Resource Contention Analysis in GPU-Accelerated Embedded Platforms

Hipert/Lab
High Performance Real Time Lab

Filippo Muzzini
University of Modena and Reggio Emilia (Italy)

CAPITAL WS - Toulouse - France, 14/06/2024

Filippo

- Bachelor Degree in Computer Science (2016) @ UNIMORE
- Master Degree in Computer Science (2018) @ UNIMORE
- PhD in Math (but with Thesis in Computer Science) (2023) @ UNIPR (but done @ UNIMORE)

Filippo

- **Post-Doc @ UNIMORE**

Hipert/Lab

High Performance Real Time Lab

- Autonomous Vehicles

- Connected Vehicles

Filippo

- **Post-Doc @ UNIMORE**

Hipert/Lab

High Performance Real Time Lab

- Embedded Systems

- Accelerators

Embedded boards for Autonomous systems

Small Size
- More suitable to be placed

Small Weight
- Less load for vehicle

Small Power Consumption
- Less power demand from battery

Embedded boards for Autonomous systems

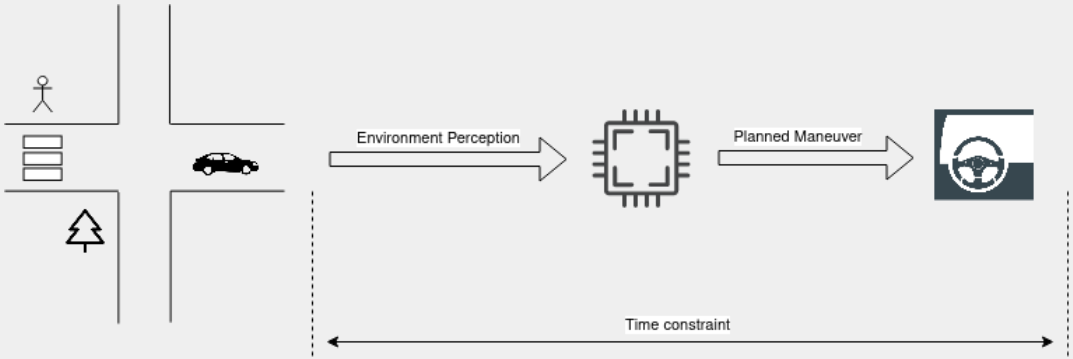
Small Power Consumption

- Less power demand from battery
- **Less computational power**
- **Longest algorithms execution time**

Embedded boards for Autonomous systems

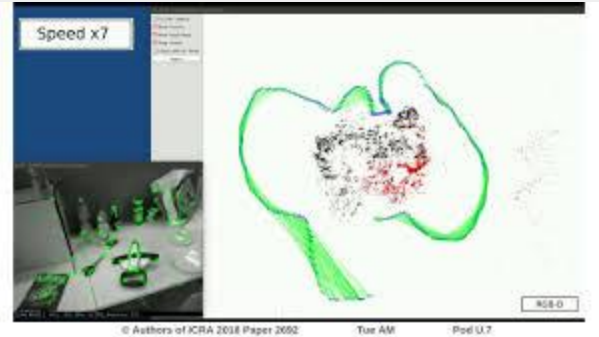
Small Power Consumption

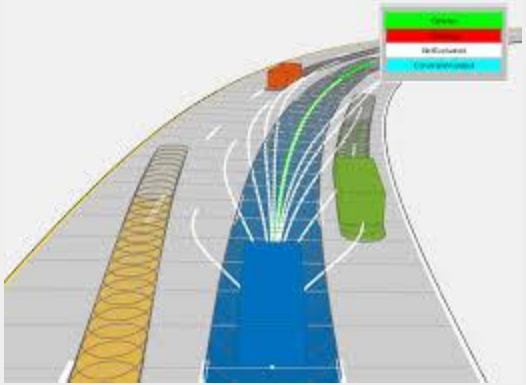
- Less power demand from battery
- **Less computational power**
- **Longest algorithms execution time**



