystem,
- $\delta\tilde{\boldsymbol{y}}_i(t)$ is the output defect of the i^{th} subsystem,
- and $\delta\tilde{u}_{ij}(t)$ is the connection defect on the j^{th} port of the i^{th} subsystem.

The numerical defects introduced in (2) give an indication of how closely the numerical solution corresponds to the underlying model it is trying to solve (1). In [29,30] it is shown that global errors.

$$\begin{aligned}&\boldsymbol{\Delta}\tilde{\boldsymbol{x}}_{\boldsymbol{i}}(\boldsymbol{t}) = \tilde{\boldsymbol{x}}_{\boldsymbol{i}}(\mathbf{t}) - \boldsymbol{x_i}(\mathbf{t}), \boldsymbol{\Delta}\tilde{\boldsymbol{y}}_{\boldsymbol{i}}(\mathbf{t}) = \tilde{\boldsymbol{y}}_{\boldsymbol{i}}(\mathbf{t}) - \boldsymbol{y_i}(\boldsymbol{t}),\\ &\boldsymbol{\Delta}\tilde{\boldsymbol{u}}_{\boldsymbol{i}}(\mathbf{t}) = \tilde{\boldsymbol{u}}_{\boldsymbol{i}}(\mathbf{t}) - \boldsymbol{u_i}(\mathbf{t})\end{aligned} \tag{3}$$

are bounded if numerical defects are bounded.

Figure 2 shows a power bond and four related variables. The effort of a power bond in the modelled system is equal to.

$$\boldsymbol{\varepsilon}(\boldsymbol{t}) = \boldsymbol{u}_{\boldsymbol{i}_1\boldsymbol{j}_1}(\boldsymbol{t}) = \boldsymbol{y}_{\boldsymbol{i}_2 j_1}(\boldsymbol{t}), \quad (\boldsymbol{i}_2, \boldsymbol{j}_1) = \boldsymbol{L}(\boldsymbol{i}_1, \boldsymbol{j}_1) \tag{4}$$

and the flow is.

$$\boldsymbol{\varphi}(\boldsymbol{t}) = \boldsymbol{u}_{\boldsymbol{i}_2\boldsymbol{j}_2}(\boldsymbol{t}) = \boldsymbol{y}_{\boldsymbol{i}_1 j_2}(\boldsymbol{t}), \quad (\boldsymbol{i}_1, \boldsymbol{j}_2) = \boldsymbol{L}(\boldsymbol{i}_2, \boldsymbol{j}_2) \tag{5}$$

The effort found by the numerical solution (2) is equal to.

$$\tilde{\boldsymbol{\varepsilon}}(\boldsymbol{t}) = \tilde{\boldsymbol{u}}_{\boldsymbol{i}_1\boldsymbol{j}_1}(\boldsymbol{t}) = \tilde{\boldsymbol{y}}_{\boldsymbol{i}_2 j_1}(\boldsymbol{t}) + \boldsymbol{\delta}\tilde{\boldsymbol{\varepsilon}}(\boldsymbol{t}), \quad (\boldsymbol{i}_2, \boldsymbol{j}_1) = \boldsymbol{L}(\boldsymbol{i}_1, \boldsymbol{j}_1) \tag{6}$$

where the effort defect $\delta\tilde{\varepsilon}$(t) is the connection defect introduced by solving the respective connection equation (2c). The flow found by the numerical solution (2) is equal to.

$$\tilde{\boldsymbol{\varphi}}(\boldsymbol{t}) = \tilde{\boldsymbol{u}}_{\boldsymbol{i}_2\boldsymbol{j}_2}(\boldsymbol{t}) = \tilde{\boldsymbol{y}}_{\boldsymbol{i}_1 j_2}(\boldsymbol{t}) + \boldsymbol{\delta}\tilde{\boldsymbol{\varphi}}(\boldsymbol{t}), \quad (\boldsymbol{i}_1, \boldsymbol{j}_2) = \boldsymbol{L}(\boldsymbol{i}_2, \boldsymbol{j}_2) \tag{7}$$

where the effort defect $\delta\tilde{\varphi}$(t) is the connection defect introduced by solving the respective connection equation (2c)..

The power residual is the difference between the power transferred between two slaves.

$$\delta\tilde{P}(t) = \tilde{u}_{i_1 j_1}(t)\tilde{y}_{i_1 j_2}(t) - \tilde{u}_{i_2 j_2}(t)\tilde{y}_{i_2 j_1}(t)$$

$$
\begin{aligned}
&= \tilde{\varepsilon}(t)[\tilde{\varphi}(t) - \delta\tilde{\varphi}(t)] - [\tilde{\varepsilon}(t) - \delta\tilde{\varepsilon}(t)]\tilde{\varphi}(t) \\
&= \delta\tilde{\varepsilon}(t)\tilde{\varphi}(t) - \tilde{\varepsilon}(t)\delta\tilde{\varphi}(t) \qquad (8)
\end{aligned}
$$

Expression (8) shows how the power residual is related to connection defects of the effort (6) and flow (7) signals. It shows that by minimizing the value of the two connection defects the power residual should also be minimized.

$$|\boldsymbol{\delta\tilde{P}(t)}| \leq |\boldsymbol{\delta\tilde{\varepsilon}(t)}||\boldsymbol{\tilde{\varphi}(t)}| + |\boldsymbol{\tilde{\varepsilon}(t)}||\boldsymbol{\delta\tilde{\varphi}(t)}| \qquad (9)$$

Minimizing the power residual may not be enough to reduce the global co-simulation error, but it is required. Since global error (3) is limited by numerical defects [29,30], for a small global error the connection defects should also be small. The flow and effort errors in (9) are connection errors. Inequality (9) suggests that the power residual is small when the corresponding flow and effort defects are small. These are the reasons why it is necessary to limit the power residual. This is not enough because the flow and power signals can be close to zero due to the simulated model and this could limit the power residual (9). In addition, poor internal FMU solvers can introduce integration errors that can lead to increased global co-simulation errors.

An advantage of monitoring the power residual instead of connection defects is that the power defect can be intuitively normalized. The power residual has a unit of measure for power and TER has a unit of measure for energy. This makes it easy to aggregate the measure across multiple power bonds. The next section shows the applicability of such an approach to the MIL simulation.

The previous analysis suggests that the total energy residual (TER) can be used to evaluate a co-simulation run.

$$\boldsymbol{\delta\tilde{E}(t_{end})} = \sum_{\substack{\boldsymbol{k\in\mathbb{N}} \\ \boldsymbol{t_k\leq t_{end}}}} \int_{\boldsymbol{t_{k-1}}}^{\boldsymbol{t_k}} |\boldsymbol{\delta\tilde{P}(t)}|\boldsymbol{dt} \qquad (10)$$

The symbol for TER includes the letter E to indicate that it has a physical unit of energy. The prefix δ is inserted to emphasize the connection to the power residual. The argument t_{end} means that the power residual is integrated until the end of the co-simulation. This criterion can be evaluated after the co-simulation run has ended, provided the corresponding input and output signals are available. The criterion can also be calculated during the run, which can possibly lead to an error detection mechanism.

