Optimal Trigger Sequence for Non-iterative Co-simulation

Franz Rudolf Holzinger[a] and Martin Benedikt[b]
VIRTUAL VEHICLE Research Center, Inffeldgasse 21/A/II, Austria

Keywords: Trigger Sequence, Execution Order, Calculation Order, Non-iterative Co-simulation, Travelling Salesman Problem.

Abstract: An execution sequence approach of interacting subsystem is presented for non-iterative co-simulation frameworks. Local behaviour of coupling signals and subsystems are used to describe a general optimization problem of co-simulation. Therefore the linking matrix is weighted by analysis of the coupling signals within fuzzy integrated expert knowledge. The weighted linking matrix is transferred to a directed co-simulation graph, which can be interpreted as an appropriate travelling sales man problem. The solution of this co-simulation graph provides an optimized trigger sequence of the subsystems.

1 INTRODUCTION

In the last decade co-simulation becomes a relevant technique in diverse system development approaches, especially in the automotive industry. The increasing complexity of the systems forces engineers to divide problems in several smaller sub-problems. These subsystems are often modelled in specific simulation environments and are solved by their own solvers. Co-simulation is a technique to combine these submodels to an overall common system and allows to run a holistic simulation (Kübler and Schiehlen, 2000).

The increasing number of subsystems and coupling signals in co-simulation presents a significant challenge for each co-simulation user and application engineer. The individual subsystems and coupling signals have different dynamic behaviour. These have to be considered in the coupling by appropriate coupling time steps of the several subsystems, proper choice of extrapolation filters and suitable trigger sequence of the subsystems. In general, application engineers have barely information about the subsystems and so it is hardly possible to prepare a co-simulation configuration with respect to simulation duration and simulation accuracy. Therefore it is necessary to support the co-simulation user by an automatically configuration of the co-simulation. This idea was already generally discussed by the authors (Benedikt and Holzinger, 2016). An automated approach for obtaining the trigger sequence for co-simulation to improve the simulation accuracy is discussed in the following.

The outline of this work is as follows. This paper starts with an introduction into non-iterative co-simulation. Reasonable information utilized for trigger sequence determination is identified based on discussion on possible sources of coupling errors. An approach to get a proper trigger sequence based on the idea to solve travelling sales man problem is presented. This approach is illustrated and discussed by an example.

[a] https://orcid.org/0000-0003-3551-6579
[b] https://orcid.org/0000-0003-2652-6812

2 NON-ITERATIVE CO-SIMULATION

There are two well-established types of co-simulation techniques: iterative and non-iterative. Compared to iterative co-simulation the non-iterative one does not allow to repeat a coupling step. In general this allows a higher simulation performance in terms of simulation duration. On the other hand the discretization caused by the coupling mechanism is much higher. Nevertheless the non-iterative co-simulation is commonly used in industry. Therefore one of the reasons is that many simulation tools and subsystems do not support external control reset of internal states and thus repeating a time step is not possible.

In general, the induced coupling errors are higher in the case of non-iterative co-simulation compared to iterative co-simulation. Bidirectional dependencies

Holzinger, F. and Benedikt, M.
Optimal Trigger Sequence for Non-iterative Co-simulation.
DOI: 10.5220/0007833800800087
In *Proceedings of the 9th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH 2019)*, pages 80-87
ISBN: 978-989-758-381-0
Copyright © 2019 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

between subsystems cause causality problems and so extrapolation filtering is needed. Low order polynomial extrapolation filters are generally used to estimate the input for the next simulation step. The commonly used filter technique is the zero-order-hold (ZOH) extrapolation. Here the last determined value of the coupling signal is set as input signal during the next calculation step. In addition to the error between the estimated signal and the actual determined signal, the resulting staircase-shaped input signal can also stimulate the subsystem and cause wrong results.

From the co-simulation configuration point of view, there are mainly three different kinds of options, to influence and improve the simulation quality:

- Coupling step size to define how often signals are exchanged between the subsystems
- Extrapolation filter to solve the causality problem of bidirectional dependent subsystems
- Trigger sequence to define the calculation order of the subsystems

The topics of extrapolation filter and coupling time-steps were already well discussed in the past (Busch and Schweizer, 2011) (Benedikt et al., 2010). There are still methods which compensate the extrapolation error (Benedikt and Hofer, 2013), (Benedikt et al., 2013). However a proper trigger sequence prevents or at least reduces coupling errors at (some) inputs.

