ISSN- 2394-5125 VOL 4, ISSUE 06, 2017

# Analyzing and Estimating Errors in Model Exchange and Co-Simulation Using FMI, Version 2.0

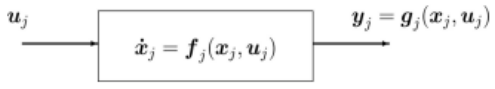
*P NAVEEN REDDY 1, BANTU SAILESH 2,*
*ASSISTANT PROFESSOR 1,2,*
*Mail ID:naveena0346@gmail.com, Mail ID: sailesh.bantu04@gmail.com,*
*Dept.: Mechanical*
*Nagole Institute of Technology and Science,*
*Kuntloor(V),Hayathnagar(M),Hyderabad,R.R.Dist.-501505.*

## Abstract

***Models of system dynamics, which might span many disciplines, are often constructed of such subsystems. The model, like large technical systems, is divided into separate modules. In commercial settings, the various subsystems are often modeled in silo using specialized software for each discipline. The Functional Mock-Up Interface (FMI) is a standard interface for coupling simulations. combines physical models from several fields and tackles issues like export The use of Industrial Simulation Tools for Importing Model Components (FMI for Model Exchange) nonlinear system dynamics, and the standardization of co-simulation interfaces For further information on co-simulation, see [10].A fresh focus on the algorithmic and numerical details of co-simulation It has prompted fresh research into error estimate and stabilizing methods in A co-simulation environment that complies with the FMI for Model Exchange and Co-Simulation v2.0 standard. In this work, we investigate effective methods of error estimation in telecommunications. Modulating the step size in this framework.***

## Introduction

The simulation of coupled technical systems and coupled physical phenomena in engineering with a focus on time-dependent issues is known as co-simulation. In the current ITEA2 project MODELISAR, theoretical and practical elements of cosimulation were investigated (2008- 2011). In order to address issues like the export and import of model components in commercial simulation tools (FMI for Model Exchange) and the standardization of cosimulation interfaces in nonlinear system dynamics (FMI for Co-Simulation), a functional mock-up interface (FMI) was developed. For more information, see [10]. Currently, the Modelica Association Project "FMI" maintains and develops the interface standard. The fourth -version of FMI for Model Exchange and Co-Simulation v2.0 was made available in August 2012 [15].




Co-Simulation takes use of the decomposable nature of linked issues by performing model setup, pre- and post-processing in parallel for each subsystem using specialized simulation software. During temporal integration, each component is simulated in isolation once again; with inter-component communication limited to a set number of discrete nodes ten (see [2]). Communication points Ten, steps TN TN+1, and step sizes HN: = TN+1 TN are also known as sampling points (synchronization points), macro steps, and

sampling rates, respectively [15]. Mathematically speaking, a modular time integration approach is produced when many numerical solvers are coupled in a co-simulation environment, with the (unknown) subsystem inputs uj(t) between the communication points Ten being replaced by a polynomial approximation.

Interpolation, or extrapolating, to fill up the gaps [2]. As a consequence of the extension of the signal, the modular time integration may become numerically unstable and suffer from new error terms. These issues are addressed in FMI for Model Exchange and Co-Simulation v2.0 through the use of interface routines that allow for higher-order extrapolation and interpolation of subsystem inputs, communication step size control (including step rejection), and Jacobean-based linearly implicit stabilization methods. Managing the size of each step in a communication and estimating errors is the subject of this research. We refer you to [1] and the more recent findings in [3], [18] for further information on stabilizing approaches based on the Jacobean. This paper follows the following structure: To rule out the possibility of algebraic loops in a block-structured coupled system, we provide a suitable condition in Section 2. In Section 3, the fundamentals of a convergence analysis in this context are laid forth. We get credible estimates of the local error by generalizing Richardson extrapolation methods from ODE and DAE theory to modular time integration. Section 4 covers both the theoretical and practical elements. Numerical experiments for a quarter-car model validate the conclusions of the theoretical research (Section 5). Section 6 is where the paper wraps up with a few last thoughts. The benchmark issue of a quarter-car was numerically tested in a Mat lab-based testing environment. Fraunhofer IIS/FMI

ISSN- 2394-5125 VOL 4, ISSUE 06, 2017

EAS's for Co-Simulation v1.0 compatible master, created for the MODELISAR project [7], [19], has recently been used to recreate these findings.

