CHALMERS

Co-Simulation in Virtual Verification of Vehicles with Mechatronic Systems

WEITAO CHEN

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2019

Thesis for the degree of Licentiate of Engineering
in
Machine and Vehicle Systems

Co-Simulation in Virtual Verification of Vehicles with Mechatronic Systems

WEITAO CHEN

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2019

Co-Simulation in Virtual Verification of Vehicles with Mechatronic Systems
WEITAO CHEN

© WEITAO CHEN, 2019

THESIS FOR LICENTIATE OF ENGINEERING no 2019:04

Department of Mechanics and Maritime Sciences
Chalmers University of Technology
SE–412 96 Göteborg
Sweden
Telephone: +46 (0)31–772 1000

Chalmers Reproservice
Göteborg, Sweden 2019

Co-Simulation in Virtual Verification of Vehicles with Mechatronic Systems

Weitao Chen

Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

In virtual verification of vehicle and mechatronic systems, a mixture of subsystems are integrated numerically in an offline simulation or integrated physically in a hardware-in-loop (HIL) simulation. This heterogeneous engineering approach is crucial for system-level development and widely spreads with the industrial standard, e.g. Functional Mock-Up Interface (FMI) standard. For the engineers, not only the local subsystem and solver should be known, but also the global coupled dynamic system and its coupling effect need to be understood. Both the local and global factors influence the stability, accuracy, numerical efficiency and further on the real-time simulation capability.

In this thesis, the explicit parallel co-simulation, which is the most common and closest to the integration with a physical system, is investigated. In the vehicle development, the vehicle and the mechatronic system, e.g. an Electrcial Power Assisted Steering (EPAS) system can be simulated more efficiently by a tailored solver and communicative step. The accuracy and numerical stability problem, which highly depends on the interface dynamics, can be investigated similarly in the linear robust control framework. The vehicle-mechatronic system should be coupled to give a smaller loop gain for robustness and stability. Physically, it indicates that the splitting part should be less stiff and the force or torque variable should be applied towards the part with a higher impedance in the force-displacement coupling. Furthermore, to compensate the troublesome low-passed and delay effect from the coupling, a new coupling method based on $\mathcal{H}_\infty$ synthesis is developed, which can improve the accuracy of co-simulation. The method shows robustness to the system dynamics, which makes it more applicable for a complex vehicle-mechatronic system.

Keywords: Vehicle and mechatronic system, explicit parallel co-simulation, coupling method, error and stability, linear robust control theorem

Acknowledgements

"Who really cares about simulation? Let's create things to make cars fast and intelligent," this is my slogan when I started my PhD. This thesis would never have happened without the support from many people in this tough journey. I would like to express my sincere gratitude to my supervisors Professor Bengt Jacobson and Dr. Shenhai Ran. Bengt, your curiosity, patience and working attitude inspired me on my way. Shenhai, thank you for the discussions on hard-core problems and the happy time traveling together as a friend.

Secondly, I would like to appreciate Adjunct Associate Professor Matthijs Klomp and Associate Professor Mathias Lidberg, who started the project and guided me at the initial stage. I have also received helps from: Adjunct Professor Fredrik Bruzelius, who is a solution toolbox when I got stuck; Mr. Edo Drenth, who opened the door of co-simulation world to me; Professor Stig Larsson, supported me to supervise a mathematical master thesis.

Next, at Volvo Cars, I would like to thank my manager Carl Sandberg, my previous and current colleagues at the entire Vehicle Dynamics CAE team. Thanks Alejandro Gonzáles for sharing the depression from simulation problems. Thanks to Canhui Wu for a great thesis work under a high-pressure.

Also, I am thankful to all my colleagues and my fellow PhD students in the division of VEAS for a friendly working environment. Thank you for digging me out from my small cave for a beer or coffee, and moreover tolerating my anti-social behavior. Special thanks to Simone, Sonja and Britta for great management and to my vehicle dynamics brothers and sister: Adi, Anton, Pär, Toheed, Tushar and Juliette.

I gratefully acknowledge the financial support and colleagues from the ITEAM project by the European Union Horizon 2020 program.

Last but not the least, I would like to thank my parents and my beloved Jingqian, without your encouragement I would have definitely given up and not be able to arrive at this stage.

Thesis

The appended papers are:

I. Chen, W., Ran, S. and Jacobson, B., *Design of Interface in Co-Simulation for Electric Power Assisted Steering System Development.* In: Proceedings of the 14th International Symposium on Advanced Vehicle Control (AVEC), Beijing, China, 2018.

II. Chen, W., Ran, S. and Jacobson, B., *Integration and Analysis of EPAS and Chassis System in FMI-based Co-Simulation.* In: Proceedings of the 13th International Modelica Conference, Regensburg, Germany, number 157, 2019.

III. Chen, W., Ran, S. and Jacobson, B., *On Explicit Co-Simulation Approach: Analysis and Improved Coupling Design Based on $\mathcal{H}_\infty$ Synthesis*, journal to be submitted in 2019.

IV. Chen, W., Ran, S., Klomp, M. and Jacobson, B., *Real-time Co-Simulation Method Study for Vehicle Steering and Chassis System.* In: Proceedings of the 15th IFAC Symposium on Control in Transportation Systems CTS, Savona, Italy, pp. 273-278, 2018.

The author of this thesis was responsible for modeling, simulation, initiating the ideas and writing the papers. The co-authors contributed with supervision and review of the manuscripts. The idea behind paper IV is initiated by Professor Matthijs Klomp and implemented by the author.

Other related work but not appended:

I. Chen, W., Klomp, M., Ran, S. and Lidberg, M.,*Design and control of the steering torque feedback in a vehicle driving simulator.* Proceedings of the 25th International Symposium on Dynamics of Vehicles on Roads and Tracks (IAVSD 2017), Rockhampton, Queensland, Australia, 2017.

Table of Contents

1 Introduction **1**
- 1.1 Background and Motivation . . . 1
- 1.2 Formulation of Research . . . 3
- 1.3 Contribution . . . 4
- 1.4 Thesis Outline . . . 5

2 Co-simulation fundamentals **7**
- 2.1 Introduction of Coupled Dynamic System . . . 7
  - 2.1.1 Calculation Schemes . . . 8
  - 2.1.2 Coupling Configurations . . . 10
  - 2.1.3 The Algebraic Loop . . . 13
- 2.2 Stability of Co-simulation . . . 14
  - 2.2.1 Numerical Analysis Framework . . . 14
  - 2.2.2 Control Theory Framework . . . 16
- 2.3 Error Analysis of Co-simulation . . . 16
  - 2.3.1 Local error . . . 17
  - 2.3.2 Global error . . . 18
  - 2.3.3 Error from the solver . . . 19

3 Virtual vehicle architecture **21**
- 3.1 Introduction . . . 21
  - 3.1.1 Chassis Model . . . 21
  - 3.1.2 EPAS Model . . . 21
- 3.2 Coupling Layout . . . 22

4 Summary **23**
- 4.1 Conclusions . . . 23
- 4.2 Limitations . . . 24

4.3 Future Work 24

Bibliography **25**

INCLUDED PAPERS

Chapter 1

Introduction

1.1 Background and Motivation

The automotive industry is evolving with a rapid trend of electrification and automation. In 2017, over 1 million new electric vehicles have been sold worldwide with a notable annual growth of 54% and 6.3% of new cars are electric in Sweden [7]. To meet the consumer demand and fulfill the policy requirements, original equipment manufacturers (OEMs) scaled up their investments on developing electrified vehicles (EVs), which includes battery electric vehicles (BEVs) and plug-in hybrid vehicles (PHEVs). According to the survey [17], to obtain profitability OEMs need greater agility and efficiency to develop new techniques (EV, automation) and the conventional platform simultaneously. Moreover, vehicle itself becomes increasingly complex with more mechatronic systems than ever, which is highly desirable to be evaluated at early design phase.

To shorten the time to market, save the development cost and enable more synergies between functions, virtual verification is widely used in automotive industry. It emerges with an increasing number of mechatronic and functional subsystems so that they can be verified in an earlier phase of the V-process for development [27]. For a vehicle-level verification, different subsystems are integrated in a heterogeneous simulation environment, e.g. model-in-the-loop (MIL), software-in-the-loop (SIL), hardware-in-the-loop (HIL) and driver-in-the-loop (DIL).

The verification tools are usually adopted in an ad-hoc manner, meanwhile their complexity grows. The integration problem of the heterogeneous

approach is not completely known but it is a primary concern for OEMs. Virtual verification, as a research subject, covers multidisciplinary fields including mathematics, computer science, system dynamics and control engineering [21][15], but they have limited sharing of findings. Academic researchers from a corresponding field may focus on the specific problem in each area. The complexity in application for engineers is that the multidisciplinary problems are bonded together. Therefore, only a systematic knowledge on the heterogeneous approach enables a full usage of the growing verification techniques. To the best knowledge of author, apart from the challenges on software and standardization, the main research focus and technical challenges in virtual verification can be classified as:

I. The integration problem of numerical systems by co-simulation, i.e. each model is solved separately and coupled by input-output variables. The simulation should be numerically stable, accurate and computationally efficient for a practical usage.

II. The integration problem of the numerical system and the physical system, e.g. the HIL or DIL simulation. The dynamic response of the controlled hardware should fulfill the verification requirement.

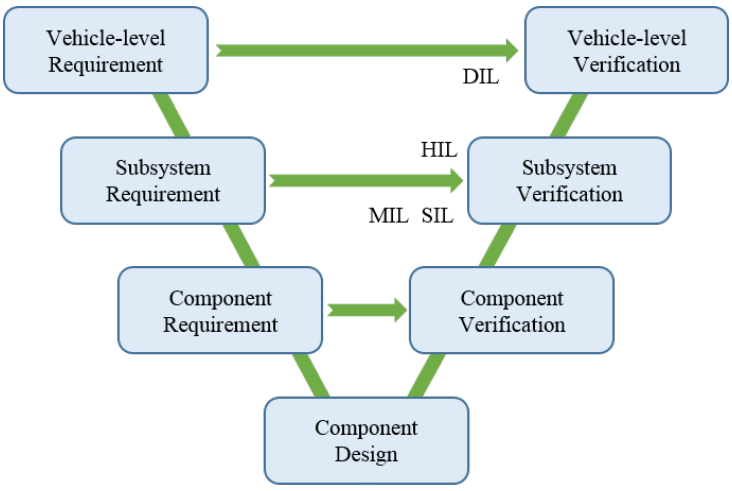


Figure 1.1: *The V-process for the development of vehicle and mechatronic systems.*

1.2 Formulation of Research

The primary research by author for Licentiate is on the co-simulation techniques for vehicle and mechatronic systems. Co-simulation is one of the cornerstone for virtual verification techniques which improves the numerical efficiency, enables a mixture of models of different fidelity levels and time scales (e.g. a vehicle model and an EPAS model). Furthermore, it allows a parallel computation in distributed processors for a real-time (HIL, DIL) and faster-than-real-time application. In fact, many subsystem models are provided from the suppliers in a white-box or black-box manner for intellectual property (IP) protection. Thus, an OEM has to integrate the models from multiple sources in different commercial software by co-simulation.

As an advanced simulation technique, co-simulation is widely used in aerospace, maritime and automotive engineering [26][4]. The formal research on issues of numerical stability, error analysis, master algorithms has been activated in the last decade [8][2][23][25][3]. Related research is growing with the need of the real-time simulation and the distributed simulation. The future solution for the simulation of vehicle with mechatronic systems might be parallelization by multiple CPUs and GPUs [11][13] rather than a monolithic approach.

The performance of co-simulation can be evaluated by the computational time, numerical stability and error. The computational time mainly depends on the dynamics of the model, the assigned solvers and how the computational burden is distributed. The associated drawbacks in numerical stability and error determine whether the simulation can proceed and whether the results are reliable. More specifically, the co-simulation performance depends on an interaction of the following factors:

- Coupling configuration: the decomposition and integration layout of the system.
- Marco-time step: the communication step of the coupled subsystem.
- Calculation scheme: the integration and communication sequence of each subsystem.

These factors will be discussed more in details in Chapter 2. Regarding these factors, the following research questions have been initiated:

I. How to integrate the vehicle and mechatronic systems to guarantee the correctness and robustness of co-simulation?

II. How to reduce the error, enhance stability and keep simulation speed fast?

1.3 Contribution

To answer the research questions, four papers have been written from different aspects. The contribution and findings are summarized as following:

- The numerical stability of an explicit parallel co-simulation is closely related to the stability of the closed-loop sampled-data system. The coupling configuration can result a different loop gain and influence the stability and robustness, which can be analyzed by the linear robust control theorem. More specifically, for a mechanical system it is desired to have a softer splitting interface with force/torque applied towards the heavier and stiffer parts, i.e. a higher impedance. *(Paper I, Paper III)*

- Co-simulation can accelerate the simulation speed by system partitioning and a relaxed communication step. A FMI-based co-simulation of vehicle and EPAS system is created as a workbench to verify the effects from different coupling configurations and co-simulation setup. The electric control unit (ECU) and mechanical part of a mechatronic subsystem are usually tightly coupled and chassis components with slower dynamics are more preferred for partitioning. *(Paper I, Paper II)*

- One extrapolation method is difficult to be the optimal for all interfaces of a complex engineering system. Therefore, robustness of the method is needed. A novel coupling design by $\mathcal{H}_\infty$ synthesis is invented with the objective to minimize the $\mathcal{L}_2$ norm of the coupling error. The method is robust and able to improve the accuracy. For real-time co-simulation on distributed machines, the delay effect can be compensated by adaptive filters. But this effect can also be potentially addressed by the recently developed $\mathcal{H}_\infty$ synthesis method. *(Paper III, Paper IV)*

1.4 Thesis Outline

The rest of the thesis comprises complementary chapters and appended papers. The objective of complementary chapters is to make the thesis more self-contained and to provide readers a more general vision on related techniques and theorems. They might dive deeply in the numerical algorithms and sometimes with an implementation difficulty. However, these techniques should be aware of in the virtual verification, especially in co-simulation research.

The thesis is organized as following. Chapter 2 provides basic information of co-simulation, fundamental analysis on numerical stability and error is given. Chapter 3 describes more specifically the vehicle and EPAS model and the co-simulation setup. In Chapter 4, conclusion, limitation and future plan is given.

Chapter 2

Co-simulation fundamentals

2.1 Introduction of Coupled Dynamic System

A model of vehicle-mechatronic system with high-fidelity is usually constituted by several multi-domain subsystems. The model of each subsystem can be coupled at three different levels [18] as illustrated in Figure 2.1. At the physical modeling level, the subsystems are modeled by a single tool or multi-domain modeling language such as Modelica [12]. At the equation level, the compiled mathematical equation codes are exported to another tool and solved by a single solver together, e.g. FMI model-exchange [6]. Alternatively, at the behavior trace level the subsystems are coupled by input-output data and co-simulation belongs to this case.

The dynamic system coupled at the behavior trace level is quite common in virtual verification techniques. In a wider sense, a system that is partially constituted by a surrogate model (e.g. a look-up table, neural-network for the vehicle suspension) is coupled by the behavior trace as well. Therefore, these systems can be analyzed similarly in the framework of co-simulation. One of their similarities is that these systems have an approximation error (by interpolation or extrapolation) at the coupling level. A distinction of co-simulation is that it has more complicated operations such as the iterative scheme in which a subsystem rollbacks to a previous time. On the contrary, a coupled physical system or a control algorithm only works forward in time in a parallel way. These features will be discussed in details in this chapter.

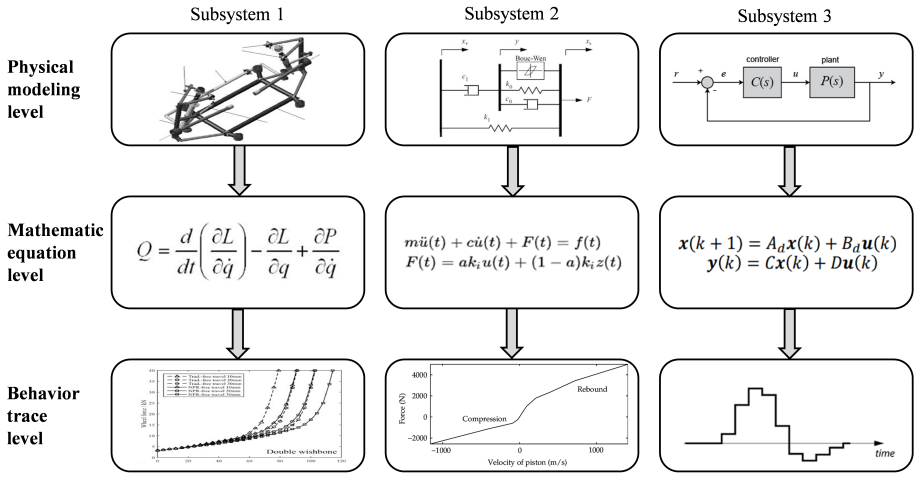


Figure 2.1: *A coupled dynamic system at different levels of coupling.*

2.1.1 Calculation Schemes

In co-simulation each subsystem is calculated by a local solver at a fixed or variable micro-step δt. The coupling by input-output communication occurs at every macro-step Δt and $\Delta t \geq \delta t$ (Figure 2.2). The calculation schemes can be classified according to the calculation sequences [8]:

- *Parallel scheme* or *Jacobi scheme*: each subsystem is calculated simultaneously. The input during the communicative interval is updated by extrapolation based on previous values.
- *Sequential scheme* or *Gauss-Seidel scheme*: the coupled subsystems are calculated in a specified sequence. The subsystem calculated *a priori* updates the input by extrapolation and its output is a prediction so that the subsystem calculated *a posteriori* can approximate the input by interpolation.
- *Iterative scheme*: the scheme starts with a *Parallel or Sequential scheme* then subsystems are calculated by iterations. Each subsystem can roll-back to the previous macro-step and the input can be interpolated from the previous iteration. The iterative process terminates until a specified error tolerance or iterative times are reached.

The calculation schemes differ drastically in terms of error, numerical stability and simulation time. In general, with a same size of Δt, the *Iterative*

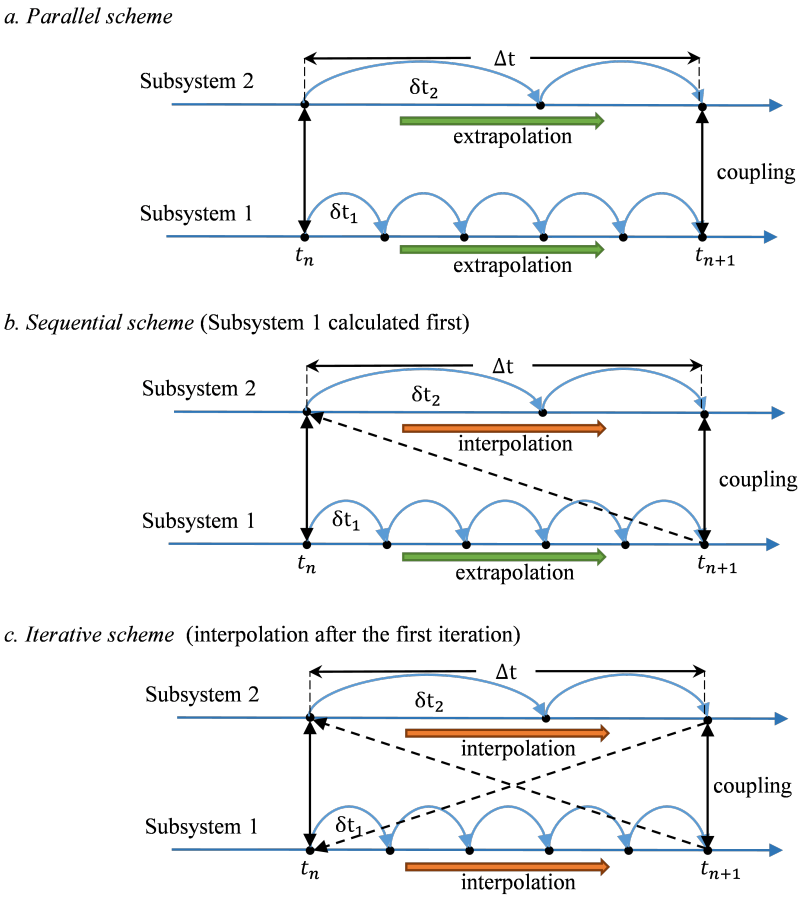


Figure 2.2: *Co-simulation of two coupled dynamic subsystems with different calculation schemes.*

scheme is most accurate and stable, whereas the *Parallel scheme* is the worst in stability and accuracy. This advantage is at a price of more computational burden: the *Iterative scheme* is the slowest and the *Parallel scheme* is the fastest, which makes the *Parallel scheme* ideal for the real-time application. Furthermore, a faster simulation speed enables a finer Δt to reduce the error. In a certain case, the *Parallel scheme* might be more precise than its alternatives, e.g. the *Sequential scheme*, if the comparison is according to the same simulation time instead of a same Δt [16]. To distribute the calculation burden and have a good balance on accuracy for a large-scale system, a mixture of those calculation schemes can be applied, namely the hierarchical

co-simulation.

The implementation difficulty of the calculation schemes should be concerned in practice. The *Parallel scheme* can be easily implemented with the commercial software. Advanced features of master-slave are required for the *Sequential scheme* and the *Iterative scheme*. First, the master is responsible to initialize the slave subsystems, vary macro-step size, specify the calculation sequence or control the iterative process. Second, the slave subsystem must be controllable by the master and capable to expose the required information, e.g. the internal states or *Jacobian* matrices for advanced master algorithms[3][25]. These requirements hinder the implementation with many commercial software and black-box subsystems.

Another interesting technique, which can be combined with different schemes and worth mentioning here, is the adaptive macro-step size control. Analogous to the variable step solver, this technique can adapt the macro-step size Δt according to the error estimation so that both the numerical efficiency and the accuracy can be improved. Several adaptive macro-step size control algorithms have been investigated by researchers [20][8][22].

To apply the advanced features, several software packages and platforms can be used such as *PyFMI*[2], *Daccosim* [14] and *FMIGo*[19], most of which are based on the FMI standard. For this research project, we only focus on co-simulation in the *Parallel scheme* at a fixed Δt. Due to the implementation reasons the *Parallel scheme* is more widely used in automotive industry and it shares more similarities to general cases such as SIL and HIL, where the sensor signals are sampled in a fixed step and the parallel nature of a physical system.

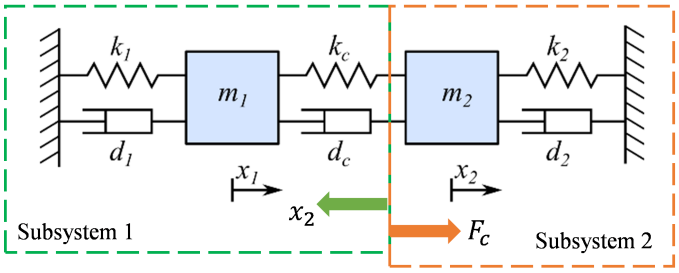
2.1.2 Coupling Configurations

The coupled dynamic system can have various configurations, i.e. how the system is partitioned and coupled. A dual mass-spring-damper system (Figure 2.3) is taken as an example to clarify the problem. It can represent a simple co-simulation case and facilitate analytical results.