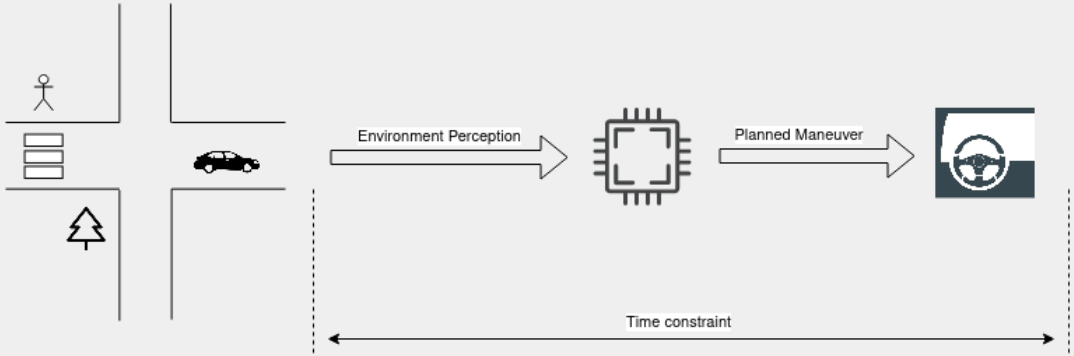
Embedded boards for Autonomous systems

Exploit GPU to reduce the execution time
- Localization
- Planning





Algorithms execution time



Autonomous Systems are Safety Critical

- **Execution time is a constraint**

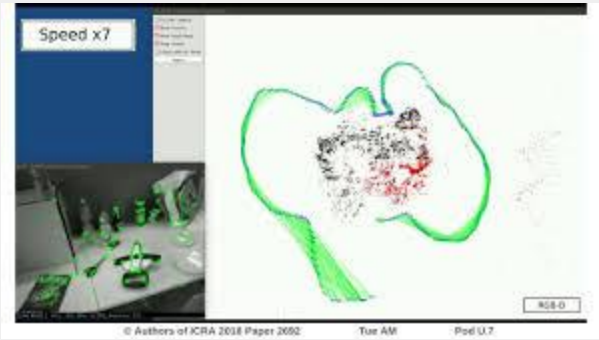
Execution time is a constraint but it depends on:

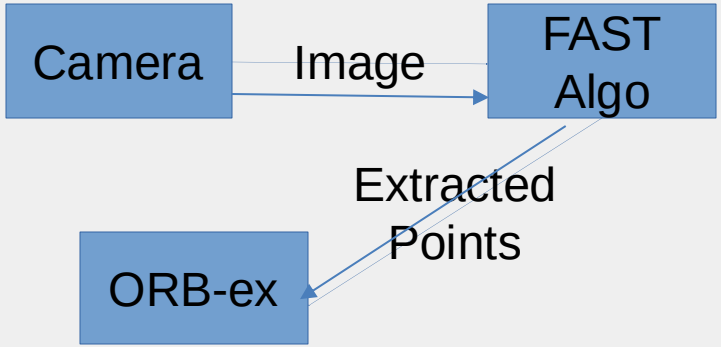
- **System State**
- **Environment State**

Algorithms execution time

Execution time is a constraint but it depends on:

- System State
- **Environment State**



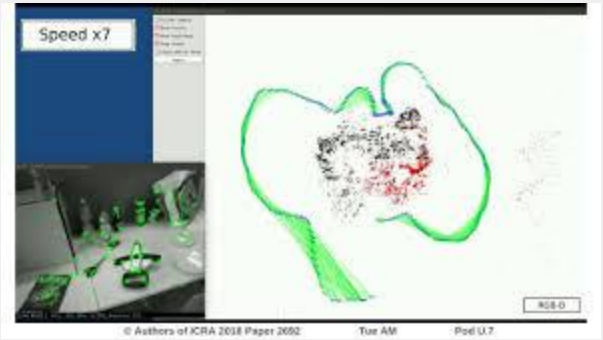


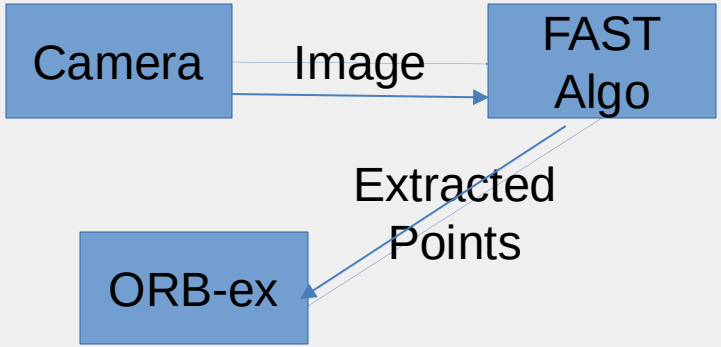
The ORB-ex execution time depends on the amount of extracted points

Algorithms execution time

Execution time is a constraint but it depends on:

- System State
- Environment State



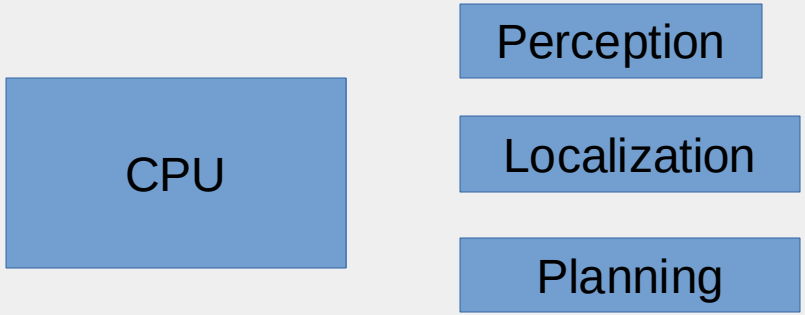


More points → More memory accesses and More computation

Algorithms execution time

Execution time is a constraint but it depends on:

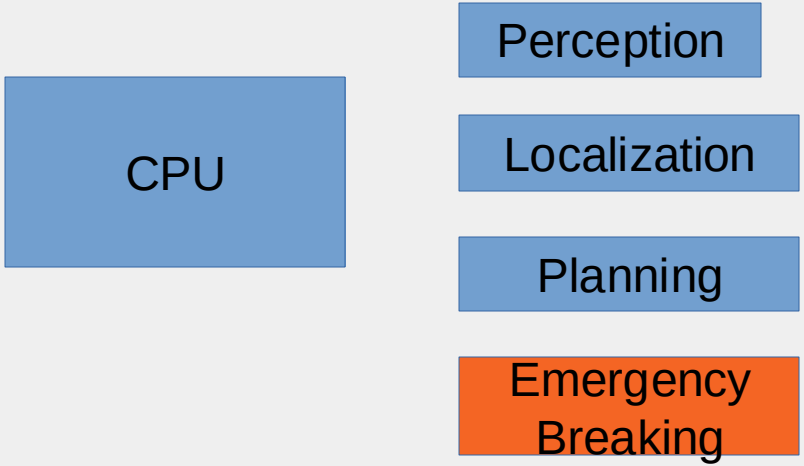
- System State
- Environment State



Algorithms execution time

Execution time is a constraint but it depends on:

- System State
- Environment State



Some processes can start in unpredictable way.

They impact the resource contention

Resources in GPU

GPU can be used to reduce the computational time:

- Parallelization of algorithms

- Less execution time
- Scalability

- Concurrent execution

- CPU can perform other tasks

But the use of the GPU increases the resource contention

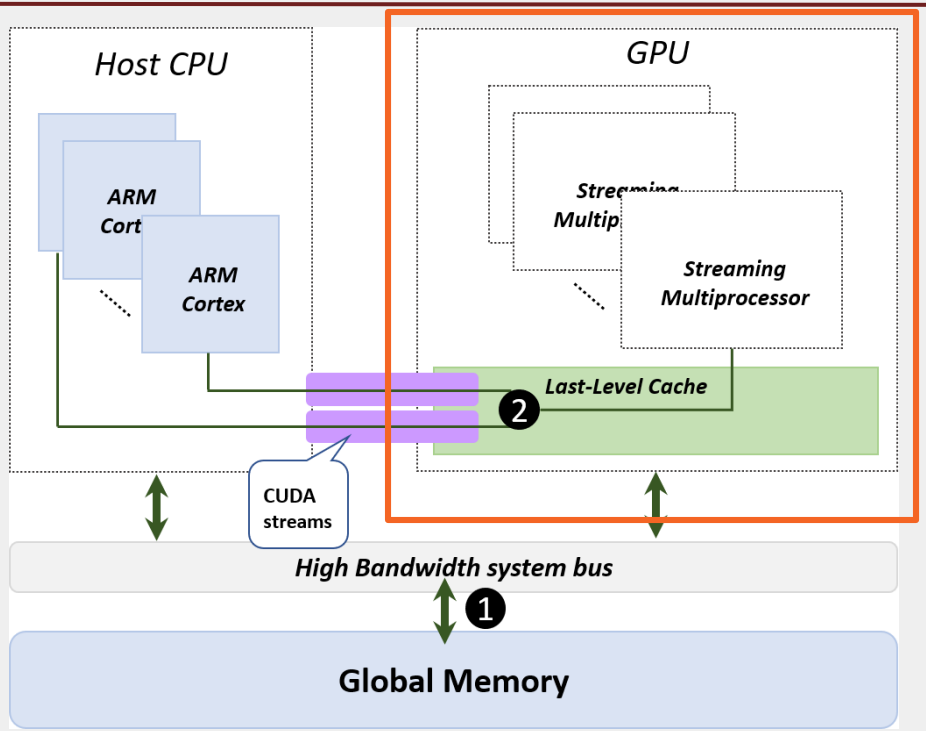
Resources in GPU

But the use of the GPU increases the resource contention:

- GPU is itself a resource of the system
- Memory
- Caches
- Streaming Multiprocessors (SMs)