The variable $k \in \mathbb{N}$ is used to enumerate communication points of both connected slaves. In this article, zero order hold is used to reconstruct input and output signals and fixed step size co-simulation is used.

$$\delta\tilde{E}(t_{end}) = h \sum_{\substack{k\in\mathbb{N} \\ kh\leq t_{end}}} |\delta\tilde{P}(t_k)| \qquad (11)$$

where h is the communication step size. It should be possible to calculate the exact value of the criterion (10) for multi-rate or variable step co-simulations using higher order extrapolation techniques. This article uses single rate co-simulation with zero order hold to ease the application of the criterion.

3. Co-simulation master comparison

This section shows how TER (11) can be used to evaluate co-simulation masters. A test model for validating such an evaluation is the co-simulation network shown in Figure 3. The Python code used for the examples in this section is available in [31]. The source code of the example co-simulation slaves is available in [32]. The slaves are co-simulation FMUs according to [13]. The modelled system is a rotational oscillator with two inertias. A similar oscillator is often used for benchmarking co-simulation master algorithms [21,33,34,35,36,37].

The example in this section uses a rotational mechanical interface that is similar to the engine test bench interface presented in the next section. There are two subsystems.

$$
\begin{aligned}
\frac{\mathrm{d}\boldsymbol{x}_1}{\mathrm{d}t}(t) &= \mathbf{A}_1\mathbf{x}_1(t) + \boldsymbol{B}_1\mathbf{u}_1(t), \\
\frac{\mathrm{d}\boldsymbol{x}_2}{\mathrm{d}t}(t) &= \mathbf{A}_2\mathbf{x}_2(t) + \mathbf{B}_2\mathbf{u}_2(t), \\
\mathbf{y}_1(\mathrm{t}) &= \mathbf{C}_1\mathbf{x}_1(t) + \mathbf{D}_1\mathbf{u}_1(t), \\
\mathbf{y}_2(t) &= \mathbf{C}_2\mathbf{x}_2(t) + \mathbf{D}_2\mathbf{u}_2(t), \\
\mathbf{x}_1(0) &= \mathbf{x}_{10}, \mathbf{x}_2(0) = \boldsymbol{x}_{20} \qquad (12)
\end{aligned}
$$

connected by.

$$u_1(t) = y_2(t), \quad u_2(t) = y_1(t) \qquad (13)$$

where

$$
\begin{aligned}
\mathbf{y}_1(t) &= [\tau_1(t)], \quad \mathbf{u}_1(t) = [\omega_2(t)], \\
\mathbf{x}_1(t) &= \begin{bmatrix}\phi_1(t) & \omega_1(t) & \phi_2(t)\end{bmatrix}^T, \\
\mathbf{x}_{10} &= \begin{bmatrix}\phi_{10}(t) & \omega_{10}(t) & \phi_{20}(t)\end{bmatrix}^T, \\
\mathbf{A}_1 &= \begin{bmatrix} 0 & 1 & 0 \\ -\dfrac{c_1 + c_k}{J_1} & -\dfrac{d_1 + d_k}{J_1} & \dfrac{c_k}{J_1} \\ 0 & 0 & 0 \end{bmatrix}, \\
\mathbf{B}_1 &= \begin{bmatrix} 0 \\ \dfrac{d_k}{J_1} \\ 1 \end{bmatrix}, \\
\mathbf{C}_1 &= \begin{bmatrix} c_k & d_k & -c_k \end{bmatrix}, \quad \mathbf{D}_1 = [-d_k] \qquad (14)
\end{aligned}
$$

and.

$$\boldsymbol{y}_2(t) = [\omega_2(t)], \quad \mathbf{u}_2(t) = [\tau_1(t)],$$

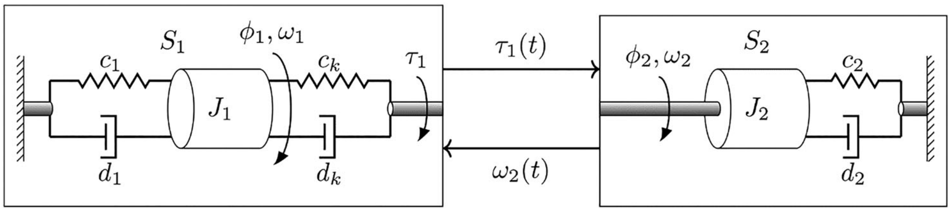
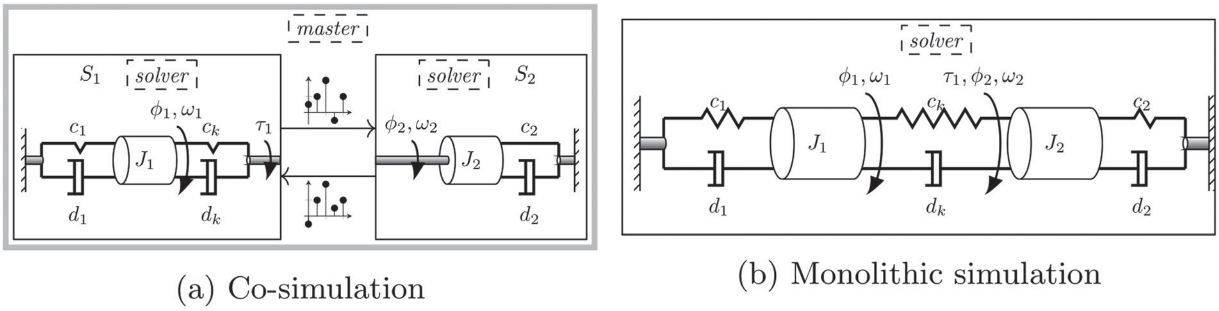


Figure 3. A rotational oscillator with two inertias is used as a test system.



(a) Co-simulation

(b) Monolithic simulation

Figure 4. During the co-simulation, each subsystem is solved with an internal solver and the co-simulation master ensures that samples of the connected signals are exchanged between subsystems. In the monolithic simulation, a solver solves the entire system of equations.

$$\mathbf{x}_2(t) = \begin{bmatrix}\phi_2(t) & \omega_2(t)\end{bmatrix}^T, \quad \mathbf{x}_{20} = \begin{bmatrix}\phi_{20}(t) & \omega_{20}(t)\end{bmatrix}^T,$$

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ -\frac{c_2}{J_2} & -\frac{d_2}{J_2} \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 \\ \frac{1}{J_2} \end{bmatrix},$$

$$\mathbf{C}_2 = \begin{bmatrix} 0 & 1 \end{bmatrix}, \quad \mathbf{D}_2 = [0] \tag{15}$$