## Structures for Coupled Systems as Block Diagrams

The FMI for Model Exchange and Co-Simulation v2.0 takes a block-based method, similar to that of Kubler and Schiehlen [14]. If no algebraic loops exist in the system, the mathematical analysis of modular time integration techniques for coupled systems of ordinary differential equations (ODEs) [1] may be used to this generic problem class. In the early days of co-simulation algorithms in system dynamics, there was a lot of interest in the specific numerical problems that can arise from a coupling by constraints or other algebraic loops in the system, but these problems have only limited practical relevance in industrial applications today. In order to rule out the possibility of algebraic loops, we will begin our theoretical research by doing a structural analysis of linked systems using a block representation. Let the equations of state and output characterizes the "j " subsystem.

$$\left.\begin{aligned} \dot{\boldsymbol{x}}_j(t) &= \boldsymbol{f}_j(\boldsymbol{x}_j(t), \boldsymbol{u}_j(t), \boldsymbol{u}_{\mathrm{ex}}(t)) \\ \boldsymbol{y}_j(t) &= \boldsymbol{g}_j(\boldsymbol{x}_j(t), \boldsymbol{u}_j(t)) \end{aligned}\right\}$$

In where xj, uj, and yj represent the current state, input, and output vectors, respectively, and uex is an external input (t). The r ≥ 2 subsystems are related by input-output relations

$$\boldsymbol{u}_j(t) = \boldsymbol{c}_j(\boldsymbol{y}_1(t), \ldots, \boldsymbol{y}_{j-1}(t), \boldsymbol{y}_{j+1}(t), \ldots, \boldsymbol{y}_r(t)), \quad (j =$$

By reducing the coupled system to its vector form, we get a summary of all of its constituents, xj, yj, uj, f j, g j, and c j.

$$\left.\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{u}_{\mathrm{ex}}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \quad \boldsymbol{u}(t) = \boldsymbol{c}(\boldsymbol{y}(t)) \end{aligned}\right\}$$

To the power of x (t):= x > 1 (t), x > 2 (t), x > r (t) >, the equations in (2) may be written as a differentialalgebraic equation (DAE) with x, y, and u as independent variables, but they can be simplified to an ordinary differential equation with little effort.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{F}(\boldsymbol{x}(t), \boldsymbol{u}_{\mathrm{ex}}(t)) := \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{c}(\boldsymbol{g}(\boldsymbol{x}(t))), \boldsymbol{u}_{\mathrm{ex}}(t))$$

, and individual equations for the results

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t)), \quad \boldsymbol{u}(t) = \boldsymbol{c}(\boldsymbol{g}(\boldsymbol{x}(t)))$$

When there is no immediate feedback from one system to another (when there is no "feed-through")

$$\boldsymbol{y}_j(t) = \boldsymbol{g}_j(\boldsymbol{x}_j(t)), \quad \frac{\partial \boldsymbol{g}_j}{\partial \boldsymbol{u}_j}(\boldsymbol{x}_j, \boldsymbol{u}_j) \equiv \boldsymbol{0}, \ (j = 1, \ldots, r).$$

## Analysis of convergence

Traditional techniques for ODE and DAE time integration change the step size at each time step to ensure that the local error does not fall beyond the user-specified error boundaries. Because local mistakes are magnified in time integration by a factor that is confined by exp(L0T), with L0 being the Lipchitz constant of the right hand side, a perturbation analysis demonstrates that this method will constrain the global error after a finite time period of length T. (in the ODE case). Complications arise when considering coupled systems in block representation (Eq. 1) since order reduction events have been described for the study of local errors [5, Appendix 2.A], without lowering the realistically seen order of the global errors. To offer theoretical reason for this realistically observed error behavior, we will now explore the foundational phases of a convergence analysis for modular time integration techniques applied to coupled systems in block representation (Eq. 1). Not all mistakes, both locally and globally, are directly proportional to one another. Although MARTIN ARNOLD, CHRISTOPH CLAUSS, and TOM SCHIERZ might be linked as in the traditional ODE and DAE instance, the communication step size control may instead be determined by estimates of local errors. For the theoretical analysis, we use the paradigm of [4] and, in order to zero in on the extra error terms generated by linking these subsystems in a co-simulation framework, we ignore the discretization errors of the time integration in all subsystems. The input vectors uj (t) are interpolated by polynomials of degree k at each communication step TN TN+1 = TN + H (see [9] for details). The extrapolation of signals using interpolation polynomials is a specific example worth noting.

$$\boldsymbol{\Psi}_j(t) = \sum_{i=0}^{\kappa} \boldsymbol{u}_j(T_{n-i}) \prod_{\substack{l=0 \\ l \neq i}}^{\kappa} \frac{t - T_{n-l}}{T_{n-i} - T_{n-l}} = \boldsymbol{u}_j(t) + O(H^{k+1}).$$

