

Hierarchical Coupling Approach Utilizing Multi-Objective Optimization for Non-Iterative Co-Simulation

Franz Rudolf Holzinger¹ Martin Benedikt¹

¹Department Electric/Electronics & Software, VIRTUAL VEHICLE Research Center, Austria,
{franzrudolf.holzinger,martin.benedikt}@v2c2.at

Abstract

A hierarchical scheduling approach for non-iterative co-simulation is presented. With an increasing number of subsystems the number of possible combinations and permutations increases dramatically, resulting in an unsolvable problem to define a proper co-simulation scheduling for application engineers. This paper shows an approach to get an optimal trade-off between simulation duration and simulation accuracy by the usage of a multi-objective optimization approach to find an optimal scheduling for hierarchical co-simulation.

Keywords: hierarchical co-simulation, co-simulation graph, multi-objective optimization

1 Introduction

Virtualization of products is common practice in industry in order to reduce costs in design, analysis and test phases. Tailored simulation tools are available for covering the different engineering domains and applications. However, when it comes to overall system considerations the different subsystems must be virtually integrated for enabling analysis of their interactions. In contrast to a time-consuming remodeling, by the co-simulation approach the individual subsystems are simulated within their dedicated simulation tools and predefined coupling variables are exchanged at specific points in time for synchronization purposes (Kübler and Schiehlen, 2000).

First activities in the field of co-simulation were published in the 1970ies and 1980ies, motivated by the idea of massive parallelization for electrical circuit simulation in analog system design (Lelarsmee et al., 1982). Algebraic constraints and non-linear behaviours of the components require implicit numerical solvers for these applications, which were directly implemented within the dedicated simulation environments. Nowadays, a lot of domain-specific simulation tools are available on the market, some of them are equipped with dedicated API's (application programming interfaces) for enabling an integration in terms of co-simulation. Most of the tools only supports the exchange of coupling variables at predefined points in time (no iterations over steps; no exchange of model information), which renders the co-simulation approach to a pure explicit numerical scheme in general. The FMI Standard (Functional Mockup Interface (Blochwitz

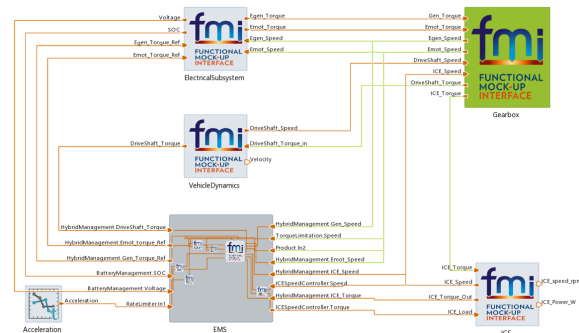


Figure 1. Co-simulation topology of the HEV example.

et al., 2012)) represents a promising path for enabling broadly horizontal subsystem integration and the application of implicit co-simulation master algorithms. By considering a system simulation in general, the individual subsystems typically possess different dynamic ranges and properties, which makes the use of different step-sizes, kinds of extrapolation and orders of execution relevant. With the increasing number of integrated subsystems an engineer is typically not able to configure the co-simulation master as required for ensuring stable and accurate co-simulation results. But, as more information about the individual subsystems is available in beforehand or may be gathered online, i.e. during the co-simulation run, as more automation is possible in order to support the user in configuration of co-simulation settings. Recently this idea was discussed (Benedikt and Holzinger, 2016) by the authors; within this contribution especially the aspect of subsystem scheduling is addressed. Scheduling relates the proper selection of the order of execution of the involved subsystems, i.e. the determination of the optional trigger-sequence.

The outline of the paper is as follows. Coupling mechanism within non-iterative co-simulation are introduced. Based on this, the trigger sequence of sub-models is discussed. After that a multi-objective optimization approach is presented to determine a scheduling configuration for hierarchical co-simulation. An example is used to illustrate the different coupling configurations as well as the multi-objective approach.

The topology of the co-simulation example is shown in Figure 1. It represents a hybrid electric vehicle (HEV).

Table 1. Subsystem of the HEV example.

	<i>Sub-Model</i>	<i>Calc. Effort d_j</i>
S1	Electrical Subsystem	17.3%
S2	EMS Battery Management	1.9%
S3	EMS Hybrid Management	4.9%
S4	EMS ICE Speed Controller	2.3%
S5	EMS Product	1.9%
S6	EMS Rate Limiter	2.0%
S7	EMS Torque Limitation	1.8%
S8	Gearbox	27.8%
S9	ICE	11.4%
S10	Vehicle Dynamics	28.7%

The HEV system is based on a Matlab/Simulink example and was split into 10 subsystems (Miller, 2017). The subsystems, compiled as FMUs, are integrated within a co-simulation framework (Model.CONNECT™, R2018a). Table 1 represents the sub-elements of the HEV example with their calculation effort d_j .

