Work-Already-Published :

Analysis and Mitigation of Shared Resource Contention on Heterogeneous Multicore: An Industrial Case Study

Michael Bechtel, Heechul Yun

https://github.com/CSL-KU/ArmArHudChallenge

Industrial Challenge 2022: A High-Performance Real-Time Case Study on Arm

Matteo Andreozzi ✉
Arm, Cambridge, United Kingdom
Giacomo Gabrielli ✉
Arm, Cambridge, United Kingdom
Balaji Venu ✉
Arm, Cambridge, United Kingdom
Giacomo Travaglini ✉
Arm, Cambridge, United Kingdom

Abstract

High-performance real-time systems are becoming increasingly common in several application domains, including automotive, robotics, and embedded. To meet the growing performance requirements of the emerging applications, these systems often adopt a heterogeneous System-on-Chip hardware architecture comprising multiple high-performance CPUs and one or more domain-specific accelerators. At the same time, the applications running on these systems are subject to stringent real-time and safety requirements. Due to the non-deterministic execution model of the compute elements involved and the co-location of the workloads, which leads to contention of the shared hardware resources, designing and orchestrating such applications is particularly challenging. In fact, the demand for novel methodologies, tools, and best practices to assist application designers working on high-performance real-time systems has never been stronger.

To stimulate innovation in this area, this document outlines an industrial case study from the automotive domain targeting an Arm-based hardware platform. The selected application is an augmented reality head-up display, which can be considered a representative example of a high-performance real-time use case. This case study will serve as the basis for a (multi-year) challenge involving real-time and embedded systems researchers across academia and industry that will be kicked off at the 34th Euromicro Conference on Real-Time Systems (ECRTS) 2022.

Andreozzi, Matteo, et al. "Industrial challenge 2022: A high-performance real-time case study on arm." In *ECRTS*, 2022

Image Source: https://www.ecrts.org/industrial-challenge-current-challenge/

Augmented reality head-up display (AR-HUD)

- ARM 2022 industrial challenge case study application
- Visual SLAM (OV²SLAM)
  - Determine orientation and trajectory + generate a map of the surroundings
  - High-criticality real-time task
- Head-pose estimation DNN (Hope-Net)
  - Estimate driver's pose for better AR rendering that accounts for the driver's viewpoint
  - high-priority real-time task
- **"Aggressor" tasks**
  - Other (synthetic) tasks that compete for the shared hardware resources of the SoC
  - Best-effort (non real-time) priority

M. Andreozzi, G. Gabrielli, B. Venu, G. Travaglini. "Industrial Challenge 2022: A High-Performance Real-Time Case Study on Arm." In *ECRTS*, 2022

Micro-architectural DoS attacks

```
1  for (int64_t i = 0; i<mem_size; i += LINE_SIZE)
2  {
3      ptr[i] = 0xff;
4  }
```

Sequential Attacker
(BwWrite)

```
1  static int* list[MAX_MLP];
2  static int next[MAX_MLP];
3
4  for (int64_t i = 0; i < iter; i++) {
5      switch (mlp) {
6      case MAX_MLP:
7          .
8          .
9      case 2:
10         list[1][next[1]+1] = 0xff;
11         next[1] = list[1][next[1]];
12         /* fall-through */
13     case 1:
14         list[0][next[0]+1] = 0xff;
15         next[0] = list[0][next[0]];
16     }
17 }
```

Parallel Linked-List Attacker
(PLLWrite)

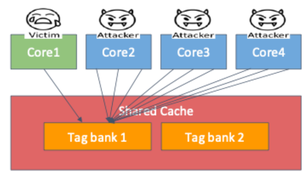
- Configurable synthetic workloads to cause resource contention
  - Sequential vs. random, read vs. write access patterns

M. G. Bechtel and H. Yun. "Cache Bank-Aware Denial-of-Service Attacks on Multicore ARM Processors." In *IEEE RTAS*, 2023

Cache bank-aware DoS attack

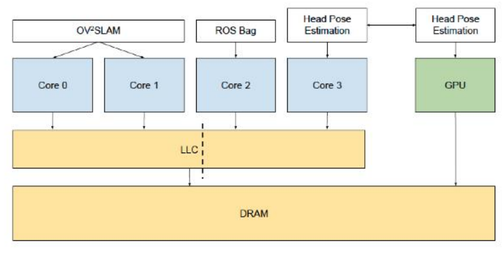
- Parallel Linked-List (PLL) attacks that target one LLC data bank

```
1  #define bit(addr,x)  ((addr >> (x)) & 0x1)
2  int paddr_to_sram_bank(unsigned long addr)
3  {
4      return ( (bit(addr, 6) << 2) |
5               (bit(addr, 5) << 1) |
6               bit(addr, 4) );
7  }
```



M. G. Bechtel and H. Yun. "Cache Bank-Aware Denial-of-Service Attacks on Multicore ARM Processors." In *IEEE RTAS*, 2023

AR-HUD mapping on Jetson Nano



M. Bechtel, H. Yun. "Analysis and Mitigation of Shared Resource Contention on Heterogeneous Multicore: An Industrial Case Study." *IEEE TC*, 2024.

