Autonomy Today: Many Delay-Prone Black Boxes (Artifact)

Sizhe Liu
University of North Carolina at Chapel Hill, NC, USA

Rohan Wagle
University of North Carolina at Chapel Hill, NC, USA

James H. Anderson
University of North Carolina at Chapel Hill, NC, USA

Ming Yang
WeRide Corp., San Jose, CA, USA

Chi Zhang
WeRide Corp., San Jose, CA, USA

Yunhua Li
WeRide Corp., San Jose, CA, USA

Abstract

Machine-learning (ML) technology has been a key enabler in the push towards realizing ever more sophisticated autonomous-driving features. In deploying such technology, the automotive industry has relied heavily on using "black-box" software and hardware components that were originally intended for non-safety-critical contexts, without a full understanding of their real-time capabilities. A prime example of such a component is CUDA, which is fundamental to the acceleration of ML algorithms using NVIDIA GPUs. In this paper, evidence is presented demonstrating that CUDA can cause unbounded task delays. Such delays are the result of CUDA's usage of synchronization mechanisms in the POSIX thread (pthread) library, so the latter is implicated as a delay-prone component as well. Such synchronization delays are shown to be the source of a system failure that occurred in an actual autonomous vehicle system during testing at WeRide. Motivated by these findings, a broader experimental study is presented that demonstrates several real-time deficiencies in CUDA, the glibc pthread library, Linux, and the POSIX interface of the safety-certified QNX Operating System for Safety. Partial mitigations for these deficiencies are presented and further actions are proposed for real-time researchers and developers to integrate more complete mitigations.

2012 ACM Subject Classification Computer systems organization → Real-time operating systems; Software and its engineering → Process synchronization

Keywords and phrases autonomous driving, CUDA programming, locking protocols, POSIX thread, operating systems, machine learning systems, real-time systems

Digital Object Identifier 10.4230/DARTS.10.1.3

Funding *Sizhe Liu, Rohan Wagle, and James H. Anderson:* supported by NSF grants CPS 2038960, CPS 2038855, CNS 2151829, and CPS 2333120.

Related Article Sizhe Liu, Rohan Wagle, James H. Anderson, Ming Yang, Chi Zhang, and Yunhua Li, "Autonomy Today: Many Delay-Prone Black Boxes", in 36th Euromicro Conference on Real-Time Systems (ECRTS 2024), LIPIcs, Vol. 298, pp. 12:1–12:27, 2024.
`https://doi.org/10.4230/LIPIcs.ECRTS.2024.12`

Related Conference 36th Euromicro Conference on Real-Time Systems (ECRTS 2024), July 9–12, 2024, Lille, France

Dagstuhl Artifacts Series, Vol. 10, Issue 1, Artifact No. 3, pp. 3:1–3:3
Dagstuhl Artifacts Series
Schloss Dagstuhl – Leibniz-Zentrum für Informatik,
Dagstuhl Publishing, Germany

1 Scope

This document introduces the *artifact* for the related article [1]. The artifact consists of the implementation for evaluating DL inference, CUDA, and glibc.

2 Content

The artifact package includes:

- `README.md` – an instruction manual for performing all tests contained in the artifact.
- **DL inference tests** – testing directed at evaluating the GPU and CPU execution times for five modern DL models commonly used to perform autonomous functions. This portion of the tests also evaluates the effectiveness of CUDA Graph in reducing locking usage in DL inference.
- **CUDA locking tests** – testing directed at tracing of `cudaLaunchKernel` and `cudaMemcpyAsync`, which are the main focus of [1]. The test will reveal the exact locking behavior within the two CUDA functions and how they can lead to delays detailed in [1].
- **glibc locking overhead evaluation** – detailed tracing experiment for glibc mutex and read-write lock functions. The trace profile compares the overhead for using the glibc mutex with the default policy and priority-inheritance policy. This section also compares overheads for using (and not using) the priority boosting with phase-fair read-write lock mentioned in [1].
- implementation of the shimming, tracing, and mitigation methods mentioned in [1], and benchmarks suite for DL inference using TensorRT.

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). In addition, the artifact is also available at: `https://github.com/sizheliu-unc/ECRTS24/tree/main/artifact` (subjected to changes deemed appropriate by the authors).

4 Tested platforms

This artifact has been tested and confirmed to work as expected with the following hardware and system setup:

- NVIDIA Titan V GPU.
- 32× Intel Xeon Silver 4110 CPUs.
- 32G Memory.
- Ubuntu 18.04 LTS and 20.04 LTS.
- Linux 5.4.0.
- glibc 2.30 and 2.38.
- NVIDIA driver 460.27.04, 525.89.02, and 550.54.14.
- CUDA 11.2 and 12.2.2.

Additional requirements for this artifact are:

- For DL inference tests: Python 3.8, NVIDIA TensorRT 8.6.1.6 GA, cuDNN 8.9.7 (CUDA 12), and NSight Systems 2024.
- For glibc tests: git and sudo permission (for `SCHED_FIFO`).

5 License

The artifact is available under the MIT license.

6 MD5 sum of the artifact

8a6b6c208b0fad54fcf0ea026764e635

7 Size of the artifact

1.70 GiB (DROPS artifact)

References

1 Sizhe Liu, Rohan Wagle, James H. Anderson, Ming Yang, Chi Zhang, and Yunhua Li. Autonomy today: Many delay-prone black boxes. In *Proceedings of the 36th Euromicro Conference on Real-Time Systems*, volume 298 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:27. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ECRTS.2024.12.