Each subsystem with a mass can be coupled by the applied-force approach or by the algebraic constraint. For the applied-force approach, the configurations can be classified as:

- *Force-Displacement (FD) coupling*, the intermediate spring-damper is

a. Force-Displacement coupling



b. Displacement-Displacement coupling

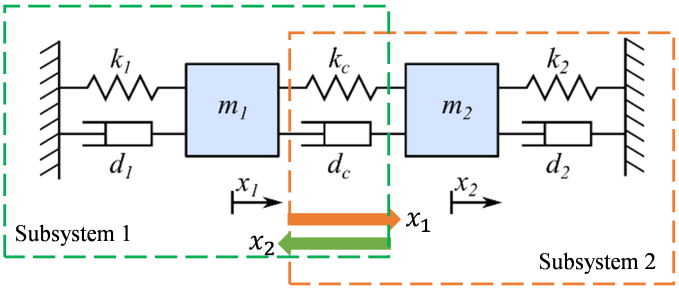


Figure 2.3: *The two subsystems coupled by the applied-force approach, the coupling force* $F_c = k_c(x_1 - x_2) + d_c(\dot{x}_1 - \dot{x}_2)$.

embedded in subsystem 1 and subsystem 2 has a force input F_c:

$$\begin{aligned} &\text{Subsystem 1} && \text{Subsystem 2} \\ &\dot{\boldsymbol{x}}_1 = f^{[1]}(\boldsymbol{x}_1, u_1) && \dot{\boldsymbol{x}}_2 = f^{[2]}(\boldsymbol{x}_2, u_2) \\ &y_1 = g^{[1]}(\boldsymbol{x}_1, u_1) && y_2 = g^{[2]}(\boldsymbol{x}_2) \\ &u_1 = y_2 && u_2 = y_1 \end{aligned} \tag{2.1}$$

- *Displacement-Displacement (DD) coupling*, both subsystems share the intermediate spring-damper with a displacement input:

$$\begin{aligned} &\text{Subsystem 1} && \text{Subsystem 2} \\ &\dot{\boldsymbol{x}}_1 = f^{[1]}(\boldsymbol{x_1}, u_1) && \dot{\boldsymbol{x}}_2 = f^{[2]}(\boldsymbol{x}_2, u_2) \\ &y_1 = g^{[1]}(\boldsymbol{x}_1) && y_2 = g^{[2]}(\boldsymbol{x}_2) \\ &u_1 = y_2 && u_2 = y_1 \end{aligned} \tag{2.2}$$

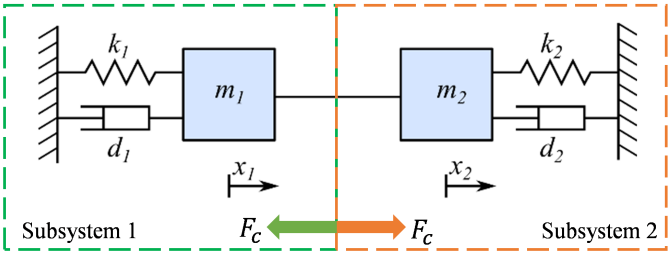
The stability and error properties of *FD coupling* and *DD coupling* have been investigated systematically by researcher [8]. For a same system, *DD coupling* is more accurate and stable than *FD coupling*.

The coupling configuration by the algebraic constraint is usually applied to the coupling of rigid bodies as illustrated in Figure 2.4, which involves *DD coupling* and *FF coupling* by the algebraic equation, in both cases the system can be written as:

$$\begin{array}{ll} \text{Subsystem 1} & \text{Subsystem 2} \\ \dot{\boldsymbol{x}}_1 = f^{[1]}(\boldsymbol{x}_1, u_1) & \dot{\boldsymbol{x}}_2 = f^{[2]}(\boldsymbol{x}_2, u_2) \\ y_1 = g^{[1]}(\boldsymbol{x}_1, u_1) & y_2 = g^{[2]}(\boldsymbol{x}_2, u_2) \\ u_1 = y_2 & u_2 = y_1 \end{array} \tag{2.3}$$

where a bi-directional dependency of input-output at the same instant exists and an algebraic loop problem needs to be solved.

a. Force-Force coupling by constraint



b. Displacement-Displacement coupling by constraint

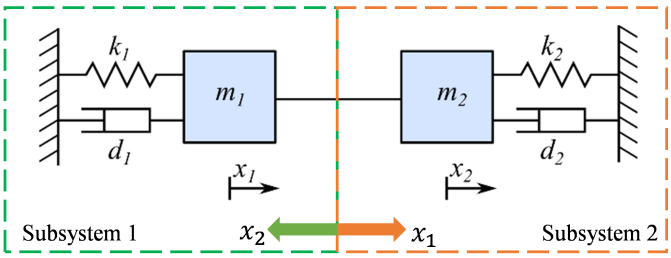


Figure 2.4: *The two subsystem coupled by the algebraic constraint.*

2.1.3 The Algebraic Loop

The algebraic loop is determined by the *feed-through* and *non feed-through* property of the subsystem. *Feed-through* connection has an output related to the input at the same instant, e.g. $y = g(t, x, u)$. If the output is not dependent on the input at the same instant, e.g. $y = g(t, x)$, then the connection is *non feed-through*.

When a closed-loop interconnection is constituted only by *feed-through* connections, an algebraic loop occurs which means there is no explicit calculation sequence. So for a system coupled by the algebraic constraint, efforts to handle the algebraic loop are needed. One can break the algebraic loop by simply adding a *non feed-through* connection, e.g. a delay or a low-pass filter. The drawback is the introduction of an additional dynamics to the original system. Alternatively, one can apply implicit methods [18], which has similar requirements on master-salve as discussed in Section 2.1.1.

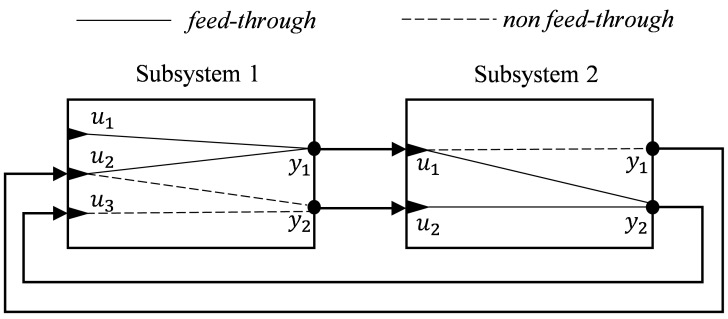


Figure 2.5: *MIMO systems can have both feed-through and non feed-through connections. There is no algebraic loop exists in this case.*

A multi-input multi-output (MIMO) system can have both *feed-through* and *non feed-through* connections as sketched in Figure 2.5. In practice, an algebraic loop may occur not only due to the numerical reason but also a software issue. Without an access to the source code or *feed-through* information of the subsystem, a *'fake'* algebraic loop problem may occur in the simulation environment.

2.2 Stability of Co-simulation

The numerical stability of co-simulation is the convergent property of the coupling error. In the numerical analysis framework, the stability property is indicated by the spectral radius of the difference equations. The co-simulation stability is also investigated as the closed-loop stability in the control theory framework by researchers [28][10]. In this section, stability analysis in both frameworks are discussed. The two analysis approaches are closely related and differ for a convenience of problem formulation.

2.2.1 Numerical Analysis Framework

For ease of analysis, a linear time-invariant (LTI) system is considered and the coupled dynamic system can be expressed by:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \\ u &= Ly \end{aligned} \tag{2.4}$$

where x is the state vector and L is the coupling matrix mapping the output vector y to the corresponding subsystem input u. In mono-simulation, the coupling $u = Ly$ suffices for all time, thus Equation 2.4 can be combined as:

$$\dot{x} = \big(A + BL(I - DL)^{-1}C\big)x \tag{2.5}$$

In co-simulation the coupling by matrix L occurs at every communicative instant t_n. So the coupled dynamic system becomes:

$$\begin{aligned} \dot{\tilde{x}}_n &= A\tilde{x}_n + B\Psi\big(\tilde{u}(\tau)\big) \\ \tilde{y}_n &= C\tilde{x}_n + D\Psi\big(\tilde{u}(\tau)\big) \\ \tilde{u}_n &= L\tilde{y}_n \end{aligned} \tag{2.6}$$

where Ψ is the extrapolation operator and $\tau \in [t_n, t_{n+1})$. The subsystem input $\Psi\big(\tilde{u}(\tau)\big)$ is updated by the extrapolation during a macro-step. In case a basic constant extrapolation method is applied:

$$\Psi\big(\tilde{u}(\tau)\big) = \tilde{u}_n, \quad \tau \in [t_n, t_{n+1}) \tag{2.7}$$

By assuming that the system is exactly solved by the solver, the updated state $\tilde{x}_{n+1}$ at next communicative instant t_{n+1} can be derived as:

$$\begin{aligned} \tilde{x}_{n+1} &= e^{A\Delta t}\tilde{x}_n + \int_{t_n}^{t_{n+1}} e^{A(t_{n+1}-\tau)} B\tilde{u}_n d\tau \\ &= e^{A\Delta t}\tilde{x}_n + K(\Delta t)BL\tilde{y}_n \\ \text{with} \quad K(\Delta t) &= \int_{t_n}^{t_{n+1}} e^{A(t_{n+1}-\tau)} d\tau \end{aligned} \tag{2.8}$$

Therefore, the difference equation of the coupled dynamic system (Equation 2.6) with a constant extrapolation (Equation 2.7) can be obtained:

$$\begin{bmatrix} \tilde{x}_{n+1} \\ \tilde{y}_{n+1} \end{bmatrix} = \underbrace{\begin{bmatrix} e^{A\Delta t} & K(\Delta t)BL \\ Ce^{A\Delta t} & CK(\Delta t)BL + DL \end{bmatrix}}_{A^*} \begin{bmatrix} \tilde{x}_n \\ \tilde{y}_n \end{bmatrix} \tag{2.9}$$

The numerical stability of co-simulation depends on the spectral radius $\rho(A^*)$. If $\rho(A^*) \leq 1$ and there is no more than one eigenvalue on the unit circle, the system is stable and the numerical result is convergent. Otherwise, the system is unstable. The equation shows that $\rho(A^*)$ is related to the following factors:

- Macro-step Δt.
- The extrapolation operator Ψ, $\rho(A^*)$ with different extrapolation method is shown in Figure 2.6.
- The system dynamics, i.e. matrices A, B, C, D and L which are based on the coupling configuration and system parameters.

The stable region $\rho(A^*) \leq 1$ is usually derived numerically with specified parameters. By this approach the numerical stability is always checked *a posteriori*. Due to a complex relation between $\rho(A^*)$ and system parameters which has a large amount of combinations. A clear and sound conclusion on the stability is difficult to make [24]. Therefore, the stability in the control framework might help to give more insights and engineering sense.

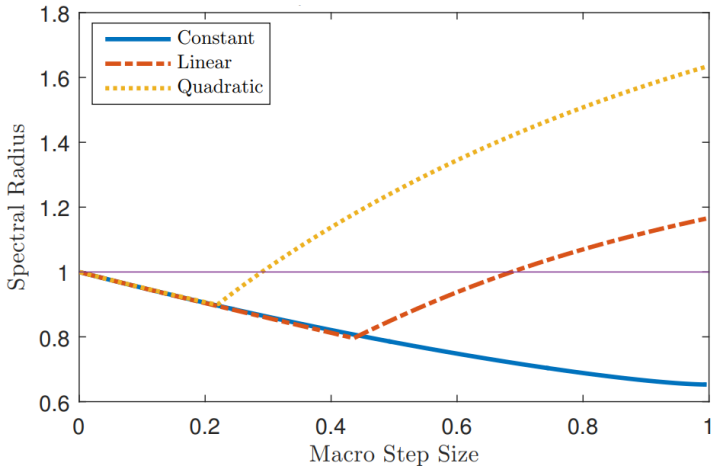


Figure 2.6: *The spectral radius of a co-simulation case with different extrapolation methods.*

2.2.2 Control Theory Framework

The coupled dynamic system, in the explicit parallel scheme with fixed macro-step Δt, is similar as the sample-data system in the digital control framework. Even though some terminologies in the related publications are different, the similarity is worth mentioning.

Same as the previous analysis, assuming that the solver is accurate enough, each LTI subsystem with zero initial condition can be represented by a transfer function. Thus, the coupled dynamic system can be emulated by a closed-loop sample-data system (Figure 2.7). The co-simulation numerical stability is closely related to the stability of the closed-loop interconnection, with the coupling error as multiplicative disturbances. The formulation and analysis has been discussed in more details in the appended *Paper III*. Based on this, Nyquist stability criterion and related theorems can be applied.

2.3 Error Analysis of Co-simulation

The numerical stability guarantee the convergence property. To understand the approximation order in co-simulation, error analysis is discussed in this section.

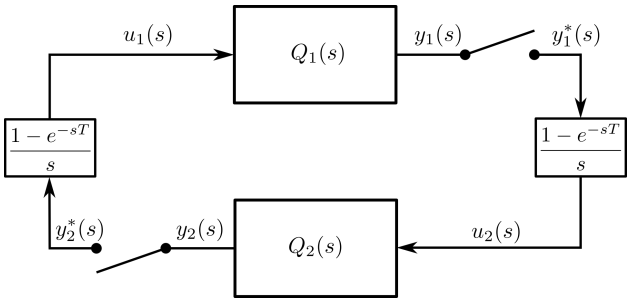


Figure 2.7: *The co-simulation emulated by a closed-loop sample-data system, constant extrapolation method has been taken.*

2.3.1 Local error

The system in Equation 2.4 is considered and same notations are used. To analyze the local error, i.e. the error in one-step numerical approximation, the state $\tilde{x}_{n-1}$ and input $\tilde{u}_{n-1}$ from the previous macro-step t_{n-1} are considered as error-free. Therefore, the local error ε_{x_n} can be derived by:

$$\begin{aligned}
\varepsilon_{x_n} &= x_n - \tilde{x}_n \\
&= e^{A\Delta t}x_{n-1} + \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}Bu(\tau)d\tau - e^{A\Delta t}\tilde{x}_{n-1} \\
&\quad - \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}B\Psi\big(u(\tau)\big)d\tau \\
&= \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}B\big(u(\tau) - \Psi(u(\tau))\big)d\tau \\
&\leq \big|B\int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}d\tau\big|\varepsilon_{u_n,max}
\end{aligned} \tag{2.10}$$

where $\varepsilon_{u_n,max} = \|u(\tau) - \Psi(u(\tau))\|_\infty, \tau \in [t_{n-1}, t_n)$, which is the maximum norm of input error during the communicative interval. The expression can be further expanded by Taylor series:

$$\begin{aligned}
\varepsilon_{x_n} &\leq \big|\frac{B}{A}(-I + e^{A\Delta t})\big|\varepsilon_{u_n,max} \\
&= \big|\frac{B}{A}(A\Delta t + \frac{(A\Delta t)^2}{2} + \frac{(A\Delta t)^3}{6} + ...)\big|\varepsilon_{u_n,max} \\
&= \big|B\mathcal{O}(\Delta t)\big|\varepsilon_{u_n,max}
\end{aligned} \tag{2.11}$$

When a k degree *Lagrange* extrapolation is used for the input, the local error ε_{x_n} is bounded by $|B|\mathcal{O}(\Delta t^{k+2})$. The output local error can be easily derived:

$$\begin{aligned} \varepsilon_{y_n} &\leq \left|C\varepsilon_{x_n} + D\varepsilon_{u_n}\right| \\ &\leq \left|C\mathcal{O}(\Delta t^{k+2}) + D\mathcal{O}(\Delta t^{k+1})\right| \end{aligned} \tag{2.12}$$

Therefore, for a *feed-through* system ($D \neq 0$), ε_{y_n} is bounded by $\mathcal{O}(\Delta t^{k+1})$, which gives an error order reduction. For a *non feed-through* system ($D = 0$), ε_{y_n} is bounded by $\mathcal{O}(\Delta t^{k+2})$. So in local error estimation, the *feed-through* and *non feed-through* conditions are considered separately [5].

2.3.2 Global error

The global error, i.e. the error by numerical approximation during the whole simulation time can be derived based on the local error and removing the error-free assumption:

$$\begin{aligned} \xi_{x_n} &= x_n - \tilde{x}_n \\ &= e^{A\Delta t}x_{n-1} + \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}Bu(\tau)d\tau - e^{A\Delta t}\tilde{x}_{n-1} \\ &\quad - \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}B\Psi(\tilde{u}(\tau))d\tau \\ &= e^{A\Delta t}\xi_{x_{n-1}} + \int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}B\big[u(\tau) - \Psi(u(\tau)) + \Psi(u(\tau)) - \Psi(\tilde{u}(\tau))\big]d\tau \\ &\leq e^{A\Delta t}\xi_{x_{n-1}} + \varepsilon_{x_n} + \Big|\int_{t_{n-1}}^{t_n} e^{A(n\Delta t-\tau)}B\big[\Psi(u(\tau)) - \Psi(\tilde{u}(\tau))\big]d\tau\Big| \end{aligned} \tag{2.13}$$

taking Ψ as a linear operator, $\Psi(u(\tau)) - \Psi(\tilde{u}(\tau)) = \Psi[u(\tau) - \tilde{u}(\tau)] = \mathcal{O}(\Delta t^{k+1})$. Equation 2.13 can be further written as:

$$\begin{aligned} \xi_{x_n} &\leq e^{A\Delta t}\xi_{x_{n-1}} + |B|(\mathcal{O}(\Delta t^{k+2}) + \mathcal{O}(\Delta t^{k+2})) \\ &\leq e^{A2\Delta t}\xi_{x_{n-2}} + |B|\mathcal{O}(\Delta t^{k+2})(1 + e^{A\Delta t}) \\ &\leq |B|\mathcal{O}(\Delta t^{k+2})(1 + e^{A\Delta t} + ... + e^{A(n-1)\Delta t}) \\ &= |B|\mathcal{O}\Delta t^{k+2})\frac{1 - e^{An\Delta t}}{1 - e^{A\Delta t}} = |B|\mathcal{O}(\Delta t^{k+1})(1 - e^{At_n}) \end{aligned} \tag{2.14}$$

then the output global error during the simulation time $[0, t_n]$ is:

$$\begin{aligned} \xi_{y_n} &= y_n - \tilde{y}_n \\ &= C\xi_{x_n} + D\xi_{u_n} \leq \mathcal{O}(\Delta t^{k+1}) \end{aligned} \tag{2.15}$$

So the global error of state and output are both in order of $O(\Delta t^{k+1})$ for *feed-through* and *non feed-through* systems.

2.3.3 Error from the solver

In previous analysis it is assumed that the approximation error from the solver can be neglected in co-simulation. In this section, the approximation error from the solver is considered. The local error ε'_{x_n} and global error ξ'_{x_n} are derived in a similar manner as before. If the subsystem is solved by a m-step numerical method:

$$\begin{aligned} \tilde{\tilde{x}}_n &= f(t_n; \tilde{\tilde{x}}_{n-m}, \tilde{\tilde{x}}_{n-m+1}, ..., \tilde{\tilde{x}}_{n-1}; \delta t) \qquad \text{explicit method} \\ \tilde{\tilde{x}}_n &= f(t_n; \tilde{\tilde{x}}_{n-m}, \tilde{\tilde{x}}_{n-m+1}, ..., \tilde{\tilde{x}}_{n}; \delta t) \qquad \text{implicit method} \end{aligned} \tag{2.16}$$

where δt is the solver integration step, i.e. the micro-step. The approximation error from a p order explicit (or implicit) method is well-known:

$$\begin{aligned} \varepsilon^n_{x_n} &= \tilde{x}_n - \tilde{\tilde{x}}_n \\ &= \tilde{x}_n - f(t_n; \tilde{x}_{n-m}, \tilde{x}_{n-m+1}, ..., \tilde{x}_{n-1}; \delta t) \end{aligned} \tag{2.17}$$

and the local error $\varepsilon^n_{x_n} = \mathcal{O}(\delta t^{p+1})$ and the global error $\xi^n_{x_n} = \mathcal{O}(\delta t^p)$. Thus, the local error of co-simulation is:

$$\begin{aligned} \varepsilon'_{x_n} &= x_n - \tilde{\tilde{x}}_n \\ &= x_n - \tilde{x}_n + \tilde{x}_n - \tilde{\tilde{x}}_n \\ &= \varepsilon_{x_n} + \varepsilon^n_{x_n} \\ &\leq \mathcal{O}(\Delta t^{k+2}) + \mathcal{O}(\delta t^{p+1}) \end{aligned} \tag{2.18}$$

By removing the error-free condition in previous steps, the global error of co-simulation can be derived:

$$\begin{aligned} \xi'_{x_n} &= x_n - \tilde{\tilde{x}}_n \\ &= x_n - \tilde{x}_n + \tilde{x}_n - \tilde{\tilde{x}}_n \\ &= \xi_{x_n} + \xi^n_{x_n} \\ &\leq \mathcal{O}(\Delta t^{k+1}) + \mathcal{O}(\delta t^{p}) \end{aligned} \tag{2.19}$$

It can be seen that the overall error is bounded by a summation of a coupling error in order $\mathcal{O}(\Delta t^{k+1})$ and an approximation error from the solver in order $\mathcal{O}(\delta t^{p})$. Considering that:

- $\Delta t \geq \delta t$, the macro-step is usually much larger than the micro-step.
- $p \geq k$, high order or multi-step numerical methods are quite common but the extrapolation of high degree is rarely used due to the drawback in stability.

one can conclude that the assumption to neglect the approximation error from the solver is quite reasonable and the coupling error is dominant which is the key to improve the co-simulation behavior.

Chapter 3

Virtual vehicle architecture

3.1 Introduction

The steering and handling requirement verification are mainly concerned in the project. The lateral dynamics of a vehicle should be accurately simulated by the chassis model. A realistic steering feel should be given by a high-fidelity EPAS model, which should capture the main mechanical characters to further enable a new conceptual control design [9]. The modeling work and co-simulation study is discussed in details in the appended *Paper II*.

3.1.1 Chassis Model

The steering response and the steering feel is simulated with a vehicle model on the flat road at different speeds. The lateral dynamics bandwidth is usually below 1.3 Hz which is relatively slow and a larger integration step might be taken by a variable step solver. However, the solver adaptability might be constrained if a fast dynamic mode is added to the model. Then the number of integration steps and simulation time will be increased.

3.1.2 EPAS Model

The EPAS system is driven by an electric motor through a transmission gear, which is a common structure of many mechatronic systems. The EPAS mechanical model requires a much smaller integration step ($\delta t < 1$ ms) than the chassis model due to:

- A large inertia (mass) ratio from the transmission gear effect.
- A under-damped behavior of the motor.
- The friction model, especially the numerically stiff dry friction.

An alternative way of friction modeling is by state event to avoid the stiff differential equation. However, the handling of event is needed for the appropriate solver and it might be challenging for real-time simulation [1].

3.2 Coupling Layout

The architecture of vehicle with an EPAS system is sketched in Figure 3.1. The subsystems are physically cascaded by feedback interconnections, e.g. *FD coupling*. The control variables, e.g. the vehicle speed and the torsion bar angle, are sent to the ECU.

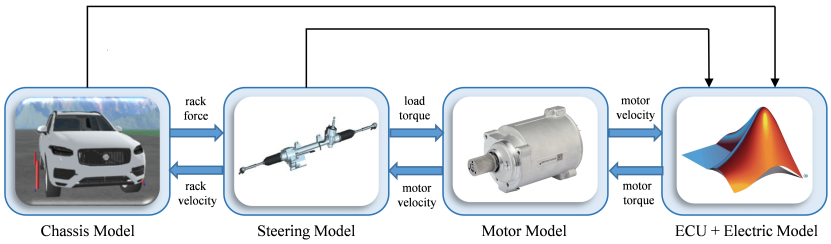


Figure 3.1: *A coupling layout of vehicle-EPAS system*

Herein, an outer loop is typically slower than an inner loop which makes the variable more robust to the effect of delay and approximation error. The layout of the coupled dynamic system should be specified according to the inner loop connections, which can be summarized as:

- Usually the lower-level controller, electric model and the mechanical part are tightly coupled. The coupling interface is more preferred in the mechanical domain with a less stiffness. (*Paper I* and *Paper II*)
- In the directional specification in *FD coupling*, force should be applied to the part with a larger impedance. So that the loop gain is smaller, which contributes to the robustness of closed-loop stability and the co-simulation numerical stability (*Paper III*).

Chapter 4

Summary

4.1 Conclusions

Co-simulation often improves the numerical efficiency of a simulation, enables the sharing and reusing of a model and further contributes to an agile system-level development. It is the most efficient and practical numerical method for the virtual verification of a complete vehicle with mechatronic systems. In an experimental case of vehicle-EPAS system, co-simulation can be 20 times faster than a monolithic approach without severe drawbacks on stability and accuracy.

To guarantee the numerical robustness, the coupling configuration and the coupling method need to be appropriately specified by the system engineers. In our interpretation, the co-simulation system can be formulated as closed-loop interconnections. Moreover, an intuitive engineering guideline following the theorems of the linear robust control framework can be applied. The open-loop transfer behavior of each subsystem needs to be known.

To further improve the stability and accuracy of a specified co-simulation system, advanced algorithms can be applied with a co-simulation master if available. Alternatively, signal reconstruction techniques can be applied to the slave subsystems. Adaptive filters, e.g. the extended Kalman filter, the recursive least square method, have been developed, as well as a novel coupling method based on $\mathcal{H}_\infty$ synthesis. These techniques can be potentially applied on a distributed simulation and a HIL simulation as well.

4.2 Limitations

In perspective of co-simulation, no advanced master algorithm such as the variable step control or iterative scheme has been experimented in the research. The FMI-based software package including these features can be used in the future work if needed.

In perspective of the engineering case, the EPAS system is the only experimental case. It may not cover some effects of mechatronic systems such as coupling by the discrete event, which can occur in gear shifting or ABS braking. It is also believed that with more systems such as the electric propulsion system and the active suspension system, the benefit from co-simulation would be even more significant.

4.3 Future Work

For the future research on virtual verification techniques, the integration problem in the HIL simulation will be investigated. The dynamics of heterogeneous systems, e.g. partially numerical and partially physical, can be potentially investigated on the unified framework as close-loop interconnections. However, the dynamics of the hardware interfacing the numerical parts and the physical parts should be taken into account. In addition, the developed experimental models should be implemented on the real-time machine.