3 TRIGGER SEQUENCE

Co-simulation typically applies parallel scheduling scheme in which all subsystems are calculated in parallel. This approach has a high simulation performance (in terms of simulation duration) because the subsystems can be calculated in the same time. Nevertheless every input signal has to be extrapolated and so in each coupling signal an error is induced. To reduce the coupling error subsystems can be executed in sequential order, i.e. each subsystem is calculated after the other.

The challenge is to define a trigger sequence or calculation order, which minimizes the coupling effects. The number of possible sequences factorially increases with the number of subsystems. A simple co-simulation example with $m = 4$ subsystems (see Fig. 1) leads already to $m! = 24$ different possible configurations with respect to the execution order. The induced coupling error and thus the results change depended on the defined trigger sequence. With increasing number of subsystems it is (even for experienced co-simulation application engineers) hardly possible to define a well-defined trigger sequence.

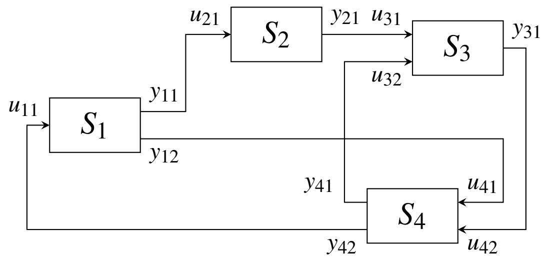

Figure 1: Co-simulation Topology.

The connection between the inputs u_i and outputs y_i of subsystems in the co-simulation is mostly the single information which is available for the configuration problem. This relation can be described with the linking matrix:

$$\mathbf{u} = \mathbf{L} \cdot \mathbf{y}. \tag{1}$$

The linking matrix $\mathbf{L}$ is an orthogonal matrix[1] of the dimension n, where n is the number of connections between the subsystems. The connections of the co-simulation topology from Figure 1 can be rewritten as follows:

$$\begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{32} \\ u_{41} \\ u_{42} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{11} \\ y_{12} \\ y_{21} \\ y_{31} \\ y_{41} \\ y_{42} \end{bmatrix}, \tag{2}$$

where $\mathbf{u} = [u_1, u_2, \ldots, u_n]^T$ is the input vector and $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$ represents the output vector. A fully description of the co-simulation is given by this relation. Nevertheless this relation does not give any information about the dependencies between the several subsystems. Therefore an other relation is needed, which describes a formal relation between each subsystem. Instead of the linking matrix $\mathbf{L}$ which describes the connection between inputs and outputs, the matrix $\mathbf{D}$ describes the dependency of the subsystems themselves. This relation can be written as follows:

$$\mathbf{D} = \left(\mathbf{T}^T \cdot \mathbf{L} \cdot \mathbf{S}\right)^T. \tag{3}$$

The matrix $\mathbf{D}$ is a $m \times m$ matrix and represents the adjacency matrix, i.e. the dependency from one subsystem to another. The matrices $\mathbf{S}$ and $\mathbf{T}$ are $n \times m$ dimensional and describe the correlation between the subsystems and the connections, where $\mathbf{S}$ describes the allocation of the source y_i to the subsystems and $\mathbf{T}$

[1] In order to ensure orthogonality for multiply connected outputs, these outputs are duplicated, i.e. each output has exactly one connection line.

the allocation of the target u_i to the subsystems. The columns represent the inputs of each subsystem and the rows their outputs.

The linking matrix is a dependency description of the connection, but it does not give any information of the dependency of the subsystems. Regarding to the trigger sequence the dependency of the subsystems is more relevant.

The system dependency $\mathbf{D}$ of the co-simulation topology from Figure 1 is given as follows:

$$\mathbf{D} = \begin{bmatrix} - & 1 & 0 & 1 \\ 0 & - & 1 & 0 \\ 0 & 0 & - & 1 \\ 1 & 0 & 1 & - \end{bmatrix} \tag{4}$$

A more general description of the dependency matrix $\mathbf{D}$ can be written as follows:

$$\mathbf{D} = \left(\mathbf{T}^T \cdot \mathbf{C} \cdot \mathbf{L} \cdot \mathbf{S}\right)^T, \tag{5}$$

with $\mathbf{C}$ as a diagonal matrix of weighted inputs $c_1, c_2, \ldots, c_n$. With this additional weights c_i it is possible to increase the relevance of several inputs y_i and thus to change the dependency between subsystems.