The system parameters are set to

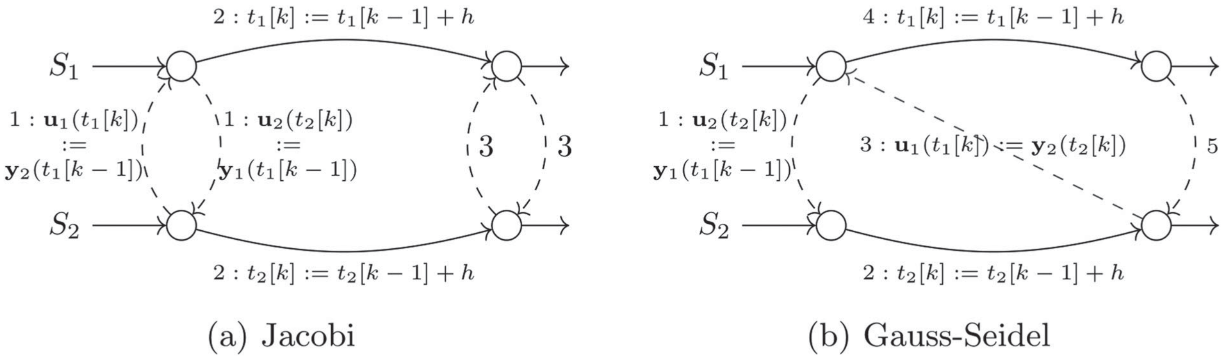
$$\begin{aligned} \mathrm{J}_1 &= 10\,\mathrm{kg\,m}^2, \quad \mathrm{c}_1 = 1\,\frac{\mathrm{N\,m}}{\mathrm{rad}}, \\ \mathrm{d}_1 &= 1\,\frac{\mathrm{N\,m\,s}}{\mathrm{rad}}, \quad \mathrm{c_k} = 1\,\frac{\mathrm{N\,m}}{\mathrm{rad}}, \\ \mathrm{d_k} &= 2\,\frac{\mathrm{N\,m\,s}}{\mathrm{rad}}, \quad \phi_{10} = 0.1\,\mathrm{rad}, \\ \omega_{10} &= 0.1\,\frac{\mathrm{rad}}{\mathrm{s}}, \quad \mathrm{J}_2 = 10\,\mathrm{kg\,m}^2, \\ \mathrm{c}_2 &= 1\,\frac{\mathrm{N\,m}}{\mathrm{rad}}, \quad \mathrm{d}_2 = 2\,\frac{\mathrm{N\,m\,s}}{\mathrm{rad}}, \\ \phi_{20} &= 0.2\,\mathrm{rad}, \quad \omega_{20} = 0.1\,\frac{\mathrm{rad}}{\mathrm{s}} \end{aligned} \tag{16}$$

The system parameters are set to the same values as the parameters of the two mass oscillator in [37]. Such a system can be described in terms of bond graphs [20], and this property connects it to a large group of physical models. The co-simulation slaves use CVODE [38] solvers with a tight tolerance bound in order to provide a solution as close as possible to the exact one. The absolute tolerance limit for all solvers used is set to 10^{-8}. This choice limits the local error so that it is several orders of magnitude smaller than the transient response of the system (16). The two subsystems shown in Figure 3 are implemented as co-simulation FMUs, *OscillatorOmega2Tau* and *OscillatorTau2Omega* (Figure 4a). A monolithic simulation of the two connected subsystems is available as a co-simulation FMU *TwoMassRotationalOscillator* (Figure 4b). The monolithic simulation is created to calculate the global co-simulation error. Since the selected solver has tight tolerance bounds, it is used as a reference solution for the co-simulation. Although the co-simulation slaves in the example co-simulation use the same solver, the co-simulation master introduces an error when exchanging the connected signal samples (Figure 4). The source of the example FMUs can be found in [32].

The test model in this section does not contain a displacement in the interface between two subsystems. A displacement is the integral of a flow. Flow and displacement are translation speed and position in the translational mechanical domain or rotational speed and angle in the rotational mechanical domain. This is the difference to the models used in [36] and [37]. The reason for this change is that the subsystem interface is similar to that used for the engine test bench presented in the next section.

Gauss-Seidel and Jacobi masters are the standard options for co-simulation masters that run at a single rate [36,39]. Their execution policies are shown in Figure 5 using the example of the co-simulation of two slaves. A Gauss-Seidel co-simulation master is a sequential co-simulation master. The assignment of the input variables and the integration of the slaves take place in the specified sequence.

In [36] it is suggested that the execution of FMUs with direct output-input dependency should follow the execution of FMUs without such a dependency. For this reason, the execution order of the Gauss-Seidel master


$2 : t_1[k] := t_1[k-1] + h$
S_1
S_2
$1 : \mathbf{u}_1(t_1[k]) := \mathbf{y}_2(t_1[k-1])$
$1 : \mathbf{u}_2(t_2[k]) := \mathbf{y}_1(t_1[k-1])$
3
3
$2 : t_2[k] := t_2[k-1] + h$
$4 : t_1[k] := t_1[k-1] + h$
S_1
S_2
$1 : \mathbf{u}_2(t_2[k]) := \mathbf{y}_1(t_1[k-1])$
$3 : \mathbf{u}_1(t_1[k]) := \mathbf{y}_2(t_2[k])$
5
$2 : t_2[k] := t_2[k-1] + h$

(a) Jacobi

(b) Gauss-Seidel

Figure 5. The actions that are performed in non-iterative co-simulation masters for the example with two slaves. The dashed lines represent the data exchange between slaves and the solid lines the integration of slaves.

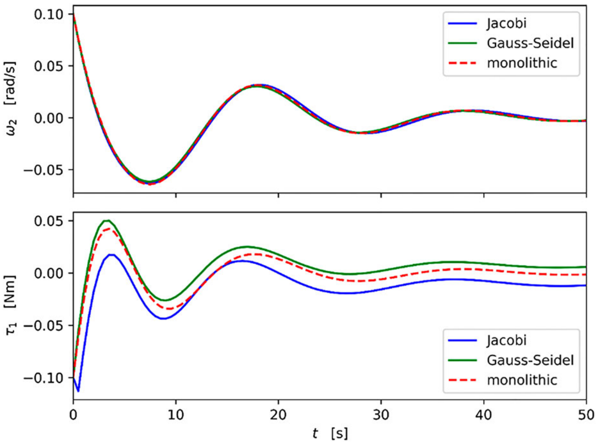


Figure 6. The simulation response for the step size $h = 1$ s. The plot shows the signals obtained from the subsystem outputs shown in Figure 3.

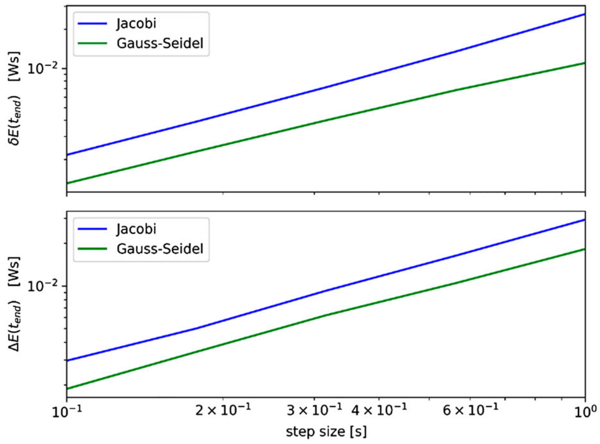


Figure 7. The TER (10) compared to the total error energy (20) for standard co-simulation masters. In both plots, both axes are logarithmic. The co-simulation masters are described in Figure 5.

is set to.

$$\begin{aligned} &1 : \mathit{OscillatorTau2Omega} \\ &2 : \mathit{OscillatorOmega2Tau} \end{aligned} \tag{17}$$

in the example.

A Jacobi co-simulation master updates the input variables of all slaves independently of one another and then integrates them. This allows slaves to be updated in parallel if several processors are available. Figure 6 compares the response of two different non-iterative co-simulation masters with the monolithic solution. The plot was obtained using the procedure *plot_signals* found in the file *demo.py* in [31].