2 Coupling Mechanism

The most common used coupling approach for co-simulation is to calculate all sub-models at the same time. Each subsystem has not to wait for each other and so this coupling mechanism has the best simulation performance. Nevertheless, this parallel coupling approach causes the most coupling errors, due to the high number of extrapolated inputs. If subsystems are calculated sequentially, i.e. a subsystem starts the calculation when the previous subsystems already finished the calculation step, no inputs have to be extrapolated. With a sequential coupling approach a minimum number of extrapolation can be reached, but the simulation performance will suffer. A hierarchical approach on the other side allows a combination of sequential and parallel scheduling. Several subsystems can be nested, where e.g. the subsystems within a group are calculated in sequential order and the several subsets (groups) are calculated in parallel.

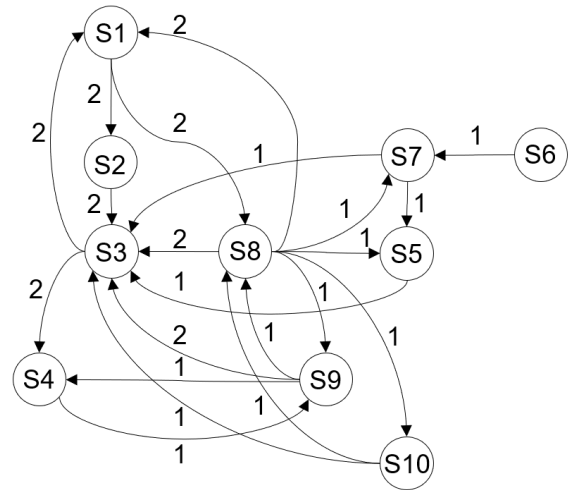
2.1 Simulation Performance

The simulation performance referenced by the real-time behaviour for different coupling mechanism related to the HEV example is shown in Table 2. The real-time factor (*RTF*) describes the relation between simulation duration and the wall-clock time. A real-time factor $RTF < 1$ means faster than wall-clock time and $RTF > 1$ means a simulation duration greater than the wall-clock time. For real-time applications it is required, that the $RTF < 1$ in each coupling time step, otherwise the real-time behaviour is not ensured.

The performance results in Table 2 show, that the real-time behaviour of the parallel coupling approach is lower than real-time. On the other hand, the sequential

Table 2. Real-time capability regarding the coupling mechanism.

<i>Coupling Mechanism</i>	<i>RTF</i>
<i>parallel</i>	0.44
<i>sequential</i>	1.1
<i>hierarchical</i>	0.5 – 0.75


Figure 2. Co-simulation graph of the HEV example.

coupling approach has a real-time factor greater than one, i.e. it is not real-time capable. From the timing point of view a parallel coupling approach represents a reasonable co-simulation scheduling setting.

2.2 Trigger Sequence

The calculation order or trigger sequence of a sequential coupling mechanism defines the extrapolated inputs and so the induced extrapolation errors.

Related to HEV example with $n = 10$ subsystems it exists $n! = 3628800$ different permutations to set the calculation order of the subsystems. With the knowledge of the topology it is possible to find at least an optimal trigger sequence with respect to a minimal number of extrapolations (Glumac and Kovacic, 2018).

Co-simulation networks can be interpreted as a directed graphs. The nodes of the graph represent the several subsystems and the edges describe the directed dependences of the subsystems. The weight of the edge describes the strength of the dependency e.g. the individual number of connections between the subsystems.

Figure 2 illustrates the co-simulation graph of the HEV example with 10 subsystems. The edges show the directed dependency of the subsystems and the weight of the edges represent the number of signals which are exchanged between the subsystems.

