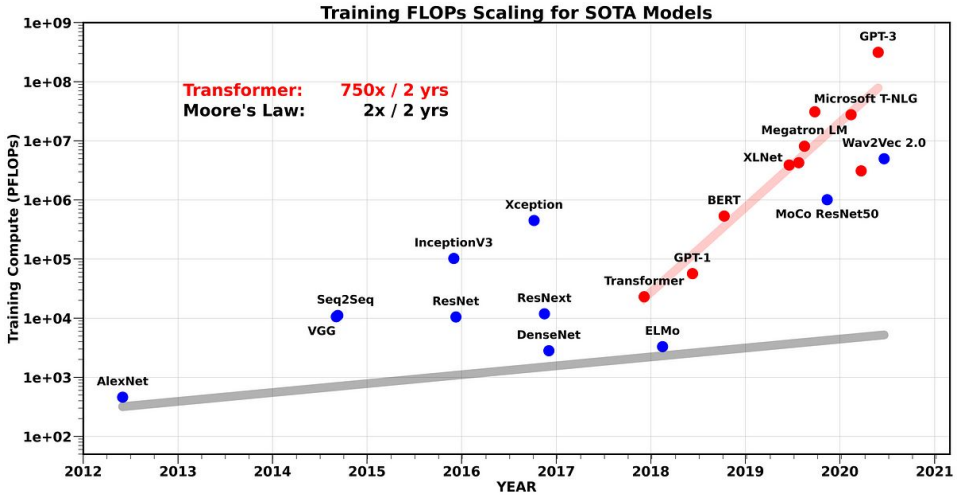
GPU Acceleration for Reinforcement Learning

Duncan Kampert & Robert Jan Schlimbach
SURF

GPUs for AI

Large NNs need many matrix multiplication operations

The more matrix operations, the better for the GPU



Conv2d and Transformer models scaled up massively in size over years

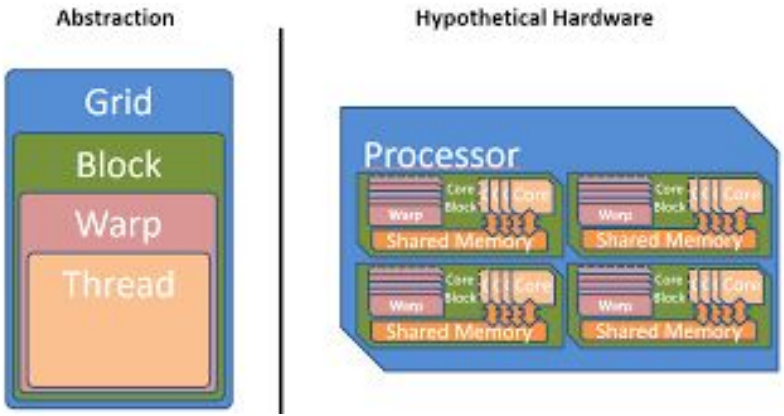
GPU architecture tailored towards those model types

