The 6th Joint International Conference on Multibody System Dynamics and
The 10th Asian Conference on Multibody System Dynamics
November 1-5, 2020, New Delhi, India

Synchronization and Scalability of the Multi-Agent Dynamics Environment SynChrono

Jay Taves, Asher Elmquist, Radu Serban, Dan Negrut

Department of Mechanical Engineering
University of Madison–Wisconsin
1513 University Ave, 53706 Madison, Wisconsin
[jtaves, amelmquist, serban, negrut]@wisc.edu

Abstract

We discuss a scalable multi-agent dynamics environment called SynChrono and focus on several key features of its operation. SynChrono's aim is to be an easy-to-use virtual prototyping solution that bridges the gap between simulation and reality for autonomous vehicle problems that involve large collections of agents. While herein the SynChrono discussion pertains only to autonomous vehicles, this platform works equally well with other autonomous agents, e.g., robots, as long as the dynamics of these agents can be captured in the physics simulation tool Chrono [1, 2]. There are already many tools, e.g., Carla and Gazebo [3, 4], that provide similar simulation capabilities, but SynChrono distinguishes itself by building on the open source, multi-physics simulation engine Chrono in order to provide realistic dynamics that accurately replicate agent behavior in real-world scenarios that include complex scenarios, e.g., deformable terrains, fluid structure interaction, etc. Through the use of a message passing interface (MPI), SynChrono scales the physics simulation capabilities of Chrono to enable tens of vehicles to operate in parallel in a time and space coherent fashion. This allows for the simulation of scenarios that are too costly to represent in a monolithic simulation but that still require accurate agent dynamics to capture real-world behavior and thus be relevant in virtual prototyping tasks.

Simulated Sensing and Vehicle Control Algorithms

Drawing on ongoing research for simulated sensing in Chrono::Sensor, SynChrono is capable of providing virtual sensor data to the control algorithms that drive the vehicle in a software-in-the-loop paradigm. Chrono::Sensor provides support for cameras, lidar, GPS, and IMU and allows further expansion and specification of user-defined sensors and noise models. The focus of simulated sensing is to provide realistic data from a virtual environment that is used by a control algorithm to assess its surroundings as well as estimate its own state. To this end, ongoing research focuses on increasing the realism of virtual sensor data to improve the testing and evaluation of autonomous vehicles and robots. In SynChrono, and in the context of autonomous vehicles, GPS and lidar give an illustrative example of how simulated data are brought into the framework. In order to follow a particular path, a control algorithm can take, as input, data from a simulated GPS sensor to estimate its position in space. It can then compare this to a target path to continually correct the vehicle's trajectory, a common task in lane-keeping and path following. If a vehicle is additionally equipped with a simulated lidar, the vehicle can use the lidar point cloud for object detection and pose estimation which is critical in object avoidance and path planning.

Multi-Agent Scalability Through MPI

Including multiple vehicles in a single Chrono experiment quickly makes the simulation run many times slower than real time as the dynamics for each additional vehicle are not computed separately, but rather in tandem with all the others. SynChrono takes the approach of distributing the vehicle dynamics computation across multiple MPI ranks which periodically synchronize their data. The rationale is that for vehicle dynamics problems not involving collisions or drafting, each vehicle in the simulation only impacts the other vehicles through the sensed data that the other vehicles may collect. Thus, the dynamics of each vehicle can be run separately, in their own Chrono simulation, without any knowledge of the dynamics of the other vehicles. At some frequency, called the heartbeat frequency, the vehicles will synchronize and communicate their new state to all the other vehicles in the network in order to maintain space and time coherence for sensing purposes. While this heartbeat frequency is still very fast relative to human perception, it is slow compared to the time step frequency of the Chrono::Vehicle dynamics simulation for each vehicle. SynChrono's approach of computing the dynamics independently scales well as vehicles are added, since the overhead of communicating limited state information between MPI

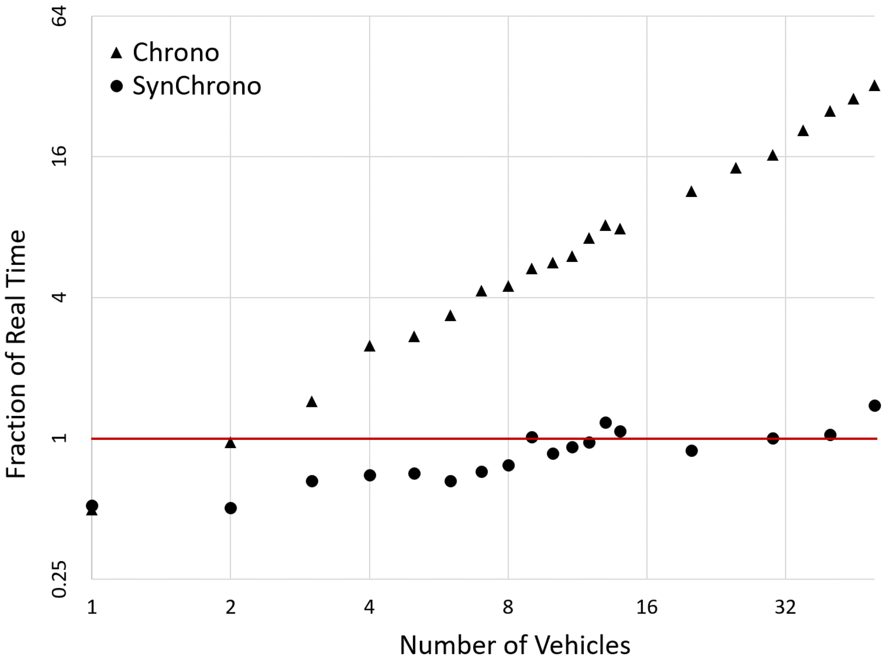

Figure 1: Scaling comparison between SynChrono and Chrono, plotted in $\log_2$-$\log_2$ scale. As more vehicles (modeled with Chrono::Vehicle [5]) are added to a monolithic Chrono simulation, the execution quickly slows down to a fraction of real-time. SynChrono is able to scale much more readily, remaining close to real-time even for tens of vehicles. For this chart both SynChrono and Chrono simulated a highway platoon, with vehicles added in lanes and driving straight ahead across flat terrain.

ranks is much lower than the overhead for computing additional vehicle dynamics in a single Chrono system, a comparison is shown in figure 1.

Conclusion and future work

In this contribution we introduced several distinguishing aspects of the multi-agent dynamics framework SynChrono, in particular its multi-agent framework which allows it to scale very well while still remaining faithful to the accurate underlying vehicle dynamics and sensing simulation of Chrono::Vehicle and Chrono::Sensor. In the full contribution we will elaborate on the operation of SynChrono and on key design decisions that have led to increased performance and scalability. Future work will focus on making SynChrono easy to use and on increasing the number of vehicle control algorithms that are immediately available to the user. Since Chrono itself has support for deformable terrain through the Soil Contact Model (SCM), in the near future we also plan to investigate the simulation of off-road multi-agent scenarios.

References

[1] 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.), pp. 19–49, Springer, 2016.

[2] Project Chrono, "Chrono: An Open Source Framework for the Physics-Based Simulation of Dynamic Systems." `http://projectchrono.org`. Accessed: 2020-03-03.

[3] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," *arXiv preprint arXiv:1711.03938*, 2017.

[4] N. P. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator.," in *IROS*, vol. 4, pp. 2149–2154, Citeseer, 2004.

[5] 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.