Bibliography

[1] M. Aberger and M. Otter, *Modeling friction in modelica with the lund-grenoble friction model*, in Proceedings of the 2nd International Modelica Conference, 2002, pp. 285–294.

[2] C. Andersson, *Methods and Tools for Co-Simulation of Dynamic Systems with the Functional Mock-up Interface*, PhD thesis, Lund Universit, 2016.

[3] M. Arnold, *Stability of Sequential Modular Time Integration Methods for Coupled Multibody System Models*, Journal of Computational and Nonlinear Dynamics, 5 (2010), p. 031003.

[4] M. Arnold, B. Burgermeister, C. Führer, G. Hippmann, and G. Rill, *Numerical methods in vehicle system dynamics: State of the art and current developments*, Vehicle System Dynamics, 49 (2011), pp. 1159–1207.

[5] M. Arnold, C. Clauss, and T. Schierz, *Error analysis and error estimates for Co-Simulation in FMI for model exhange and Co-Simulation V2.0*, The Archive of Mechanical Engineering, Vol. LX, n (2013), pp. 75–94.

[6] T. Blochwitz, M. Otter, J. Akesson, M. Arnold, C. Clauss, H. Elmqvist, M. Friedrich, A. Junghanns, J. Mauss, D. Neumerkel, et al., *Functional mockup interface 2.0: The standard for tool independent exchange of simulation models*, in Proceedings of the 9th International Modelica Conference, no. 076, 2012, pp. 173–184.

[7] T. Bunsen, P. Cazzola, M. Gorner, L. Paoli, S. Scheffer, R. Schuitmaker, J. Tattini, and J. Teter, *Global ev outlook 2018: Towards cross-modal electrification*, (2018).

[8] M. Busch, *Zur effizienten Kopplung von Simulationsprogrammen*, PhD thesis, Kassel University, 2012.

[9] T. Chugh, W. Chen, M. Klomp, S. Ran, and M. Lidberg, *Design and control of model based steering feel reference in an electric power assisted steering system*, Dynamics of Vehicles on Roads and Tracks, 1 (2017), p. 43.

[10] E. Drenth, *Robust co-simulation methodology of physical systems*, in 9th Graz Symposium Virtual Vehicle, 2016.

[11] H. Elmqvist, S. E. Mattsson, and H. Olsson, *Parallel Model Execution on Many Cores Algorithms for Parallelization*, in Proceedings of the 10th International Modelica Conference, 2014, pp. 363–370.

[12] P. Fritzson, *Principles of object-oriented modeling and simulation with Modelica 3.3: a cyber-physical approach*, John Wiley-IEEE Press, 2014.

[13] A. Gallrein, M. Baecker, M. Burger, and A. Gizatullin, *An advanced flexible realtime tire model and its integration into fraunhofer's driving simulator*, SAE Technical Papers, 1 (2014).

[14] V. Galtier, S. Vialle, C. Dad, J.-P. Tavella, J.-P. Lam-Yee-Mui, and G. Plessis, *Fmi-based distributed multi-simulation with dacosim*, in Proceedings of the Symposium on Theory of Modeling & Simulation, Society for Computer Simulation International, 2015, pp. 39–46.

[15] C. Gomes, C. Thule, D. Broman, P. G. Larsen, and H. Vangheluwe, *Co-simulation: State of the art*, (2017).

[16] A. B. Khaled, M. B. Gaid, D. Simon, and G. Font, *Multicore simulation of powertrains using weakly synchronized model partitioning*, IFAC Proceedings Volumes, 45 (2012), pp. 448–455.

[17] S. M. Knupfer, R. Hensley, P. Hertzke, P. Schaufuss, N. Laverty, and N. Kramer, *Electrifying insights: How automakers*

can drive electrified vehicle sales and profitability, McKinsey & Company, (2017).

[18] R. Kübler and W. Schiehlen, *Two Methods of Simulator Coupling*, Mathematical and Computer Modelling of Dynamical Systems, 6 (2000), pp. 93–113.

[19] C. Lacoursière and T. Härdin, *FMI Go! A simulation runtime environment with a client server architecture over multiple protocols*, in Proceedings of the 12th International Modelica Conference, vol. 132, 2017, pp. 653–662.

[20] S. Sadjina, L. T. Kyllingstad, S. Skjong, and E. Pedersen, *Energy conservation and power bonds in co-simulations: non-iterative adaptive step size control and error estimation*, Engineering with Computers, 33 (2017), pp. 607–620.

[21] W. Schamai, P. Helle, P. Fritzson, and C. J. Paredis, *Virtual verification of system designs against system requirements*, in International Conference on Model Driven Engineering Languages and Systems, Springer, 2010, pp. 75–89.

[22] T. Schierz, M. Arnold, and C. Clauss, *Co-simulation with communication step size control in an fmi compatible master algorithm*, in Proceedings of the 9th International Modelica Conference, no. 076, Linköping University Electronic Press, 2012, pp. 205–214.

[23] B. Schweizer, P. Li, and D. Lu, *Implicit co-simulation methods: Stability and convergence analysis for solver coupling approaches with algebraic constraints*, ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift fur Angewandte Mathematik und Mechanik, 96 (2016), pp. 986–1012.

[24] B. Schweizer, P. Li, D. Lu, and T. Meyer, *Stabilized implicit co-simulation methods: solver coupling based on constitutive laws*, Archive of Applied Mechanics, 85 (2015), pp. 1559–1594.

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

[26] S. Skjong, *Modeling and Simulation of Maritime Systems and Operations for Virtual Prototyping using Co-Simulations*, PhD thesis, Norwegian University of Science and Technology, 2017.

[27] H. Van Der Auweraer, J. Anthonis, S. De Bruyne, and J. Leuridan, *Virtual engineering at work: The challenges for designing mechatronic products*, Engineering with Computers, 29 (2013), pp. 389–408.

[28] A. Viel, *Implementing stabilized co-simulation of strongly coupled systems using the functional mock-up interface 2.0*, in Proceedings of the 10 th International Modelica Conference, no. 096, 2014, pp. 213–223.

Paper I

Design of Interface in Co-simulation for Electric Power Assisted Steering System Development

Design of Interface in Co-simulation for Electric Power Assisted Steering System Development

Weitao Chen*/, Shenhai Ran*, Bengt Jacobson****
***Volvo Cars Corporation, ****Chalmers University of Technology**
E-mail: weitao.chen@volvocars.com

Interface and causality have important effects on the co-simulation used in vehicle system development. In this paper we analyzed these effects by modeling a co-simulated dual mass-spring-damper system as a sampled-data system. By stability and frequency domain analysis we find that the co-simulation interface needs be designed where the coupling interface is less stiff and the force variable should be applied in the direction of larger mass, natural frequency and damping ratio. The analytical results have been verified through two test cases of co-simulation in electric power assisted steering (EPAS) system development: a multi-rate offline co-simulation and a hardware-in-loop (HIL) simulation. Both test cases showed more consistent and stable results using the suggested interface design.

Topics/Vehicle Dynamics, EPAS System, Co-simulation

1. INTRODUCTION

Vehicle development is getting more complex due to the trend of electrification and automation. In the development process, multi-disciplinary systems such as the suspension mechanism, the electric actuator and the control unit are developed in parallel and integrated in different design phases. To achieve the system integration and reduce development time, co-simulation has been widely used.

Co-simulation can be divided into off-line and online simulation depending on whether the simulation is in real-time [1]. The off-line co-simulation usually has only virtual models calculated separately by domain-specific solvers and coupled by exchange variables at specified communicative instant t_K as shown in Fig.1. The online co-simulation usually also has physical parts in the loop, the measured variables are exchanged to the virtual models at the sensor sampling rate. In both cases the overall system has a modular integration problem [2]: the exchange variables are available at specified communicative instant or sampling time but unknown at local integration step δt_1, δt_2 for each subsystem. So the co-simulated system is called weakly-coupled [1]: the subsystems are not coupled by the exact values.

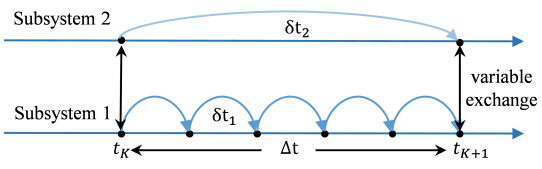


Fig. 1: Modular integration of two coupled subsystems

The stability and accuracy of the weakly-coupled system vary according to the communication step size Δt, extrapolation methods from the latest exchanged variables and the co-simulation schemes: calculate in parallel, in sequential or by iteration. All these factors have been investigated by researchers [3][4][5]. In general a compromise needs to be made between the numerical accuracy and the computational burden. From numerical tests in [3] it shows that the dynamics of the system itself also influences the stability and accuracy. So the interface design is important as well since it changes the dynamics of the decoupled subsystem. Although the interface problem is very practical, the related research seems still lacking.

The concept of interface and causality exists when formulating the co-simulation or control algorithms: the exchange variables and their directions between subsystems need to be specified. Mechanical subsystems are generally coupled by the force-displacement (or velocity) variables and a feedback loop is constituted. In force-displacement (or velocity) coupling usually either one side is a non-feedthrough system which means the output is not explicitly dependent on the input [5], so an algebraic loop problem can be avoided.

Due to quite many variants in co-simulation, an optimal solution is rather difficult to find. In this paper we only focus on the factors of interface design in co-simulation. Because understanding how the system can be decoupled in a more robust way is the first step before overcome the weak-coupling drawbacks by stabilization algorithms. The parallel calculation scheme is taken in this work since it is more general, easier to apply and how online co-simulation behaves.

We observed that the interface should be where the subsystems are not so tightly coupled and the force variable should be applied towards the subsystem with larger mass, natural frequency and damping ratio. This is shown analytical in Section 2. In Section 3 two co-simulation cases in EPAS development have been introduced and tested with different scenarios of interface which verified our finding. Further discussion and conclusion are given in Section 4.

2. SYSTEM MODELING

To model a co-simulated system we assume that the virtual subsystems can be solved accurately by the local solvers. Involving the numerical stability of local solver is a troublesome starting point and might be unnecessary since it just means the local integration step is insufficient and needs to be reduced. How the extrapolation error and instability propagate between the weakly-coupled subsystems is more interesting. So the accurately-simulated subsystems can be emulated by continuous systems sampled by the communicative time step Δt which is similar to a physical system measured at the sensor sampling rate.

In conclusion for both off-line and online co-simulation, the overall system can be seen as a combination of continuous and discrete subsystems.

2.1 Dual mass spring damper system

To derive general guidelines for more complex problem, we first analyze a dual mass-spring-damper system as shown in Fig. 2. It can represent many complex mechanical systems which mainly show second-order behaviors.

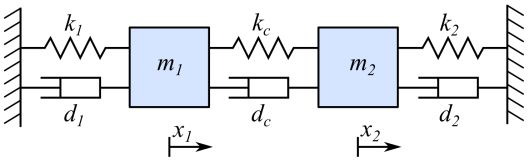


Fig. 2: The dual mass spring damper system

In mono-simulation the system is solved by integrating the ordinary-differential equations (ODE):

$$\begin{aligned} m_1\ddot{x}_1 &= -k_c(x_1 - x_2) - d_c(\dot{x}_1 - \dot{x}_2) - k_1x_1 - d_1\dot{x}_1 \\ m_2\ddot{x}_2 &= -k_c(x_2 - x_1) - d_c(\dot{x}_2 - \dot{x}_1) - k_2x_2 - d_2\dot{x}_2 \end{aligned} \tag{1}$$

When the system is solved on different simulators, equation 1 can be decoupled to equation 2 and 3. In this case the second mass is defined to take the force F_c as input and its output $[x_2 \ \dot{x}_2]^T$ is sent to the other subsystem. The subsystem dynamic matrices A_1, A_2, B_1 and B_2 vary according to the interface selection.

$$\begin{bmatrix} \dot{x}_1 \\ \ddot{x}_1 \end{bmatrix} = A_1 \begin{bmatrix} x_1 \\ \dot{x}_1 \end{bmatrix} + B_1 \begin{bmatrix} x_2 \\ \ddot{x}_2 \end{bmatrix}, \quad y_1 = F_c \tag{2}$$

$$\begin{bmatrix} \dot{x}_2 \\ \ddot{x}_2 \end{bmatrix} = A_2 \begin{bmatrix} x_2 \\ \dot{x}_2 \end{bmatrix} + B_2F_c, \qquad y_2 = \begin{bmatrix} x_2 & \dot{x}_2 \end{bmatrix}^T \tag{3}$$

2.2 Sampled-data system

The emulated local subsystems in co-simulation can be expressed by continuous transfer functions according to equations above, where ω_1 and ω_2 are the natural frequencies of the uncoupled systems and ζ_1 and ζ_2 are the respective mechanical damping ratios.

$$\begin{aligned} Q_1(s) &= -\frac{1}{s\left(\frac{1}{m_1(s^2+2\zeta_1\omega_1 s+\omega_1^2)} + \frac{1}{k_c+d_c s}\right)} \\ Q_2(s) &= \frac{s}{m_2(s^2 + 2\zeta_2\omega_2 s + \omega_2^2)} \end{aligned} \tag{4}$$

Here using displacement x_2 as exchange variable makes the transfer function $Q_1(s)$ improper. It means that the system has more zeros than poles which results undesired high-pass character. So the velocity is used instead. It can be noticed that: when the interface is selected k_c and d_c are then fixed; when the causality changes the uncoupled system parameters m_1, m_2, ω_1, ω_2, ζ_1 and ζ_2 are just swapped in equation 4.

The local subsystem output $y(t)$ is sampled at each communicative step Δt and this operation can be expressed by the starred transform [6].

$$y(t)^* = \sum_{k=-\infty}^{\infty} y(t)\delta(t - k\Delta t) \tag{5}$$

The subsystem input $u(t)$ at every local time step is obtained from the extrapolation based on the latest sampled variable $y(t)^*$. A basic zero-order hold (ZOH) method is used in this work. Due to the extrapolation error, even the sampling time Δt is selected to avoid aliasing effect, it is still not sufficient to guarantee the system stability and accuracy.

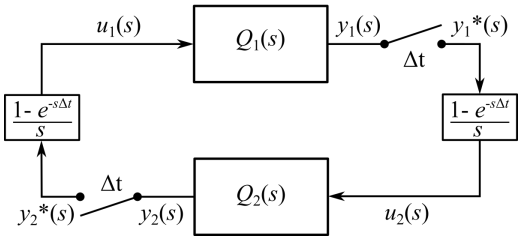


Fig. 3: Block diagram of the co-simulated system

The local subsystems together with the sampling and hold operation are shown in Fig. 3. The open-loop transfer function from $y_2(s)^*$ to $y_1(s)^*$ is given in equation 6. The other half transfer function is derived in a similar way.

$$\begin{aligned} y_1^*(s) &= \left\{y_2^*(s)\frac{1 - e^{-s\Delta t}}{s}Q_1(s)\right\}^* \\ &= y_2^*(s)(1 - e^{-s\Delta t})\left\{\frac{Q_1(s)}{s}\right\}^* \end{aligned} \tag{6}$$

and the starred transform can be further written into discrete form as:

$$y_1(z) = \mathbf{Z}\left\{y_1^*(s)\right\} = y_2(z)(1 - z^{-1})\mathbf{Z}\left\{\frac{Q_1(s)}{s}\right\} \tag{7}$$

Therefore the discrete open-loop transfer function $H_{open}(z)$ of the closed-loop system in Fig. 3 is derived:

$$H_{open}(z) = (1 - z^{-1})^2\mathbf{Z}\left\{\frac{Q_1(s)}{s}\right\}\mathbf{Z}\left\{\frac{Q_2(s)}{s}\right\} \tag{8}$$

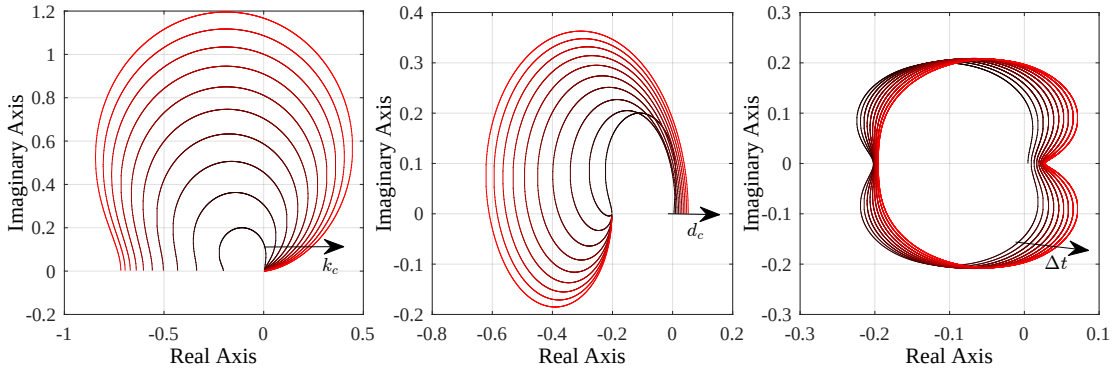


Fig. 4: Half view of Nyquist plot of $H_{open}(z)$ from low frequency to high frequency with increasing k_c, d_c, Δt (black to red).

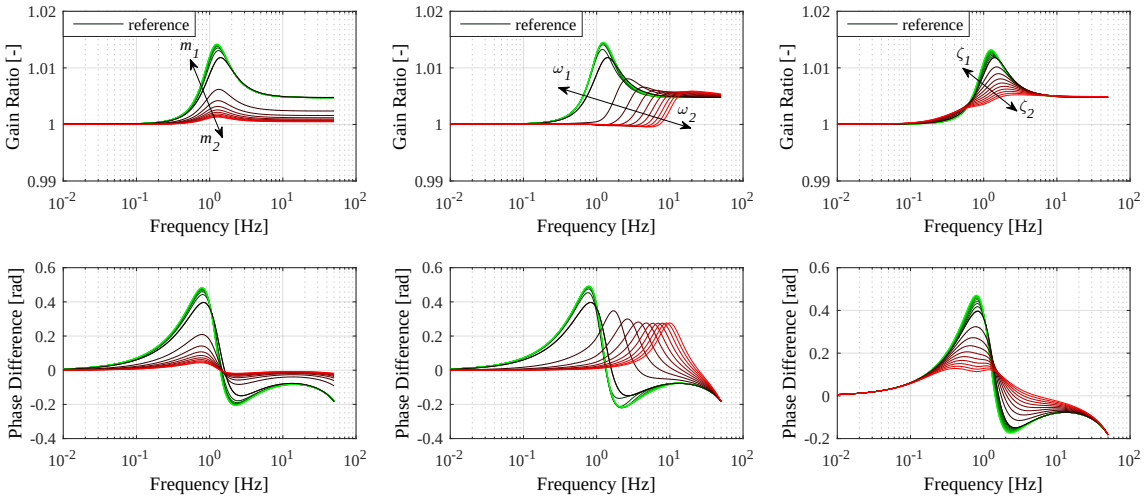


Fig. 5: Frequency response of the normalized function $G(s)$ with increasing m_1, ω_1, ζ_1 (green) and m_2, ω_2, ζ_2 (red).

the remaining parts can be obtained by the Z-transform tables and are not extended in detail here. In this way we have modeled the co-simulated system by a sample-data system and its behavior can be analyzed.

2.3 Interface effect analysis

First we look into the system behavior with variations of parameters k_c and d_c at the coupling interface. Later on the causality is checked by varying the uncoupled system parameters. Each parameter varies while the others are fixed. The parameter variation range are given in Table 1.

Table 1: Parameter variations.

Parameters	domain	units
Δt	[0.01, 0.1]	s
k_c	[10, 100]	N/m
d_c	[1, 10]	Ns/m
m_1 , m_2	[1, 10]	kg
ω_1 , ω_2	[1, 10]	Hz
ζ_1 , ζ_2	[0.5, 1.5]	-

As $z = e^{s\Delta t}$ half view of the Nyquist plot of $H_{open}(z)$ is drawn in Fig. 4. No unstable open-loop poles exists. It can be seen that when k_c and d_c increase the stability of the system gets reduced with a smaller margin to point $(1, 0)$. It means that an interface designed where the systems are coupled with high stiffness is undesired. A larger communicative step Δt also results a less stable system unsurprisingly. Because the first harmonics of ZOH process is equivalent to a delay of $\Delta t/2$ [6]. Moreover the low frequency component can be distorted by aliasing effect from a large Δt.

The dynamics of the mono-simulation reference system and its own stability are changed with varying parameters. So in addition to check by Nyquist plot, a normalized function $G(s)$ has been introduced in equation 9 by dividing the closed-looped transfer function of the continuous system to the sampled-data system.

$$G(s) = \frac{1 - Q_1(s)Q_2(s)}{1 - \mathcal{L}\{H_{open}(z)\}} \tag{9}$$

The magnitude of $G(s)$ is the gain ratio of the two

Table 2: Interface and causality.

cases	subsystem 1	subsystem 2	force/torque direction
I	chassis	rack-pinion + EPAS motor	to subsystem 2
II	chassis	rack-pinion + EPAS motor	to subsystem 1
III	chassis + rack-pinion	EPAS motor	to subsystem 2
IV	chassis + rack-pinion	EPAS motor	to subsystem 1

systems which ideally should equal to one and the phase angle of $G(s)$ is the phase difference which ideally should be zero. It can indicate how much the sampled-data system deviates from the reference as parameters change. The magnitude and phase angle of $G(s)$ has been plotted in Fig. 5. It can be seen that the reference system is less distorted both in gain and phase with an increasing m_2. The increase of m_1 gives an opposite trends. A natural frequency ω_2 higher than ω_1 reduces the gain change in lower frequency range and it shifts the range where the phase change occurs. In usual case less phase change might be preferred in low frequency range. A higher damping ratio ζ_2 contributes less gain and phase change compared to ζ_1.

In perspective of stability, the system gets more stable by larger m_2, ω_2 and ζ_2 which shows similar direction indicated by $G(s)$. To avoid lengthy results the Nyquist plots of these parameters are not given.

From this analysis above we can conclude that:
1.The interface should be selected where the models are coupled by lower stiffness and damping.
2.The force should be applied to the side with larger mass, natural frequency and damping ratio while the displacement or velocity should be applied to the other side.

Mass can be a straightforward indicator as it is usually more intuitive than the others. Although this analysis is based on a linear time-invariant system. The results can still give an insight on nonlinear systems. In next section tests for more complicated problems are shown.

3. CO-SIMULATION CASES OF EPAS SYSTEM DEVELOPMENT

Experiments have been done through two co-simulation cases that have been used in the EPAS development to verify the analytical results.

3.1 EPAS and chassis co-simulation

The first test case is an off-line co-simulation with a chassis model and a high-fidelity EPAS model. The EPAS model has been validated through experiment and it consists of rack, pinion, electric motor and the ECU. The rack m_{rack} connects to the chassis model through tie-rods. The electric motor J_{motor} is connected to the rack through a belt transmission and a ball-nut gear transmission. The hysteresis friction effect on each mechanical part has been added and the layout is shown in Fig. 6.

The EPAS model is created using modelica, an acausal modeling tool, so the system can be easily decomposed with different interfaces and causality. Each subsystem is exported as a Functional Mock-up Unit (FMU) with an embedded solver and connects to other subsystems [7]. Four different designs of interface and causality are taken as in Table 2. In case I and II the system is decoupled at the rack and tie-rods connection while in case III and IV the system is decoupled at the motor and rack connection. The subsystem 1 is simulated at fixed-time step of 5 ms and the subsystem 2 is simulated at 1 ms.

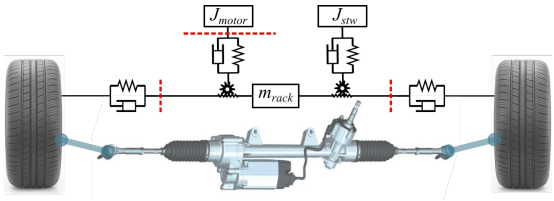


Fig. 6: The EPAS and chassis system and an approximated analytical layout.

A slow sine swept steer with vehicle speed of 50 km/h has been simulated, divergent results occur in case II and case IV. The reason of instability is that a less stable interface has been used according to our previous analysis. Same as the massive equivalent mass from motor inertia J_{motor} which deteriorates the steering feel [8]. Due to the ratio from the belt transmission and the ball-nut gear transmission, the equivalent masses in subsystem 1 from the rack mass m_{rack} and chassis lateral dynamics are hugely scaled down and smaller than the motor inertia J_{motor}. Due to this reason case II and case IV gets stable as expected when the gear ratio value has been reduced.