Here's a Third Sample By only holding the input vectors uj(t) constant at each communication step Tn

ISSN- 2394-5125 VOL 4, ISSUE 06, 2017

Tn+1 = Tn + H, we have k = 0 and an extrapolation polynomial j(t) uj(Tn), (t [Tn, Tn+1]), that is constant. Interpolation using the linear (k = 1) and quadratic (k = 2) methods is used when precision is paramount.

$$\Psi_j(t) = u_j(T_n) + \frac{u_j(T_n) - u_j(T_{n-1})}{T_n - T_{n-1}}(t - T_n), \ (k = 1),$$

$$\Psi_j(t) = u_j(T_n) + \frac{u_j(T_n) - u_j(T_{n-1})}{T_n - T_{n-1}}(t - T_n) + \beta_{j,n}(t - T_n)(t - T_{n-1}), \ (k = 2),$$

*with*

$$\beta_{j,n} := \left(\frac{u_j(T_n) - u_j(T_{n-1})}{T_n - T_{n-1}} - \frac{u_j(T_{n-1}) - u_j(T_{n-2})}{T_{n-1} - T_{n-2}}\right)/(T_n - T_{n-2}).$$

## Consensus on the Error Level

An acceptable estimate of the local error is compared to user-defined error limitations (tolerances) in order to regulate the size of the communication steps (see [4]). Temporal-consuming but accurate, Richardson extrapolation may be used to estimate local errors in the time integration of ODEs and DAEs. By modifying the approach to fit the needs of the FMI co-simulation framework, significant processing time may be saved. In this subsection, we do an asymptotic error analysis on both strategies. Section 5 presents the results of numerical tests performed on the quarter car benchmark challenge.

To make the nomenclature manageable, we just conduct and analyze a communication step of Tn Tn+1 = Tn + H for all subsystems simultaneously and limit the theoretical analysis of the error estimates to pure polynomial signal extrapolation in all subsystems. Interpolating the analytical solution u(t) of (Eq. 2) at the k + 1 equally spaced prior communication points TN, (= 0, 1, k) is considered, as in (Eq. 9). For the inaccuracy of polynomial interpolation, classical estimates [9] show

$$\Delta_{\bar{u}}(t) := \bar{u}(t) - u(t) = -\frac{u^{(k+1)}(T_n)}{(k+1)!}\prod_{i=0}^{k}(t - T_{n-i}) + O(H^{k+2})$$

In the range [TN, TN+2]. By substituting u(t) for the input function in (Eq. 2), we get approximate solutions x(t), y(t) with x (Tn) = 0 and y(t) with x (Tn) = 0.

$$\left.\begin{aligned}\dot{\Delta}_{\bar{x}}(t) &= \mathbf{A}_n \Delta_{\bar{x}}(t) + \mathbf{B}_n \Delta_{\bar{u}}(t) + O(H^{k+2}) \\ \Delta_{\bar{y}}(t) &= \mathbf{C}_n \Delta_{\bar{x}}(t) + \mathbf{D}_n \Delta_{\bar{u}}(t) + O(H^{k+2})\end{aligned}\right\}$$

For

$$\Delta_{\bar{x}}(t) := \bar{x}(t) - x(t), \ \Delta_{\bar{y}}(t) := \bar{y}(t) - y(t), \ (t \in [T_n, T_{n+2}]).$$

System matrices An, Bn, Cn, Dn denote the Jacobeans f x, f u , gx , gu evaluated at x = x(Tn), u = u(Tn). The leading error term in Δx is obtained as solution of a linear time invariant system resulting in

$$\Delta_{\bar{x}}(T_n + aH) = -\mathbf{B}_n \int_0^a \prod_{i=0}^{k}(t + s)\,\mathrm{d}s \cdot \frac{u^{(k+1)}(T_n)}{(k+1)!}H^{k+2} + O(H^{k+3}),$$

## A Block Representation of Coupled Systems

FMI for Model Exchange and Co-Simulation v2.0 uses a block model of coupled systems, similar to the method proposed by Kubler and Schiehlen [14]. If the system does not include any algebraic loops, the mathematical analysis of modular time integration techniques for coupled systems of ordinary differential equations (ODEs) [1] may be applied to this more wide problem class. In the early days of co-simulation algorithms in system dynamics, people were very interested in the numerical problems that could arise as a result of a coupling by constraints or other algebraic loops in the system (for examples, see [4], [14]), but these problems now have relatively little bearing on real-world applications. We will begin our theoretical research by doing a structural analysis of linked systems using a block representation, so excluding any systems that include algebraic loops. Let the equations for the state and the output of " j " subsystem be used to characterize the subsystem.