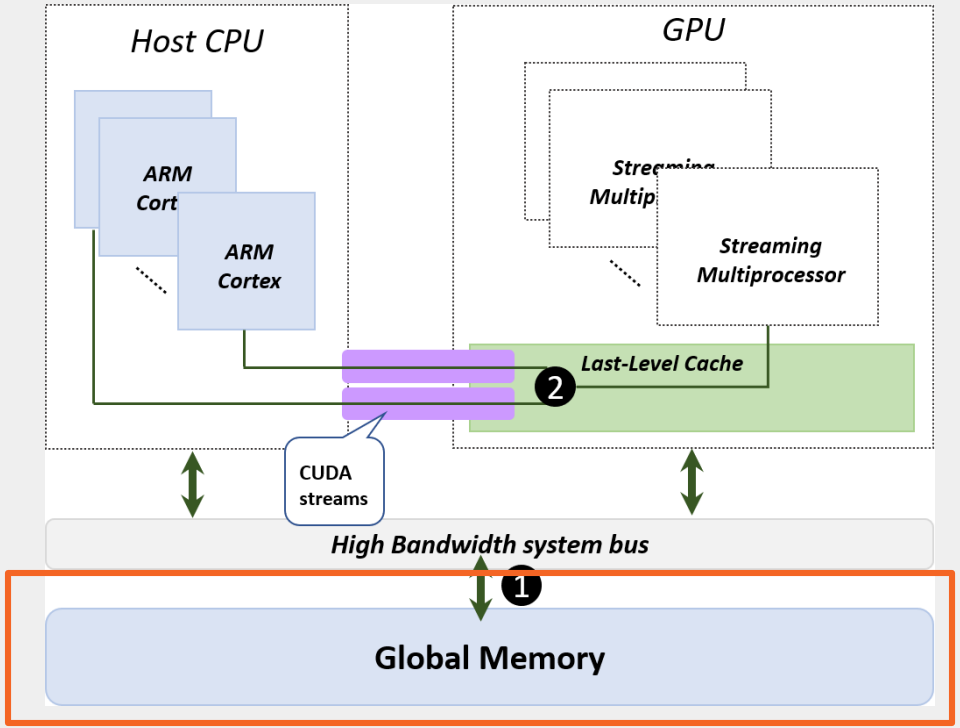
Resources in GPU

- **GPU as resource**
- Memory
- Caches
- SMs



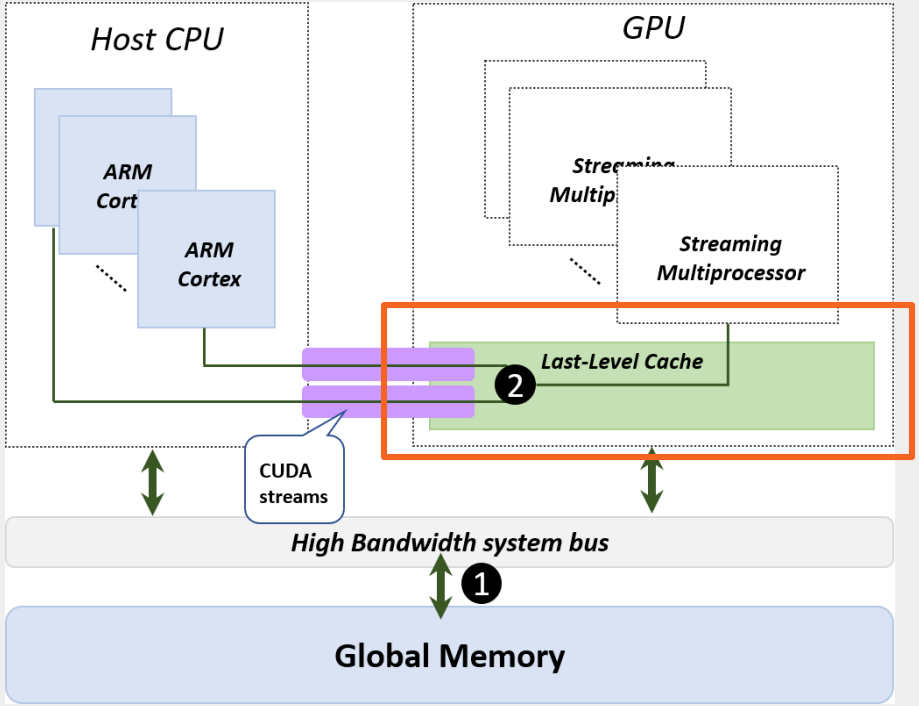
Resources in GPU

- GPU as resource
- **Memory**
- Caches
- SMs



