A flexible co-simulation framework for vehicle-terrain interaction

Radu Serban, Dan Negrut, Wei Hu

Department of Mechanical Engineering
University of Wisconsin-Madison
Madison, WI, 53706, USA
serban@wisc.edu, negrut@wisc.edu, whu59@wisc.edu

EXTENDED ABSTRACT

1 Introduction

Vehicle simulation is inherently a multi-physics problem. Indeed, different physical phenomena present in a complex vehicle simulation (including different vehicle subsystems, but also interaction with the terrain and environment) involve different time and space scales. Mathematical models used to represent the different physical phenomena involved can be significantly distinct in their nature (e.g., multibody dynamics vs. solid mechanics vs. Navier-Stokes), resulting in systems of equations of different types (e.g., differential-algebraic equations vs. partial differential equations vs. differential variational inequalities, etc.). As such, numerical simulation of the resulting equations of motion require different solution techniques (implicit vs. explicit integration, Eulerian vs. Lagrangian discretization, etc.) and parallelization paradigms (distributed, multicore, GPU, etc.).

A prototypical example is that of off-road vehicles and the vehicle-terrain interaction. There are various ways of modeling deformable terrain, each of those relying on different mathematical models, each of which is different from the multibody dynamics formulation which is the common high-fidelity approach for modeling and simulation the vehicle mechanical system. Other examples is the mechatronic vehicle system itself which combines a mechanical system with electrical, hydraulic, and control subsystems. Monolithic simulations of such coupled multi-physics, multi-scale problems are sometimes possible, but co-simulation [1] allows much more flexibility and freedom. Besides addressing the problem of disjunct physics, distinct numerical formulations, and different implementation strategies, a co-simulation approach to vehicle simulation also allows interfacing dedicated simulation packages thus offering the possibility of higher-fidelity and/or higher-performance overall simulation

2 The Chrono::Vehicle co-simulation framework

The open-source multi-physics Chrono software [2] provides a module dedicated to high-fidelity modeling, simulation, and visualization of ground vehicles [3]. For off-road simulations with ground vehicle models, wheeled or tracked, as well as for extraterrestrial rover simulations, Chrono provides a suite of different formulations for terramechanics, i.e., simulation of deformable terrain and vehicle-terrain interaction. Techniques for modeling deformable terrain span an entire spectrum varying in complexity, representation accuracy, and ensuing computational effort. Chrono provides support for several of these formulations, ranging from arguably the most accurate but also computationally-intensive granular dynamics representation (DEM), to continuum representations (CR) derived through homogenization, to FEA-based soil models, and to real-time capable, but lower-fidelity semi-empirical approaches, based on Bekker theory and its generalization through the Soil Contact Model. The list of deformable terrain formulations offered through Chrono is certainly not exhaustive and different approaches are possible and available through external software libraries (such as the hierarchical multiscale approach for concurrent simulation of deformable tires and deformable terrain described in [4]).

The goal of Chrono::Vehicle off-road co-simulation framework presented here is to allow seamless simulation of wheeled or tracked systems – full Chrono::Vehicle models, various wheeled robots (e.g., extraterrestrial rovers), or test rigs – using any of the formulations provided in Chrono for modeling and simulating deformable terrain and interaction with vehicles, implements, or other objects, or with third-party deformable terrain simulation software.

Features From a high vantage point, the principal features this framework provides are as follows:

- Allows simulation of various rigid and deformable terrain types, including rigid, SCM, CR with SPH, DEM (multicore or GPU-based solvers), as well as interfacing to external, third-party terramechanics libraries.
- Offers a suite of vehicle multi-body system, specified as Chrono models or Chrono::Vehicle models, including wheeled and tracked vehicles (with arbitrary number of axles and wheels or track shoes, respectively), as well as single-tire test rigs.
- Allows simulation of different tire types, including rigid or FEA-deformable Chrono::Vehicle tire models. Offers a "by-pass" mode in which the tire co-simulation nodes are short-circuited to allow integration of external terramechanics libraries that concurrently simulate the vehicle tires.
- Allows attaching instrumentation rigs to any of the simulated vehicles (e.g., drawbar pull rigs of different topologies corresponding to different measuring methods).

Implementation considerations The salient characteristics of architecture and design of this co-simulation framework for terramechanics are listed below. The overall architecture is illustrated schematically in Fig. 1.

