Efficient extrapolation for parallel co-simulation of coupled systems (CHOPtrey)

A. Ben Khaled-El Feki, L. Duval, M. Ben Gaid

IFP Energies nouvelles
1 et 4 avenue de Bois-Préau, 92852 Rueil-Malmaison, France
[abir.el-feki, laurent.duval, mongi.ben-gaid]@ifpen.fr

Abstract

Building high-fidelity system-level models of Cyber-Physical Systems (CPS) is a challenging duty. A first problem is the diversity of modeling and simulation environments used by the various involved multi-disciplinary teams. Particular environments are preferred for a specific use, due to distinctive strengths (modeling language, libraries, solvers, cost, etc.). The Functional Mock-up Interface (FMI) specification has been proposed to improve this issue [1]. A second problem is the growing complexity of such high-fidelity models and their induced prohibitive CPU execution time. Indeed, major system-level simulation softwares are relying on sequential ODE/DAE[1] solvers. They are currently unable to efficiently exploit the available parallelism provided by multi-core chips. In addition, CPS are commonly modeled as hybrid models where the major challenge resides in their numerous discontinuities. Indeed, discontinuities usually prevent high integration speeds with variable-step solvers. We propose a modular co-simulation [2] of a split model, where each sub-model is integrated with its own solver. Thanks to splitting, high integration speeds can be reached when using LSODAR[2],a variable step solver with a root-finding capability.

Nevertheless, partitioning a complex model into several lesser complex sub-models also brings some difficulties that need to be managed. First, partitioning may add virtual algebraic loops, therefore involving delayed outputs, even with an efficient execution order. To avoid the latter, we propose in [3] a new co-simulation method based on a refined scheduling approach. This technique, denoted "RCosim", retains the speed-up advantage of modular co-simulation thanks to the parallel execution of the sub-models. Furthermore, it improves the accuracy of simulation results through an offline scheduling of operations that takes care of model input/output dynamics. Second, partitioning and even co-simulation require synchronization between coupled models to exchange updated data to reduce numerical error propagation in simulation results.

Thus, tight synchronization, using small communication steps, is required between blocks. This greatly limits the possibilities to accelerate the simulation. Adaptive communication steps may better handle changes in model dynamics [4]. Meanwhile, stability of multi-rate simulators needs to be carefully assessed. Data extrapolation over steps is expected to enhance the precision over large communication steps. However, complex models usually present non-linearities and discontinuities, entangling forecasts from past observations only [5].

We propose a Computationally Hasty Online Prediction framework (CHOPred) to stretch out synchronization steps with negligible precision changes in the simulation, at low-complexity. It allows to improve the trade-off between speed-ups, needing large communication steps, and precision, needing frequent updates for model inputs. It is based on a Contextual & Hierarchical Ontology of Patterns (CHOPatt) that handles the discontinuities of exchanged signals by selecting appropriate

[1] Ordinary Differential Equation/Differential Algebraic Equation.

[2] Livermore Solver for Ordinary Differential equations, with Automatic method switching for stiff and nonstiff problems, and with Root-finding.

Causal Hopping Oblivious Polynomials (CHOPoly). CHOPtrey [6] uses a time-depending oblivion faculty for polynomial extrapolation with an independent power weighting on past samples. It accounts for memory depth changes required to adapt to sudden variations. Computations are performed on frames of samples hopping at synchronization steps.It is implemented on xMOD [7] tool in combination with model splitting and parallel simulation. It is applied on a hybrid dynamical engine model where the split parts are exported as FMUs from Dymola.

Test results show effective simulation speed-up with imperceptible computational overheads. In addition, sustained or even improved simulation precision is obtained without noticeable instability. Table 1 shows the speed-up compared with the single-threaded reference and the relative error variation compared with zeroth-order hold (ZOH) at 100 µs. Firstly, the speed-up is supra-linear w.r.t. the number of cores when the model is split into 5 threads integrated in parallel on 5 cores. Indeed, the containment of event detection and handling inside small sub-systems allows for solver accelerations, enough to over-compensate the multi-threading costs. Secondly, model splitting combined with enlarged communication steps, from $H = 100$ µs to $H = 250$ µs, allows around +13 % extra speed-up. Unfortunately this extra speed-up is obtained at the cost of a relative error increase (e.g. +341 % for the fuel density). Thirdly, the combination of model splitting at expanded communication steps ($H = 250$ µs) with CHOPtrey allows to keep the same extra speed-up while decreasing the relative error to values close to, or below, those measured with $H = 100$ µs and ZOH.

Table 1: CHOPtrey performance: speed-up vs. accuracy.

Communication time	Prediction	Speed-up factor	Relative error variation	
			Burned gas density	Fuel density
100 µs	ZOH	8.9	–	–
250 µs	ZOH	10.01	+7 %	+341 %
	CHOPtrey	10.07	−26 %	+21 %

References

[1] T. Blochwitz *et al.*, The Functional Mockup Interface for tool independent exchange of simulation models, in: C. Clauß (Ed.), Proc. Int. Modelica Conf., Linköping Univ. Electronic Press, Dresden, Germany, 2011.

[2] A. Ben Khaled *et al.*, Multicore simulation of powertrains using weakly synchronized model partitioning, in: IFAC Workshop on Engine and Powertrain Control Simulation and Modeling ECOSM, Rueil-Malmaison, France, 2012, pp. 448–455.

[3] A. Ben Khaled *et al.*, Fast multi-core co-simulation of cyber-physical systems: Application to internal combustion engines, Simul. Model. Pract. Theory 47 (2014) 79–91.

[4] T. Schierz, M. Arnold, C. Clauß, Co-simulation with communication step size control in an FMI compatible master algorithm, in: Proc. Int. Modelica Conf., Linköping University Electronic Press, Munich, Germany, 2012, pp. 205–214.

[5] M. Arnold, C. Clauß, T. Schierz, Error analysis and error estimates for co-simulation in FMI for model exchange and co-simulation V2.0, Arch. Mech. Eng. LX (1) (Jan. 2013).

[6] A. Ben Khaled-El Feki *et al.*, CHOPtrey: contextual online polynomial extrapolation for enhanced multi-core co-simulation of complex systems, Simulation: Trans. Soc. Mod. Sim. 93 (Mar. 2017) 185–200.

[7] M. Ben Gaïd *et al.*, Heterogeneous model integration and virtual experimentation using xMOD: Application to hybrid powertrain design and validation, in: 7th EUROSIM Congress on Modeling and Simulation, Prague, Czech Republic, 2010.