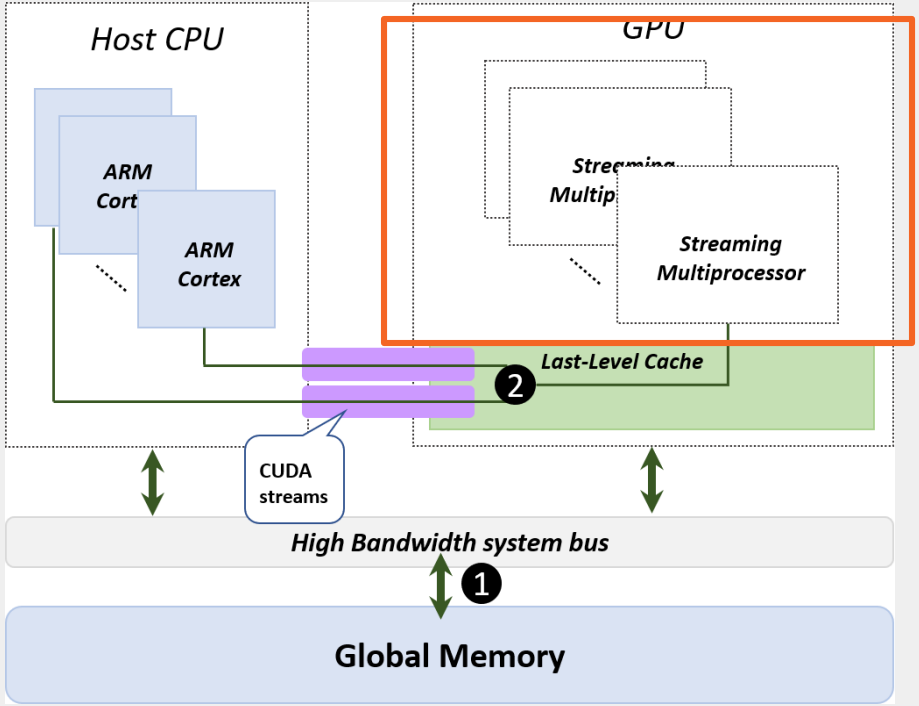
Resources in GPU

- GPU as resource
- Memory
- **Caches**
- SMs



Resources in GPU

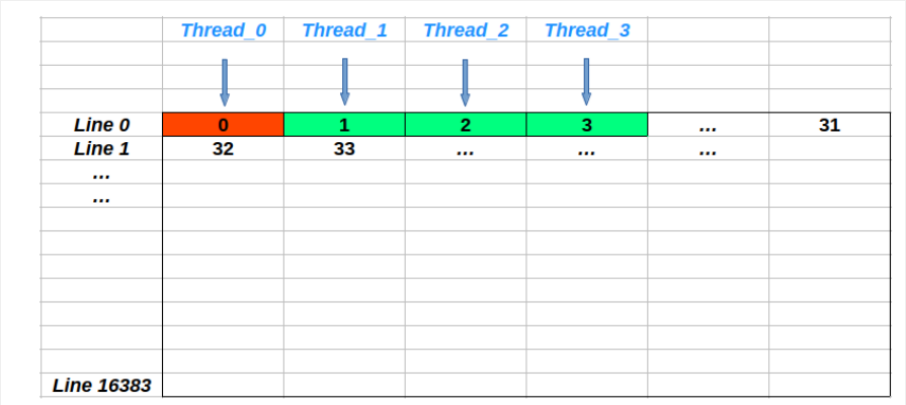
- GPU as resource
- Memory
- Caches
- **SMs**