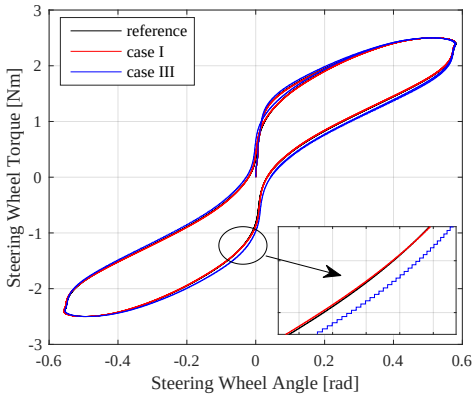


Fig. 7: Steering torque to steering angle.

On the contrary the force or torque is applied to the subsystem 2 in case I and case III and the simulations are stable. The resulting steering torque to steering wheel angle characteristic is plotted in Fig. 7. The normalized root-mean-square error (RMSE) of several variables are given in Table 3. Case I shows less error than case III.

Table 3: RMSE of co-simulation results.

variables	case I	case III
steering wheel angle	0.0178	0.1042
assist force	0.0431	0.1313
motor torque	0.2412	0.2963
motor speed	1.4101	3.9603
rack velocity	5.9446	6.1630

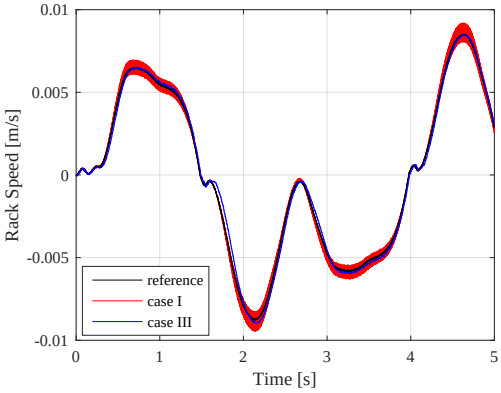


Fig. 8: Rack speed in mono-simulation and co-simulation.

However, the rack velocity in case I is more noisy than case III in Fig. 8 even the subsystem 1 is simulated at a bigger time step in case III. If we review the dynamic equation on the rack:

$$m_{rack}\ddot{x}_R = F_{rod} + F_{assist} + F_{friction} + F_{pinion} \quad (10)$$

If the coupling forces are directly applied to the rack, it might induce fast-changing rack states. In case III the force is applied to the stiffer EPAS motor therefore the rack states are less noisy.

3.2 A motor-in-loop simulation

The second case is an online simulation which has physical EPAS motor and ECU. The chassis and the other EPAS mechanical parts (rack, pinion, ball-nut gear and belt) are modeled in virtual environment. Online co-simulation is one step further as it requires real-time calculation of subsystem and synchronization of coupling variables, which can be achieved by adaptive filters [9]. In the HIL system the ECU receives vehicle speed and torsion bar angle from the virtual model; the motor is connected to a load motor which is controlled by the virtual model as control reference as in Fig. 9.

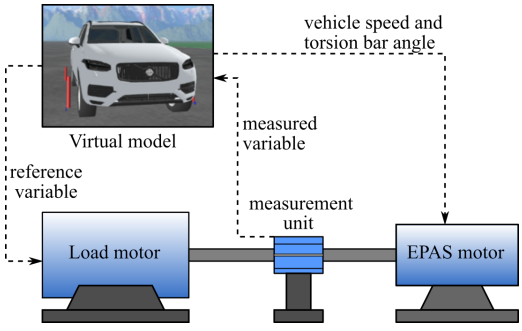


Fig. 9: The HIL system of EPAS motor in the loop

Similarly two different options of causality exist at the physical-virtual interface:
I. the virtual model takes torque measurement and sends the reference angle to the load motor.
II. the virtual model takes motor speed measurement and sends the reference torque to the load motor. (using angle input to the virtual model induces huge torque value.)

So the load motor can be either torque-controlled or position-controlled. A feed-forward and feedback control has been used for the reference angle tracking. Although different control strategy differs in bandwidth, the bandwidth should be much higher than the hardware or virtual reference. So the subsystem dynamics is still dominated by the hardware or virtual reference. Therefore the causality can be analyzed in a similar way.

Sine-wave steering tests at 0.5 Hz and 1 Hz have been applied. From Fig. 10 it can be seen that compared with a mono-simulation reference, the case II using torque-controlled is better than case I using position-controlled especially in a higher frequency. From our previous analysis: the virtual rack-pinion and the chassis model have a much smaller equivalent inertia compared to the hardware part especially a load motor and a measurement unit are added to the EPAS motor. So case II has a more robust and less distorted behavior since the velocity variable is applied to the virtual part and torque variable is applied to the stiffer hardware.

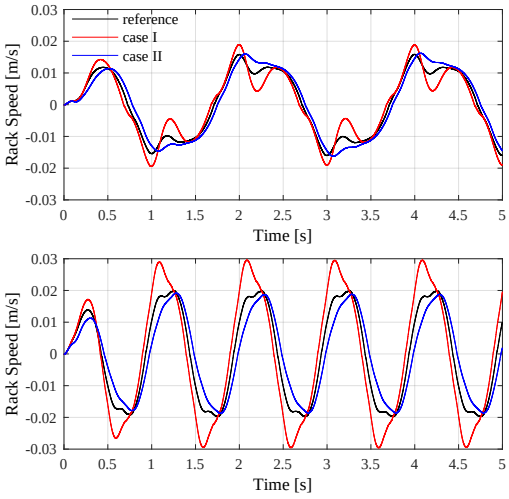


Fig. 10: Rack speed in mono-simulation and HIL simulation with two causality.

Some researchers have already discussed torque-control and position-control strategies. In literature they are called respectively impedance control and admittance control according to different causality. In contract to impedance control, admittance control, which the force is applied to the control reference, can result instability during dynamic interaction with stiff environments [10]. It is interesting to notice the consistency with our work in co-simulation: admittance control in stiff environment is a non-robust interface design as the displacement or velocity variable is transmitted to the stiffer side.

4. CONCLUSIONS

In this paper we have discussed the interface and causality effects in co-simulation. It is shown that the design of interface and causality influences significantly the results. A general guideline of the interface design has been given: the coupling interface should be designed where the system is less stiff and the force or torque variable should be sent to the heavier and stiffer side with the displacement or velocity variable feedback to the other side.

The analytical study is based on a sampled-data dual mass-spring-damper model and its stability and frequency domain analysis. Even this method is limited to linear systems, the result can still give an insight to complex problems. Two different co-simulation cases in EPAS system development have been tested. In both cases high fidelity and validated non-linear models are used. The comparisons of different interface designs have shown that better results can be obtained following the suggested rule.

The interface selection and other variants like calculation schemes and extrapolation method influence the co-simulation result comprehensively. From point of view of author, the interface design should come first and it is very practical in engineering work when model is prepared. So the information in this work is quite useful. Co-simulation stabilization algorithms may be investigated in the future work.

ACKNOWLEDGEMENT

The authors would like to thank ITEAM project in the European Union Horizon 2020 research and innovation program under Marie Sklodowska-Curie Grant Agreement No. 675999.

REFERENCES

[1] Stettinger, G. et al., "Extending Co-Simulation to the Real-Time Domain", SAE World Congress & Exhibition, 2012.

[2] Arnold, M. et al., "Numerical methods in vehicle system dynamics: State of the art and current developments", Vehicle System Dynamics, 2011, pp. 1159-1207.

[3] Busch, M. et al., "Zur effizienten Kopplung von Simulationsprogrammen", PhD thesis, Kassel University, Germany, 2012.

[4] Arnold, M. et al., "Stability of Sequential Modular Time Integration Methods for Coupled Multibody System Models", Journal of Computational and Nonlinear Dynamics, 2010, pp. 031003.

[5] Kübler, R. et al., "Two Methods of Simulator Coupling", Mathematical and Computer Modelling of Dynamical Systems, 2010, pp. 93-113.

[6] Franklin, G. F.et al., "Digital control of dynamic systems"(Vol. 3), Menlo Park, CA: Addison-wesley, 1998.

[7] Blochwitz, T. et al., "Functional Mockup Interface 2.0: The Standard for Tool independent Exchange of Simulation Models", Proceedings of the 9th International MODELICA Conference, 2012, pp. 173-184.

[8] Chugh, T. et al., "Design and control of model based steering feel reference in an electric power assisted steering system", Proc. of the 25th IAVSD Symposium,2017, pp. 43-48.

[9] Chen, W. et al., "Real-Time Co-Simulation Method Study for Vehicle Steering and Chassis System", 15th IFAC Symposium on Control in Transportation Systems, 2017.

[10] Ott, C. et al., "Unified Impedance and Admittance Control", IEEE International Conference on Robotics and Automation, 2010, pp. 554-561.

Paper II

Integration and Analysis of EPAS and Chassis System in FMI-based co-simulation

Integration and Analysis of EPAS and Chassis System in FMI-based Co-Simulation

Weitao Chen[1,2] Shenhai Ran[1] Bengt Jacobson[2]

[1]Vehicle Dynamics CAE, Volvo Cars, Sweden,
{weitao.chen,shenhai.ran}@volvocars.com
[2]Vehicle Dynamics, VEAS, Chalmers University of Technology, Sweden,
bengt.jacobson@chalmers.se

Abstract

The vehicle steering characteristics and active functions can be virtually developed with a high-fidelity electric power assisted steering (EPAS) model and a multibody chassis model. The simulation of the EPAS model requires small integration step due to high stiffness and interfacing with the controller. The multibody chassis model is computationally heavy for each integration step due to calculation of large matrices. A mono-simulation based on a single solver is not efficient for this case. Instead a co-simulation (solver coupling) approach has been used to overcome the drawbacks.

In this paper the EPAS system and chassis system are modeled in Dymola and further exported as separate functional mockup units (FMUs) and integrated with the control algorithms in Matlab. A co-simulation based on the explicit parallel calculation scheme (Jacobi scheme) has been used. A huge simulation speed-up has shown the potential and effectiveness of the approach. To understand its accuracy and tolerance, analysis on the numerical error and dynamics of the coupled-system are given.

Keywords: EPAS system, Chassis system, Co-Simulation, FMU

1 Introduction

Modern vehicles involve more electric and functional subsystems with a trend of electrification and automation. Multi-domain subsystems need to be modeled and integrated by co-simulation for a holistic development. This modular approach is quite common because the models might be from multiple sources (e.g., OEM-suppliers relationship) in different tools. Furthermore, it enables each model efficiently solved by a domain-specific numerical method. The approach has been applied in many engineering cases such as an integration of large-scale pantograph-catenary system (Arnold, 2010), a distributed simulation of a 4 cylinder engine (Saidi et al., 2016).

For accurate simulation of vehicle handling, steering and active function tests, a mechanical multibody chassis model and an EPAS model are needed. The chassis model has hundreds of degrees of freedom and its dynamics is relatively slow especially for handling and steering simulation on the flat road. The EPAS model has much faster dynamics because of the lightweight components, friction elements, electric parts and the control algorithms. Its fidelity is critical for the steering feel. As the chassis model and EPAS model differ in terms of dynamics and requirements. A mono-simulation based on a single solver might not be the optimal solution. In this paper, a FMI-based co-simulation has been tested. The coupled-system is constituted by FMUs of a chassis model, an EPAS mechanical model, a S-function for the EPAS electrics and control algorithms.

The modeling work in Dymola is presented in Section 2 and Section 3. The integration based on FMI standard and the co-simulation setup are shown in Section 4. In Section 5 the co-simulation results and analysis on simulation speed and system dynamics are discussed.

2 EPAS System

2.1 EPAS mechanism

The EPAS mechanism comprises mainly a steering wheel and column, a steering rack and an EPAS motor as shown in Figure 1. The steering column is connected to the rack and pinion by a compliant torsion bar. A belt transmission connects the motor and the ball screw which transfers the motor rotation into the rack translation.

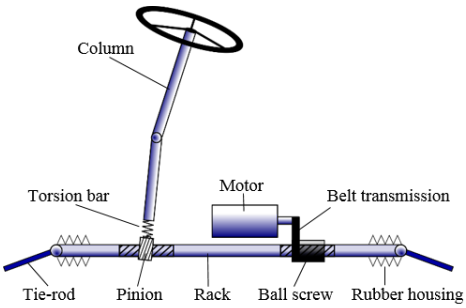


Figure 1. The EPAS system with axle-parallel drive.

The steering rack is articulated to the vehicle chassis through the suspension tie-rods and steers the front wheels. This mechanical chain builds a direct interaction

between the driver and the road. The auxiliary electric motor can deliver an assist torque according to the torsion bar deflection and vehicle speed to reduce the steering effort. The steering feel defined by the introduced mechanisms is a key metric in vehicle development and needs to be accurately simulated and evaluated on a driving simulator.

The EPAS mechanism is modeled by 3 degrees of freedom: 2 degrees of rotation for the column and motor, 1 degree of translation for the rack. Different from the Modelon Vehicle Dynamics Library (VDL) steering template, the non-linear effect from the column CV-joint has been neglected and no multibody components have been used for simplicity. Instead, the friction and motor dynamics are very important for EPAS in terms of vehicle steering response, subjective feeling for the driver and the stability of EPAS controller (Harrer and Pfeffer, 2017), more detailed effects have been considered.

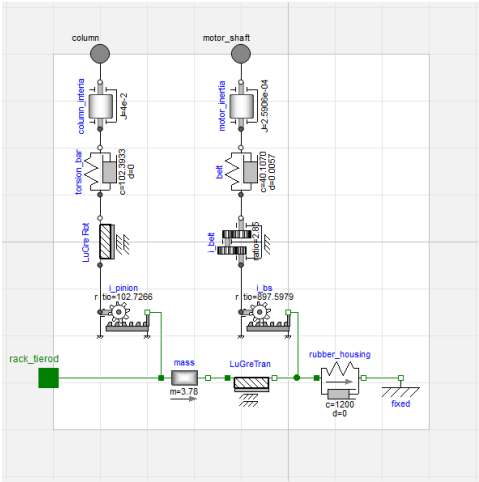


Figure 2. The EPAS mechanism modeled in Dymola.

The model based on basic Modelica mechanical components and detailed friction elements, shown in Figure 2, is created according to the dynamics on the column, the motor and the rack:

$$J_{column}\ddot{\delta}_s = T_s - T_{pinion} - T_{c_{friction}} \quad (1)$$

$$J_{motor}\ddot{\delta}_m = T_{motor} - T_{belt} \quad (2)$$

$$m_{rack}\ddot{x}_R = F_{pinion} + F_{assist} - F_{rod} - F_{r_{friction}} - F_{housing} \quad (3)$$

The states and parameters of the model are given in Table 1. The force F_{pinion} and F_{assist} are calculated from the respective torques and transmission ratios:

$$F_{pinion} = T_{pinion}/i_{pinion} \quad (4)$$

$$F_{assist} = T_{belt}/(i_{belt}i_{bs}) \quad (5)$$

where the torque T_{pinion} and T_{belt} are calculated based on the deflection of the torsion bar and belt.

Table 1. States and parameters of the EPAS model.

Notation	*Definition*
δ_s, δ_m	angle of the steering wheel, motor
x_R	rack displacement
J_{column}, J_{motor}	inertia of the column and wheel, motor
m_{rack}	rack mass
$i_{pinion}, i_{belt}, i_{bs}$	transmission ratios of the rack pinion, the belt, the ball screw
T_s	steering torque
T_{pinion}	torsion bar torque
$T_{c_{friction}}$	column friction torque
T_{motor}	applied torque from the motor
T_{belt}	load torque on the output shaft
F_{pinion}	force transmitted by the rack pinion
F_{assist}	assist force from the ball screw
F_{rod}	tie-rod force along the rack
$F_{r_{friction}}$	friction on the rack
$F_{housing}$	a spring-damper force from the housing

2.2 Friction elements

The mechanical friction is mainly divided into the upstream element $T_{c_{friction}}$ and the downstream element $F_{r_{friction}}$. The friction elements are modeled by the LuGre friction model (Astrom and Canudas de Wit, 2008). In Modelica the standard friction element is implemented by discrete events switching between stuck and slide mode. An appropriate numerical method is needed for this continuous/discrete approach. The LuGre friction model adds the hysteresis effect and it expresses the friction by differential equations:

$$\dot{z} = v - \sigma_0 z/g(v)|v| \quad (6)$$

$$g(v) = F_c + (F_s - F_c)e^{-(v/v_s)^2} \quad (7)$$

$$F_{friction} = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v \quad (8)$$

where v is the sliding velocity, z is the internal state. The bristle stiffness σ_0 and micro-damping σ_1 produce a spring-like behavior in small displacements. σ_2 is the viscous friction coefficient. $g(v)$ is a velocity-dependent term relating to the Coulomb friction F_c, the static friction F_s and the Stribeck velocity v_s.

Numerical methods for continuous system can be used to solve this model. However, its dynamics is so stiff that small tolerance value for variable step solver or small time steps for fixed step solver is needed. As a result, the simulation speed gets slow. A detailed implementation and analysis of the LuGre friction model in Modelica has been introduced in (Aberger and Otter, 2002).

The friction model parameters have been partially identified from experiments using a steering system test rig with a steering robot connecting the steering wheel and two linear actuators connecting the rack. Pull-by-torque and pull-by-rack tests (Harrer and Pfeffer, 2017) have been taken with the EPAS controller deactivated. The

steering system is excited accordingly either by velocity-controlled steering wheel input or rack input in free load condition. Thanks to the acausal modeling, the recorded data can be conveniently taken as input to the EPAS model. The comparison of the simulation results and the measurement data are given in Figure 3 and Figure 4.

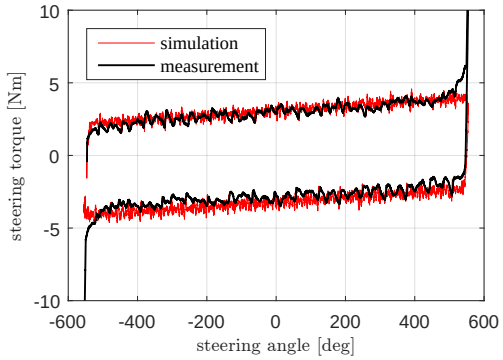


Figure 3. Steering torque in a pull-by-torque test with a steady steering velocity input of 13 deg/s.

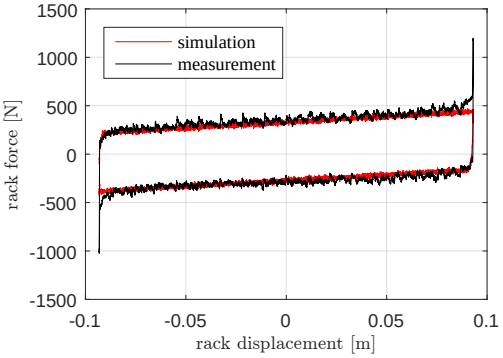


Figure 4. Rack force in a pull-by-rack test with a steady rack speed of 2 mm/s.

2.3 EPAS control

The large inertia, high friction and less damped behavior from the EPAS mechanism is counterbalanced by the basic steering functions involving the inertia compensation, friction compensation, active damping and power-assist. The advanced driving functions like the lane keeping aid (LKA) and Pilot-Assist are added to the motor torque request $T_{request}$ in Figure 5, which is further delivered to the electric motor.

The detailed models of the control algorithm, ECU and electrics are provided from the supplier as black-box S-functions with inputs of vehicle speed $V_{vehicle}$, torsion bar angle δ_{pinion}, motor speed $\dot{\delta}_m$ and the external request from the advanced functions. So that the system needs to run in the Simulink environment with a forward Euler method with 1 ms integration step.

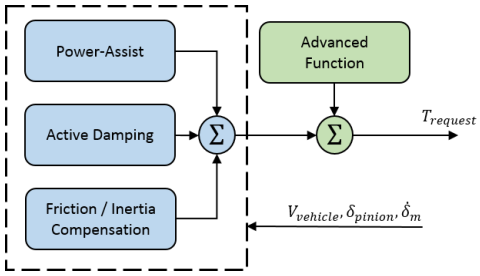


Figure 5. A block diagram of the EPAS control architecture.

3 Chassis Model in Dymola

A chassis model based on the Modelon VDL has been used. It is constituted by the car body, Pacejka tire models and suspensions (Figure 6). To facilitate the computation, the suspension linkages are represented by kinematic tables. The wheel orientation and translation varies according to the wheel jounce and steering input. A validated model of Volvo XC90 has been used in the work.

To integrate the chassis model with the created EPAS model, 1D translation interface is attached to the rack and the original VDL steering model is disconnected as a dummy part. In this way the translation is still relative to the front subframe whose compliance may have a great impact on the steering feel.

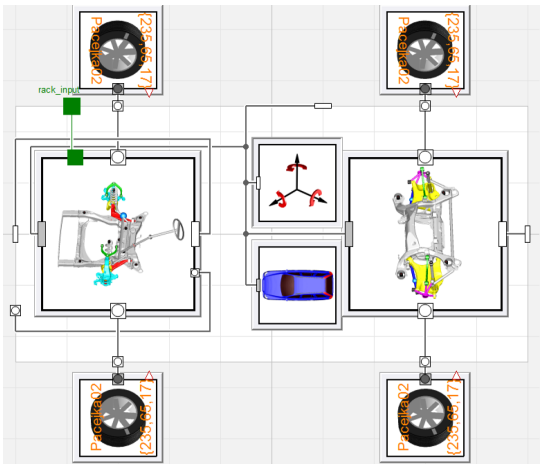


Figure 6. The multibody chassis model in Dymola.

4 Co-Simulation Setup

The EPAS model and chassis model are compiled to separate FMUs embedded with variable step and variable order Dassl solvers. The EPAS FMU, chassis FMU and EPAS control S-function are coupled by specified input-output signals (Figure 7). At the coupling interface the chassis FMU takes the rack velocity $\dot{x}_R$ as input and EPAS model takes the force F_{rod} as input. The decision is based on our analysis from a previous work (Chen et al., 2018), briefly:

- The force variable should be applied towards the heavier and stiffer part for robustness. The EPAS system due to the gear ratio effect is much heavier and stiffer than the lateral dynamics of chassis system.
- The displacement input results an improper dynamic system (more zeros than poles in the transfer function). Thus, the derivation of input variable is needed and this might generate noisy or incorrect results.

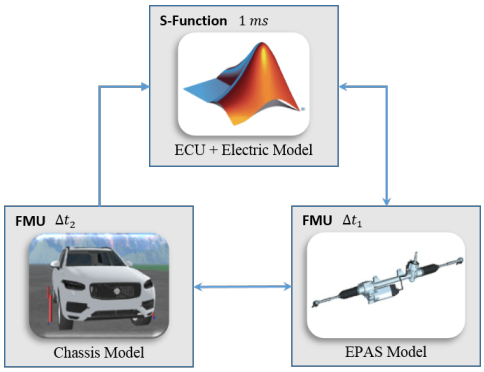


Figure 7. The layout of the co-simulation setup.

The chassis FMU is setup with a communicative step of Δt_2 which is the time size that the local solver updates the input and output. The communicative step of EPAS FMU is Δt_1 with a default value of 1 ms because of the coupling with the controller.

For simplicity, the co-simulation is implemented with an explicit parallel calculation scheme (i.e., during the communicative interval each model is integrated independently and the input is approximated from extrapolation). In this work, a common constant extrapolation (zero-order hold) has been used. Although this calculation scheme suffers from the numerical stability and coupling errors. It is advantageous for less computational burden and easy implementation in practice because no control of computation sequence or iterative process is needed from a master algorithm (Busch, 2012).

5 Co-Simulation Results

The co-simulation have been tested with various scenarios as given in Table 2. For comparison, a mono-simulation reference, denoted by *Ref-1*, is made by compiling the EPAS and chassis model together as a whole FMU with the same solver. The tests are performed on a laptop with 32GB RAM and one Intel Core i7 processor which runs 8 cores at 2.70 GHz.

5.1 Simulation speed-up

A 5 seconds steering maneuver with a sine wave steering torque input has been simulated. From the CPU time of each simulation case (Figure 8), one can see that comparing with *Ref-1* the co-simulation cases are much faster

Table 2. Simulation Cases

Case	*Communicative step*
Ref-1	$\Delta t_1 = 1ms$ no Δt_2
Ref-2	$\Delta t_1 = 5ms$ no Δt_2
CS-1	$\Delta t_1 = 1ms$ $\Delta t_2 = 1ms$
CS-2	$\Delta t_1 = 1ms$ $\Delta t_2 = 5ms$
CS-3	$\Delta t_1 = 1ms$ $\Delta t_2 = 10ms$
CS-4	$\Delta t_1 = 1ms$ $\Delta t_2 = 15ms$
CS-5	$\Delta t_1 = 1ms$ $\Delta t_2 = 20ms$

especially when Δt_2 gets larger. In mono-simulation case *Ref-1*, the chassis model needs to take a small integration step due to the stiff EPAS model. Instead, in co-simulation each solver can adapt to the local dynamics more efficiently.