The trigger sequence can be interpreted as a Hamilto-

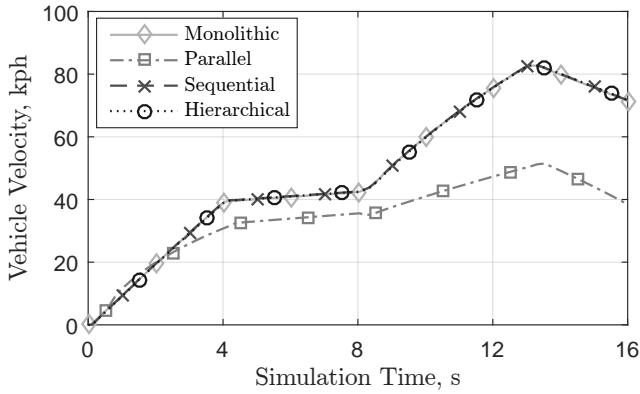


Figure 3. Simulation result (Vehicle Velocity) regarding different coupling mechanisms.

nian cycle, where each node has to be visited once. The weights of the edges represent the number of inputs to be extrapolated. The shortest way to visit all nodes represents the optimal trigger sequence with respect to minimal number of to be extrapolated inputs. Nevertheless, the connections of already visited nodes do not need an extrapolation and so the weights of these edges (connections) becomes zero.

A comparison of sequential and parallel coupling approach (in contrast to the monolithic simulation) is shown in Figure 3. The sequential simulation delivers almost the same results than the monolithic simulation. The results of the parallel coupling approach clearly differ from the reference.

3 Optimal Hierarchical Approach

Parallel scheduling turns out the best behaviour with respect to the simulation duration but induces, on the other hand, the most coupling errors into the co-simulation. A minimal number of extrapolated inputs can be reached by a sequential coupling approach at an expense of the simulation duration. A hierarchical coupling approach allows to find an optimal trade-off between simulation duration and accuracy.

The permutations of the subsystems with sequential method enlarges by the number of possible combinations. An upper estimate of possible combinations in consideration of the simple nested scheme (parallel scheduling of sequential calculated groups) is as follows:

$$\sum_{j=0}^{n-1} \binom{n-1}{j} n! = 2^{n-1} n!. \quad (1)$$

For a number of $n = 10$ subsystems there are almost $1.8 \cdot 10^9$ possible combinations and permutations for setting the calculation order. This huge number makes it practically impossible for co-simulation application engineers to find a proper set without any automated optimization approach.

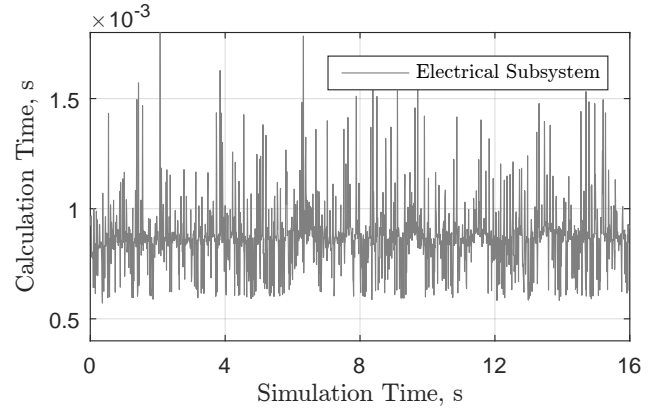


Figure 4. Calculation time of the Electrical Subsystem.

Beforehand some assumptions have to be applied:

- The calculation time (the time which the subsystem needs to compute the results) is significantly bigger than the synchronisation time (time which is needed to exchange the coupling signals). So the synchronisation time is unattended.

Note: For a high number of coupling signals or fast calculating subsystems, the synchronisation time is not negligible. In this case both, the synchronisation time and the calculation time have to be considered.

- The calculation effort has an ergodic behaviour. Figure 4 shows the calculation duration of the model *Electrical Subsystem* (S1) during the simulation. The mean computation duration is about 9ms and is almost constant throughout the simulation.

- The coupling error is directly associated to the number of extrapolations and degrades overall co-simulation accuracy.

Note: In general this is not the case. The introduced coupling error depends on the subsystem dynamic as well as the coupling signal behaviour. Nevertheless, if no additional subsystem information is available and the system behaviour is unknown, the introduced coupling error can be assumed equal for each input.

The relative calculation effort of the several subsystems is illustrated in Table 1. The *Vehicle Dynamics* (S10), *Gearbox* (S8) and the *Electical Subsystem* (S1) together need about 75% of the computation time. It is obvious to combine several subsystems and calculate the different sets in a parallel way, to reduce the overall calculation effort.