In [36] a two-mass oscillator test system is analyzed to find out which standard co-simulation master is better. This section shows that the TER can be used for the same purpose. The power signal.

$$p(t) = \varphi(t)\,\varepsilon(t) = \omega_2(t)\,\tau_1(t) \tag{18}$$

is the basis for comparing co-simulation masters. In the example presented in this section, the above power signal is calculated using the monolithic simulation. Its source is available in the repository [32]. The error power is equal to

$$\Delta\tilde{p}(t) = |\tilde{p}(t) - p(t)| \tag{19}$$

The total error energy.

$$\Delta\tilde{E}(t_{end}) = \sum_{\substack{k\in\mathbb{N} \\ t_k \le t_{end}}} \int_{t_{k-1}}^{t_k} |\Delta\tilde{p}(t)| \, dt \approx h \sum_{\substack{k\in\mathbb{N} \\ kh \le t_{end}}} |\Delta\tilde{p}(t_k)| \tag{20}$$

is compared to the TER (11) in Figure 7. The figure compares the Gauss-Seidel co-simulation master and the Jacobi co-simulation master. The plot was obtained using the procedure *residual_analysis* found in the file *demo.py* in [31].

The plot shows that the total error energy is linear with respect to the step size. This is expected because zero order hold is used to extrapolate output and input signals. In [29] it was shown that the global error is bounded in terms of integration and algebraic defects. If zero order hold is used for extrapolation, the output defect (2b) is asymptotically linear with respect to the

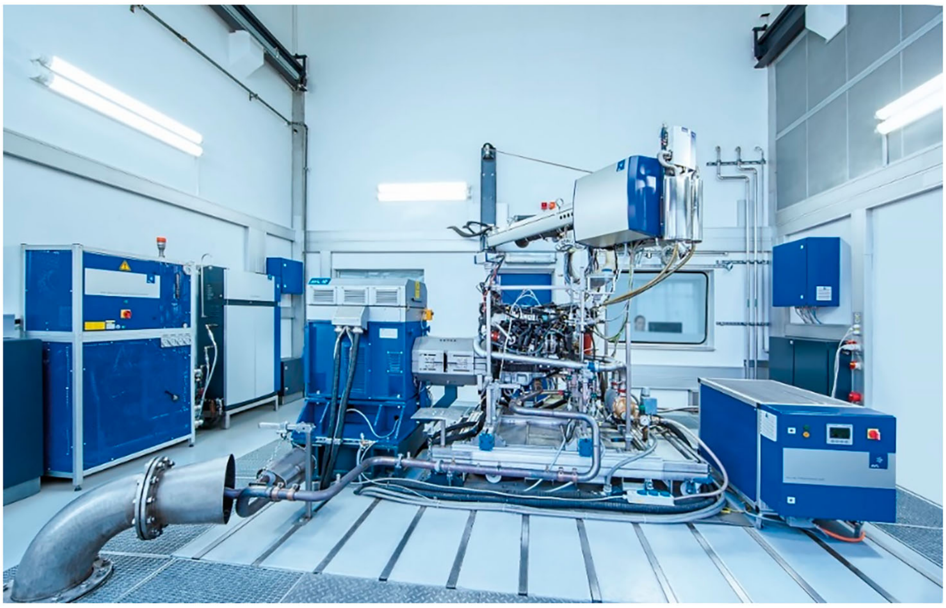


Figure 8. The engine test bench can be used to simulate how the rest of the vehicle interacts with a real engine.

step size. A connection with a delay in the transfer of the signals (Figure 5) has the connection defect (2c) asymptotically linear with respect to the step size. In addition, it was shown in [19] that the total energy residual (TER) can be used to control the total error energy by varying the communication step size.

However, the main purpose of Figure 7 is not to confirm the expected results. It shows an indication that the TER can be used to compare two non-iterative co-simulation masters (Figure 5). The total error energy (20) shows that the Gauss-Seidel co-simulation master (Figure 5b) is better than the Jacobi master (Figure 5a) for the example system (16) shown in Figure 3. The TER (11) shows the same result. This comparison together with the analysis carried out in [19] suggests that (11) can be used to evaluate a co-simulation master. Ideally, the error would be used for that purpose. In practice, however, an ideal reference solution is almost never available to enable global error calculation.

4. Hardware in the loop simulation setup

This section describes how the TER (11) can be used to evaluate the simulation with an engine test bench in the loop (Figure 8). The tests were carried out on an engine test bench with the automation system AVL Puma Open 2 R5.4 [40] and the 200 kW dynamometer AVL DynoForce ASM500 [41]. The engine used in the experiments is a 4-cylinder, 1395 cc, 103 kW (@4500 – 6000 rpm) gasoline engine. The speed was measured with the rotary encoder ROD426—2048 [42] and the torque with the torque converter HBM T12HP [43]. The sensor data was transferred to the automation system via EtherCAT [44]. This section argues that control system, actuators and control system influence on the HIL simulation can be evaluated using the TER.

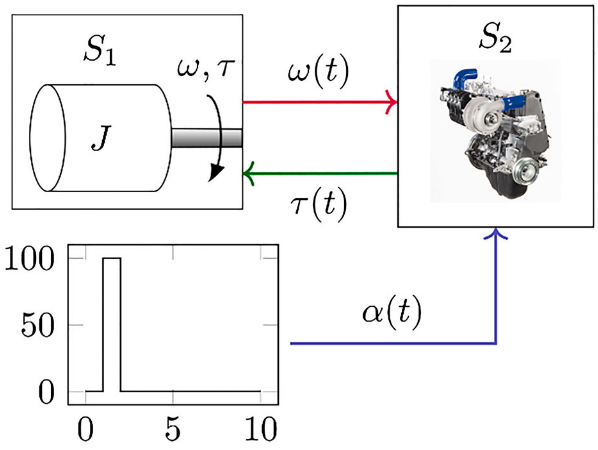


Figure 9. The underlying model of the example system that is simulated by the HIL simulation.

The model of the system simulated with the engine test bench is shown in Figure 9. The system consists of the gasoline engine, the first order load subsystem.

$$J\frac{d\omega}{dt}(t) = \tau(t) \tag{21}$$

and the following percentage gasoline flow signal

$$\alpha(t) = \begin{cases} 100\,\%, & 1 \le t < 2 \\ 0, & \mathit{otherwise} \end{cases} \tag{22}$$

The load model (21) is a simple inertia that was set to $J = 1\ \mathrm{kg\,m^2}$ during the experiments in this section.