In another mono-simulation case *Ref-2* with increased Δt_1, the CPU time reduces a lot as well but the time saving is not so effective as the co-simulation cases with a same or larger Δt_2 setup. It can be observed that a big time saving is from a relaxation of communication with the chassis model.

The co-simulation case *CS-1* does not show an obvious advantage in the simulation speed. Because the adaptability of the local solver is constrained by a very frequent communication of 1 ms. In such a case the speed-up capability of co-simulation cannot be fully used even though the stiff part has been decoupled.

For other co-simulation cases, a further relaxation of Δt_2 does increase the simulation speed but the improvement gets reduced at a larger step. If a rather large Δt_2 has been taken, the two models can be seen as nearly decoupled and calculated independently. Therefore, the simulation time might just depend on the dynamics and solver of each part. In practice, the Δt_2 size setup needs to be compromised considering the stability and coupling error which is discussed in the following section.

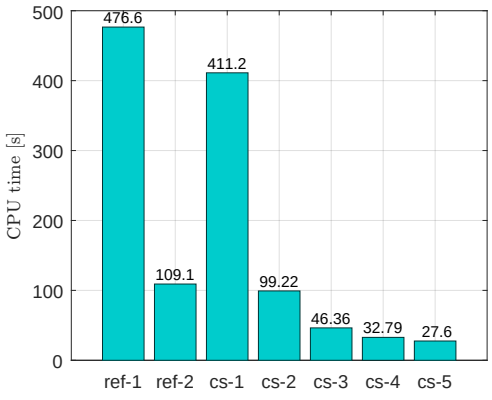


Figure 8. CPU time for a 5 seconds simulation.

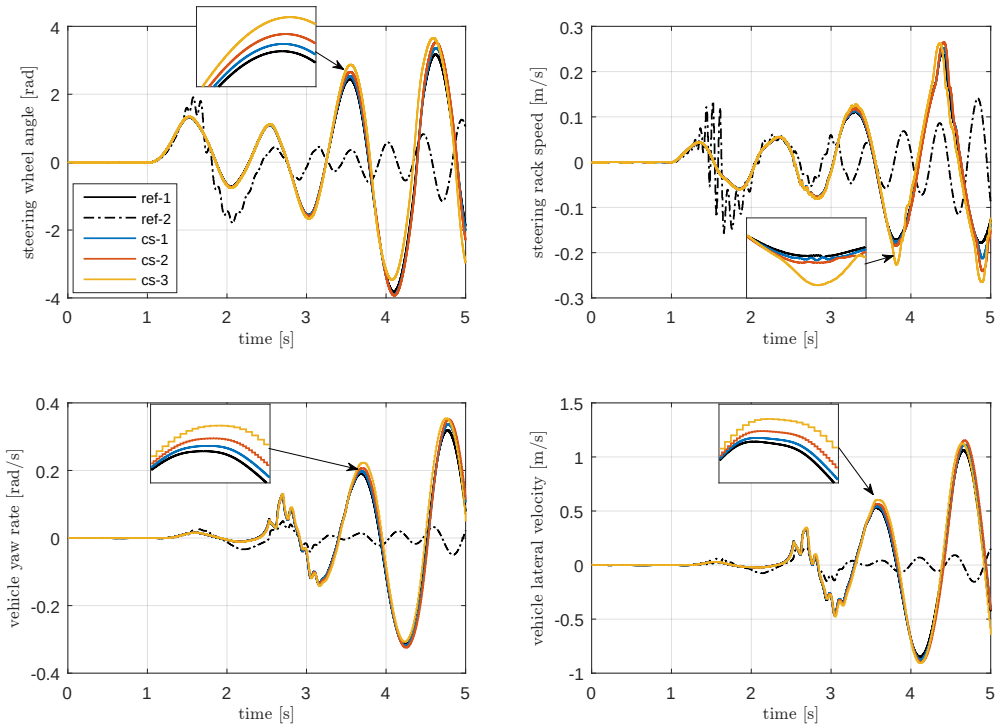


Figure 9. Simulation results of the EPAS system and chassis states.

5.2 Error analysis

Two EPAS system states (steering wheel angle δ_s, rack speed $\dot{x}_R$) and two chassis states (yaw rate, lateral velocity) from the previous simulation tests are plotted in Figure 9. It can be seen that *Ref-2* gives inaccurate and useless results although it can run really fast in the previous analysis. Because the high bandwidth coupling between the EPAS control, electric and the mechanism, the coupling variables are poorly approximated by extrapolation and the simulation error gets quite large.

The co-simulation cases, due to a more robust integration, have shown more stable and consistent results even their simulation speeds are faster. Case *CS-3* in Figure 9 shows larger stepwise signals from the chassis model. The co-simulation results deviate more at the peaks which is very intuitive since the accuracy of extrapolation is worse when the signal changes direction.

The relative global error $\varepsilon_{g,x}$ of selected state x are computed by the normalized root-mean-square error as:

$$\varepsilon_{g,x} = \frac{\sqrt{\sum_{t=0}^{T}\left(\left(x_{cs}(t) - x_{ref}(t)\right)^2/T\right)}}{x_{ref}^{max} - x_{ref}^{min}} \tag{9}$$

where x_{ref}^{max} and x_{ref}^{min} are the maximum and minimum reference state value during simulation time $t \in [0, T]$. The relative global error $\varepsilon_{g,x}$ is plotted in Figure 10. One can see that $\varepsilon_{g,x}$ in case *Ref-2* is clearly the worst and the error increases as the step Δt_2 grows, which limits the relaxation of communication for the simulation speed-up. To reduce the error and enable a further relaxation, some explicit coupling methods from (Khaled et al., 2014) and (Benedikt et al., 2013) can be potentially applied, which is out of the scope of this paper. Thus, only a basic constant extrapolation is presented in this work.

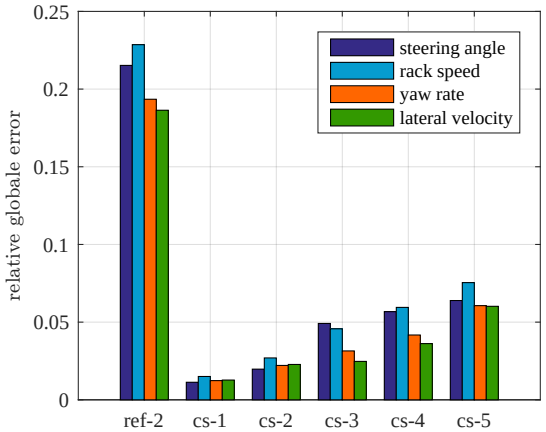


Figure 10. Comparison of the relative global error $\varepsilon_{g,x}$.

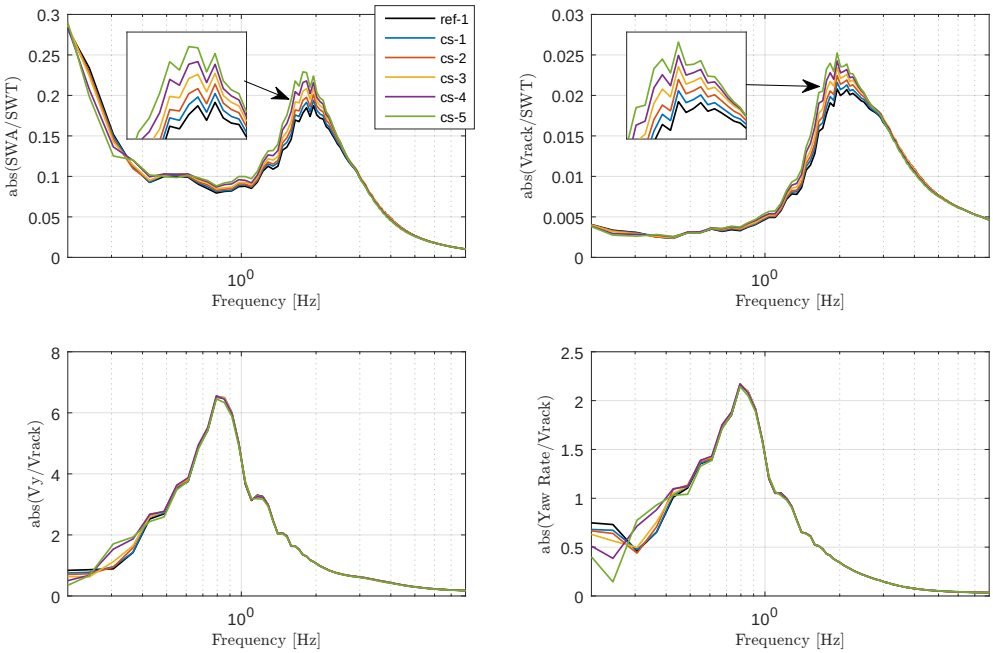


Figure 11. The identified transfer behaviors of EPAS and chassis.

5.3 System dynamics analysis

To investigate the system dynamics in multiple conditions, a frequency domain comparison is more intuitive as a second analysis. In this analysis, a steering torque input from low frequency up to high frequency has been applied. The simulation time is long enough that the system can be excited sufficiently within the frequency range of interest. Two pairs of transfer functions are identified from the simulation results. The first is steering torque (SWT) to steering wheel angle (SWA) and rack speed (Vrack), which is more relevant to steering behavior. The second is from rack speed (Vrack) to chassis lateral speed (Vy) and yaw rate which mainly shows the chassis lateral dynamics.

The magnitudes of the transfer functions are plotted in Figure 11. The steering feedback character (SWA/SWT) and the EPAS transfer behavior (Vrack/SWT) are influenced in a certain range. The deviation gets larger around 1.1 Hz which is close to the chassis yaw eigenfrequency. As Δt_2 increases, more delayed rack force resistant to the rack motion gives an increased steering wheel angle and rack speed.

The chassis transfer behaviors are relatively more consistent to the reference. It might be the reason that the chassis has a slower dynamics and more robust to the coupling effect. The deviation of chassis dynamics occurs mainly below 0.5 Hz and the magnitude of deviation is correlated to the relaxation condition.

Furthermore, the dynamics of the EPAS and chassis system limited the bandwidth of the coupling signals. In the high frequency range of steering input, the only excitation to chassis system has been filtered out and the coupling effect gets minor.

6 Conclusion

In this paper a FMI-based co-simulation of EPAS and vehicle chassis system has been presented. The solver coupling approach is used for mechanical-functional system integration and also for mechanical systems in large time scale. The accelerated simulation speed makes the simulation tool more useful for design optimization and control tuning work. A controllable coupling error without severe numerical instability is induced by the explicit parallel calculation scheme. The approach can also be applied on real-time applications where the simulation speed is crucial. However, the CPU time from the current test is still huge that model order reduction might be needed to make each system real-time capable first.

The approach is quite promising for vehicle chassis and other mechatronic systems (e.g., active suspension, electric propulsion and automated driving system).

Acknowledgement

The authors would like to thank ITEAM project funded by the European Union Horizon 2020 research and innovation program under Marie Sklodowska-Curie Grant Agreement No. 675999.

References

Martin Aberger and Martin Otter. Modeling friction in Modelica with the Lund-Grenoble friction model. *Proceedings of the*

2nd International Modelica Conference, 3:285–294, 2002.

Martin Arnold. Stability of Sequential Modular Time Integration Methods for Coupled Multibody System Models. *Journal of Computational and Nonlinear Dynamics*, 5(3):031003, 2010.

Karl Johan Astrom and Carlos Canudas de Wit. Revisiting the LuGre model; Stick-slip Motion and Rate Dependence. *IEEE Control Systems Magazine*, 6:101–114, 2008.

Martin Benedikt, Daniel Watzenig, Josef Zehetner, and Anton Hofer. NEPCE - A nearly energy-preserving coupling element for weak-coupled problems and co-simulations. *V International Conference on Computational Methods for Coupled Problems in Science and Engineering*, pages 1–12, 2013.

Martin Busch. *Zur effizienten Kopplung von Simulationsprogrammen*. PhD thesis, Kassel University, 2012.

Weitao Chen, Shenhai Ran, and Bengt Jacobson. Design of Interface in Co-simulation for Electric Power Assisted Steering System Development. *Proceedings of the 14th International Symposium on Advanced Vehicle Control (AVEC' 18)*, 2018.

Manfred Harrer and Peter Pfeffer. *Steering Handbook*. 2017.

Abir Ben Khaled, Laurent Duval, Mohamed El Mongi Ben Gaïd, and Daniel Simon. Context-based polynomial extrapolation and slackened synchronization for fast multi-core simulation using fmi. In *International Modelica Conference*, pages 225–234. Linköping University Electronic Press, 2014.

Salah Eddine Saidi, Nicolas Pernet, Yves Sorel, and Abir Ben Khaled. Acceleration of FMU Co-Simulation On Multi-core Architectures. *The First Japanese Modelica Conferences*, (124):106–112, 2016.

Paper III

On Explicit Co-Simulation Approach: Analysis and Improved Coupling Design Based on $\mathcal{H}_\infty$ Synthesis

Journal manuscript

On explicit parallel co-simulation approach: analysis and improved coupling design based on $\mathcal{H}_\infty$ synthesis

Weitao Chen · Shenhai Ran · Canhui Wu · Bengt Jacobson

Received: date / Accepted: date

Abstract Explicit parallel co-simulation approach is widely used in industry for simulation of a large-scale system. The stability and error analysis has been well-investigated by many researchers in a numerical analysis framework. The performance of different extrapolation techniques are checked by a numerical test with an experimental model and evaluated *a posteriori*. However, due to a combination effect of multiple factors of system dynamics and extrapolation methods, the results are less intuitive, especially for a complex engineering system. In this manuscript, the explicit parallel co-simulation approach is reviewed in the linear robust control framework. The numerical stability and error is interpreted by the stability and robustness of the closed-loop interconnection. Some intuitive engineering guidelines can be given by the small-gain theorem and the passivity theorem.

In addition, a novel coupling method to minimize the worst-case $\mathcal{L}_2$ norm of the coupling error is introduced. The method comprises two dynamic feed-forward terms, one smoother and one compensator to preserve the passivity, which are designed based on the $\mathcal{H}_\infty$ synthesis. It has advantages of transparent parameter tuning, robustness and easy implementation. The proposed method has been verified by two test cases. The first test case is a dual mass-spring-damper system with random system parameters and the second case is a FMI-based co-simulation of a multi-body dynamics (MBD) vehicle chassis model with an electric power assisted steering (EPAS) model. The results show that the developed method can improve the stability and accuracy for explicit parallel co-simulation, which enables a larger communication step to speed-up the computation.

Keywords Explicit co-simulation · Parallel calculation scheme · Control theory · $\mathcal{H}_\infty$ synthesis

Weitao Chen · Shenhai Ran
Vehicle Dynamics CAE, Volvo Cars, Torslanda,40531 Gothenburg, Sweden
E-mail: weitao.chen@volvocars.com, shenhai.ran@volvocars.com
Canhui Wu
Fraunhofer Institute for Industrial Mathematics ITWM, 67663 Kaiserslautern, Germany
E-mail: canhui.wu@itwm.fraunhofer.de
Bengt Jacobson
Vehicle Dynamics, VEAS, Chalmers University of Technology, 41296 Gothenburg, Sweden
E-mail: bengt.jacobson@chalmers.se

1 Introduction

Co-simulation is a cornerstone of the virtual development of mechatronic systems. Especially in automotive industry, the holistic vehicle system is multi-disciplinary with subsystems of high-fidelity and part of them might be from the suppliers in different software tools. A Functional Mock-up Interface (FMI) co-simulation standard has been initiated by the academia and industry to enable the modular engineering approach [6]. Moreover, the solver-coupling approach, i.e. co-simulation, is advantageous numerically since each solver can adapt more efficiently to the local subsystems. A multi-core distributed simulation of combustion engine is presented by Khaled [3], the simulation is accelerated by partitioning different cylinder models with discrete events. Christian has partitioned a race car model to achieve an accelerated parallel co-simulation[1]. Gallrein has used a co-simulation of high-fidelity tire models and a MBD chassis model to achieve a real-time application on driving simulator [12].

The investigation on numerical stability, error and master algorithms of co-simulation has been active in the last decade. An extensive state-of-the-art review on co-simulation has been given by Gomes [13]. The subject comprises the coupling of continuous systems, discrete systems and hybrid systems. In terms of calculation scheme, it can be classified by explicit (non-iterative) and implicit (iterative and semi-explicit) types or parallel scheme(*Jacobi scheme*), sequential scheme (*Gauss-Seidel scheme*) and iterative scheme. Furthermore, for mechanical system the model coupling configuration can be differentiated by the algebraic constraint approach or the applied force approach. The applied force approach, which the coupling variables are force-displacement (*FD coupling*) or displacement-displacement (*DD coupling*), is more preferred if possible in engineering work since an algebraic loop problem can be avoided [7].

The explicit parallel co-simulation, i.e. each subsystem integrates once in parallel and exchange the coupling variables at specified macro-step Δt, is more common than its alternatives due to a less implementation difficulty. In this approach, no advanced master algorithm to control the iterative process or calculation sequence is required. Additionally, each slave subsystem is neither required to be controllable for rollback nor to be observable for the internal states or *Jacobian* matrices. This makes the approach more implementable with off-the-shelf commercial software and black-box models for the intellectual property (IP) protection. In general, the explicit parallel co-simulation is the fastest thanks to a less computational burden, which makes it more useful for the optimization and real-time application, e.g. hardware-in-loop (HIL) simulation. As a result, the explicit parallel approach is more popular in the industrial virtual engineering work.

However, it is well-known that the explicit parallel co-simulation has drawbacks in terms of accuracy and numerical stability. For each subsystem the input during the communicative interval Δt is frozen or approximated by some extrapolation methods. This unavoidable approximation error, namely, the coupling error, is significant to the simulation accuracy and numerical stability. Without the capability of predictor-corrector method or rollback like the iterative techniques [20], efforts on improving the explicit coupling method have been made by some researchers. Busch [7] has analyzed systematically the extrapolation method by *Lagrange* polynomials and *Hermite* polynomials which contains first-order derivatives. It shows that a higher extrapolation degree gives a higher error order but the numerical

stability might suffer and the coupling methods depends on the system parameters. Benedikt [5] developed an energy-based coupling method which corrects the coupling values to nearly preserve the generalized energy of a power bond. Drenth [10] used a similar concept for a dual sampler-hold design to preserve the energy in bond graph theory. Actually, these approaches preserve the energy by reducing the error of effort-flow variables separately. If the whole physical energy error is considered alone, the result might be incorrect which has been shown in the work of González [14] and Wu [22]. Furthermore, some adaptive extrapolation methods have been developed. The energy concept has been taken as an error estimator by Sadjina [18] for an explicit variable macro-step control. Stettinger [21] has developed a model-based coupling approach by extended *Kalman* filter and the *Recursivel least square* algorithm. Khaled [15] used a context-based heuristic method to adapt the extrapolation polynomial.

Usually the parameter tuning of these improved coupling methods is less transparent and their performance might be evaluated by numerical test *a posteriori*. A combination effect of system parameters and coupling configurations, such as *DD coupling* and *FD coupling* might make the tuning of the method difficult to conclude [19]. Furthermore, the system can be much more complex in the real engineering case. The main goal of this manuscript is to review the explicit parallel co-simulation subject in the control theory framework. The error analysis and stability analysis is given equivalently in the *Laplcae* domain instead of in the time domain, which can potentially bring more engineering insights. In addition a novel coupling method based on $\mathcal{H}_\infty$ synthesis is introduced to minimize the $\mathcal{L}_2$ norm of the coupling error.

The manuscript is organized as following: first, the co-simulation system is formulated as a closed-loop interconnection in Section 2. The co-simulation stability and robustness can be analyzed by the Nyquist stability criterion and related theorems. Then, the new coupling design formulated as a $\mathcal{H}_\infty$ synthesis problem is introduced in Section 3. The synthesis problem is solved by an optimization routine based on the linear matrix inequality (LMI) method with a pole-placement constraint. In Section 4 the proposed improved coupling method has been verified on a dual mass-spring-damper system and applied on a real engineering case of vehicle and EPAS system co-simulation. Further discussions and conclusions are given in Section 5.

2 Analysis based on control theory

2.1 Co-simulation system description

A co-simulation system can be simplified as two linear time-invariant (LTI) subsystems coupled by input-output variables, which are mapped by a matrix L at communicative instant t_n:

$$\begin{aligned} \dot{\tilde{z}}_n &= A\tilde{z}_n + B\Psi\big(\tilde{u}(\tau)\big) \\ \tilde{y}_n &= C\tilde{z}_n + D\Psi\big(\tilde{u}(\tau)\big) \\ \tilde{u}_n &= L\tilde{y}_n \end{aligned} \tag{1}$$

where $\tilde{z}_n$ is the augmented state vector and Ψ is the extrapolation operator during the communicative interval $\tau \in [t_n, t_{n+1})$. Each subsystem can be taken as exactly

solved that the approximation error from the solver is minor compared to the coupling error [11]. If a constant extrapolation is taken, i.e. $\Psi\big(\tilde{u}(\tau)\big) = \tilde{u}_n, \tau \in [t_n, t_{n+1})$. The system can be discretized at macro-step Δt and augmented as:

$$\begin{aligned} \begin{bmatrix} \tilde{z}_{n+1} \\ \tilde{y}_{n+1} \end{bmatrix} &= \underbrace{\begin{bmatrix} e^{A\Delta t} & K(\Delta t)BL \\ Ce^{A\Delta t} & CK(\Delta t)BL + DL \end{bmatrix}}_{A^*} \begin{bmatrix} \tilde{z}_n \\ \tilde{y}_n \end{bmatrix} \\ \text{with} \quad K(\Delta t) &= \int_{t_n}^{t_{n+1}} e^{A(t_{n+1}-\tau)} d\tau \end{aligned} \tag{2}$$

In numerical analysis framework, the numerical stability of co-simulation is given by the spectral radius $\rho(A^*)$ of the difference equation above. The necessary and sufficient condition of numerical stability is that $\rho(A^*) \leq 1$ and there is no more than one eigenvalue on the unit circle. However, the relation between the spectral radius $\rho(A^*)$ and the system parameters is complicated and needs to be checked graphically by a numerical test. Different normalization methods have been taken for an intuitive conclusion [14][19]. In this manuscript, the co-simulation system is formulated as a close-loop system in control theory framework so the problem can be interpreted in a different way.

2.2 Closed-loop interconnection formulation

A nominal mono-simulation reference can be formulated by a closed-loop interconnection of two subsystems. The subsystems, with zero initial condition, are emulated by transfer functions $Q_1(s)$ and $Q_2(s)$, s denoting the *Laplace* domain. $Q_1(s)$ and $Q_2(s)$ are usually proper or strictly proper, i.e. the degree of the numerator polynomial is equal or less than that of the denominator, which indicates weather the connection is *feed-through* (proper) or *non feed-through* (strictly proper)[17]. Improper system is not implementable in physical modeling as a derivation is needed. A subsystem represented by the transfer function might have some internal states. But the stability is guaranteed by an appropriate local solver and it has no influence on the input-output behavior and the closed-loop interconnection.

In co-simulation, sampler and hold devices are added to the nominal system as shown in Figure 1. The sampled input $u^*(t)$ is a product of the continuous input $u(t)$ and a periodic impulse train. Its frequency domain expression is well-known:

$$u^*(t) = \sum_{n=-\infty}^{\infty} u(t)\delta(t - n\Delta t), \quad u^*(j\omega) = \frac{1}{\Delta t} \sum_{n=-\infty}^{\infty} u(j\omega - jn\omega_s) \tag{3}$$

then the approximated input $\tilde{u}(j\omega)$ is obtained by holding $u^*(s)$ using a linear extrapolation operator $H(s)$, e.g. the *Lagrange* extrapolation:

$$\tilde{u}(s) = \frac{H(s)}{\Delta t} \sum_{n=-\infty}^{\infty} u(s - jn\omega_s) \tag{4}$$

the system becomes a sample-data closed-loop interconnection, which induces error and stability issues. There are two main crucial characters of co-simulation need to be investigated by the formulated system:

1. The input coupling error ξ_u from extrapolation and its influence on the simulation accuracy.
2. The numerical stability and robustness of co-simulation.

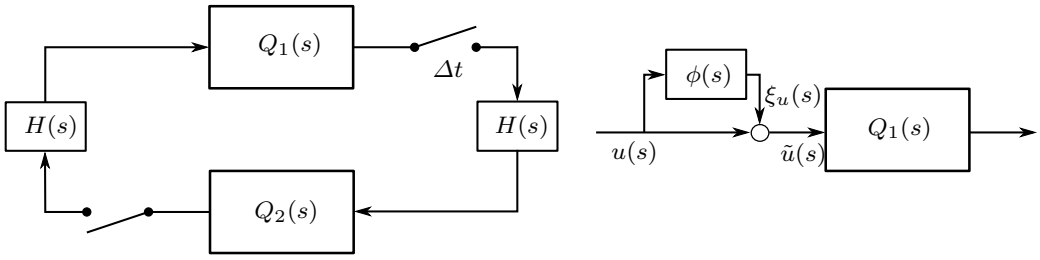


Fig. 1 (a) A co-simulation system formulated as a closed-loop interconnection. (b) A truncated subsystem on one side and ξ_u is an input multiplicative disturbance.