A combination set $C = \{C_1, C_2, \dots, C_i\}$ consists of several subsets C_j and includes all subsystem indices. The duration D_j of a subset C_j is calculated as follows:

$$D_j(C_j) = \sum_{i \in C_j} d_i, \quad (2)$$

where $d_i \in \mathbb{R}$ is the calculation effort of the individual subsystems S_i (see Table 1). The objective function J_D concerning the calculation effort can be written as the maximum duration of all subsets scaled by the sum of all elements:

$$J_D = \frac{1}{\sum_i d_i} \max_j D_j. \quad (3)$$

With respect to the HEV example a reasonable combination C consists of the subsets $C_1 = \{6, 10\}$, $C_2 = \{1, 2, 3, 8\}$ and $C_3 = \{4, 5, 7, 9\}$. Based on the effort of the combinations $D_1 = 30.7$, $D_2 = 51.9$ and $D_3 = 17.4$ a cost value $J_D = 0.519$ is determined.

For each possible subset C_j an optimal calculation order can be found, where the number of extrapolated inputs is minimized. Therefore the adjacency matrix A is used.

$$A = \begin{bmatrix} - & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 2 & - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & - & 0 & 1 & 0 & 1 & 2 & 2 & 1 \\ 0 & 0 & 2 & - & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & - & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & - & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & - & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & - & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & - & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & - \end{bmatrix} \quad (4)$$

The adjacency matrix A in (4) represents the graph of the HEV example (see Figure 2). The number of required extrapolated inputs, which is directly associated to the simulation accuracy, can be calculated as follows:

$$e_k = \sum_{i=1}^N A_{ik}. \quad (5)$$

Beginning with one element of the subset $k \in C_j$ the whole column is summed up. This represents the whole number of extrapolated inputs for the index. Regarding to the HEV example for an index $k = 3$ the whole column is summed up and $e_3 = 4$.

In contrast to the calculation effort, the number of extrapolated inputs is depending on the execution order of the subsystems. Therefore the entire row of the considered index k (or subsystem) has to be set to zero.

$$A_{ki} = 0, \quad i \in C_j \quad (6)$$

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

$$A = \begin{bmatrix} - & 0 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 2 & - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & - & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & - & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & - & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & - & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & - & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & - & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & - \end{bmatrix} \quad (7)$$

The extrapolation effort E_j of the subset C_j is given as sum of the required to be extrapolated inputs.

$$E_j(C_j) = \sum_{i \in C_j} e_i \quad (8)$$

The normalized objective function regarding the number of extrapolations can be written as follows:

$$J_E = \frac{1}{\sum_j \sum_i A_{ij}} \sum_j E_j. \quad (9)$$

A multi-objective optimization problem with minimization of number of extrapolated inputs and minimization of the simulation duration can finally formulated as follows:

$$\min \{(1-w)J_E + wJ_D\}, \quad (10)$$

where the factor w enables to set the focus of the optimization to the extrapolation error J_E or to the calculation duration J_D . A small factor w weights the optimization in the direction of the minimum extrapolation error and so sequential calculation is preferred. On the other hand

Table 3. Optimized hierarchical scheduling with respect to weighting factor w .

w	Trigger Sequence
0.0	(S6),S10,S9,S8,S7,S5,S3,S4,S1,S2
0.25	(S6),S10 S9,S8,S7,S5,S3,S4,S1,S2
0.5	(S6),S10 S9,S8,S7,S5,S3,S4 S1,S2
0.75	(S6),S10 S9 S8 S7 S5,S3,S4,S1,S2
1.0	(S6),S10 S9 S8 S7 S5,S4,S3,S2,S1

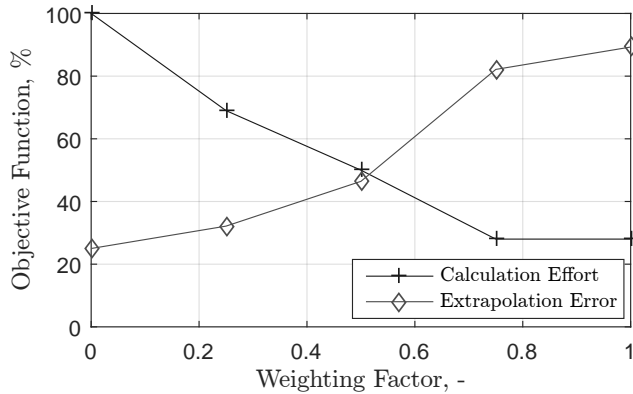


Figure 5. Relative calculation effort and relative extrapolation error regarding weighting factor w .

the factor $w = 1$ is the focus on the optimization of the simulation duration and so parallel approach is selected.