AR-HUD mapping on Jetson Nano

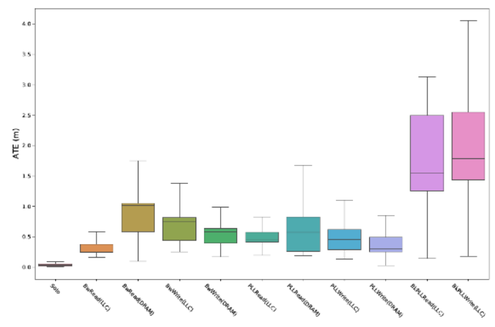
Task	Thread	Core(s)	Real-Time Priority	Rate (Hz)
OV²SLAM	Front-End	0,1	2	20
	Mapping			-
	State Optimization			-
ROS bag	-	2	2	20
Head Pose Est.	-	3,GPU	1	20

Real-time tasks (Linux SCHED_FIFO) threads/core mapping and scheduling parameters

- *Aggressor (DoS attack)* tasks are scheduled on all cores as *best-effort* tasks (using Linux CFS scheduler) to fully load the system
- L2 cache is *partitioned* w/ page coloring (*): OV²SLAM vs. all else

(*) H. Yun, R. Mancuso, Z. Wu, R. Pellizzoni. "PALLOC: DRAM Bank-Aware Memory Allocator for Performance Isolation on Multicore Platforms." In *IEEE RTAS*, 2014

Impact of DoS attacks on OV²SLAM



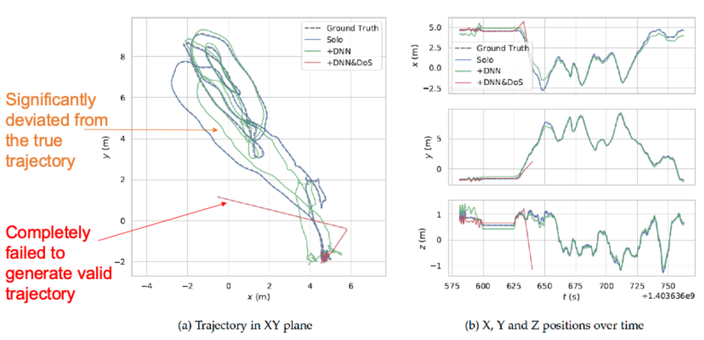
Working-set sizes:
(LLC) -> LLC fitting
(DRAM) -> DRAM fitting

Y-axis: Absolute Trajectory Error (ATE) = a standard measure of accuracy of SLAM. Lower is better.
X-axis: different micro-architectural denial-of-service (DoS) attacks

Impact of DNN and DoS attacks on OV²SLAM

Significantly deviated from the true trajectory

Completely failed to generate valid trajectory



(a) Trajectory in XY plane

(b) X, Y and Z positions over time

Our approach: RT-Gang++

Cache bandwidth throttling
- Throttle attacker's access (from CPU) to the shared LLC
- Using per-core performance counters (based on MemGuard*)
- To limit cache (bank) bandwidth contention

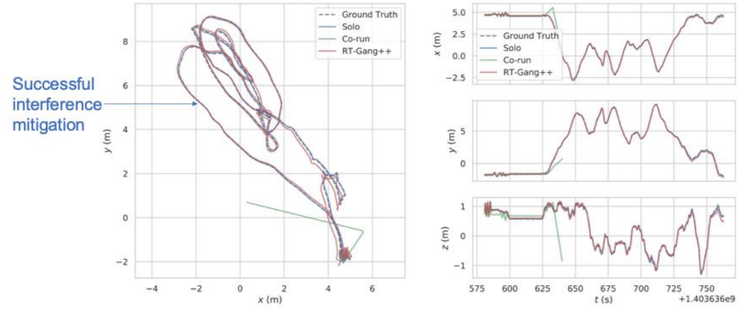
GPU bandwidth throttling
- Throttle HopeNet DNN's access (from GPU) to the shared DRAM
- Using NVIDIA's memory controller level throttling mechanism
- To limit GPU induced memory b/w interference on CPU (running SLAM)

Partitioned gang scheduling
- To avoid inter-application interference on multiple multi-threaded RT apps.

(*) H. Yun, G. Yao, R. Pellizzoni, M. Caccamo, and L. Sha. "MemGuard: Memory Bandwidth Reservation System for Efficient Performance Isolation in Multi-core Platforms." In *IEEE RTAS*, 2013

Impact of RT-Gang++ on OV²SLAM

Successful interference mitigation



(a) Trajectory in X-Y plane

(b) X, Y, and Z positions over time.

Summary

- Consolidating multiple RT/NRT tasks on heterogeneous multicore is challenging due to interference on shared hardware resources
- Cache bank-aware DoS attacks are especially effective in impacting performance of the real-time SLAM task in the AR-HUD case-study
- Executing a DNN task on the integrated GPU also significantly impact the performance of the SLAM on the CPU
- RT-Gang++ mitigates the interference problem via (1) software-based cache bandwidth throttling, (2) hardware-based GPU bandwidth throttling, and (3) partitioned real-time gang scheduling.

- A lot more details in the paper (e.g., results on RPi4, the effect of other DoS attacks, etc.); come talk to us at the poster session.

Publications

Michael Garrett Bechtel and Heechul Yun. "Cache Bank-Aware Denial-of-Service Attacks on Multicore ARM Processors." In *RTAS*, 2023.
Michael Bechtel, Heechul Yun. Analysis and Mitigation of Shared Resource Contention on Heterogeneous Multicore: An Industrial Case Study. *IEEE Transactions on Computers*, 2024.
Connor Sullivan, Alex Manley, Mohammad Alian, Heechul Yun. Per-Bank Bandwidth Regulation of Shared Last-Level Cache for Real-Time Systems. In *RTSS,* 2024.

KU-CSL
COMPUTER SYSTEMS LAB

Supported in part by NSF CNS-1815959, CPS-2038923 and NSA Science of Security initiative contract no. H98230-18-D-0009