2.3 Analysis of co-simulation error

The input coupling error ξ_u can be seen as a multiplicative disturbance to the input (Figure 1) and is derived as:

$$\begin{aligned}\xi_u(s) &= \tilde{u}(s) - u(s) \\ &= \frac{1}{\Delta t}\Big(H(s) - 1\Big)u(s) + \frac{H(s)}{\Delta t}\sum_{n=1}^{\infty} u(s \pm jn\omega_s)\end{aligned} \tag{5}$$

where the first part is the component in the lower frequency and the second part is the component in the higher frequency, which can be mirrored into the lower frequency part if the input signal frequency ω is high and sampling frequency ω_s is not high enough (i.e. the aliasing effect). In this case, the two terms in $\xi_u(s)$ are combined and not possible to be decoupled. Thus, to avoid aliasing effect the macro-step Δt needs to be sufficiently small.

The linear operator $H(s)$ varies according to the extrapolation degree k. For simplicity, zero-order hold H_{zoh} $(k = 0)$, first-order hold H_{foh} $(k = 1)$, second-order hold H_{soh} $(k = 2)$ are considered and their *Laplace* expressions are given as [4]:

$$\begin{aligned}H_{zoh}(s) &= \frac{1 - e^{-s\Delta t}}{s} \\ H_{foh}(s) &= \frac{1 + \Delta t}{\Delta t}\Big(\frac{1 - e^{-s\Delta t}}{s}\Big)^2 \\ H_{soh}(s) &= \Big(\frac{0.5 - e^{-s\Delta t} + 0.5e^{-2s\Delta t}}{\Delta t^2}\Big)\Big(\frac{2 - \Delta t^2 s^2 e^{-s\Delta t} - 2\Delta t s e^{-s\Delta t} - 2e^{-s\Delta t}}{s^3}\Big) \\ &\quad + \Big(\frac{1.5 - 2e^{-s\Delta t} + 0.5e^{-2s\Delta t}}{\Delta t}\Big)\Big(\frac{1 - e^{-s\Delta t} - s\Delta t e^{-s\Delta t}}{s^2}\Big) + \frac{1 - e^{-s\Delta t}}{s}\end{aligned} \tag{6}$$

which can be further expanded by Taylor series using symbolic calculation tools and $\xi_u(s)$ can be respectively further approximated by the low frequency part that:

$$\begin{aligned}
\xi_{u,zoh}(s) &\approx \frac{1}{\Delta t}\Big(H_{zoh}(s)-1\Big)u(s) = u(s)\Big(-\frac{1}{2}s\Delta t + s^2\mathcal{O}(\Delta t^2)\Big)\\
\xi_{u,foh}(s) &\approx \frac{1}{\Delta t}\Big(H_{foh}(s)-1\Big)u(s) = u(s)\Big(-\frac{5}{12}(s\Delta t)^2 + s^3\mathcal{O}(\Delta t^3)\Big)\\
\xi_{u,soh}(s) &\approx \frac{1}{\Delta t}\Big(H_{soh}(s)-1\Big)u(s) = u(s)\Big(-\frac{1}{4}(s\Delta t)^3 + s^4\mathcal{O}(\Delta t^4)\Big)
\end{aligned} \tag{7}$$

where $\mathcal{O}(\Delta t^k)$ indicates residual terms of order equal or higher than Δt^k. With k degree extrapolation, ξ_u is in order of Δt^k. In addition, the frequency property can be seen in this formulation: the error grows with the input frequency. For the co-simulation system and nominal system respectively we have:

$$\tilde{u}(s)Q(s) = \tilde{y}(s) \quad \text{and} \quad u(s)Q(s) = y(s) \tag{8}$$

then the output error $\xi_y(s) = \tilde{y}(s) - y(s)$ is a linear projection of $\xi_u(s)$ in s domain sharing the same character of order $\mathcal{O}(\Delta t^k)$, which is consistent to the global error analysis in the time-domain [7]. Similarly, the state error $\xi_x(s)$ by a similar projection of state $Q_x(s)$ has the same order. If the respective dynamics by $Q(s)$ or $Q_x(s)$ is really fast, the output y or state x might be excited by the high frequency components in the coupling error.

2.4 Analysis of stability and robustness

In perspective of the numerical stability of co-simulation, the error ξ_y, ξ_u should be convergent. For a co-simulation system, the closed-loop interconnection can be represented by:

$$\tilde{y}_1 = (1+\phi)Q_1\tilde{y}_2 \quad \text{and} \quad \tilde{y}_2 = (1+\phi)Q_2\tilde{y}_1 \tag{9}$$

notation s for *Laplace* domain is dropped for clarity and ϕ is the operator for the multiplicative disturbance. For the error-free nominal system, the closed-loop interconnection can be represented by:

$$y_1 = Q_1 y_2 \quad \text{and} \quad y_2 = Q_2 y_1 \tag{10}$$

and can be further extended as:

$$y_1 + \phi Q_1 y_2 = (1+\phi)Q_1 y_2 \quad \text{and} \quad y_2 + \phi Q_2 y_1 = (1+\phi)Q_2 y_1 \tag{11}$$

Then the co-simulation output error ξ_y is derived from the difference:

$$\xi_{y_1} = (1+\phi)Q_1\xi_{y_2} + \phi Q_1 y_2 \quad \text{and} \quad \xi_{y_2} = (1+\phi)Q_2\xi_{y_1} + \phi Q_2 y_1 \tag{12}$$

and in a more compact representation:

$$\xi_y = \frac{1}{1 - Q_1(1+\phi)Q_2(1+\phi)}\begin{bmatrix}\phi(1+\phi)Q_1Q_2 & \phi Q_1\\ \phi Q_2 & \phi(1+\phi)Q_1Q_2\end{bmatrix} y \tag{13}$$

where $\xi_y = \begin{bmatrix}\xi_{y_1} \ \xi_{y_2}\end{bmatrix}^T$ and $y = \begin{bmatrix}y_1 \ y_2\end{bmatrix}^T$. y is always bounded for a nominal stable system. The convergent property of ξ_y is determined by the rest parts of the equation, which is actually the transfer function of the closed loop system in Figure 1 and Nyquist stability criterion can apply.

Given the subsystems $Q_1(1+\phi)$ and $Q_2(1+\phi)$ are stable, if the Nyquist plot of the open loop transfer function does not encircle the $(1,0)$ point on the complex plane, then the positive feedback interconnection is stable. The criterion is a graphical approach which might be less intuitive as well. But two related theorems of sufficient condition on the stability can be used:

Small gain theorem: the closed loop interconnection (9) is input-output stable, if the maximum norm of its loop gain suffices $||Q_1(1+\phi)Q_2(1+\phi)||_\infty < 1$ [16].

Geometrically, it means that the Nyquist plot is bounded inside the unit circle on the complex plane. If a nominal system suffices the condition, its co-simulation stability can be guaranteed with a constant extrapolation method (ZOH), of which the gain never exceeds 1, regardless of the macro-step Δt. This conservative condition is not always fulfilled by the nominal system. But one can reduce the loop gain from the configuration design to enhance the stability and the robustness of co-simulation. For example in mechanical system co-simulation, if the splitting interface is less stiff and the force is applied towards the part with a higher impedance (larger mass and stiffness value) in *FD coupling*, the simulation is more stable [8]. Furthermore, one can reduce the input-output variables to obtain an incorrect but convergent co-simulation system, which is better than a divergent system for an initial setup by trials and errors.

Passivity theorem: the closed loop interconnection (9) is stable if each subsystem $(1+\phi)Q_1$, $(1+\phi)Q_2$ is either strictly passive or output strictly passive with zero-state observable [16].

If each subsystem is stable and its Nyquist plot is within the right half plane. Geometrically the open loop transfer function of the interconnection has a phase-lag less than 180°, so the Nyquist plot of a positive feedback system does not encircle the $(1,0)$ point on the complex plane. Because the coupling of co-simulation subsystem induces an unavoidable phase-lag behavior, the passivity cannot be preserved. Physically, an energy is added to the system which makes co-simulation unstable.

The passivity theorem can also apply to a large scale system since a system is passive if it is constituted by passive subsystems in a parallel or closed loop form. However, the passivity cannot be guaranteed by a cascaded dynamic term. Indeed, an explicit coupling method in co-simulation is usually in a cascaded dynamic form. But we can conclude that in order to improve the co-simulation stability, the phase-lag behavior should be compensated by introducing phase-lead dynamics.

3 Improved coupling method by $\mathcal{H}_\infty$ synthesis

3.1 Problem formulation

The output and state error ξ_y, ξ_x are projections of the only error source ξ_u as discussed in Section 2. As a result, the design of improved coupling method in co-simulation can thus be peeled off from the closed-loop system. The idea is to

introduce dynamic feed-forward terms at the input-output connection that can somehow minimize ξ_u. The concept is illustrated by an error system in Figure 2, which is equivalent to Equation 5. This structure is inspired from the research work in modern signal reconstruction [23]. The dynamic terms K_1 and K_2 to be designed are embedded inside each simulation model and are solved by each local solver. Considering K_1 ,K_2 and the sampling-hold process can be in different steps ($\delta t_1, \delta t_2$ and Δt) and also variable-step solvers might be used, the system is therefore formulated in the continuous domain. The sampling-hold process can be approximated by a linear function $H^*(s)$ using a *Padé* approximation.

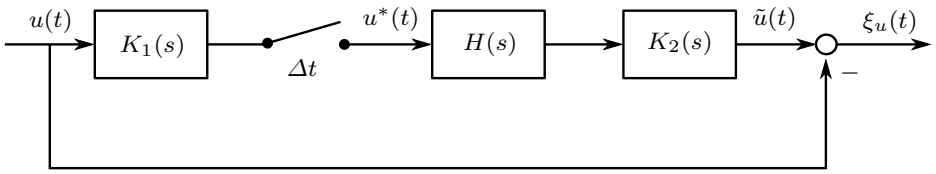


Fig. 2 The formulation of an input coupling error system.

It should be noticed that the exact input u is neither specified nor accessible. So the actual value of error ξ_u is unknown in practice. However, it can be known that if the introduced terms are projections in the *Laplace* domain fulfilling that $K_1(s)K_2(s)H^*(s) = 1$, then the error $\xi_u = 0$. Unfortunately the solution cannot be simply given by the inverse of $H^*(s)$ because the results are unstable and improper. Instead, we formulate the problem as minimization of the $\mathcal{L}_2$ norm of error $||\xi_u||_2$. The transfer function of the error system is denoted by T_{ue} such that:

$$||\xi_u||_2 = ||T_{ue}u||_2 \leq ||T_{ue}||_\infty ||u||_2 \tag{14}$$

the $\mathcal{L}_2$ norm of error $||\xi_u||_2$ is upper-bounded by the supremum norm of the error system. The optimal design problem of K_1, K_2 can thus be solved by the $\mathcal{H}_\infty$ synthesis framework to minimize the worst-case gain from the energy of u to ξ_u.

3.2 $\mathcal{H}_\infty$ synthesis for the coupling design

For $\mathcal{H}_\infty$ synthesis problem the error system needs to be transformed into a generalized plant P as shown in Figure 3. The input u is the disturbance input to the plant and ξ_u is the error output. The $\mathcal{H}_\infty$ synthesis problem can be formulated.

$\mathcal{H}_\infty$ Synthesis Problem: Given a LTI system P, find a feedback gain K such that the closed-loop system is stable and the following objective is satisfied:

$$||\xi_u||_2 < \gamma ||u||_2, \quad ||T_{ue}||_\infty := \sup_{Re\{s\}>0} ||T_{ue}(s)|| < \gamma \tag{15}$$

scalar γ is the level of guaranteed $\mathcal{L}_2$ gain performance to be minimized. The synthesized control law K is always stable, proper and causal, i.e. it only depends on the previous inputs, so it satisfies the condition for implementation.

Although a linear approximation is made for the sampling and hold $H^*(s)$, the solution is improved by partially fixing the structure. In this case, K_2 is a specified

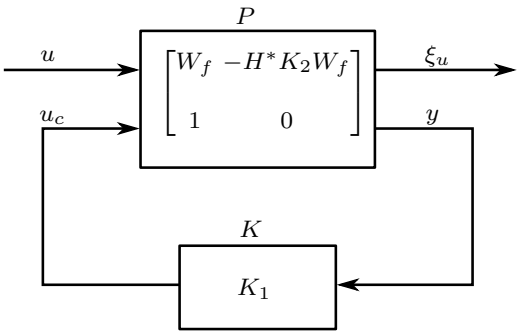


Fig. 3 The generalized plant P embedded with fixed terms H^*, K_2, W_f for $\mathcal{H}_\infty$ synthesis.

second order low-pass filter to smooth the constant piecewise signal. The feedback law K is actually the feed-forward term K_1 in the error system.

In addition, a weighted penalty function W_f, which is a low-pass filter, is added to the error output so that $||\xi_u||_2$ is more minimized in its lower frequency range. Since the low frequency content is more important considering the bandwidth of u and the existence of smoother K_2. In addition, it is crucial to introduce W_f for a feasible solution. Because $H^*(s)$ is always low-passed, the term $||\xi_u||_2$ cannot be minimized in the whole frequency range. Otherwise, either K_1 or K_2 must be improper and the control law cannot be implemented.

With the introduced dynamic terms expressed by the state-space realizations:

$$H^*(s) = \left[\begin{array}{c|c} A_h & B_h \\ \hline C_h & D_h \end{array}\right], \quad W_f(s) = \left[\begin{array}{c|c} A_w & B_w \\ \hline C_w & D_w \end{array}\right], \quad K_2(s) = \left[\begin{array}{c|c} A_{k2} & B_{k2} \\ \hline C_{k2} & D_{k2} \end{array}\right] \tag{16}$$

the generalized plant P is derived as:

$$P = \left[\begin{array}{c|c} A_p & B_p \\ \hline C_p & D_p \end{array}\right] = \left[\begin{array}{cccc|cc} A_w & 0 & 0 & 0 & B_w & 0 \\ 0 & A_h & 0 & 0 & 0 & B_h \\ 0 & B_{k2}C_h & A_{k2} & 0 & 0 & B_{k2}D_h \\ 0 & B_wC_hD_{k2} & B_hC_{k2} & A_w & 0 & B_wD_hD_{k2} \\ \hline C_w & -C_hD_{k2}D_w & -C_{k2}D_w & -C_w & D_w & -D_hD_{k2}D_w \\ 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right] \tag{17}$$

The solution of $\mathcal{H}_\infty$ synthesis problem is found using the Matlab robust control toolbox. The optimization is solved by the linear matrix inequalities (LMI) method with a pole-placement constraint. So the pole of the closed-loop system:

$$\frac{\xi_u}{u} = W_f - W_f H^* K_1 K_2 \tag{18}$$

can be bounded inside the design region. From the equation above, the pole of K_1 is a subset and bounded inside the region as well. In this case the optimal performance level γ is restricted but it gives two advantages for implementation:

1. The optimal term K_1 is always stable with the assigned solver. For example, K_1 can be constrained with its poles, i.e. eigenvalues, λ in the region $|1 + h\lambda| < 1$ so it can be solved by the Forward Euler method with step h, as well as by other methods of higher error order.

2. The fast mode of K_1 can be removed so that the method will not require small integration steps and increase much the computational time. Since simulation speed-up is a main goal to run co-simulation.

The synthesized dynamic feed-forward term K_1 is a transfer function of high-order. Mathematically it is similar to a linear combination of terms of multiple orders and their weights are given by the solution of the optimization problem. Indeed, the optimal K_1 is phase-lead to compensate the delay and low-passed behavior. K_1 and K_2 are similar as lead-lag compensators for loop-shaping in the classical control framework. The frequency behavior of the coupling error by the new coupling method $\xi_{u,h_\infty}(s)$ and other extrapolation methods $\xi_{u,zoh}(s)$, $\xi_{u,foh}(s)$ and $\xi_{u,soh}(s)$ are shown in Figure 4.

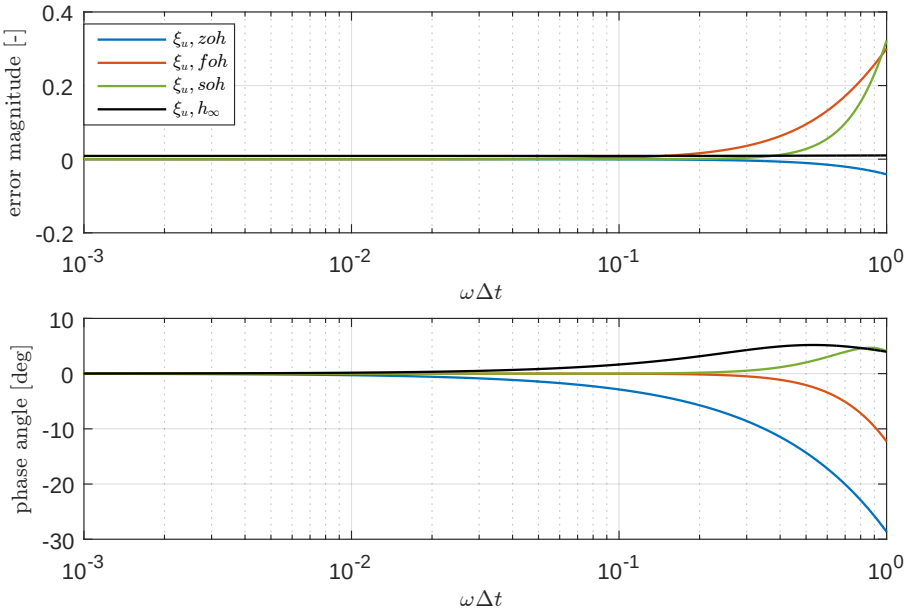


Fig. 4 Transfer behavior of the error with different coupling methods. Frequency is normalized by the macro-step Δt.

With a specified low-pass filter W_f and a constraint $Re(\lambda) > -500$, the achieved performance level $\gamma = 0.0138$. It should be noted that $\xi_{u,h_\infty}(s)$ is not attenuated in the whole frequency range, which means that at certain frequency the error might be still larger than the alternatives without a proper tuning. But the worst-case $||\xi_u||_2$ is upper bounded. Although the formulation of the synthesis problem is a bit sophisticated, the implementation of the method is not challenging: the $\mathcal{H}_\infty$ synthesis problem can be solved by Matlab functions. Two transfer functions K_1, K_2 can be added to each subsystem as plug-in to the model.

The parameter tuning of the method is very transparent. Smoother K_2 is tuned based on the bandwidth of input signal u. The weighting function W_f relates to the interesting frequency range of ξ_u which depends on input u and Δt. When a large macro-step Δt is taken, minimization of ξ_u in a high-frequency range should not be expected. The LMI pole-placement constraint of K_1 is designed based on the local solver and the desired modes. These needed information is usually well-known by the simulation engineers which enables the coupling design *a priori*.

4 Case Study

4.1 Co-simulation of a coupled dual mass-spring-damper system

The improved coupling design is verified first on a general experimental case: a dual mass-spring-damper system in two partitioned models. Each model is solved by Forward Euler solver at step $\delta t_{1,2} = 1$ ms. The coupling variables are force F_c and velocity $\dot{x}_2$ same as the configuration of *FD Coupling* and the macro-step $\Delta t = 50$ ms. Here the velocity variable $\dot{x}_2$ is taken instead, because it avoids the derivation error by the piecewise constant input. The mono-simulation with a single solver is generated as a reference and co-simulation with other coupling methods (ZOH, FOH and SOH) are implemented for a comparison. The $\mathcal{H}_\infty$ synthesis method is designed based on the size of Δt and W_f is low-passed above 3 Hz.

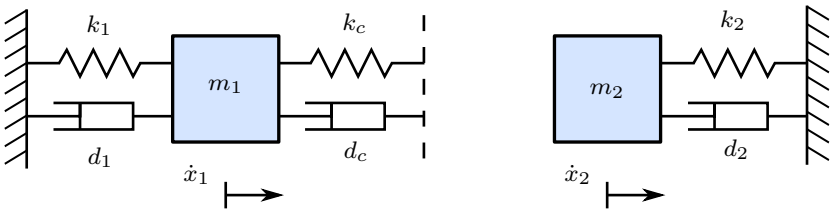


Fig. 5 The coupled dual mass damper spring system.

Since the performance of the coupling method is highly dependent on the system dynamics, a specific method maybe just optimal for one parameter setup but much worse in other cases. Especially in most engineering cases, the system dynamics is nonlinear or time-varying. Considering the combination effect of system parameters, the model is specified in a stochastic way for an unbiased setting. Each parameter is taken as an uniform distributed random variable. The stiffness $k_1, k_c, k_2 \in [0, 1000]$ N/s, masses $m_1, m_2 \in [0, 100]$ kg and damping coefficient d_1, d_c, d_2 are calculated giving a damping ratio in the range of $[0.3, 1.3]$. So it is possible to have systems with different stiffness, over-damped or under-damped behavior, highly asymmetric subsystems, etc. 2000 random cases have been simulated with the coupling methods fixed. Some cases which are unsolvable by the local solver due to an extremely small mass value are excluded. A simulation of impulse response is taken and two external force impulse is applied to m_2 during the simulation period of 5 s.

The results of the coupling variables $\dot{x}_2$ and F_c are compared. As the output is available every Δt, the error compared with the mono-simulation reference is unavoidable. The $\mathcal{L}_2$ norm of the error by each coupling method is compared. The criterion for accuracy comparison is that, if one method fulfills:

$$||\xi_{\dot{x}_2}||_2 > \eta_1 ||\xi_{\dot{x}_2}||_{2,min} \quad \text{or} \quad ||\xi_{F_c}||_2 > \eta_1 ||\xi_{F_c}||_{2,min} \tag{19}$$

an inaccurate case is counted, where $||\xi_{\dot{x}_2}||_{2,min}, ||\xi_{F_c}||_{2,min}$ are the minimum error of all the methods and η_1 is a threshold value. The consideration is that when different methods give very similar results, their differences are less critical. But if the method has an error much larger than the optimum, this indicates a quite different case with unreliable results. The statistic results with different threshold value η_1 are given in Table 1. It can be seen that $\mathcal{H}_\infty$ method is robust to the change of system parameters and has fewer cases with inaccurate results. The

alternative methods have more unreliable cases, which might be due to imprecision by a low degree extrapolation (ZOH) or less robustness by a high degree extrapolation (SOH).

Table 1 Number of cases violating the accuracy criterion.

Coupling method	ZOH	FOH	SOH	$\mathcal{H}_\infty$
$\eta_1 = 1.5$	1005	194	1570	111
$\eta_1 = 3$	86	41	243	32
$\eta_1 = 5$	15	38	118	22
$\eta_1 = 10$	4	34	90	16

The stability of all the simulation cases is checked by comparing the results in the end condition. Because an unstable simulation would have divergent simulation results. The comparing criterion is that if one method fulfills:

$$|\dot{x}_2(T)| > \eta_2 |\dot{x}_{2,ref}(T)| \tag{20}$$

then the simulation is likely to be unstable, where T is the end time and η_2 is the threshold value. The statistic result with different threshold value η_2 is shown in Table 2. A small η_2 gives a more conservative criterion and might involve stable but less accurate cases. It can be seen that $\mathcal{H}_\infty$ synthesis method has less likely unstable cases than the other methods ($\eta_2 = 5, 10$) or similar number of largely divergent cases as the ZOH method ($\eta_2 = 100, 1000$). It can also be seen that, the unstable cases increases with the extrapolation degree in general.

Table 2 Number of cases violating the stability criterion.

Coupling method	ZOH	FOH	SOH	$\mathcal{H}_\infty$
$\eta_2 = 5$	95	83	134	56
$\eta_2 = 10$	64	71	122	46
$\eta_2 = 100$	26	54	101	32
$\eta_2 = 1000$	18	41	82	23

The time plot of $\dot{x}_2$. F_c of several simulation cases are shown in Figure 6-8. Another observation from the time plot is that even in a single case with deterministic system parameters, one coupling method is difficult to be the optimum for both coupling variables. Because the interface dynamics can be quite different which may need different coupling methods. This adds the complexity to evaluate their performance with the close-loop system or a more complex case. Instead, in real application the robustness of the coupling method is desired.