Figure 9 shows the underlying model of the system that is simulated with HIL simulation. However, the behaviour of such a simulation is not completely

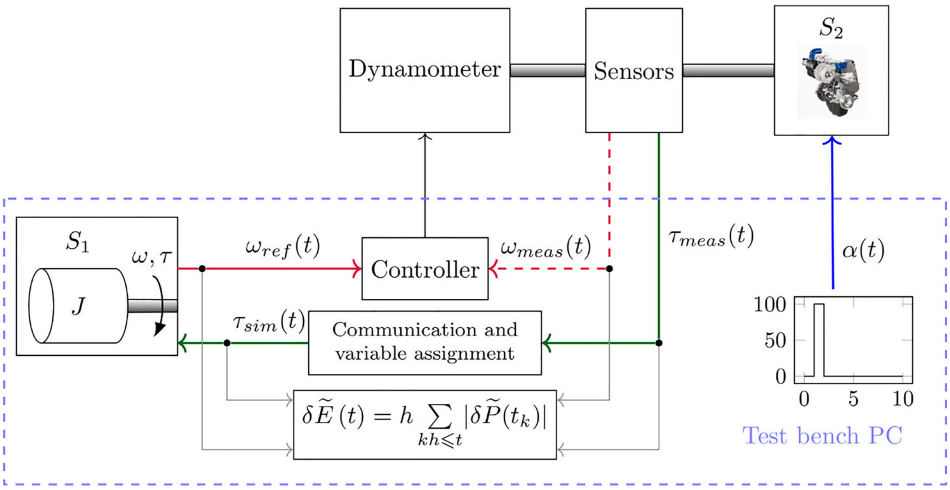


Figure 10. The TER measurement (11) is implemented in Simulink and deployed as a co-simulation FMU [13]. Coloured thick lines correspond to the thick lines with the matching colour in Figure 9. A red dashed line is added to highlight the difference in information flow in the HIL simulation compared to the underlying model.

determined by that model. The results of the HIL simulation are influenced by the communication network, the assignment of variables, controllers and actuators (Figure 10). This article argues that the diagram in Figure 10 corresponds to the numerical solution (2) and the model in Figure 9 corresponds to the analytical solution (1). This analogy suggests that the TER (10) can be used to compare different settings for a HIL simulation.

There is another analogy between the example HIL and the numerical simulation. The error in the controlled engine speed corresponds to the numerical defect of the flow signal (7). During the HIL simulation (Figure 10), the controllers and dynamometer are used to control engine speed. As a result, both the speed control error and the flow defect is equal to the difference between the reference and measured engine speed.

$$\delta\tilde{\varphi}(\mathrm{t}) = \delta\tilde{\omega}(\mathrm{t}) = \omega_{\mathrm{ref}}(\mathrm{t}) - \omega_{\mathrm{meas}}(\mathrm{t}) \qquad (23)$$

Figure 10 is a more detailed schematic representation of Figure 1b. It shows which software and hardware components are used in the data flow of the HIL simulation and how the TER is measured. Figure 9 shows the underlying model that is simulated by the HIL simulation. TER can be viewed as a measure of the influence of software and hardware components on the ideal underlying model.

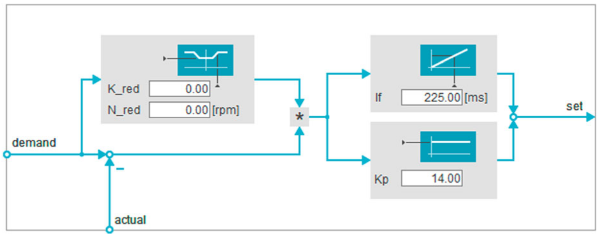
The effort defect in the HIL simulation (Figure 10) is equal to.

$$\delta\tilde{\varepsilon}(\mathrm{t}) = \delta\tilde{\tau}(\mathrm{t}) = \tau_{\mathrm{meas}}(\mathrm{t}) - \tau_{\mathrm{sim}}(\mathrm{t}) \qquad (24)$$

The power residual of the mechanical shaft in the HIL simulation is

$$\delta\tilde{\mathrm{P}}(\mathrm{t}) = \omega_{\mathrm{ref}}(\mathrm{t})\,\tau_{\mathrm{sim}}(\mathrm{t}) - \omega_{\mathrm{meas}}(\mathrm{t})\,\tau_{\mathrm{meas}}(\mathrm{t}) \qquad (25)$$

In the example HIL simulation, the zero order hold is used for signal reconstruction. This enables the TER to be calculated using (11). The calculation is implemented in a Simulink model and compiled as a co-simulation FMU. Grey lines in Figure 10 point to the FMU. This FMU runs with the fixed communication step size of 0.1 ms. The communication step size should be small enough to capture all the changes in the connected signals. The measurement signals are sampled with the step size of 0.1 ms and the inertia FMU is running with the communication step size of 1 ms. This setup is used in the next section to tune the parameters of the speed controller.



K_red 0.00
N_red 0.00 [rpm]
If 225.00 [ms]
Kp 14.00

Figure 11. The user interface of the PI speed controller in Puma Open. The experiment in this section searches for the optimal controller gain and integral time constant.

5. Controller tuning

This section presents how the TER (11) can be used to set the speed controller parameters (Figure 11). In Section 3, the TER is used to compare two non-iterative co-simulation masters, Jacobi and Gauss-Seidel (Figure 5). Section 4 shows how the TER is calculated for the HIL simulation using the engine test bench. The HIL simulation setup can be optimized by such a quality evaluation. The proportional gain $\mathrm{K_P}$ and the integral time constant $\mathrm{T_I}$ of the speed controller are varied in this section to adjust the HIL simulation.

The HIL simulation is described in Section 4 and shown in Figure 10. The simulation experiments are

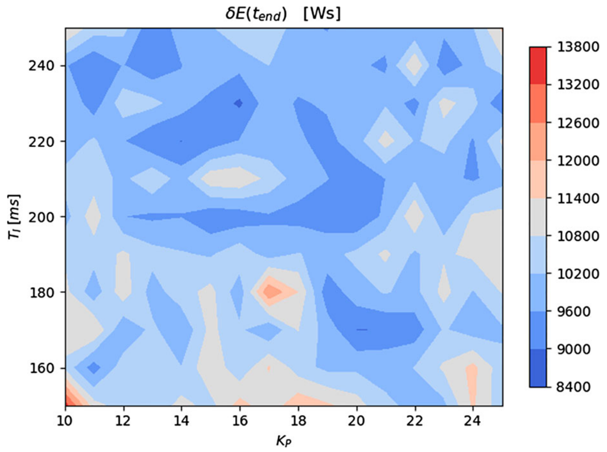


Figure 12. The HIL simulation shown in Figure 10 is carried out with various speed controller parameters. The image above shows the heat map with the values of the TER at the end of the simulation.

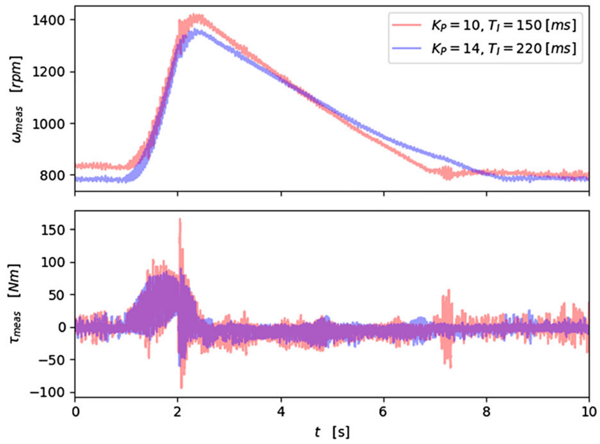


Figure 13. The speed and torque signals measured on the engine shaft when running a simple inertia model (Figure 10) with various controller parameters.

carried out for $t_{end} = 10$ s. The speed controller parameters are varied in the range that is considered safe from the practical commissioner's experience. The parameters of the speed controller vary in the range that has been verified in practice for the engine test bench used.

$$K_P \in \{10, 11, \ldots 25\},$$
$$T_I \in \{150\text{ ms}, 160\text{ ms}, \ldots, 250\text{ ms}\} \qquad (26)$$

A simulation experiment is performed for each combination of controller parameters and the TER (11) is recorded.