3.1 Extended TSP Problem

The co-simulation network can be interpreted as an asymmetric, directed graph. The subsystems represent the nodes and the edges are the connections between the subsystems. The number of signals from one subsystem to another weights the edge, and so the number of extrapolated inputs. The co-simulation graph is depicted in Figure 2. The nodes represent the several subsystems from Figure 1 and the edges the coupling signals between the subsystems.

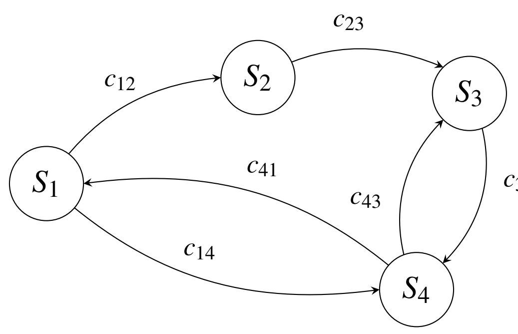

Figure 2: Co-simulation Graph.

In this context the trigger sequence can be interpreted as a Hamiltonian cycle, where every node or subsystem is exactly visited once. In the case that the $\mathbf{C}$ is the identity matrix (i.e. all coefficients in the diagonal $c_{ij} = 1$), the value of the edges represents the number of extrapolated inputs. If for example node 3 is visited (i.e. subsystem S_3 is calculated) the sum of incoming edges represents the number of extrapolated coupling signals for this subsystem. Regarding to the adjacency matrix $\mathbf{D}$ that means the sum of the node's column, see (6).

$$\mathbf{D} = \begin{bmatrix} 0 & c_{12} & 0 & c_{14} \\ 0 & 0 & c_{23} & 0 \\ 0 & 0 & 0 & c_{34} \\ c_{41} & 0 & c_{43} & 0 \end{bmatrix} \tag{6}$$

If the node 3 is already visited, the subsystem has calculated and the results are available. There is no extrapolation needed any more for these coupling signals, i.e. the row of the node has to set to zero, see (7).

$$\mathbf{D} = \begin{bmatrix} 0 & c_{12} & 0 & c_{14} \\ 0 & 0 & c_{23} & 0 \\ 0 & 0 & 0 & 0 \\ c_{41} & 0 & c_{43} & 0 \end{bmatrix} \tag{7}$$

The more nodes or subsystems are visited the more coefficients become zero and the less inputs have to be extrapolated.

Based on the graph representation of the co-simulation network, the optimal trigger sequence results from the shortest path between all nodes. A general description of the optimization problem can be written as follows:

$$\min \sum_i \sum_{j \backslash i} \left(\sum_{k \backslash I} c_{kj} \right) x_{ij} \tag{8}$$

The coefficients c_{ij} represent the value of the edges and x_{ij} is 0 or 1 according if a path or node is visited. Starting from each row i, every column j is considered. The coefficients c_{kj} along the column j are accumulated up. Already visited nodes or columns I are not considered.

Nowadays a typical co-simulation problem includes less than ten subsystems and so in general the dimension of the adjacency matrix is lower than ten. Therefore the brute-force solving strategy for the co-simulation graph is sufficient and can be solved with low effort. With less than 5 nodes or subsystems the brute-force solving method is faster than alternative solving algorithms. The calculation effort of the brute-force and other algorithm to solve the TSP problem is compared in Figure 3.

Nevertheless, with an increasing number of subsystems the calculation effort of the brute-force approach significantly increases and so other solving strategies with lesser computation effort should be used.

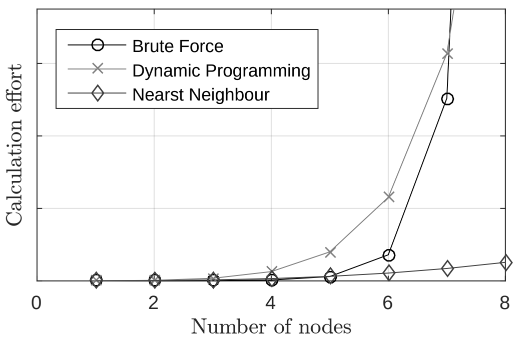

Figure 3: Calculation effort to solve a TSP problem.