GPU Refresher

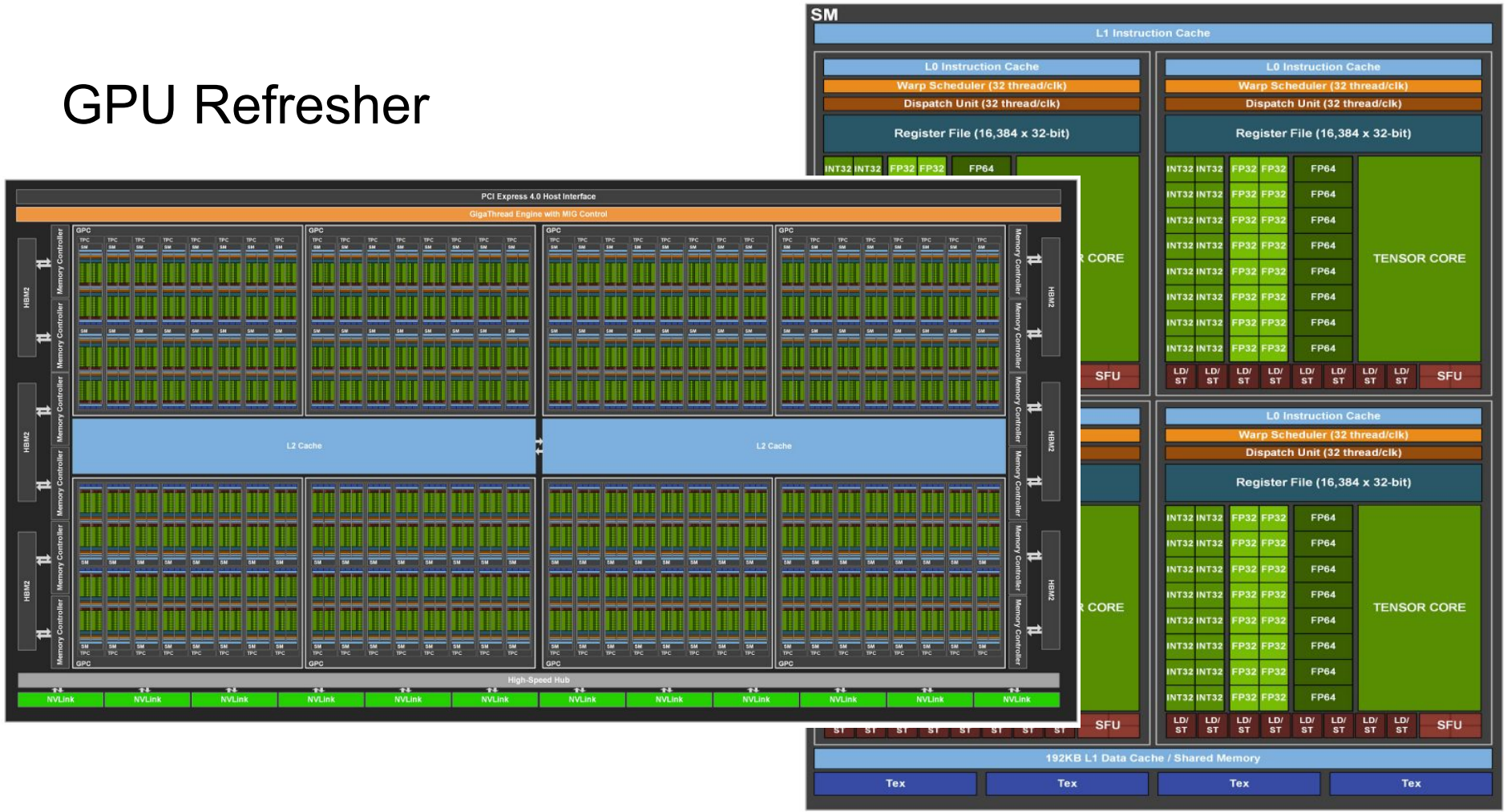
SIMD device

Tens of ‘threads’ run in lockstep.

Same instruction required for all threads in *block*.



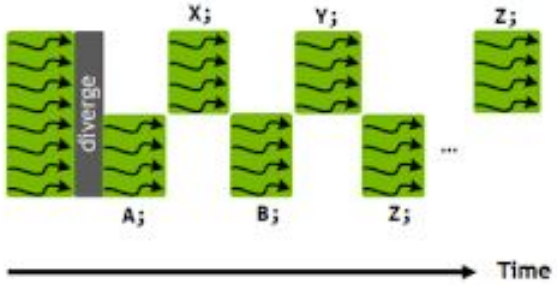
GPU Refresher



Why everything cannot run on the GPU

Lockstep requires the **same instruction**, otherwise you get *warp divergence*

```
if (threadIdx.x < 4) {
    A;
    B;
} else {
    X;
    Y;
}
Z;
```



Can have half the throughput

Rule of thumb: don’t use conditional statements

What makes it hard for RL to run efficiently

Typically:

- Smaller model size
- Mix of inference and training interleaved (AlphaGoZero)
- Mix of simulator (CPU) and training (GPU)

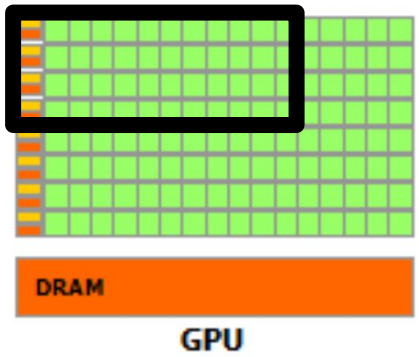
GPU upgrades (for everybody else)

GTX2080TI → 27 TFlops (bf16) with 11GB memory

A100 → 312 TFlops (!) with 40GB memory

Even though the GPU got 10x+ faster, you are unlikely to see this boost

GPU is not being used fully



How to work around this?