The results are presented in Figure 12 as a heat map. The heat map is not perfectly smooth since each experiment is performed only once and physical devices are used in the simulation loop. However, it shows areas with good controller parameters. The red area in the lower left corner shows parameters with higher TER values.

The results in Section 3 indicate that parameters in red areas should result in a larger global error.[1] This indicates that the combination of the parameters $K_P = 10$ and $T_I = 150$ ms should lead to a worse response than the combination $K_P = 14$ and $T_I = 220$ ms

$$K_P = 10,\ T_I = 150\text{ ms}: \quad \delta\tilde{E}(t_{end}) = 13.5\text{ kWs}$$
$$K_P = 14,\ T_I = 220\text{ ms}: \quad \delta\tilde{E}(t_{end}) = 9.0\text{ kWs} \qquad (27)$$

The output signals of the two experiments are shown in Figure 13. The global error of this hardware in the loop simulation is not known because the engine is a physical system. This is the main reason to suggest the TER for evaluating different experiments.

There is no guarantee that the controller parameters will be robust to changing the simple inertia model. The inertia model was replaced by a more complex vehicle model (Figure 14) in the HIL simulation (Figure 10). The vehicle model[2] was developed in AVL Cruise 2020 R1 [45]. The gear box within the vehicle model was set to first gear at the beginning of the simulation and to second gear at t = 2 s. The second gear remains constant until the end of the simulation. The throttle signal was replaced by.

$$\alpha(t) = \begin{cases} 15\,\%, & 1 \le t < 2 \\ 50\,\%, & 10 \le t < 13 \\ 60\,\%, & 23 \le t < 23 \\ 0, & \textit{otherwise} \end{cases} \qquad (28)$$

in order to observe additional dynamics in the experiment. The experiments were carried out for $t_{end} = 35$ s.

The experiments have been performed with the two combinations of controller parameters $K_P = 10, T_I = 150$ ms and $K_P = 14, T_I = 225$ ms

$$K_P = 10,\ T_I = 150\text{ ms}: \quad \delta\tilde{E}(t_{end}) = 44.4 \pm 1.1\text{ kWs}$$
$$K_P = 14,\ T_I = 225\text{ ms}: \quad \delta\tilde{E}(t_{end}) = 48.3 \pm 1.3\text{ kWs} \qquad (29)$$

each combination 5 times. Samples of speed and torque signals on the engine shaft are shown in Figure 15. The presented results suggest two topics for future work.

Inequality (9) suggests that by minimizing torque and speed signals the TER should be minimized. When the torque and speed are minimized, the mechanical power exchanged between the dynamometer and the engine should also be minimized. One topic of future work is the question of whether the minimization of TER leads to a minimization of the mechanical power on the shaft. Figure 15 shows a slight decrease in the measured speed for $K_P = 10, T_I = 150$ ms compared to $K_P = 14, T_I = 225$ ms in the small time window around the fifth second. At about the same time, however, there is an increase in the torque oscillations for $K_P = 10, T_I = 150$ ms compared to $K_P = 14, T_I = 225$ ms can be observed. These two observations do not allow any statement to be made as to

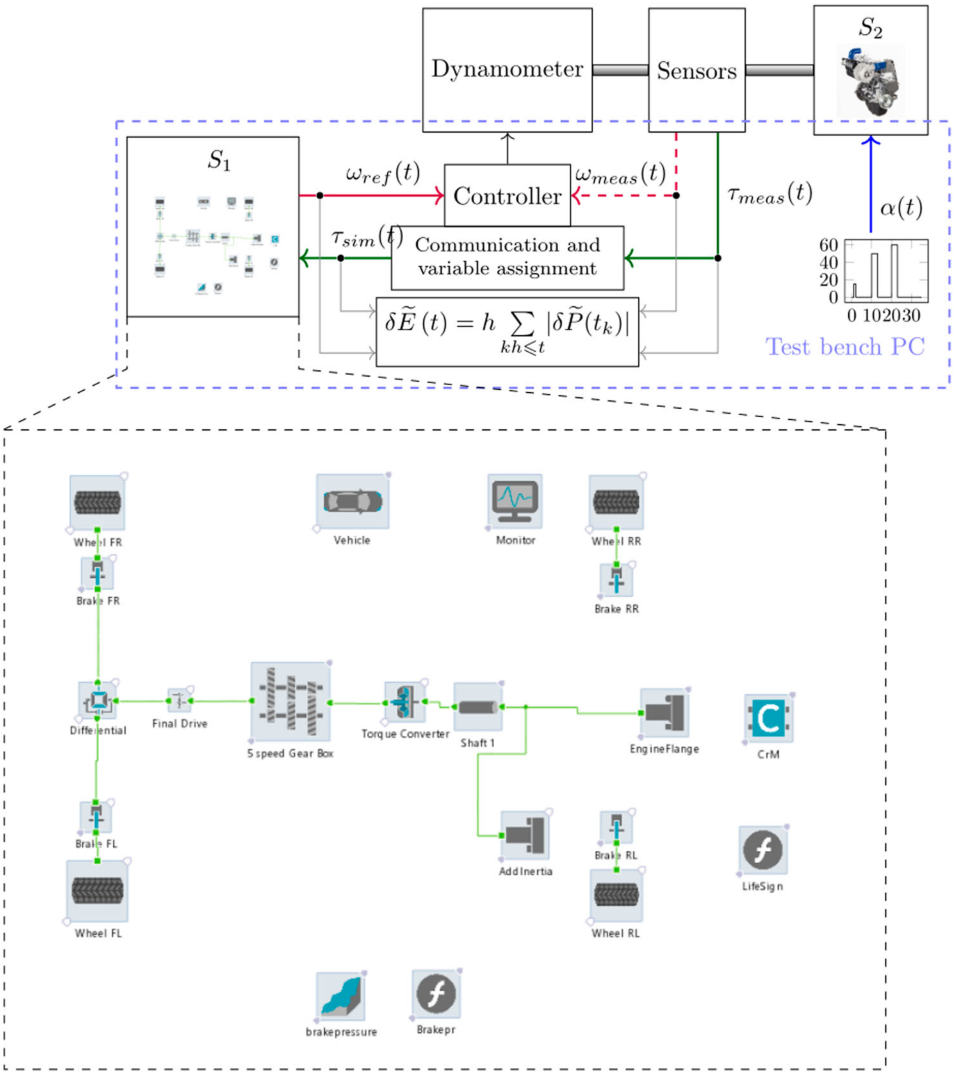


Figure 14. An additional HIL simulation is developed by replacing the simple inertial model with a more complex vehicle model. The vehicle model developed in AVL Cruise is a front-wheel drive car 1530 kg.

whether there is a connection between mechanical performance and TER. In future work, the experiments will be planned in such a way that this problem is thoroughly addressed. If minimizing TER minimizes the mechanical power transmitted through the shaft, this can be beneficial for the longevity of the test bench.