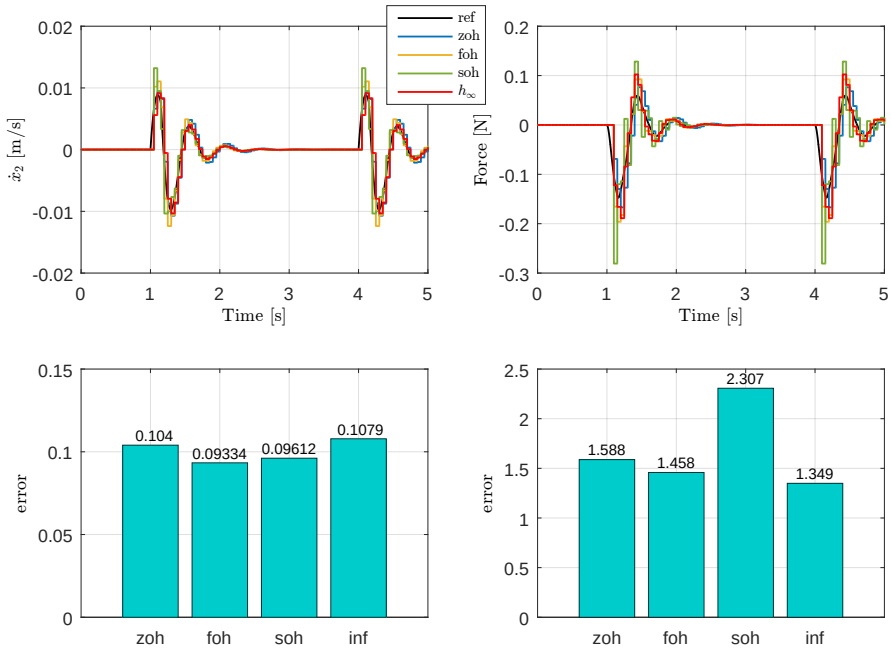


Fig. 6 Time response of $\dot{x}_2, F_c$ and the $\mathcal{L}_2$ norm of the error. In this case SOH violates the criterion with $\eta_1 = 1.5$).

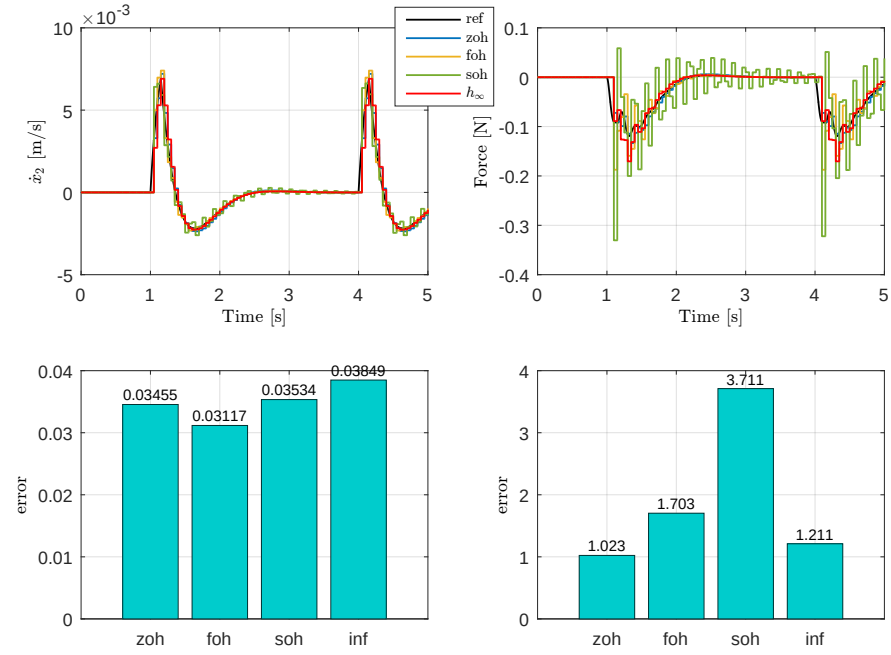


Fig. 7 Time response of $\dot{x}_2, F_c$ and the $\mathcal{L}_2$ norm of the error. In this case SOH violates the criterion with $\eta_1 = 3$ and FOH violates the criterion with $\eta_1 = 1.5$.

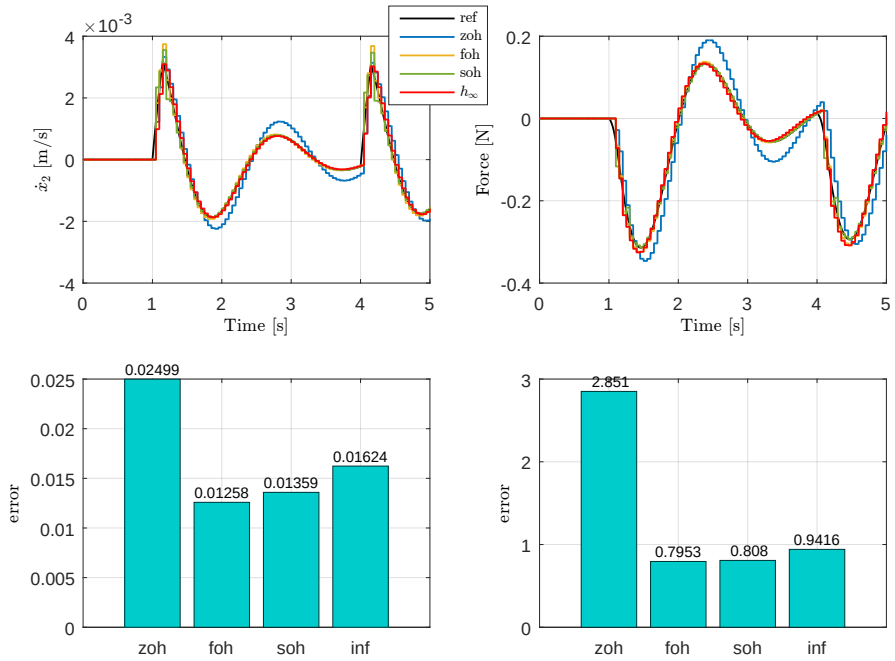


Fig. 8 Time response of $\dot{x}_2, F_c$ and the $\mathcal{L}_2$ norm of the error. In this case ZOH violates the criterion with $\eta_1 = 3$.

4.2 Co-simulation of a MBD vehicle chassis and EPAS model

The second experimental case is a co-simulation of MBD chassis model and an EPAS model. The MBD chassis model is computationally heavy for each integration step due to a large number of states. However, like many mechatronic systems the EPAS model is numerically stiff due to components with a large inertia ratio, highly under-damped dynamics and friction elements, which requires $\delta t < 1$ ms for a fixed-step solver. In addition, the EPAS model is tightly coupled with the ECU software which is calculated every 1 ms.

Therefore, co-simulation is applied in this typical case of vehicle-mechatronic systems coupling. The numerical efficiency can be improved by assigning tailored solver and macro-step to each model, which is compiled into a FMU for co-simulation [2]. The setup of co-simulation is shown as Figure 9. The vehicle chassis model and EPAS model are coupled by steering rack force and rack velocity due to the same reason as the first case. More detailed information on the system modeling and analysis on the simulation speed-up is explained in the previous work by author [9]. In this manuscript, the new coupling method is applied to see if the error can be reduced when a large macro-step is taken.

For the test, a steering torque input is applied to the steering wheel of EPAS model at both low frequency up to 1 Hz and high frequency up to 3 Hz which is close to the limit of a human driver. The steering wheel angle, steering rack speed, vehicle yaw rate and vehicle lateral velocity of each simulation cases are shown in Figure 11 and Figure 12. For a better comparison, a normalized root mean square

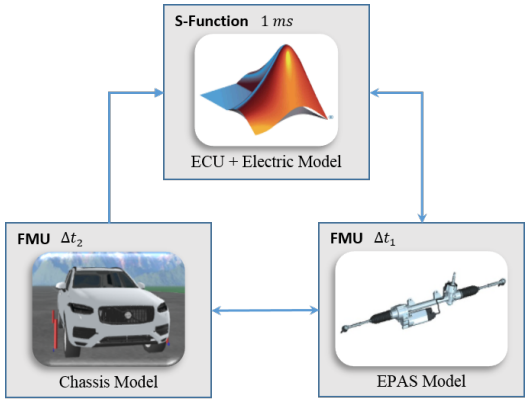


Fig. 9 The layout of co-simulation of vehicle chassis model, EPAS model in FMU and ECU software in matlab s-function.

(NRMS) error of each variable is calculated:

$$\varepsilon_{nrms,x} = \frac{\sqrt{\sum_{t=0}^{T}\left((x(t) - x_{ref}(t)\right)^2/T}}{x_{ref,max} - x_{ref,min}} \tag{21}$$

where $x_{ref,max}$, $x_{ref,min}$ are the maximum and minimum value of the state from the mono-simulation reference and the results are plotted in Figure 10. The SOH method gives inaccurate results with large deviations showing a lack of robustness and stability. The method by $\mathcal{H}_\infty$ synthesis gives more accurate results than the other two methods both in the low frequency and the high frequency steering test. In the low frequency test (Figure 11), there is no significant error for all the methods, so the improvement on accuracy looks a bit saturated by the inherent error between data in a larger step and reference data in a smaller step. In the high frequency test (Figure 12), FOH method gives an oscillatory result. However, the new method shows both an oscillation depression and accuracy improvement.

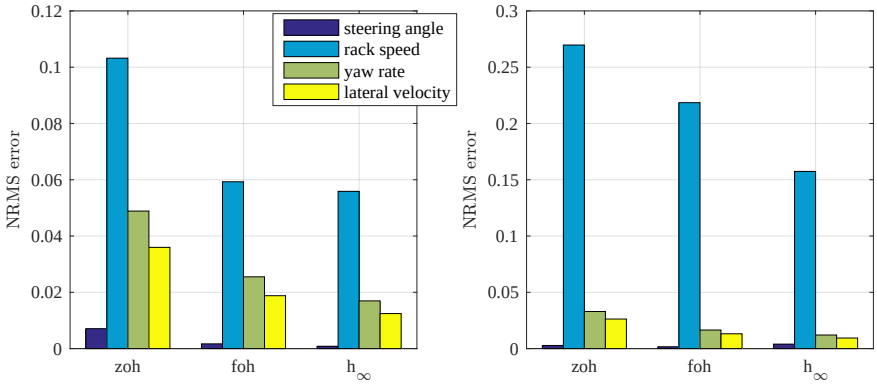


Fig. 10 NRMS error of the simulation results with a low frequency (left) and a high frequency (right) steering torque input. Results of SOH method is not plotted due to a large deviation $\varepsilon_{nrms,x} > 1$.

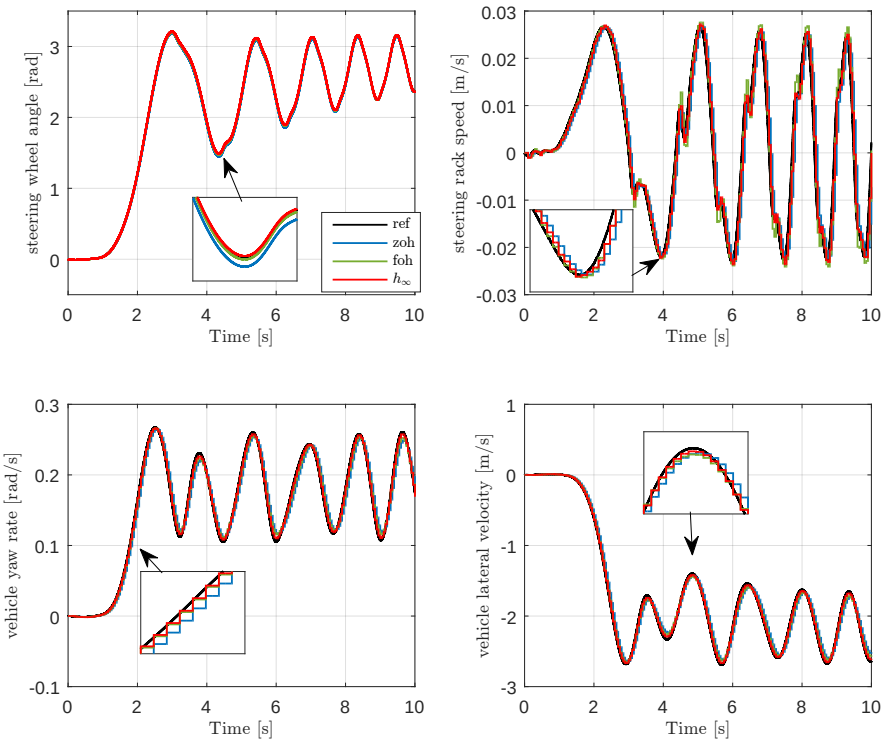


Fig. 11 Simulation results of low frequency steering torque input. Results of SOH method is not plotted due to a large deviation.

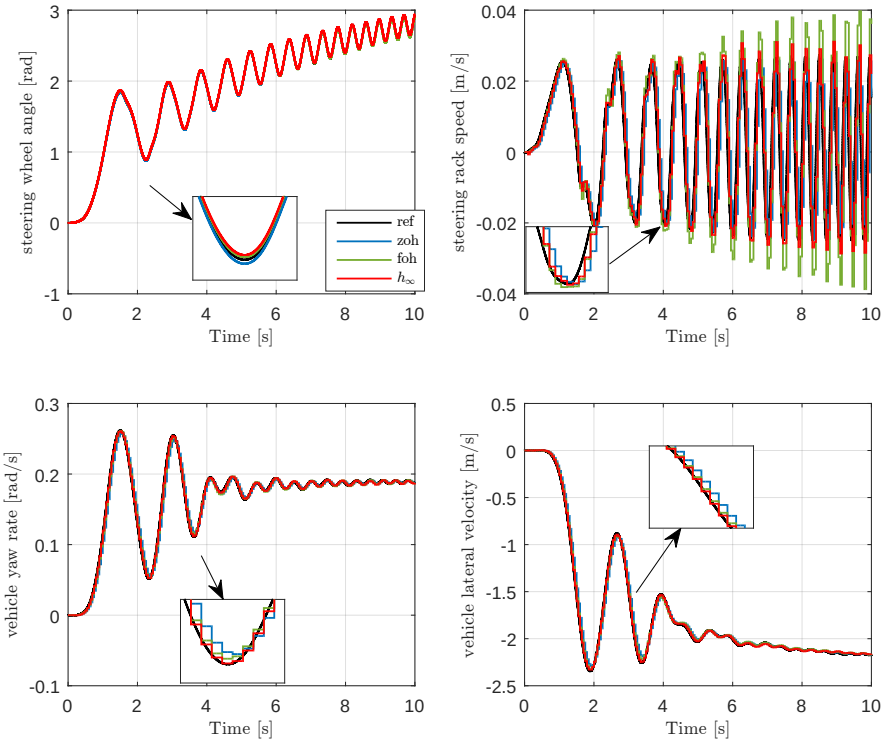


Fig. 12 Simulation results of high frequency steering torque input. Results of SOH method is not plotted due to a large deviation.

5 Conclusion

In this manuscript, the explicit parallel co-simulation approach has been reviewed. The numerical stability and robustness has been interpreted by formulating the co-simulation system as a closed-loop interconnection with a multiplicative disturbance, which is the coupling error. The subsystem dynamics can be analyzed by the apparent transfer behavior rather than single system parameters. In the linear robust control framework, the stability and robustness depends on the loop gain of the co-simulation system and several complete theorems can apply. This might bring more engineering sense as an intuitive guide tool.

Based on the understanding from above, a new explicit coupling method has been developed. We formulated the coupling design as a signal reconstruction problem by $\mathcal{H}_\infty$ synthesis to reduce the worst-case $\mathcal{L}_2$ norm of the coupling error in *Laplace* domain. From two experimental cases, the new method has shown potential in accuracy improvement and also robustness to system dynamics, which has not been addressed explicitly in many other methods. However, robustness is usually needed in co-simulation of a complex system. The method will be experimented on more co-simulation cases in the future work.

Disclosure statement

No potential conflict of interest was reported by the authors.

Acknowledgement

The project leading to this study has received funding from ITEAM project in the European Union Horizon 2020 research and innovation program under Marie Sklodowska-Curie Grant Agreement No. 675999. The author would like to show gratitude to Prof. Fredrik Bruzelius and Dr.Maliheh Sadeghi Kati for discussions on $\mathcal{H}_\infty$ synthesis problem.

References

1. Andersson, C.: Methods and Tools for Co-Simulation of Dynamic Systems with the Functional Mock-up Interface. Ph.D. thesis, Lund University (2016)
2. Arnold, M., Clauss, C., Schierz, T.: Error analysis and error estimates for co-simulation in fmi for model exchange and co-simulation v2. 0. In: Progress in Differential-Algebraic Equations, pp. 107–125. Springer (2014)
3. Ben Khaled, A., Ben Gaid, M., Simon, D., Font, G.: Multicore simulation of powertrains using weakly synchronized model partitioning, vol. 45. IFAC (2012)
4. Benedikt, M., Watzenig, D., Hofer, A.: Modelling and analysis of the non-iterative coupling process for co-simulation. Mathematical and Computer Modelling of Dynamical Systems **19**(5), 451–470 (2013)
5. Benedikt, M., Watzenig, D., Zehetner, J., Hofer, A.: NEPCE - A nearly energy-preserving coupling element for weak-coupled problems and co-simulations. V International Conference on Computational Methods for Coupled Problems in Science and Engineering pp. 1–12 (2013)

6. Blockwitz, T., Otter, M., Akesson, J., Arnold, M., Clauss, C., Elmqvist, H., Friedrich, M., Junghanns, A., Mauss, J., Neumerkel, D., Olsson, H., Viel, A.: Functional Mockup Interface 2.0: The Standard for Tool independent Exchange of Simulation Models pp. 173–184 (2012)
7. Busch, M.: Zur effizienten Kopplung von Simulationsprogrammen. Ph.D. thesis, Kassel University (2012)
8. Chen, W., Ran, S., Jacobson, B.: Design of Interface in Co-simulation for Electric Power Assisted Steering System Development. Proceedings of the 14th International Symposium on Advanced Vehicle Control (AVEC' 18) (2018)
9. Chen, W., Ran, S., Jacobson, B.: Integration and Analysis of EPAS and Chassis System in FMI-based co-simulation. In: 13th International Modelica Conference 2019 (2019)
10. Drenth, E.: Robust co-simulation methodology of physical systems. In: 9th Graz Symposium Virtual Vehicle (2016)
11. Feki, A.B.K.e.: Distributed real-time simulation of numerical models : application to power-train. Ph.D. thesis, University of Grenoble (2014)
12. Gallrein, A., Baecker, M., Burger, M., Gizatullin, A.: An advanced flexible realtime tire model and its integration into fraunhofer's driving simulator. SAE Technical Papers **1** (2014)
13. Gomes, C., Thule, C., Broman, D., Larsen, P.G., Vangheluwe, H.: Co-simulation: State of the art (2017)
14. González, F., Naya, M.Á., Luaces, A., González, M.: On the effect of multirate co-simulation techniques in the efficiency and accuracy of multibody system dynamics. Multibody System Dynamics **25**(4), 461–483 (2011)
15. Khaled, A.B., Duval, L., Gaid, M.B., Simon, D., Bois-préau, D.: Context-based polynomial extrapolation and slackened synchronization for fast multi-core simulation using FMI pp. 225–234 (2014)
16. Khalil, H.K.: Nonlinear systems, vol. 3. Prentice hall (2002)
17. Kübler, R., Schiehlen, W.: Two Methods of Simulator Coupling. Mathematical and Computer Modelling of Dynamical Systems **6**(2), 93–113 (2000)
18. Sadjina, S., Pedersen, E.: Energy Conservation and Coupling Error Reduction in Non-Iterative Co-Simulations (7491) (2016)
19. Schweizer, B., Li, P., Lu, D.: Explicit and Implicit Cosimulation Methods: Stability and Convergence Analysis for Different Solver Coupling Approaches The. ZAMM Zeitschrift fur Angewandte Mathematik und Mechanik **96**(8), 986–1012 (2016)
20. Schweizer, B., Lu, D.: Predictor/corrector co-simulation approaches for solver coupling with algebraic constraints. ZAMM Zeitschrift fur Angewandte Mathematik und Mechanik **95**(9), 911–938 (2015)
21. Stettinger, G., Horn, M., Benedikt, M., Zehetner, J.: Model-based coupling approach for non-iterative real-time co-simulation. In: 2014 European Control Conference (ECC), pp. 2084–2089. IEEE (2014)
22. Wu, C.: Co-simulation Methods for EPAS and Chassis Systems Development. Master's thesis, Chalmers University of Technology (2018)
23. Yamamoto, Y., Nagahara, M., Khargonekar, P.P.: Signal Reconstruction via H-infinity Sampled-Data Control Theory: Beyond the Shannon Paradigm. IEEE Transactions on Signal Processing **60**(2), 613–625 (2012)

Paper IV

Real-time Co-simulation Method Study for Vehicle Steering and Chassis System

Real-time Co-simulation Method Study for Vehicle Steering and Chassis System

W. Chen*/. M. Klomp*/**. S. Ran***

** Department of Vehicle Dynamics CAE, Volvo Car Corporation, Gothenburg, Sweden.*
*** Chalmers University of Technology, Gothenburg, Sweden.*
(e-mail: weitao.chen@volvocars.com).

Abstract: Real-time co-simulation is widely used in complex system development. This paper presented an application of our driver-in-the-loop simulator, the vehicle steering and chassis system are implemented on different real-time machines coupled with input-output signals. Results inconsistent with the mono-simulation reference may be generated due to the real-time co-simulation drawbacks: the delay effect and coupling errors from modular integration. To overcome the drawbacks, a coupling element as an additional subsystem has been proposed by the author. The coupling element consists of a delay compensation part based on adaptive filters and a coupling error correction part based on the energy-preserving method. This coupling method has been tested with the vehicle steering and chassis model. More stable and consistent results are obtained. The frequency domain analysis and implementation method have been provided.

Keywords: Real-time co-simulation, adaptive filters, steering and chassis system

1. INTRODUCTION

Vehicle development has become more complex than ever due to the increase of electrification and driving automation. Mono-simulation is unable to fully satisfy the system-level development work. Instead, co-simulation has gain significant interest in the industry as it allows parallel development work and an efficient usage of the domain-specific solvers for multidisciplinary problems. For instance, a vehicle could be co-simulated by a multibody chassis system and a high-fidelity electric power assisted steering (EPAS) system. In our driver-in-the-loop simulator the two systems are implemented on different real-time machines which adds a synchronization problem to the modular time integration problem in co-simulation.

The co-simulated systems are coupled by the input-output data exchange at given time instants as shown in Figure1. During the communication interval, the macro-step Δt, the subsystems are integrated at the micro-steps δt with unknown inputs. These systems are called weakly-coupled and their inputs during a macro-step need to be extrapolated from the latest exchanged values, which causes errors. Therefore, even without an algebraic loop, co-simulation may not give reliable and consistent results when compared to the mono-simulation. The stability and accuracy depends not only on the local solvers but also the co-simulation scheme, macro-step Δt size and the extrapolation method, see Busch, M et al. (2012).

An iterative scheme with recalculation for the whole macro-step can get stable and precise results, see Kübler, R (2000). By exchanging additional Jacobian matrices of subsystems and implicit integration, macro-step Δt can be increased for stable co-simulation, see Viel, A (2014). However, iterative methods are challenging for real-time application due to a high computational burden and it is difficult to implement with limited software support. The non-iterative scheme, especially the Jacobi scheme (Fig.1), although is less accurate and stable, is still more preferred in the industry for easy implementation and fast calculation. Interesting extrapolation solutions for non-iterative scheme have been provided by Benedikt, M et al. (2013) and Drenth, E (2017) based on the concept of preserving the energy of power bond. In this approach only input-output data at macro-step Δt is needed. Sadjina, S et al. (2017) has further extended the method to an adaptive macro-step size control for accuracy and calculation efficiency. The energy approach is used by the author for modular integration problem and will be discussed in the later section.

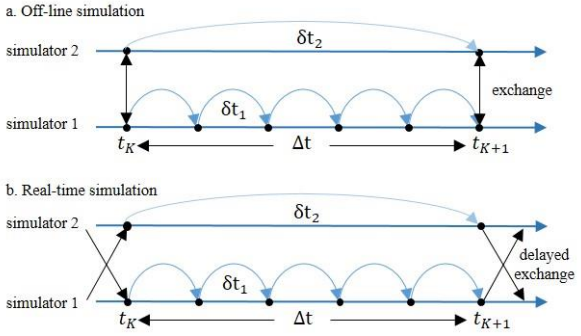


Fig. 1. Off-line and real-time co-simulation of two coupled systems by the Jacobi scheme.