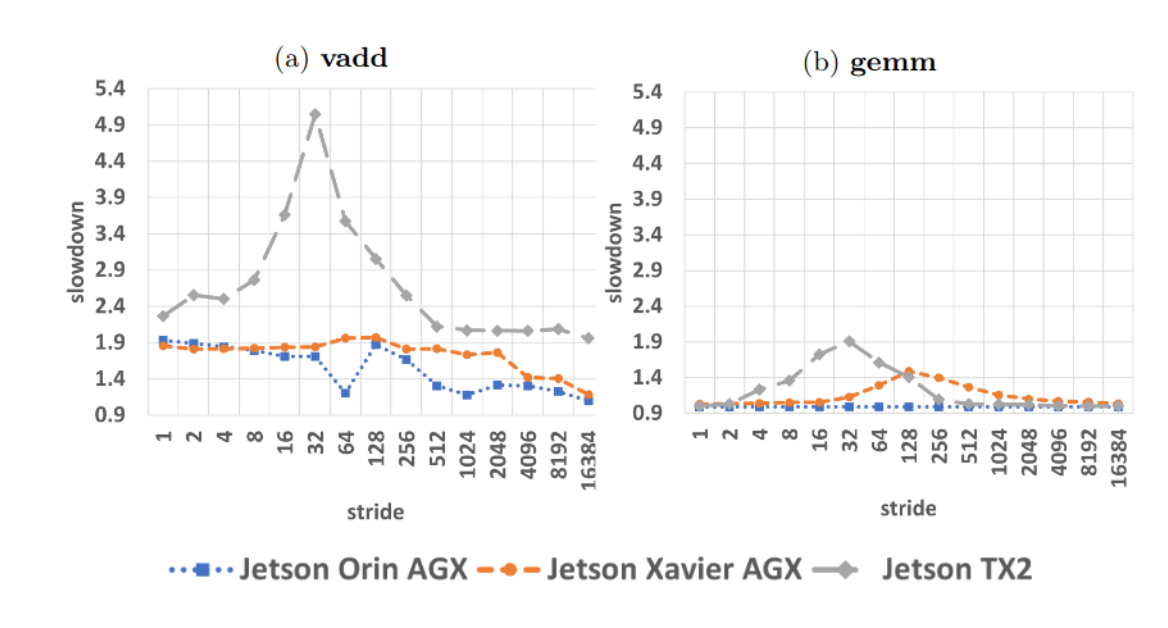
LLC (L2) interference experiments

- One task (vadd or gemm) on 1 SM
- Interference task on another SM

Interference task access memory with a stride to dirty the caches

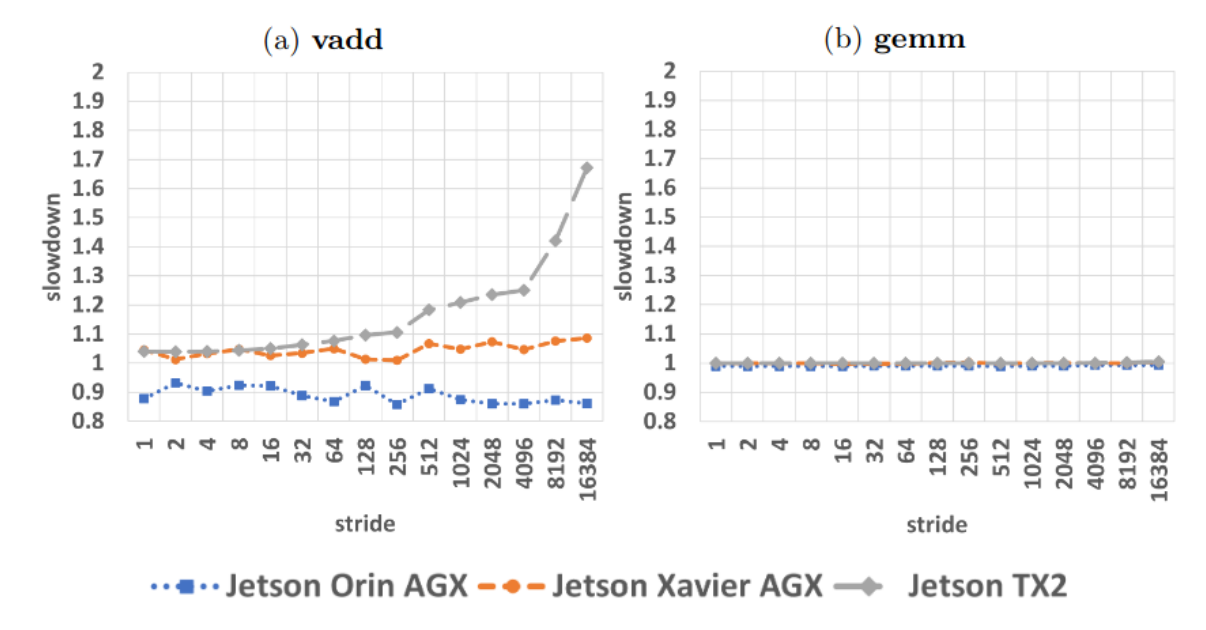


LLC (L2) Interference results



Slowdown caused by interference on L2 cache (Kernel)

LLC (L2) Interference results



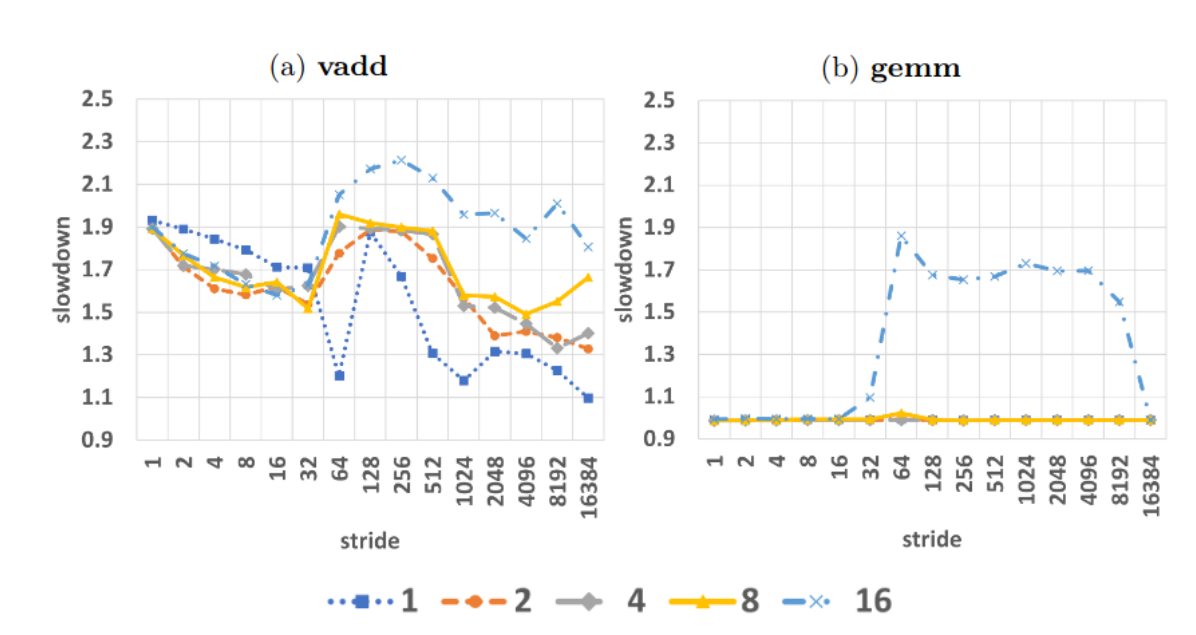
Slowdown caused by interference on L2 cache (Copy Engine)