In this section, two sets of controller parameters are evaluated in two different HIL simulations, which are shown in Figure 10 and Figure 14. The respective TER values are shown in (27) and *(29)*. This was done to see if a good combination of parameters according to the heat map in Figure 12 is still good when a more complex vehicle model is used. It can be seen that one parameter set is better according to (27) and the other is better according to *(29)*. This is expected because the dynamics of two HIL simulations are different. Even if the results in (27) and *(29)* are different, the control parameters seem to deliver stable results in both cases. This result suggests the possibility of building a HIL simulation with a simpler model. However, to ensure that this is safe, a rigorous analysis of the approach will be carried out as part of future work. It is a plan to analyze how the simple experiment can be made most robust to the changes in the simulation model.

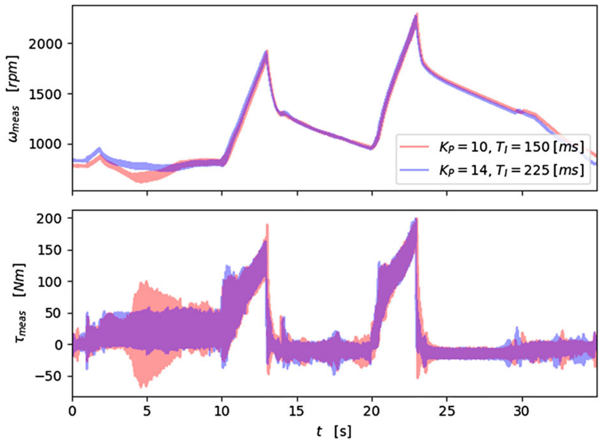


Figure 15. The speed and torque signals measured on the engine shaft when running a complex vehicle model (Figure 14) with various controller parameters.

6. Conclusion and future work

This article shows an analogy between a standard co-simulation and the HIL simulation. This analogy enables the use of co-simulation quality measurements such as TER (10). In [19] this criterion was used to solve a standard co-simulation with a variable communication step size. Section 3 of this article shows that the proposed criterion can be used to compare standard co-simulation masters. The criterion (10) is generally applicable to variable step, multiple rate co-simulations using higher order extrapolation techniques. In this article, this criterion is used on a co-simulation system with a small number of components running with a single rate and zero order hold (11). In future work, the system simulation should be divided into more subsystems, which should lead to a multi-rate co-simulation.

Section 4 shows the structure of the HIL simulation with an engine test bench and a simple inertia model (Figure 10). This section presents the underlying model simulated by the HIL simulation in Figure 9. The HIL simulation (Figure 10) corresponds to the numerical solution (2) and the underlying model of the HIL simulation (Figure 9) corresponds to the analytical model (1). This comparison makes it possible to measure with the TER how much the results of the HIL simulation differ from the ideal results of the underlying model. Actuators, controllers, sensors, the communication system and the co-simulation master can influence the results and contribute to the TER. Figure 10 shows the calculation of the TER with a Simulink model that was compiled as an FMU.

Section 5 shows an experimental approach for tuning the speed controller parameters of such a HIL simulation. The approach consists of repeating experiments with different parameters and evaluating them for the simulation quality using TER. The approach repeats experiments with different parameters and uses TER to evaluate the simulation quality (Figure 12). Such an approach can require a prohibitive number of experiments, and there are methods for selecting controller parameters [46,47] that can be more efficient. The presented experimental approach was used to demonstrate the possibility of using TER to set up an engine test bench. The main goal of Section 5 was to show that the TER can be used to evaluate a HIL simulation setup and not just a MIL simulation running on the office computer. Such an approach could possibly be generalized to other parts of the HIL simulation.

An example of a valuable benefit of such an approach would be to evaluate the influence of a communication network (Figure 10). In the work described in this article EtherCAT [44] is used to transfer power bond signals between the test bench and the computer running the FMUs. This computer is usually physically located next to the test bench. In practice it is interesting to see if a personal computer dislocated in a office building can be used for the same purpose. One of the potential solutions for this problem would be to replace EtherCAT protocol with UDP. If power residual introduced to a power bond transferred via communcation network could be calculated, it would allow to assess the impact of such a change on the HIL simulation quality. The work presented in this paper suggests that TER can be used in evaluating controllers. The topic of future work is to confirm that such application can be generalized to other parts of the HIL simulation and see if repeated experimentation can be completely avoided.

In Section 5 a simple inertia model (Figure 10) was replaced by a more complex vehicle model (Figure 14). Two pairs of parameters that were tested with the simpler HIL simulation (27) were tested with the more complex one *(29)*. The TER analysis shows that in the first case one pair is better, in the second case the other. Such results are not surprising due to the different dynamics of the simulated parts of the HIL simulation. However, this approach gives a suggestion on how to prepare an engine test bench for a complex simulation. If the setup found with a simpler simulation is robust to parameter changes in such a simulation, such a setup could potentially be very useful. One topic of the future work would be to rigorously define and analyze the robustness of the HIL simulation with respect to the TER.

The goal of this article is to enable the comparison of two co-simulation setups without internal knowledge of the simulation models. Such a quality evaluation enables a commissioning engineer to set up the HIL simulation without being familiar with the simulation models. This enables a faster initial setup of a HIL simulation with an engine in the loop and reduces the time and costs involved in preparing the experiments. This article presents an initial attempt to use the total energy residual (TER) for this purpose. The usefulness of such a measure was demonstrated in this article using an experimental approach. In the future, the robustness, the influence on the power consumption and the possibility of tuning different parts of an engine test bench will be investigated.

Notes

1. It is important to note that the integration error of the co-simulation slaves is ignored in this section.
2. This is a vehicle model without an engine because a physical engine is present during the HIL simulation.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by AVL List.

ORCID

Slaven Glumac http://orcid.org/0000-0002-2855-3307

References

[1] Isermann R, Schaffnit J, Sinsel S. Hardware-in-the-loop simulation for the design and testing of engine-control systems. Control Eng Pract. 1999;7:643–653.

[2] Fathy HK, Filipi ZS, Hagena J, et al. Review of hardware-in-the-loop simulation and its prospects in the automotive area). Modeling and simulation for military applications; 2006; Orlando.

[3] Vijaykumar A, Prashanth V, Sreehari S, et al. Simulation of Road Vehicle on Engine Test Bed). Internal Combustion Engine Division Fall Technical Conference; 2005; Ottawa.

[4] Perez F, Moulin P, Del Mastro A. Vehicle simulation on an engine test bed). Proc. SIA Conference on Diesel Engines–The low CO2 & Emissions Reduction Challenge; 2008; Rouen.

[5] Denger D, Hirsch A, Hochmann G, et al. The virtual vehicle at the engine test bed. MTZ worldwide. 2010;71(9):32–36.

[6] Mayr C, Merl R, Gigerl HP, et al. Test emissionsrelevanter Fahrzyklen auf dem Motorprüfstand). Simulation und Test 2018; 2018; Hanau.

[7] Klein S, Savelsberg R, Xia F, et al. Engine in the Loop: Closed Loop Test Bench Control with Real-Time Simulation. SAE International Journal of Commercial Vehicles. 2017;10(1):95–105.