In real-time co-simulation the subsystems must be solved within a fixed time frame, usually by fixed-step solvers, so that the simulation time is synchronized with the wall-clock time. The subsystems are coupled together by input-output signals transmitted at finite transmission speeds. The resulting time delay may induce unsynchronized communication and induce oscillatory or even unstable results. To synchronize the data, a model-based coupling approach has been applied for real-time co-simulation, see Stettinger, G et al. (2014).

1.1 Steering System and Chassis System Co-Simulation

In our driving simulator the EPAS system and the chassis system are weakly-coupled. They are running in parallel on different real-time machines due to different software tools. The EPAS model includes the steering mechanism, a detailed electric motor and an ECU model in black-box manner. The EPAS model is integrated at a fixed-time step of 1 ms (micro-step δt_1). The detailed chassis model is integrated at a fixed-time step of 5 ms (micro-step δt_2) due to the complexity of the multibody model. The two systems are coupled by steering rack velocity and the steering rack force. The steering rack force is taken as input to the EPAS system with steering rack velocity taken as input to the chassis system. This flow direction contributes to the numerical stability since the EPAS system rack has a much higher impedance than the chassis system, see Drenth, E (2016). The communication macro-step Δt is chosen as its feasible minimum value, namely the bigger micro-step δt_2. The coupling signals are transmitted in real-time with a delay τ.

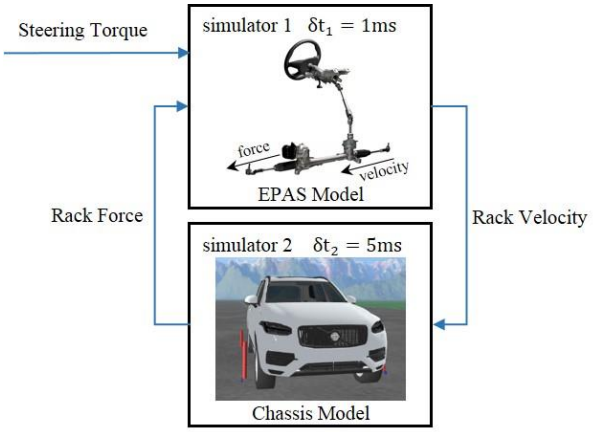


Fig. 2. EPAS steering and chassis system co-simulation.

1.2 Paper Structure

Two main problems needs to be addressed to achieve reliability and consistency in the presented real-time co-simulation case: unsynchronized signals due to delay τ and errors from modular integration. Other issues such as packet loss will not be discussed in this paper. The paper provides a comprehensive solution: a coupling element consists of a delay compensation part and a coupling error correction part. The element is implemented on the real-time machine with the EPAS system and able to run at fixed-time step of 1 ms as shown in Figure 3.

The delay compensation part based on the online identification by adaptive filters will be explained in Section 2. After the signals are synchronized by the compensation part, a linear interpolation and an error correction based on the energy preserving method is used which will be discussed in Section 3. In the end, the implementation method and the simulation results will be shown in Section 4.

2. TRANSMISSION DELAY COMPENSATION

2.1 Adaptive Filters for System Identification

The time delay will vary in both directions of signal transmission. For simplicity the time delay is approximated as being constant $\tau = 2\Delta t$. At the receiving end, the delayed input $u_{k-\tau}$ should be extrapolated to $\hat{u}_k$ for the integration at the current step (Fig. 3). At the sending end, the output y_k also needs to be extrapolated to $\hat{y}_{k+\tau}$ to compensate the transmission time to the receiving simulator.

The signals can be extrapolated based on the dynamics of the coupled systems which is only partially known. As the input-output signals are accessible, the system dynamics can be online identified by adaptive filters. Recursive least squares (RLS) and the extended Kalman filter (EKF) are well-studied algorithms, see Ljung, L (1979) and Ljung, L (2002). To extrapolate the outputs from both systems, the chassis system is identified by an adaptive filter at receiving end and EPAS system is identified by a second filter at the sending end.

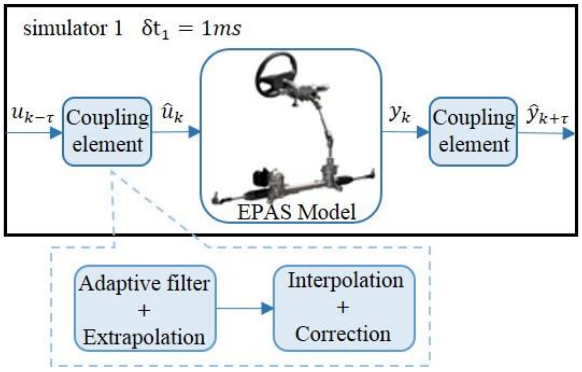


Fig. 3. Coupling elements added to the EPAS model.

2.2 RLS Algorithm

Because most mechanical systems show behaviours like second-order systems. To online identify the unknown system, a simplified baseline model is defined as (1).

$$y_k = -a_{1,k}y_{k-1} - a_{2,k}y_{k-2} + b_{1,k}u_{k-1} + b_{2,k}u_{k-2} = \varphi_k^{\mathrm{T}}\theta_k \quad (1)$$

with $\varphi_k = [-y_{k-1}, -y_{k-2}, u_{k-1}, u_{k-2}]^T$

$\theta_k = [a_{1,k}, a_{2,k}, b_{1,k}, b_{2,k}]^T$

φ_k contains observed signals which are inputs and outputs of the system. θ_k contains unknown baseline model parameters. By minimizing the weighted linear least squares cost function (2), θ_k can be estimated at every time step k so that the time varying model (1) can locally track the input-output behaviour of the identified system.

$$V_k = \sum_{i=1}^{k} \lambda^{k-i}\left(y_i - \varphi_i^{\mathrm{T}}\theta_i\right)^2 \quad (2)$$

The minimization is achieved by RLS algorithm summarized as (3), see Ljung, L (2002).

$$L_k = P_{k-1}\varphi_k(\lambda + \varphi_k^T P_{k-1}\varphi_k)^{-1} \tag{3a}$$

$$P_k = \lambda^{-1}(P_{k-1} - L_k\varphi_k^T P_{k-1}) \tag{3b}$$

$$\hat{\theta}_k = \hat{\theta}_{k-1} + L_k(y_k - \varphi_k^{\mathrm{T}}\hat{\theta}_{k-1}) \tag{3c}$$

P_k is the covariance matrix of the parameter estimation error. $\hat{\theta}_k$ is the estimated parameter vector. λ is the forgetting factor and smaller λ gives less weight to older observations.

2.3 EKF Algorithm

The predefined baseline model (1) can be represented in state-space (4) with state vector $x_k = [y_{k-2}, y_{k-1}]^T$ and input vector $u'_k = [u_{k-2}, u_{k-1}]^T$.

$$x_{k+1} = \begin{bmatrix} 0 & 1 \\ -a_{2,k} & -a_{1,k} \end{bmatrix} x_k + \begin{bmatrix} 0 & 0 \\ b_{2,k} & b_{1,k} \end{bmatrix} u'_k \tag{4a}$$

$$y_k = [0\ \ 1]x_k \tag{4b}$$

The extended Kalman filter (EKF) can be used for parameter estimation by augmenting the state vector x_k with parameter vector θ_k, see Ljung, L (1979). Then it becomes a state estimation problem. The augmented state-space is written as (5):

$$z_{k+1} = \begin{bmatrix} x_{k+1} \\ \theta_{k+1} \end{bmatrix} = f_k(z_k, u'_k) + w_k = \begin{bmatrix} z_{2,k} \\ -z_{4,k}z_{1,k} - z_{3,k}z_{2,k} + z_{6,k}u'_{1,k} + z_{5,k}u'_{2,k} \\ z_{3,k} \\ z_{4,k} \\ z_{5,k} \\ z_{6,k} \end{bmatrix} + w_k \tag{5a}$$

$$y_k = h_k(z_k) + v_k = z_{2,k} + v_k \tag{5b}$$

The first two states are driven by the process model and the augmented states vary with time as a sort of random walk. w_k and v_k are process and measurement noises which are assumed to be zero mean with covariance matrix Q_k and R_k. The EKF algorithm consists of the prediction and correction step (6):

$$\hat{z}_k^- = f_{k-1}(\hat{z}_{k-1}, u'_{k-1}) \tag{6a}$$

$$P_k^- = F_{k-1}P_{k-1}F_{k-1}^T + Q_{k-1} \tag{6b}$$

$$K_k = P_k^- H_k^T(H_k P_k^- H_k^T + R_k)^{-1} \tag{6c}$$

$$P_k = (I - K_k H_K)P_k^- \tag{6d}$$

$$z_k = \hat{z}_k^- + K_k(y_k - h_k(\hat{z}_k^-)) \tag{6e}$$

F and H are Jacobian matrices of f and h with predicted states in every step. The augmented state vector z_k is unobservable but the purpose is to estimate $\hat{\theta}_k$ that provides the same input-output behaviour. So whether each parameter is distinguishable should not be a concern.

2.4 RLS and EKF Tuning

The rate of convergence of the adaptive filters influences the simulation stability. In initial condition, parameter estimate $\hat{\theta}_0$ is set as zero with a large estimate covariance P_0 which means the initial guess is highly uncertain and the convergence to the correct value is fast. When the estimator is triggered by fast-changing signals, a divergent co-simulation may be generated. So it is desired that the estimator is slowly-excited at the beginning.

The estimator should be set with high tracking ability for systems with high-frequency outputs. A forgetting factor $\lambda < 1$ for the RLS filter is preferred so the estimation $\hat{\theta}$ could be fast adapted. Similarly the EKF is tuned with a small measurement noise covariance R_k to increase the feedback to achieve a higher bandwidth. If observed signals are test-rig measurements corrupted with noises, noise suppression is needed. In this case the forgetting factor λ should be increased or equals to 1 for RLS method. For EKF method the measurement covariance R_k could be derived from measurements and the terms corresponding to x_{k+1} in covariance matrix Q_k should be small. Then the prediction is more trusted and the filter behaves more as an open-loop predictor with compromised tracking ability.

In the presented case the estimator is tuned in an ad-hoc manner due to unknown noise structure of uncertain dynamics. The observed signals are noise-free simulation data so the adaptive filter are tuned for higher rate of convergence.

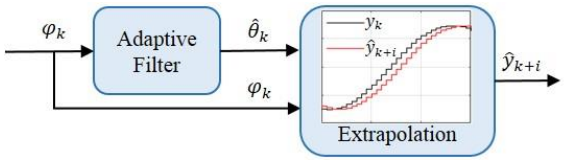


Fig. 4. Parameter identification and signal extrapolation.

2.5 Signal Extrapolation

The signal extrapolation is based on the identified baseline model (1) and input-output data which can be written as:

$$\hat{y}_{k+i} = [-\hat{y}_{k-1+i}, -\hat{y}_{k-2+i}, u_{k-1}, u_{k-2}]\,\hat{\theta}_k \tag{7}$$

i is the number of extrapolation step and $\hat{y}_{k+i}$ can be calculated iteratively from the previous values.

It must be noticed that the parameter $\hat{\theta}_k$ estimated by the RLS method (3) or the EKF method (6) varies with the sample time of the filter. At the receiving end, rack force data is updated every micro-step $\delta t_2 = 5$ ms. So $\hat{\theta}_k$ is correctly estimated at the respective sample time and two extrapolation steps ($i = 2$) are needed for the time delay $\tau = 10$ ms. At the sending end, $\hat{\theta}_k$ is estimated according to rack velocity data which is updated every 1 ms. An extrapolation for 10 steps ($i = 10$) is therefore needed. The smaller sample time allows more precise identification but on the other hand the extrapolation error increases due to more extrapolation steps.

To investigate if a more precise identification has a gain over more extrapolation steps, the method has been tested with feasible sample times of 1 ms, 5 ms and 10 ms. For an easy comparison, same RLS filter has been used. Random inputs updated at different sample times have been sent to a black-box model. The reference output is compared to the extrapolated values from the filters given delayed signals. The normalized root-mean-square (RMS) errors are shown in Figure 5. It can be seen that the extrapolation error is smaller when the sample time of the filter and signals are the same. The difference is less obvious at small sample time because the precision of identification is reduced by more extrapolation steps.

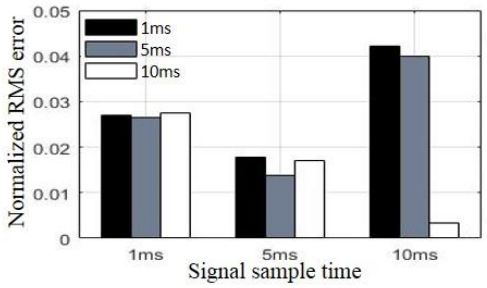


Fig. 5. Extrapolation errors of adaptive filters.

Besides the errors, the extrapolation method has been further checked in frequency domain. Same input has been sent to filters updated at 1ms and 5ms. As shown in Figure 6, the extrapolation at smaller sample time adds more high-frequency components and more energy to the reference signal. The overall system would be less stable. Consequently, 5ms sample time is chosen for extrapolation at both receiving and sending ends.

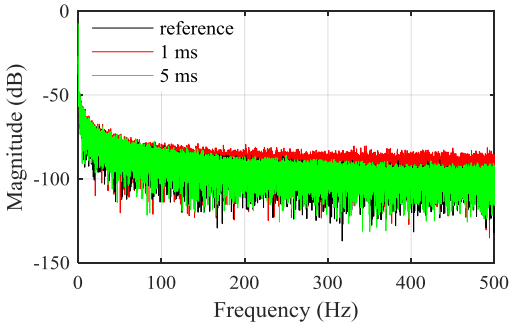


Fig. 6. Frequency spectrum of extrapolated signals.

3. INTERPOLATION AND ERROR CORRECTION

3.1 Linear Interpolation and correction

After signal synchronization by the delay compensation part, the modular integration problem remains: the input to the EPAS model during a macro-step Δt is unknown. Zero-order hold (ZOH) gives a large piecewise input which may excite the fast dynamics of the system and make the result oscillatory. A smooth input signal $\hat{y}_{k|n}$ can be obtained by a linear interpolation based on the current large step value $\hat{y}_k$ and the prediction for next large step $\hat{y}_{k+1}$, as shown in (8).

$$\hat{y}_{k|n} = \hat{y}_k + n(\hat{y}_{k+1} - \hat{y}_k)\delta t_1/\Delta t \tag{8}$$

n denotes the smaller step number within a macro-step. The interpolation behaves as a feedforward smoother. To avoid the change of signal energy, the integration of the interpolated signal in a specified time range should be same as the large step signal. So an error feedback based on the backward cumulative moving average is added (Fig.7). The feedback term has been applied by Benedikt, M et al. (2013) in the nearly energy preserving coupling element (NEPCE) for a stable co-simulation but it requires more previous coupling data.

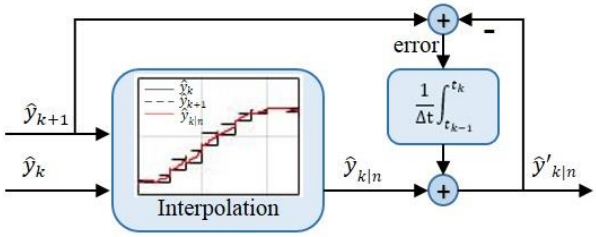


Fig. 7. Linear interpolation and error correction.

3.2 Energy Preserving Method in Co-Simulation

The error estimation in non-iterative co-simulation is difficult to implement because normally the system dynamic equations and the reference are unavailable. The residual energy of a power bond in energy preserving method can be used as an error order indicator, see Sadjina, S et al. (2017). The energy-preserving method is based on the conservation law of energy: the power of input-output is same. In our case the mechanical power sent by the chassis system should be equal to the power received by the steering system, which is also correct for a strongly-coupled system because coupled input-output signals are always unchanged.

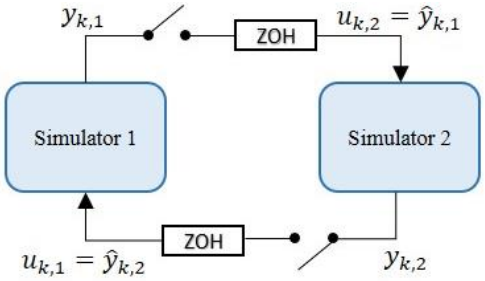


Fig. 8. Block diagram of co-simulated system.

For a weakly-coupled system (Fig. 8), however, input and output can be different due to multiple rates. The conservation law of energy is therefore violated: a residual energy δE_k can be added to the system virtually and make the simulation result incorrect or unstable. The residual power δP_k and the residual energy δE_k during the macro-step k can be calculated as:

$$\delta P_k = u_{k,1}y_{k,1} - u_{k,2}y_{k,2} = \hat{y}_{k,2}y_{k,1} - \hat{y}_{k,1}y_{k,2} \quad (9a)$$

$$\delta E_k = \int_{t_k}^{t_{t+1}} \delta P_k \, dt = \int_{t_k}^{t_{t+1}} (\hat{y}_{k,2}y_{k,1} - \hat{y}_{k,1}y_{k,2}) \, dt \quad (9b)$$

When a small integration of each signal is same, the integration of their product is consistent so the residual energy δE_k gets reduced.

4. SIMULATION RESULTS AND DISCUSSION

4.1 Coupling Element Implementation

The coupling element designed from a single-input single-output (SISO) model (1) usually connects with a multi-input multi-output (MIMO) system. It is proposed that the two coupling elements should first be connected in a way less dependent on each other for high robustness. For example the element at the sending end can take the steering torque as input rather than the extrapolated rack force signal. Secondly, a continuous-changing signal is preferred to make the adaptive filter work properly. The steering angle output is more robust than the steering torque input because the latter has higher bandwidth and may have a sudden change.

4.2 Simulation Test and Results

The proposed real-time co-simulation method has been tested with a detailed EPAS model and a chassis model solved at different micro-steps δt_1 and δt_2. The tests are done offline, so in order to mimic the transmission delay effect, a constant delay τ has been added to the coupled signals.

As can be seen from Figure 8, a sine swept steering torque has been applied to the EPAS model. The rack force in the co-simulation (b) has a larger oscillation. The co-simulation with coupling elements (c) shows a more consistent result with the mono-simulation reference (a).

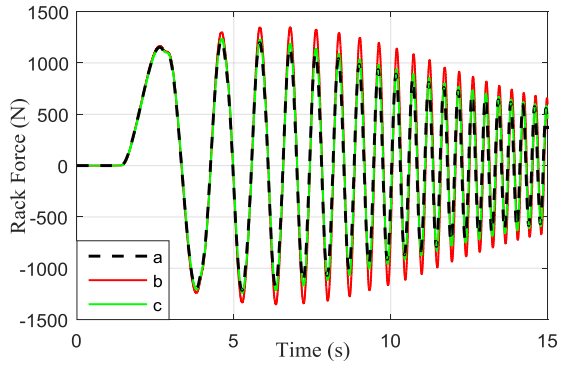


Fig. 8. Rack force in mono-simulation reference (a), co-simulation (b) and co-simulation with coupling elements (c).

In the second test, the transfer function from steering torque to steering angle has been identified (Fig. 9). The co-simulation with coupling elements (c) shows a same transfer behaviour as the reference (a). It can be noticed that the coupling element is able to adapt progressively to higher frequency. An interesting finding is that the error of the transfer function is small also in the high frequency range. It might be a reason of the EPAS model bandwidth. The high frequency steering input has been low-pass filtered by the steering mechanism. The rack movement has not been excited so the effect at the coupling interface is quite small in high frequency.

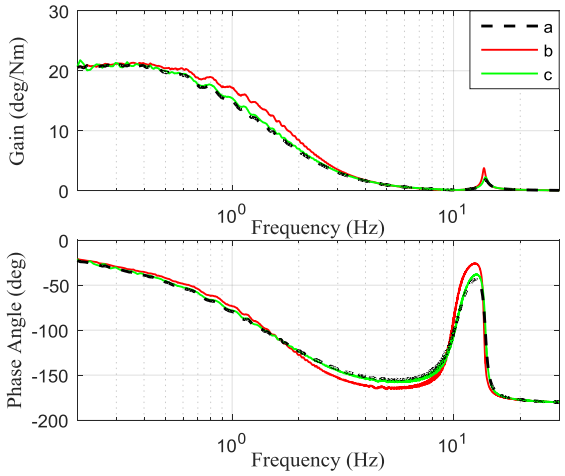


Fig. 9. Identified transfer function of steering torque to steering angle in mono-simulation reference (a), co-simulation (b) and co-simulation with coupling elements (c).

4.3 Effect of Interpolation and Correction

The delay compensation part is more critical in the test case due to the large delay value. The linear interpolation and correction part is highly dependent on the extrapolated data as well. To see the effect of interpolation and correction part, a higher bandwidth EPAS model has been used. A fast-switching steering input has been given as shown in Figure 10.

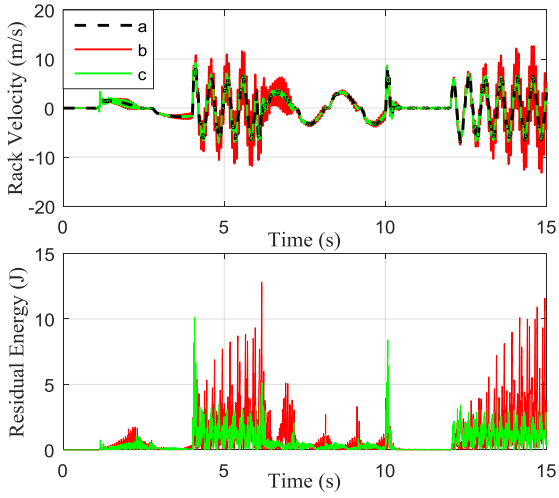


Fig. 10. Rack velocity and residual energy δE_k in mono-simulation reference (a), without (b) and with interpolation and correction (c) in coupling elements.

The system becomes more sensitive to the modular integration problem. An oscillatory result occurs even though the delay has been nearly compensated. The co-simulation gets more stable with a further refinement by interpolation and correction part. As can be seen from the plot, the reduced error can be indicated by less residual energy.

5. CONCLUSIONS

In this paper a coupling method has been developed for stable and consistent real-time co-simulation. The coupling element has been designed based on two functional parts to deal with the combined effects of transmission delay and modular simulation. From a benchmark test using a vehicle steering and chassis system, improvement by the approach has been clearly shown. The approach is easy and flexible to implement since only input-output signals are used. One limitation of the approach is that the delay compensation part needs to be always excited for the identification. Another limitation is that a varying time delay can make the approach less efficient.

Because the real-time co-simulation problems interact and influence together the simulation stability and accuracy. It is quite complex to draw a general conclusion on how each functional part can work optimally. It is more practical to tune each functional part separately based on frequency domain analysis and energy-preserving concept.

For the future investigation on real-time co-simulation, fundamental numerical analysis and systematic signal analysis are going to be taken. Questions from the modelling point of view, for example the system coupling method and the causality of the interface, are interesting to be investigated. This may further extend the co-simulation capability.

ACKNOWLEDGEMENT

The authors would like to thank ITEAM project in the European Union's Horizon 2020 research and innovation program under Marie Sklodowska-Curie Grant Agreement No. 675999.

The authors would like to thank Prof. Antonella Ferrara and Mr. Enrico Regolin from ICDS Lab, University of Pavia for support on parameter identification.

The authors would like to further thank Prof. Bengt Jacobson from Chalmers University of Technology and Mr. Edo Drenth from Volvo Cars for precious feedback.

REFERENCES

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

Busch, M. (2012). Zur effizienten Kopplung von Simulationsprogrammen, *Thesis (PhD)*, Kassel University, Germany.

Drenth, E. (2016). Robust co-simulation methodology of physical systems, *9th Graz Symposium Virtual Vehicle*.

Drenth, E. (2017). Method and system for control and co-simulation of physical systems, *Patent US 20170061069 A1*, Volvo Car Corporation.

Ljung, L. (1979). Asymptotic behaviour of the Extended Kalman Filter as parameter estimator for linear systems, *IEEE Transaction on Automatic Control*, volume 24, page 36-50.

Ljung, L. (2002). Recursive Identification Algorithms, *Circuits Systems Signal Processing*, volume 21, page 57–68.

Kübler, R. and Schiehlen, W. (2000). Two methods of simulator coupling, *Mathematical and Computer Modelling of Dynamical Systems*, volume 6:2, page 93-113.

Sadjina, S., Skjong, S. and Pedersen, E. (2017). Energy conservation and power bonds in co-simulations: non-iterative adaptive step size control and error estimation, *Engineering with Computers*, volume 33, page 607.

Stettinger, G., Horn, M., Benedikt, M. and Zehetner, J. (2014). Model-based coupling approach for non-iterative real-time co-simulation, *European Control Conference (ECC)*, IEEE.

Viel, A. (2014). Implementing stabilized co-simulation of strongly coupled systems using the Functional Mock-up Interface 2.0. *Proceedings of the 10th International Modelica Conference*, page 213-223.