The solutions of the hierarchical optimization problem for the HEV example is shown in Table 3. The optimization is analysed by different weighting factors $w = [0, 0.25, 0.5, 0.75, 1]$. The order of the subsystems in a row indicates the trigger sequence. The several rows within the w cases mean the parallel calculation of these bundles. In the case of $w = 0.5$ the combined models $S6, S10$ are calculated parallel to the sequential calculated group $S9, S8, S7, S5, S3, S4$ and $S1, S2$. The subsystem $S6$ is considered separately and therefore in Table 3, it is written in parentheses because it has no dependencies on other subsystems.

The overall simulation duration increases with increasing w and, on the other hand, the number of extrapolated inputs decreases. The behaviour of the extrapolation error and the calculation effort regarding the weighting factor w for the HEV example is shown in Figure 5. A proper trade-off between simulation duration and accuracy for this example is at $w = 0.5$.

The comparison of the different coupling approaches is illustrated in Figure 6. The sequential and hierarchical ($w = 0.5$) approach are equal to the monolithic simulation result. The result of the parallel scheduling shows a different behaviour.

The HEV example in Table 3 shows the identical results regarding the calculation order for different weighting factors. Only the parallelization of subsystems changes dependent on w , because in contrast to the optimization part regarding the extrapolation error J_E , the calculation effort J_D is not dependent on the execution order. Therefore it is conceivable that the execution order is calculated first and after that the parallelization is determined. This will at least reduce the computation effort to find an optimal trigger sequence.

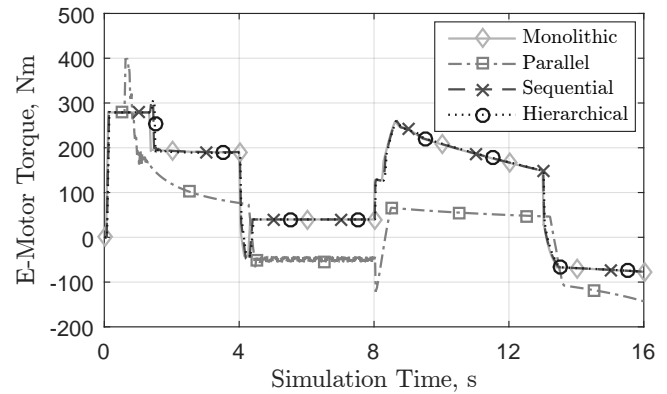


Figure 6. Simulation result (Electric Motor Torque) regarding different coupling mechanisms.

4 Conclusion and Future Work

Especially with increasing number of subsystems hierarchical co-simulation can bring a balance between simulation duration and simulation accuracy. A high number of subsystems makes it almost impossible to find a proper configuration set regarding the execution order. The HEV example in this work shows that the simulation duration of parallel scheduling is better than sequential coupling but the simulation result on the other hand differs significantly from the monolithic simulation caused by the introduced extrapolation errors. The presented hierarchical approach allows a real-time capable simulation with accurate simulation results. In the discussed HEV example all subsystems have the same coupling time-steps. For future works hierarchical co-simulation with different coupling time-steps is considered. Therefore subsystems with similar dynamics and coupling time-steps are clustered together. So they can be interpreted as a separate hierarchical layer.

Acknowledgements

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

- M. Benedikt and F. R. Holzinger. 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, April 2016. doi:10.1109/EuroSimE.2016.7463355.

T. Blochwitz, M. Otter, J. Åkesson, M. Arnold, C. Clauss, H. Elmqvist, M. Friedrich, A. Junghanns, J. Mauss, D. Neumerkel, H. Olsson, and A. Viel. 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, 2012. ISBN 978-91-7519-826-2. URL <http://dx.doi.org/10.3384/ecp12076173>.

Slaven Glumac and Zdenko Kovacic. Calling sequence calculation for sequential co-simulation master. In *Proceedings of the 2018 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation, SIGSIM-PADS '18*, pages 157–160, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5092-1. doi:10.1145/3200921.3200924. URL <http://doi.acm.org/10.1145/3200921.3200924>.

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

E. Lelarasmee, A. E. Ruehli, and A. L. Sangiovanni-Vincentelli. The waveform relaxation method for time-domain analysis of large scale integrated circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1(3):131–145, July 1982. ISSN 0278-0070. doi:10.1109/TCAD.1982.1270004.

Steve Miller. Hybrid-electric vehicle model in simulink. (<https://www.mathworks.com/matlabcentral/fileexchange/28441>) MATLAB Central File Exchange, 2017. Accessed: 2017-08-30.

Model.CONNECT™. Hybrid electric vehicle (hev). (<https://www.avl.com/-/model-connect>), R2018a. Accessed: 2018-04-11.