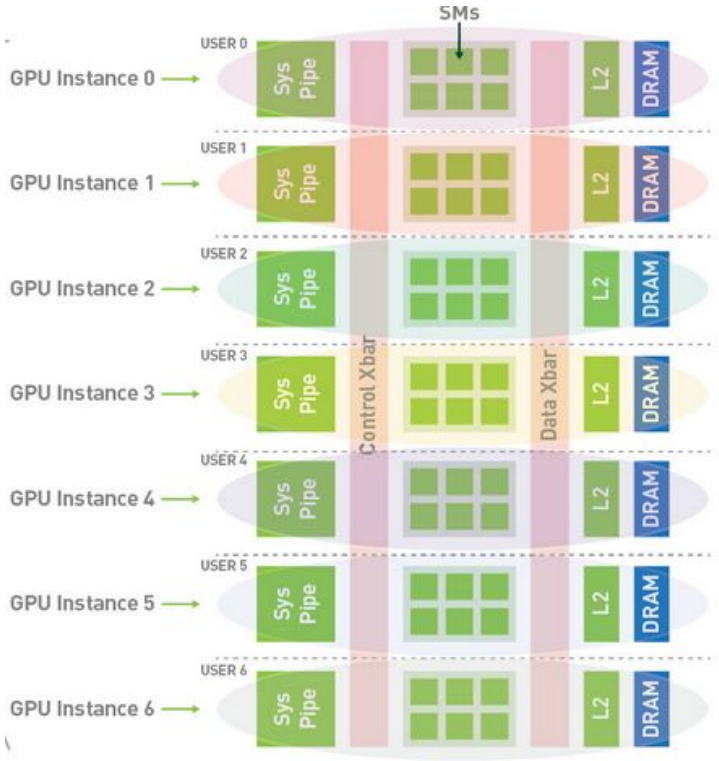
Multi-instance GPU (MIG)

CUDA MPS server

Multi-instance GPU

Easiest to work with, depends on sysadmins

- Splits the GPU into up to 7 parts
- A100 can be:
  - 2 small A100s with 20GB ram
  - 7 small A100s with 5.7GB ram
- Installed on Snellius in 2-split configuration
- Test it out!

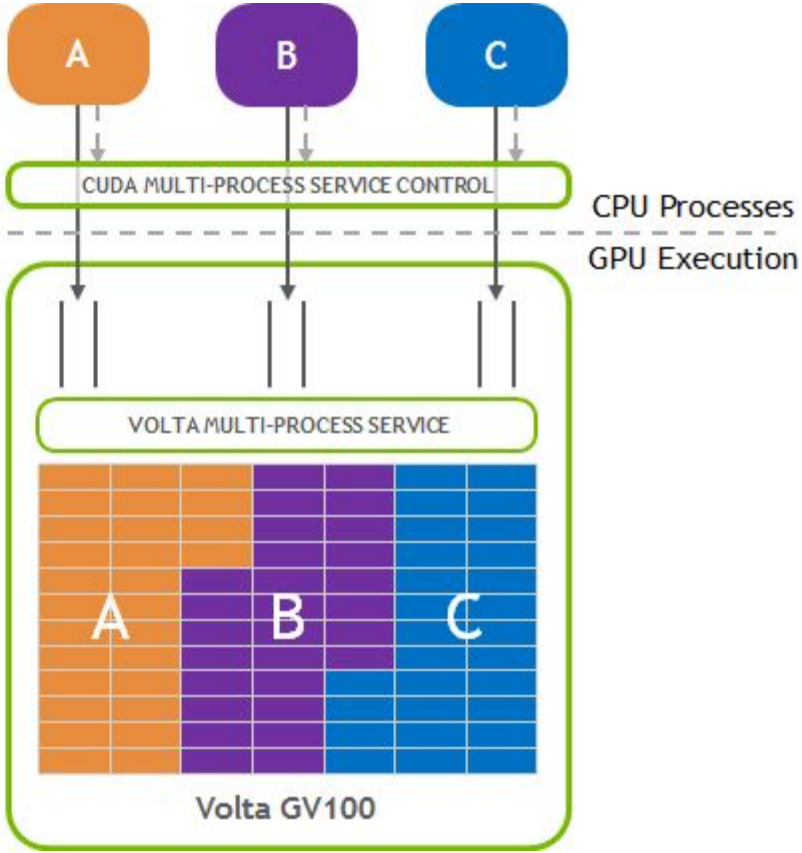


CUDA MPS

Requires some code change

- Allows for different processes to fill up the GPU

- Combine with parallel simulation → duplicate your model easily



Nothing vs MIG vs MPS

Test case

- AlphaGoZero on Connect4
- Very small model
- Run many simulations in parallel for data generation
- 500 Inference passes per data sample (MCTS)

	2 Processes	7 Processes	16 Processes
Nothing	0.50 / s	1.20 / s	1.07 / s
MIG (2 dev)	0.38 / s	1.39 / s	1.19 / s
MPS	0.74 / s	2.22 / s	5.10 / s