However, it turns out that without any other information the subsystem within the lowest number of inports should be calculated first (Holzinger and Benedikt, 2019).

It is obvious that, if no more information is available, the minimal number of extrapolated inputs is a desirable goal for a trigger sequence. To find this sequence the shortest way to connect all nodes has to be found. The optimum trigger sequence can be described as a modified travelling sales man problem (TSP). In contrast to the original TSP, the (outgoing) edges of already visited nodes have no impact to the other nodes and thus these edges are set to zero.

3.2 Connection Properties

Depending on the relevance or impact of a connection the weight of the signal c_{ij} can influence the result of the graph and consequently the trigger sequence. This weighting can be influenced by different coupling signal and subsystem properties (Benedikt and Holzinger, 2016). In the following chapters some relevant properties are discussed.

3.2.1 Subsequence

The subsequence is a subsystem property affected by the subsystem solver, coupling time-step and the subsystem interface. If the internal solving step size of a subsystem, the so called micro step-size, is smaller than the coupling step-size the subsystem provides subsequence. Instead of a single value at the coupling time-step, a sequence with the micro-steps is given. These additional samples in the coupling signals generally allow more accurate simulation results compared to single value coupling signals (coupling time step equal micro time step).

3.2.2 Signal Types

The characteristic of the coupling signals is an important information for the choice of a proper extrapolation filter. The prediction of continuous signals is easier than the estimation of a discontinuous or discrete coupling signal. The most advanced extrapolation and compensation techniques are based on continuous signal and so better to prevent the extrapolation of discontinuous and discrete signals.

3.2.3 Direct Feed-through

The direct impact of an input to an output signal is one of the most important properties of subsystem. Especially if the direct feed-through channels are formed in a closed loop and affects an algebraic loop. An direct feed-through indicates that an uncertainty at the input signal (e.g. caused by an extrapolation filter) means an uncertainty at the output signal. The extrapolation of such inputs should be avoided.

3.2.4 Subsystem Dynamic

An other indicator for the extrapolation and trigger sequence of a subsystem is its dynamical behaviour. A high system dynamic indicates that uncertainty at the input of a system has a high impact to the output. And on the other hand that low dynamic systems barely react to any changes and discontinuities to the input. So the extrapolation of inputs of slow input-output dynamics should be preferred instead of inputs of high dynamic subsystems.

3.2.5 Coupling Signal Frequency

In addition to the system dynamic the frequency of the coupling signal represents a further information for the extrapolation filter setting and trigger sequence. It is obvious that it is easier to extrapolate a signal with low frequency components. However, coupling signals with high frequency components are not implicitly signals which should be prevented to extrapolate. If the following system has a low pass characteristic, high frequencies and uncertainty at the input signal are not critical. The combined consideration of the individual input-output dynamics and the signal frequency determines if extrapolation shall be applied.

3.3 Contribution of Expert Knowledge

To combine the connection properties fuzzy logic is used. The output of the fuzzy algorithm represents the weight of the input signal c_{ij} and for the inputs the results of the signal quantities are used. Expert

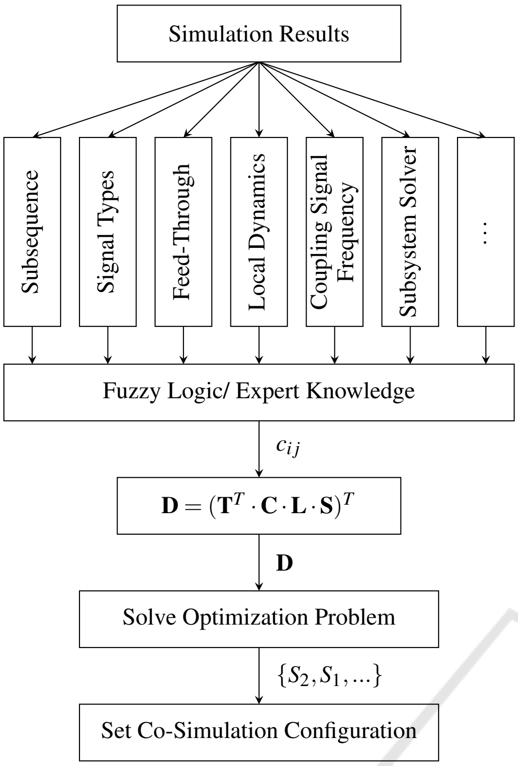

Figure 4: Schematic procedure to get the trigger sequence.

knowledge is used to describe rules and transfer the properties of each coupling signal to a weight of the co-simulation graph.