SMs Contentions experiments

- One task (vadd or gemm) on 1 SM
- Interference tasks on other SMs

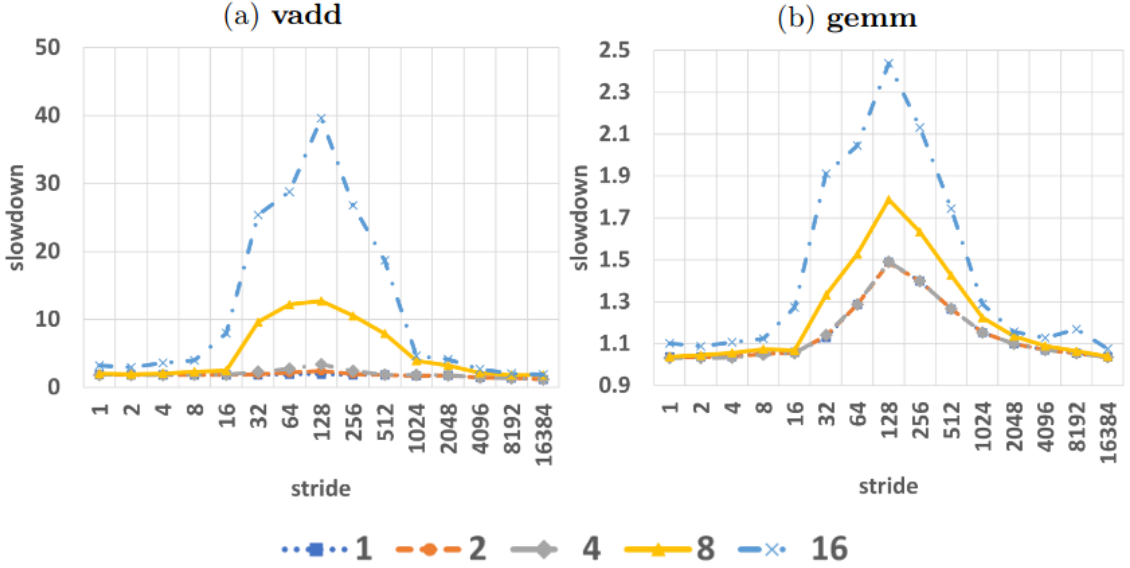
Varying the number of interferences tasks

SMs Contentions results



Slowdown caused by interference on L2 cache + SMs contentions (Jetson Orin)

SMs Contentions results



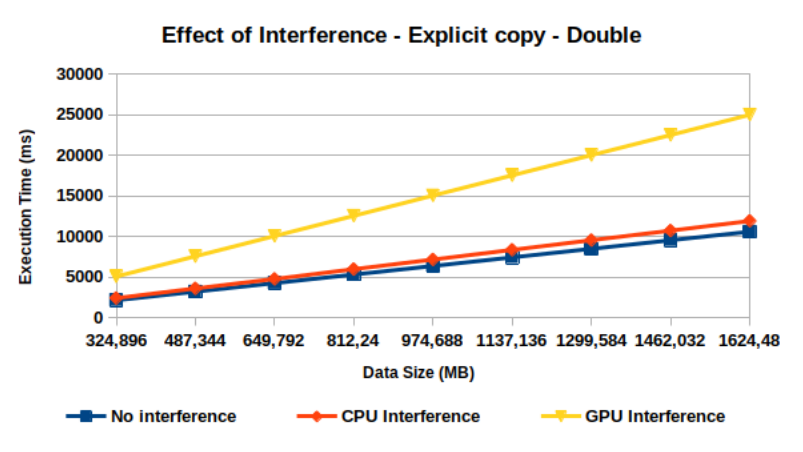
Slowdown caused by interference on L2 cache + SMs contentions (Jetson Xavier)

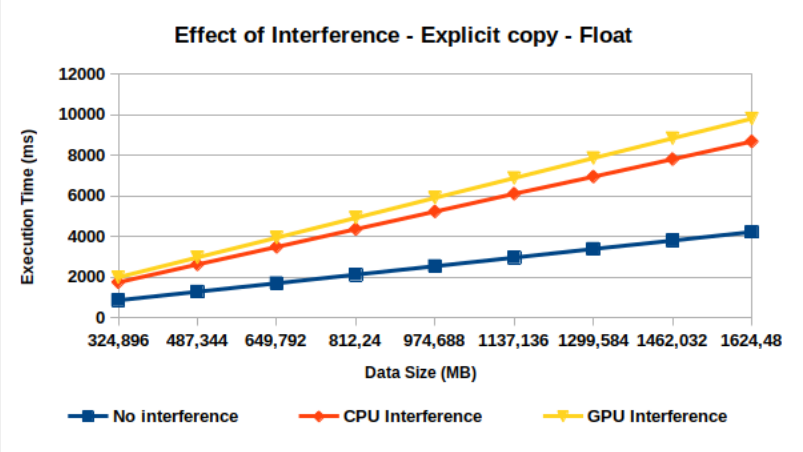
Global Memory Interference experiments

- Path planner that runs on GPU
- Interference tasks on other CPU cores
- Interference tasks on GPU copy engine

Interference tasks are memory intensive

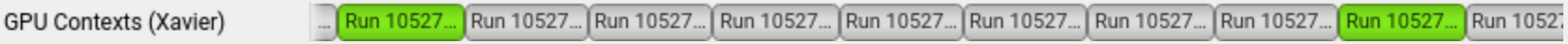
Global Memory results





Execution time of the Frenet Path Planner Algorithm with CPU/GPU interference (Jetson Xavier)

GPU contention



Context switch in GPU execution. One task at time is executed (unless you use streams)
(Jetson Xavier)

Resource contention in GPU (Recap)

- GPU is itself a resource of the system
- Memory
- Caches
- Streaming Multiprocessors (SMs)

- Context switch effect
- CPU processes interference
- Other kernels interference
- Other kernels interference

Resource contention in GPU (Recap)

- GPU is itself a resource of the system
- **Memory**
- Caches
- Streaming Multiprocessors (SMs)

- Context switch effect
- **CPU processes interference**
- Other kernels interference
- Other kernels interference

Resource contention in GPU (Recap)

- GPU is itself a resource of the system
- Memory
- **Caches**
- Streaming Multiprocessors (SMs)