$$\begin{aligned}m_c\ddot{x}_c &= F_{susp}(x_c, \dot{x}_c, x_w, \dot{x}_w) \\ m_w\ddot{x}_w &= F_{tire}(x_w, \dot{x}_w, u_{ex}(t)) - F_{susp}(x_c, \dot{x}_c, x_w, \dot{x}_w) \\ u_{ex}(t) &= (z(t), \dot{z}(t))^\top, \ z(t) = \begin{cases} 0 & \text{if } t \le 0 \\ 0.1 & \text{if } t > 0 \end{cases} \\ F_{susp} &= k_c(x_w - x_c) + d_c(\dot{x}_w - \dot{x}_c) \\ F_{tire} &= k_w(z - x_w) + d_w(\dot{z} - \dot{x}_w)\end{aligned}$$

ISSN- 2394-5125 VOL 4, ISSUE 06, 2017

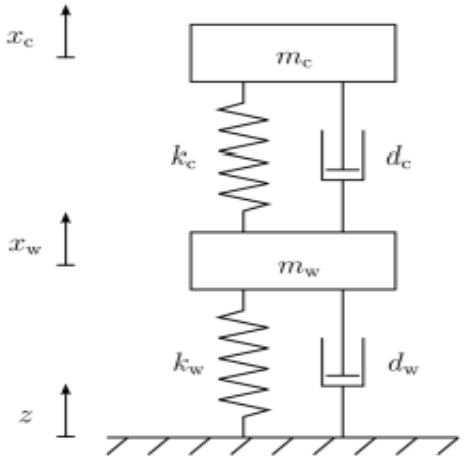



Fig .1 The novel alternative error estimate ESTmod performs comparably to the traditional estimate ESTRich in both scenarios.

| | |
|---|---|
| $m_c$ | 400.0 kg |
| $m_w$ | 40.0 kg |
| $k_c$ | 15000.0 N/m |
| $k_w$ | 150000.0 N/m |
| $d_c$ | 1000.0 Ns/m |
| $d_w$ | 0.0 Ns/m |

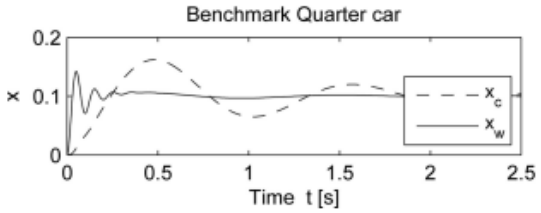
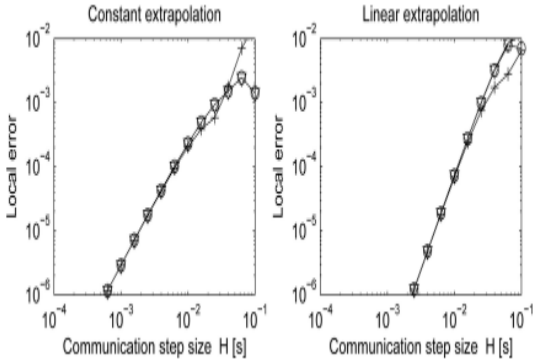



Fig. 2 Benchmark Quarter car: Model parameters and solution [12]




Co-simulation of a Quarter-Standard automobile using a displacement-displacement coupling is shown in Fig. 3. Error estimations and local errors ("+") ESTmod, ESTrich ("", ""). However, the end-user cares more about global faults than local ones. The convergence analysis in Section 3 demonstrates that the global errors x n, y n, u n do not undergo order reduction for displacement-displacement coupling or force displacement coupling. Further, as the state vector x has the same components in both scenarios (xc, xc, xw, and xw), the global errors x n are predicted to be in the same order of magnitude.

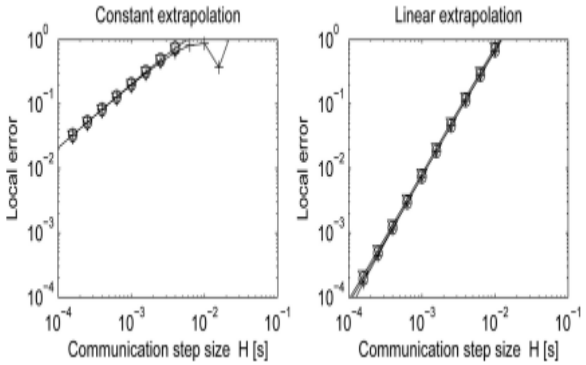



Fig. 4 Benchmark Quarter car, co-simulation with force-displacement coupling: Local error (“+”) and error estimates ESTRich, ESTmod (“◦”, “∇”)

## Note

The discretization errors of the (micro)time integration in the separate subsystems can no longer be ignored for a meaningful application of local error estimates like ESTRich and ESTmod in communication step size management. The investigation of this question still at hand is the focus of ongoing studies; for a glimpse at some early findings, check out [19].