A schematic procedure to determine the trigger sequence is shown in Figure 4. The procedure starts with the results of a successfully simulated co-simulation. The results are used for the signal analysis, where the properties of the signals are computed, like signal types, direct feed-through, etc. The information is merged into the fuzzy logic, where the expert knowledge is integrated. The fuzzy algorithm support a weight for every connection. The weighted matrix delivers with the source and target transfer matrix the adjacency matrix $\mathbf{D}$, which is solved to get an optimized trigger sequence.

In the case, that no results are available to calculate the weight of the connections (e.g. after a new configuration), all coefficients of the weighted matrix $\mathbf{C}$ are set to a default value $c_{ij} = 1$. The resulting default trigger sequence has minimized the number of inputs to extrapolate.

4 EXAMPLE

The following section shows an example to demonstrate the described approaches. Therefore three strategies are used to determine trigger sequences. The first strategy considers the number of extrapolated inputs. The second uses the property of direct feed-through to determine the trigger sequence and the third strategy or advanced strategy weights the coupling signals by subsystem and signal behaviour.

The example consists of four subsystems S_1, S_2, S_3 and S_4 which are connected to another as shown in Figure 1. The subsystems S_1, S_3 and S_4 are based on an example in (Benedikt and Drenth, 2018). The subsystems describe the behaviour of spring-damper-mass systems. The additional model S_2 is a superposed controller, which controls the output y_{42} of the subsystem S_4.

4.1 Subsystem Description

The four subsystems are solved with a fixed-step size solver (Euler) and a step size $\delta T = 0.1\,ms$. The coupling time steps is constant for all subsystems $\Delta T = 1\,ms$. The extrapolation filters for all coupling signals is set to ZOH.
The mathematical description of the several subsystems is as follows:

4.1.1 Subsystem 1

The subsystem S_1 is a second order linear model with one input u_{11} and two outputs $\mathbf{y} = [y_{11}, y_{12}]^T$. The parameters are set $c = 1000$ and $J_1 = 0.1$. There is a direct feed-through $d = 44.27$ from the input u_{11} to y_{12}.

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} 0 & -\frac{1}{J_1} \\ c & -\frac{d}{J_1} \end{bmatrix} \mathbf{x} &+ \begin{bmatrix} 1 \\ d \end{bmatrix} u \\ \mathbf{y} &= \begin{bmatrix} 0 & \frac{1}{J_1} \\ c & -\frac{d}{J_1} \end{bmatrix} \mathbf{x} &+ \begin{bmatrix} 0 \\ d \end{bmatrix} u \end{aligned} \tag{9}$$

4.1.2 Subsystem 2

The second subsystem describes a PI-controller with the parameter $k_p = 100$ and $k_i = 0.1$ of the form:

$$\begin{aligned} \dot{x} &= k_i \cdot x &+ (r-u) \\ y &= x &+ k_p \cdot (r-u) \end{aligned} \tag{10}$$

The set value $r = 10$ and the initial states of all subsystems are zero.

4.1.3 Subsystem 3

The subsystem S_3 has two inputs $\mathbf{u} = [u_{31}, u_{32}]^T$ and an output y_{31}. The model has an integrative behaviour, there is no direct feed-through from the inputs to the output. The parameter $J_3 = 0.9$.

$$\begin{aligned} \dot{x} &= \frac{1}{J_3} \cdot x &+ \begin{bmatrix} 1 & -1 \end{bmatrix} \mathbf{u} \\ y &= x & \end{aligned} \tag{11}$$