[8] Rahikainen J, González F, Naya MÁ. An automated methodology to select functional co-simulation configurations. Multibody Sys Dyn. 2020;48(1):79–103.

[9] Wu X, Monti A. Methods for partitioning the system and performance evaluation in power-hardware-in-the-loop simulations. Part I). 31st Annual Conference of IEEE Industrial Electronics Society; 2005; Raleigh.

[10] Wu X, Monti A. Methods for partitioning the system and performance evaluation in power-hardware-in-the-loop simulations. Part II). 31st Annual Conference of IEEE Industrial Electronics Society; 2005; Raleigh.

[11] Ren W, Steurer M, Baldwin TL. Improve the stability and the accuracy of power hardware-in-the-loop simulation by selecting appropriate interface algorithms. IEEE Trans Ind Appl. 2008;44(4):1286–1294.

[12] Lentijo S, D'Arco S, Monti A. Comparing the dynamic performances of power hardware-in-the-loop interfaces. IEEE Trans Ind Electron. 2009;57(4):1195–1207.

[13] Functional Mock-up Interface for Model Exchange and Co-Simulation, Version 2.0. (3 January 2021). [Online]. Available: https://github.com/modelica/fmi-standard/releases/download/v2.0.2/FMI-Specification-2.0.2.pdf.

[14] Tools |Functional Mock-up Interface. (3 January 2021).. [Online]. Available: https://fmi-standard.org/tools/.

[15] Mosterman PJ, Prabhu S, Erkkinen T. An industrial embedded control system design process). Proceedings of the Canadian Engineering Education Association; 2004; Montreal.

[16] Bringmann E, Krämer A. Model-based testing of automotive systems). 2008 1st international conference on software testing, verification, and validation; 2008; Lillehammer.

[17] Broenink JF, Ni Y. Model-driven robot-software design using integrated models and co-simulation). 2012 International Conference on Embedded Computer Systems; 2012; Samos.

[18] Drenth E, Törmänen M, Johansson K, et al. Consistent Simulation Environment with FMI based Tool Chain). Proceedings of the 10 th International Modelica Conference; 2014; Lund.

[19] Sadjina S, Kyllingstad LT, Skjong S, et al. Energy conservation and power bonds in co-simulations: non-iterative adaptive step size control and error estimation. Eng Comput. 2017;33(3):607–620.

[20] Borutzky W. Bond graph modelling of engineering systems. New York: Springer; 2011.

[21] Busch M, Schweizer B. An explicit approach for controlling the macro-step size of co-simulation methods). Proceedings of the 7th European Nonlinear Dynamics Conference; 2011; Rome.

[22] Benedikt M, Watzenig D, Zehetner J, et al. NEPCE-A nearly energy-preserving coupling element for weak-coupled problems and co-simulation). International Conference on Computational Methods for Coupled Problems in Science and Engineering; 2013.

[23] González F, Naya MÁ, Luaces A, et al. On the effect of multirate co-simulation techniques in the efficiency and accuracy of multibody system dynamics. Multibody Sys Dyn. 2011;25(4):461–483.

[24] Schierz T, Arnold M, Clauß C. Co-simulation with communication step size control in an FMI compatible master algorithm). Proceedings of the 9th International MODELICA Conference; 2012; Munich.

[25] Arnold M, Clauß C, Schierz T. Error analysis and error estimates for co-simulation in FMI for model exchange and co-simulation V2.0). Progress in Differential-Algebraic Equations; 2014.

[26] Verhoeven A, Tasic B, Beelen T, et al. BDF compound-fast multirate transient analysis with adaptive step-size control. J. Numer. Anal. Ind. Appl. Math. 2008;3: 275–297.

[27] Günther M, Kværnø A, Rentrop P. Multirate partitioned runge-kutta methods. BIT Numerical Mathematics. 2001;41(3):504–514.

[28] Viel A. Implementing stabilized co-simulation of strongly coupled systems using the Functional Mock-up Interface 2.0). Proceedings of the 10 th International Modelica Conference; 2014; Lund.

[29] Nguyen PH. Interpolation and error control schemes for algebraic differential equations using continuous implicit Runge-Kutta methods. Toronto: University of Toronto, Department of Computer Science; 1995.

[30] Enright WH. Continuous numerical methods for ODEs with defect control. J Comput Appl Math. 2000;125(1–2):159–170.

[31] Glumac S. "cs_perspective," 22 1 2021. [Online]. Available: https://github.com/sglumac/cs_perspective.

[32] Glumac S. "BenchmarkFMUs," 22 1 2021. [Online]. Available: https://github.com/sglumac/cs_perspective.

[33] Benedikt M, Watzenig D, Hofer A. Modelling and analysis of the non-iterative coupling process for co-simulation. Math Comput Model Dyn Syst. 2013;19(5): 451–470.

[34] Schweizer B, Li P, Lu D. Explicit and implicit cosimulation methods: Stability and convergence analysis for different solver coupling approaches. J Comput Nonlinear Dyn. 2015;10(5):051007.

[35] Schweizer B, Lu D. Predictor/corrector co-simulation approaches for solver coupling with algebraic constraints. ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift fúr Angewandte Mathematik und Mechanik. 2015;95(9):911–938.

[36] Busch M. Continuous approximation techniques for co-simulation methods: Analysis of numerical stability and local error. ZAMM - Journal of Applied Mathematics and Mechanics. 2015;96(9):1061–1081.

[37] Glumac S, Kovačić Z. Relative consistency and robust stability measures for sequential co-simulation). Proceedings of the 13th International Modelica Conference; 2019; Regensburg.

[38] Hindmarsh AC, Brown PN, Grant KE, et al. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. ACM Trans Math Softw. 2005;31(3):363–396.

[39] Gomes C, Thule C, Broman D, et al. Co-Simulation: A Survey. ACM Comput Surv. 2018;51(3):1–33.

[40] AVL Puma 2 TM. [Online]. Available: https://www.avl.com/-/avl-puma-2. [Accessed 30 1 2021].

[41] AVL DynoFORCE TM ASM for Engine Testing - Load systems for Engine Testing. [Online]. Available: https://www.avl.com/load-unit-for-engine-testing/-/asset_publisher/gYjUpY19vEA8/content/avl-dynoroad?inheritRedirect = false. [Accessed 30 1 2021].

[42] Rotary Encoders - 349529–2 K. [Online]. Available: https://www.heidenhain.de/fileadmin/pdb/media/img/349529–2K_Rotary_Encoders_en.pdf. [Accessed 30 1 2021].

[43] T12HP Torque Transducer: High Precision Over Entire Range | HBM. [Online]. Available: https://www.hbm.com/en/6384/t12hp-torque-transducer-with-maximum-precision/. [Accessed 30 1 2021].

[44] Jansen D, Buttner H. Real-time Ethernet: the EtherCAT solution. Computing and Control Engineering. 2004;15(1):16–21.

[45] AVL CRUISE TM. [Online]. Available: https://www.avl.com/cruise. [Accessed 30 1 2021].

[46] Leonhard W. Control of electrical drives. Berlin, Heidelberg: Springer Science & Business Media; 2001.

[47] Dorf RC, Bishop RH. Modern control systems. Upper Saddle River, New Jersey: Pearson; 2011.