## Conclusions

Higher-order approximations of subsystem inputs and variable communication step sizes may significantly increase the numerical efficiency of co-simulation techniques. If there are no algebraic loops in the coupled system, then the global error may be shown to be confined in terms of local errors by means of a rigorous mathematical analysis. For systems without direct feed-through, such as mechanical systems with displacement-displacement coupling, the efficacy of local error estimates based on Richardson extrapolation methods and certain variations has been explored in depth. Direct feed-through in one of the subsystems due to a force-displacement coupling

might reduce the advantageous asymptotic features of conventional error estimating algorithms.

## REFERENCES

[1] M. Arnold: Multi-rate time integration for large scale multibody system models. In P. Eberhard, editor, IUTAM Symposium on Multiscale Problems in Multibody System Contacts, pages 1-10. Springer, 2007.

[2] M. Arnold: Numerical methods for simulation in applied dynamics. In M. Arnold and W. Schiehlen, editors, Simulation Techniques for Applied Dynamics, volume 507 of CISM Courses and Lectures, p. 191-246. Springer, Wien New York, 2009.

[3] M. Arnold: Stability of sequential modular time integration methods for coupled multibody system models. J. Comput. Nonlinear Dynam. 5:031003, 2010.

[4] M. Arnold, M. Gunther: Preconditioned dynamic iteration for coupled differential-algebraic ¨ systems. BIT Numerical Mathematics, 41:1-25, 2001.

[5] M. Busch: Zur effizienten Kopplung von Simulationsprogrammen. PhD thesis, Universitat¨ Kassel, Fachbereich Maschinenbau, 2012.

[6] M. Busch, B. Schweizer: Numerical stability and accuracy of different co-simulation techniques: Analytical investigations based on a 2-DOF test model. In Proc. of The 1 st Joint International Conference on Multibody System Dynamics, May 25-27, 2010, Lappeenranta, Finland, 2010.

[7] C. Clauß, M. Arnold, T. Schierz, J. Bastian: Master zur Simulatorkopplung via FMI. In X. Liu-Henke, editor, Tagungsband der ASIM/GI-Fachgruppen STS und GMMS, Wolfenbuttel, ¨ 23.02.-24.02.2012, Wolfenbuttel, 2012. Ostfalia Hochschule f ¨ ur Angewandte Wissenschaften. ¨

[8] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein: Introduction to Algorithms. The MIT Press, Cambridge, MA, 2nd edition, 2001.

[9] P. Deuflhard, A. Hohmann: Numerical Analysis in Modern Scientific Computing: An Introduction. Number 43 in Texts in Applied Mathematics. Springer, 2nd edition, 2003.

[10] FMI: The Functional Mockup Interface. https://www.fmi-standard.org/.

[11] E. Hairer, G. Wanner: Solving Ordinary Differential Equations. II. Stiff and DifferentialAlgebraic Problems. Springer-Verlag, Berlin Heidelberg New York, 2nd edition, 1996.

[12] S. Knorr: Multirate-Verfahren in der Co-Simulation gekoppelter dynamischer Systeme mit Anwendung in der Fahrzeugdynamik. Master Thesis, University Ulm, Faculty of Mathematics and Economics, 2002.

[13] R. Kubler: ¨ Modulare Modellierung und Simulation mechatronischer Systeme. FortschrittBerichte VDI Reihe 20, Nr. 327. VDI-Verlag GmbH, Dusseldorf, 2000. ¨

[14] R. Kubler, W. Schiehlen: Two methods of simulator coupling. ¨ Mathematical and Computer Modelling of Dynamical Systems, 6:93-113, 2000.

[15] Modelica Association Project FMI: Functional Mockup Interface for Model Exchange and Co-Simulation v2.0 beta 4. https://www.fmi-standard.org/downloads, August 2012.

[16] H. Olsson: Private communication, June 2011.

[17] P. Popp, W.O. Schiehlen: Ground Vehicle Dynamics. Springer-Verlag, Berlin Heidelberg, 2010.