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

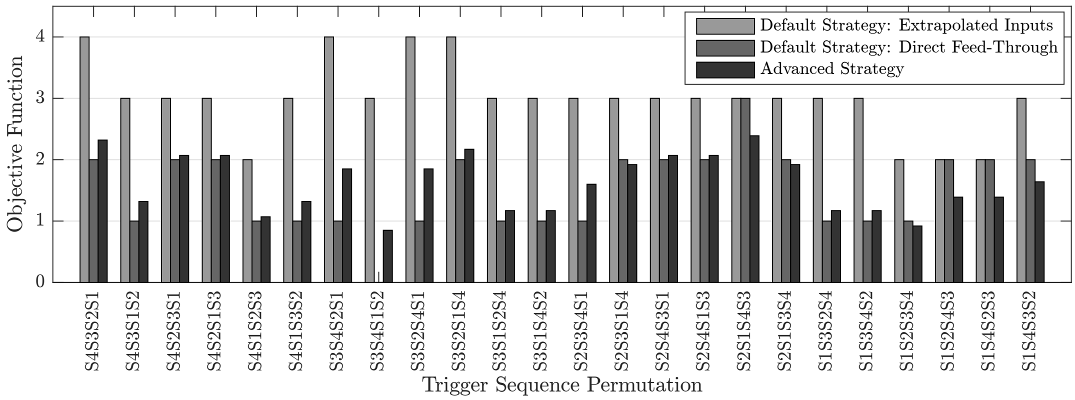

Figure 5: Evaluation of the objective function regarding different strategies: default strategy based on the extrapolated inputs, default strategy based on the direct feed-through and advanced strategy.

4.1.4 Subsystem 4

The model S_4 has two inputs $\mathbf{u} = [u_{41}, u_{42}]^T$ and two outputs $\mathbf{y} = [y_{41}, y_{42}]^T$ with a direct feed-through $d = 44.27$ from the input u_{42} to y_{41} and the parameter are set $c = 1000$ and $J_4 = 0.5$.

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} 0 & -\frac{1}{J_4} \\ c & -\frac{d}{J_4} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 & 1 \\ -1 & d \end{bmatrix} \mathbf{u} \\ \mathbf{y} &= \begin{bmatrix} c & -\frac{d}{J_4} \\ 0 & \frac{1}{J_4} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 & d \\ 0 & 0 \end{bmatrix} \mathbf{u} \end{aligned} \tag{12}$$

4.2 Default Strategies

The default strategies describe the possibility to define a trigger sequence without knowledge or with reduced knowledge of the subsystem behaviours. Therefore no additional analysis of the coupling signals is needed to determine a trigger sequence for the subsystems.

4.2.1 Extrapolated Inputs

In a first step without any additional information the weight of all coupling signals is set to one, $c_i = 1$, and so the adjacency matrix $\mathbf{D}$ of the graph is given by (4). In this case the linking matrix $\mathbf{L}$ is the single required information to determine a trigger sequence. The resulting trigger sequence represents the minimum number of extrapolated inputs. The best solutions of the default matrix are $\{S_4, S_1, S_2, S_3\}$, $\{S_1, S_2, S_3, S_4\}$, $\{S_1, S_2, S_4, S_3\}$ and $\{S_1, S_4, S_2, S_3\}$. All of these solution delivers a minimum according to (8). Each for each of the resulting trigger sequences two inputs have to be extrapolated in each simulation step.

4.2.2 Direct Feed-through

If the direct feed-through of a system is known by the application engineer, this information can be used as prior knowledge to define a trigger sequence. Some subsystems also provide this information, e.g. the FMI standard supports the information of direct feed-through (Blochwitz et al., 2012).

The subsystems of the considered example have three inputs with a direct feed-through to an output. A reduced adjacency matrix $\mathbf{D}$ with the direct feed-through is given as follows:

$$\mathbf{D} = \begin{bmatrix} - & 1 & 0 & 0 \\ 0 & - & 0 & 0 \\ 0 & 0 & - & 1 \\ 1 & 0 & 0 & - \end{bmatrix}, \tag{13}$$

where each input with a feed-through to an output is set to one. Based on the adjacency matrix D from (13) the best solution is $\{S_3, S_4, S_1, S_2\}$. The resulting trigger sequence represents the minimum number of inputs with a direct feed-through to an output. In the example exits one configuration, where no input with a direct feed-through has to be extrapolated.

4.3 Advanced Strategy

By the advanced strategy the default weights are modified based on the identified connection properties. The adjacency matrix $\mathbf{D}$ can be rewritten as follows:

$$\mathbf{D} = \begin{bmatrix} - & 1 & 0 & 0.35 \\ 0 & - & 0.25 & 0 \\ 0 & 0 & - & 0.72 \\ 0.67 & 0 & 0.25 & - \end{bmatrix} \tag{14}$$

The integrative behaviour of subsystem S_3 reduces the coefficients $c_{23} = c_{43} = 0.25$. On the other hand, the

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

coefficients c_{12},c_{34} and c_{41} of the coupling inputs with direct feed-through behaviour are still high.