- For efficiency and ease of implementation, the Chrono::Vehicle off-road co-simulation framework relies on the Message Passing Interface (MPI) for the top-level communication layer. Each co-simulation subsystem runs in a separate process (as an MPI rank), but is free to further parallelize its own workload (including by spanning an MPI sub-communicator for distributed parallel computation)

- For wheeled vehicles, the framework implements a 3-way co-simulation framework with the vehicle system, tire subsystems, and deformable terrain acting as separate co-simulation nodes. For tracked vehicles, due to the tighter coupling between the vehicle and the running-gear, the framework adopts a 2-way co-simulation between the vehicle and the terrain.
- The framework only implements an explicit co-simulation scheme, of the force-displacement type [5]. In such a scheme, subsystems evolve their dynamics between communication meta-steps in parallel and independently of each other. At each communication step, one subsystem sends *displacement*-type information (for example full state information of rigid bodies or flexible meshes) and receives *force*-level information (for example, generalized forces and moments acting on solids due to their interaction with the terrain).
- The structure of MPI messages and the communication pattern is fixed and specified by the framework. Wrapped vehicle systems must adhere to a predefined API.
- Provided as an optional Chrono module, with features enabled based on user-selected Chrono modules Like everything else in Chrono, provided as a middleware library (mention advantages)

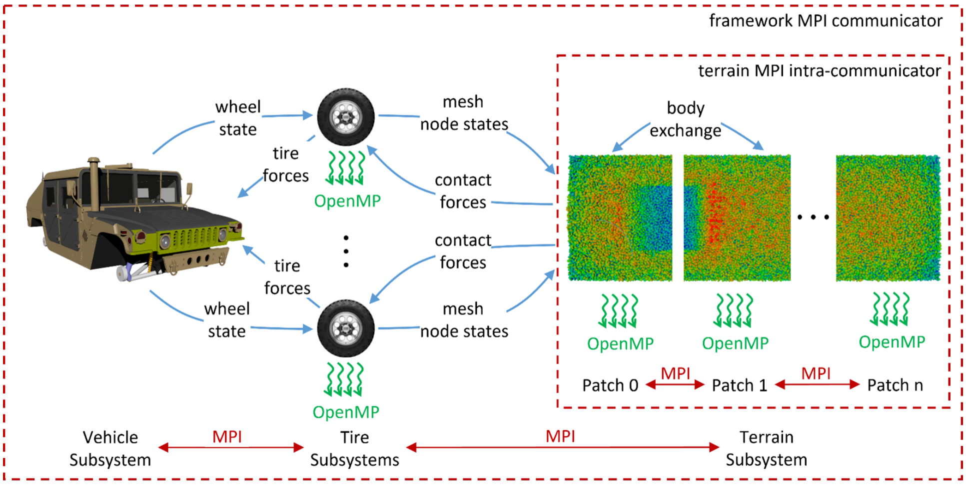

Figure 1: Schematic of the 3-way co-simulation scheme for wheeled vehicles with tires simulated independently of the terrain. For tracked vehicles or terramechanics modules that incorporate the tire-terrain interaction, the framework adjust to a 2-way co-simulation scheme.

3 Example simulations

In this presentation, after describing the overall co-simulation framework, its capabilities and architecture, we will provide representative simulations (with both wheeled and tracked vehicles, as well as various extraterrestrial rovers, such as the Curiosity Mars rover and the Moon Viper rover). We will also present results extracted from these simulations (such as drawbar pull curves).

Finally, we will demonstrate the ability of this co-simulation framework to interface full Chrono multibody vehicle models with external terramechanics packages. For this, we will present results of a collaboration with the developers of the IMBD package which implements the method described in [4]. During this exercise, a Chrono 4WD off-road vehicle was co-simulated with IMBD and results validated against an IMBD monolithic simulation of the same scenario showing perfect agreement.

The Chrono::Vehicle terramechanics co-simulation framework as described here is part of release 8.0 of Chrono. However, it is under active development with planned extensions including support for more flexible terrain patch definitions (in terms of topology and surface initialization), as well as further formalization by adopting the Functional Mock-Up Interface standard.

References

[1] C. Gomes, C. Thule, D. Broman, P. G. Larsen, and H. Vangheluwe, "Co-simulation: A survey," *ACM Comput. Surv.*, vol. 51, no. 3, may 2018. [Online]. Available: https://doi.org/10.1145/3179993

[2] A. Tasora, R. Serban, H. Mazhar, A. Pazouki, D. Melanz, J. Fleischmann, M. Taylor, H. Sugiyama, and D. Negrut, "Chrono: An open source multi-physics dynamics engine," in *High Performance Computing in Science and Engineering – Lecture Notes in Computer Science*, T. Kozubek, Ed. Springer International Publishing, 2016, pp. 19–49.

[3] R. Serban, M. Taylor, D. Negrut, and A. Tasora, "Chrono::Vehicle template-based ground vehicle modeling and simulation," *Intl. J. Veh. Performance*, vol. 5, no. 1, pp. 18–39, 2019.

[4] H. Yamashita, G. Chen, Y. Ruan, P. Jayakumar, and H. Sugiyama, "Hierarchical multiscale modeling of tire–soil interaction for off-road mobility simulation," *Journal of Computational and Nonlinear Dynamics*, vol. 14, no. 6, 2019.

[5] M. Busch, "Continuous approximation techniques for co-simulation methods: Analysis of numerical stability and local error," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 96, no. 9, pp. 1061–1081, 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/zamm.201500196