- Context switch effect
- CPU processes interference
- **Other kernels interference**
- Other kernels interference

Resource contention in GPU (Recap)

- GPU is itself a resource of the system
- Memory
- Caches
- **Streaming Multiprocessors (SMs)**

- Context switch effect
- CPU processes interference
- Other kernels interference
- **Other kernels interference**

Resource contention in GPU (Recap)

- Concurrent execution of CPU processes
- Concurrent execution of GPU processes

- Context switch effect
- CPU processes interference
- Other kernels interference
- Other kernels interference

Resource contention in GPU (Recap)

- Concurrent execution of CPU processes
- Concurrent execution of GPU processes
(spawned by CPU processes)

- Context switch effect
- CPU processes interference
- Other kernels interference
- Other kernels interference

Resource contention in GPU (Recap)

- Concurrent execution of GPU processes (spawned by a single CPU process using streams)

- Context switch effect
- CPU processes interference
- Other kernels interference
- Other kernels interference (SMs contentions)

Resource contention in GPU (Recap)

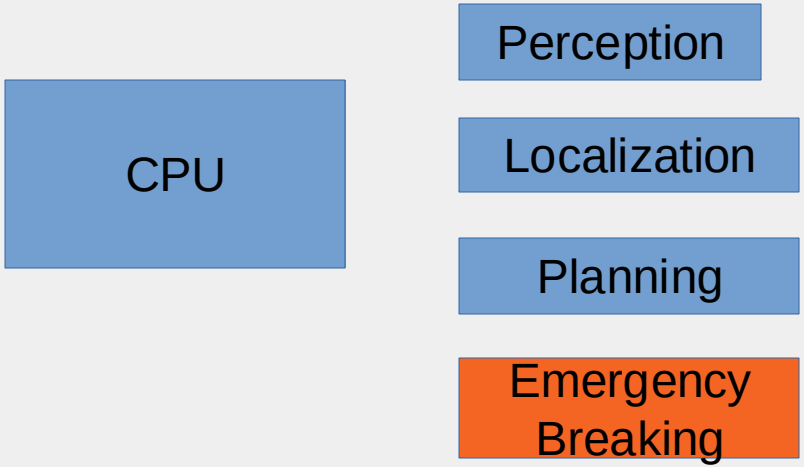
We can manage the GPU execution using streams but some aspect are unpredictable. Including CPU processes interference.

Resource contention in GPU (Recap)

We can manage the GPU execution using streams but some aspect are unpredictable. Including CPU processes interference.

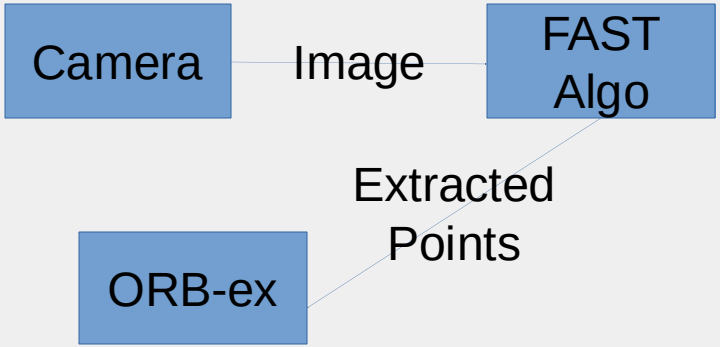
- System State

And it impact the execution time



Resource contention in GPU (Recap)

Moreover the System state can be affected by the environmental state. And it affect itself the execution time.



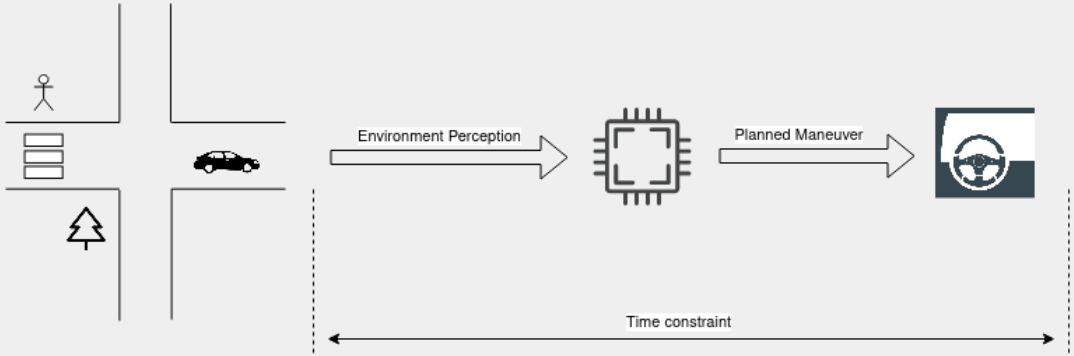
Resource contention in GPU (Recap)

- Environment State
- System state

Impact the execution time:
- Input of computation
- Resource contention

Resource contention in GPU (Recap)

It is difficult to estimate the execution time!



Estimation of interference

- Estimate the execution time of a kernel based on:
  - Input (Environment State)
  - other running task (System state)

It needs a large set of experiments
  - limit the set to a significant states
  - couple the estimation with scheduler aware of execution time and possible interferences