The cost to solve the objective function for all possible trigger sequences with respect to the three discussed strategies are illustrated in Figure 5. The two default strategies show the evaluation of the objective function based on the minimum number of extrapolated inputs with respect to (4) and the minimum number of extrapolated direct feed-through channels with the adjacency matrix (13). The advanced strategy show the results of (14). The best solution delivers the trigger sequence $\{S_3,S_4,S_1,S_2\}$.

The default strategy based on the direct feed-through and the advanced strategy provides the same trigger sequence. This is due the fact, that the subsystem properties are mainly characterized by the direct feed-through. Nevertheless, in examples with more coupling signals or if the property of direct feed-through is not known, the presented (advanced) approach helps to get a proper trigger sequence.

The determined solution $\{S_3,S_4,S_1,S_2\}$ is not from the optimal set, which was calculated with the default strategy based on the minimum number of extrapolated inputs $\{S_4,S_1,S_2,S_3\}$, $\{S_1,S_2,S_3,S_4\}$, $\{S_1,S_2,S_4,S_3\}$ and $\{S_1,S_4,S_2,S_3\}$. The solutions from the default strategy have a minimal number of extrapolated inputs, but all of the solutions extrapolate an input signal of a subsystem with a direct feed-through property. The direct feed-through can be interpreted as an additional extrapolation of corresponding output signal and will cause additional coupling errors.

The results of the subsystem S_1 are shown in Figure 6. The dashed plot in both figures represents the monolithic simulation result, where all subsystems were simulated assembled and with one solver in a single simulation environment. The grey lines show the results of all possible permutations of the trigger sequence. The black solide line is the result of the trigger sequence $\{S_3,S_4,S_1,S_2\}$ from the presented approach.

The result of the determined trigger sequence is next to the monolithic simulation and has smaller difference compared to the remaining permuted sequences. In the determined solution no input with direct feed-through behaviour was extrapolated and so the coupling error was reduced although that an additional input has to be extrapolated compared to the default solutions.

The other way around the solutions with high efforts depicts a discrepancy to the monolithic solution. Nevertheless, even the best solution show discrepancy between the monolithic solution caused by the delay of the extrapolation filter. To compensate these remaining difference advanced extrapolation techniques can be used (Benedikt and Hofer, 2013) (Benedikt et al., 2013).

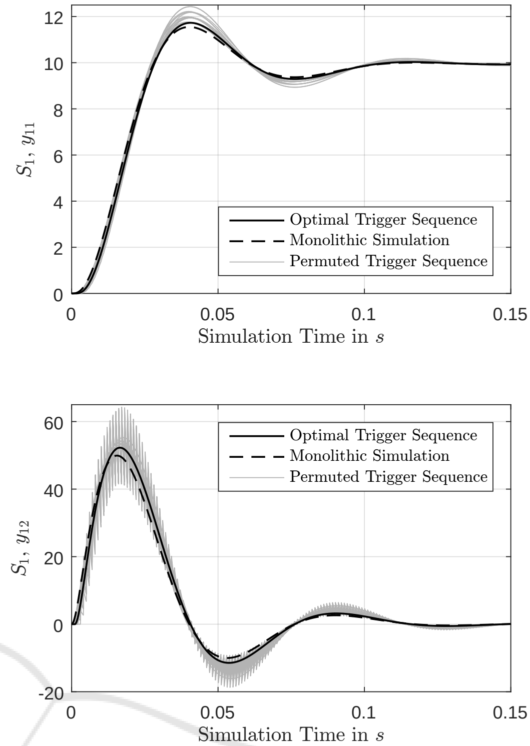

Figure 6: Simulation results of the permutation, monolithic simulation and the optimal trigger sequence $\{S_3,S_4,S_1,S_2\}$.

The presented approach provides an optimized trigger sequence for sequential co-simulation. Nevertheless it is obvious that the simulation duration for sequential co-simulation is higher compared to parallel co-simulation, where each subsystem can calculate at the same time. On the other hand, generally, sequential co-simulation delivers better simulation results, due the fact, that less inputs have to be extrapolated. Hierarchical co-simulation is a method to get a trade-off between simulation accuracy from and simulation duration. A configuration approach for hierarchical co-simulation based on the number of extrapolated inputs and calculation effort of the several subsystems was already discussed by the authors (Holzinger and Benedikt, 2019).