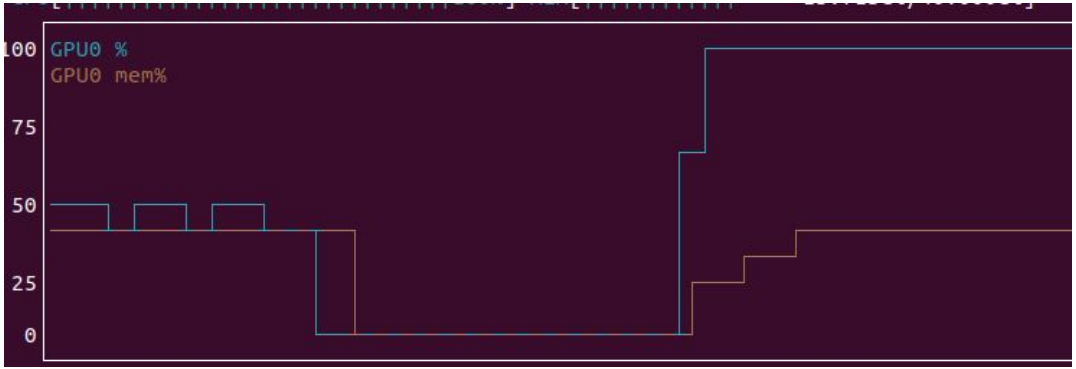
`nvidia-smi` and `nvtop`

These are two runs:

Nothing/MIG/MPS

Quiz:

- What uses the GPU better?
- What is on the left?
- What is on the right?



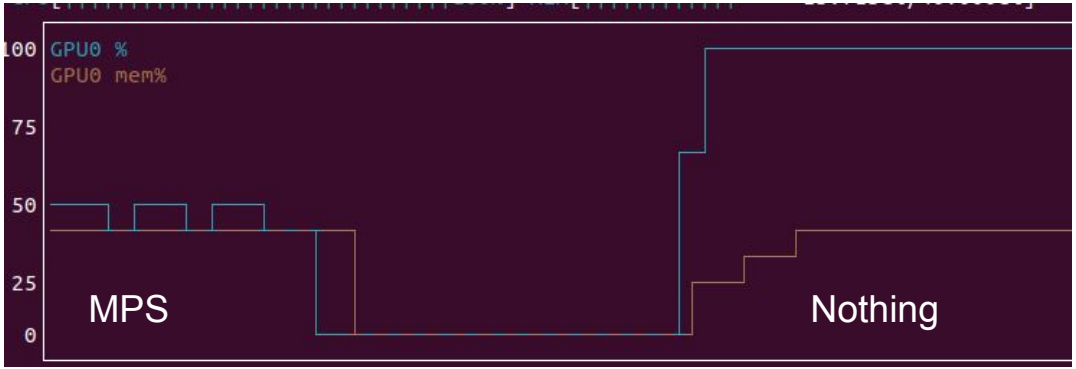
`nvidia-smi` and `nvtop`

These are two runs:

Nothing/MIG/MPS

Quiz:

- What uses the GPU better?
- What is on the left?
- What is on the right?



`nvidia-smi` and `nvtop`

The %GPU *only* tells you the amount of time *something* was running

Profiling is hard: there is no one-size-fits-all

- `import time; start = time.now(); train(); print(time.now() - start)`
- pytorch profiler → good overview
- dcgmi → specific GPU hardware information (quite cryptic)
- nsys → more for kernel programming, but can indicate which layers are slow

Conclusion

- Many instances of smaller models → MPS
- Small/medium sized models but don’t want to spend time → MIG

Ask us if you’re unsure what would be the best fit for your program/model

Documentation

MPS: https://docs.nvidia.com/deploy/mps/index.html

MIG: https://docs.nvidia.com/datacenter/tesla/mig-user-guide/index.html

nsys: https://docs.nvidia.com/nsight-systems/UserGuide/index.html

pytorch profiling: https://servicedesk.surf.nl/wiki/display/WIKI/PyTorch+Profiling

dcgmi: https://docs.nvidia.com/datacenter/dcgm/latest/user-guide/feature-overview.html

dcgmi-simple: https://servicedesk.surf.nl/wiki/display/WIKI/How+to+run+efficient+jobs

local nvme: --constraint=scratch-node

Robert jan links

rj slides - https://github.com/sara-nl/MLonHPC_2day_Okt2023/blob/main/Day2/slides/hardware.pdf

notebook - https://github.com/sara-nl/MLonHPC_2day_Okt2023/blob/main/Day2/notebooks/PyTorch_profiling.ipynb

module env - https://servicedesk.surf.nl/wiki/display/WIKI/Loading+modules

servicedesk: https://servicedesk.surf.nl/

best practices AI- https://servicedesk.surf.nl/wiki/pages/viewpage.action?pageId=74227856

a100 tutorial - https://servicedesk.surf.nl/wiki/display/WIKI/Deep+Learning+on+A100+GPUs

entire course- https://github.com/sara-nl/MLonHPC_2day_Okt2023

wiki general: https://servicedesk.surf.nl/wiki/display/WIKI/Snellius

events: https://www.surf.nl/agenda/onderzoek-en-ict?page=0

mailing list signup: https://www.surf.nl/en/training-courses-for-research (note: this will be used more starting next year, this year you will have to look at the event list manually)