5 CONCLUSIONS

The article addresses an approach to determine an optimized trigger sequence for non-iterative co-simulation. Therefore expert knowledge e.g. in form of fuzzy rules is used to assess the extrapolation quality of each input, based on the properties of the coupling signals. The so resulting weighted connections are used to describe a co-simulation graph. The solution of the graph delivers an optimized trigger sequence.

If no additional information in terms of simulation results is available to calculate the connection properties, the approach delivers a default solution related to the minimum number of extrapolated coupling signals. After each simulation run the simulation results can be used to calculate the connection properties and finally weight the graph to get an optimal trigger sequence. This information can also be used to set the time-step and the extrapolation filter or at least to assess the co-simulation.

In a further work, the presented approach will be extended for the configuration of hierarchical co-simulation, so that some subsystems can be calculated in parallel and others sequentially. This allows a trade-off between simulation accuracy and simulation duration.

REMARK PATENT

The presented work describes a part of a novel automatic configuration approach for co-simulation of distributed components. Protected by a pending European patent (Benedikt et al., 2016) the outlined schemes are supported by the co-simulation platform Model.CONNECT™ (AVL, 2018) from AVL.

ACKNOWLEDGMENT

This work was accomplished at the VIRTUAL VEHICLE Research Center in Graz, Austria. The authors would like to acknowledge the financial support of the COMET K2 - Competence Centers for Excellent Technologies Programme of the Austrian Federal Ministry for Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry of Science, Research and Economy (bmwfw), the Austrian Research Promotion Agency (FFG), the Province of Styria and the Styrian Business Promotion Agency (SFG).

REFERENCES

AVL (2018). Model.CONNECT™, the neutral model integration and co-simulation platform connecting virtual and real components. (https://www.avl.com/-/model-connect). [Online; accessed 31.01.2019].

Benedikt, M., Bernasch, J., Holzinger, F., and Watzenig, D. (2016). Method for configuration a co-simulation for a total system. EP3188053 (A1) — 2017-07-05.

Benedikt, M. and Drenth, E. (2018). Relaxing stiff system integration by smoothing techniques for non-iterative co-simulation. In *IUTAM Symposium on Co-Simulation and Solver-Coupling*.

Benedikt, M. and Hofer, A. (2013). Guidelines for the application of a coupling method for non-iterative co-simulation. In *Modelling and Simulation (EUROSIM), 2013 8th EUROSIM Congress on*, pages 244–249.

Benedikt, M. and Holzinger, F. R. (2016). Automated configuration for non-iterative co-simulation. In *2016 17th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*, pages 1–7.

Benedikt, M., Stippel, H., and Watzenig, D. (2010). An adaptive coupling methodology for fast time-domain distributed heterogeneous co-simulation. In *SAE 2010 World Congress and Exhibition*. SAE International.

Benedikt, M., Watzenig, D., Zehetner, J., and Hofer, A. (2013). Macro-step-size selection and monitoring of the coupling errof for weak coupled subsystems in the frequency-domain. In *Proceedings*, pages 1–12. .

Blochwitz, T., Otter, M., Åkesson, J., Arnold, M., Clauss, C., Elmqvist, H., Friedrich, M., Junghanns, A., Mauss, J., Neumerkel, D., Olsson, H., and Viel, A. (2012). Functional mockup interface 2.0: The standard for tool independent exchange of simulation models. In *Proceedings of the 9th International Modelica Conference*, pages 173–184. The Modelica Association.

Busch, M. and Schweizer, B. (2011). An explicit approach for controlling the macro-step size of co-simulation methods. In Bernadini, D., editor, *Proceedings of the 7th European Nonlinear Dynamics Conference (ENOC 2011): July 24 - 29, 2011, Rome, Italy*, pages 1–6.

Holzinger, F. R. and Benedikt, M. (2019). *Hierarchical Coupling Approach Utilizing Multi-Objective Optimization for Non-Iterative Co-Simulation*, volume 1348, pages 735–740.

Kübler, R. and Schiehlen, W. (2000). Modular simulation in multibody system dynamics. *Multibody System Dynamics*, 4.