Lecture: Accelerators

- Topics: GPU basics, accelerators for machine learning
- Wednesday: review session
- Next Monday 12/13, 1pm – 3pm: Final exam
SIMD Processors

- Single instruction, multiple data

- Such processors offer energy efficiency because a single instruction fetch can trigger many data operations

- Such data parallelism may be useful for many image/sound and numerical applications
GPUs

• Initially developed as graphics accelerators; now viewed as one of the densest compute engines available

• Many on-going efforts to run non-graphics workloads on GPUs, i.e., use them as general-purpose GPUs or GPGPUs

• C/C++ based programming platforms enable wider use of GPGPUs – CUDA from NVidia and OpenCL from an industry consortium

• A heterogeneous system has a regular host CPU and a GPU that handles (say) CUDA code (they can both be on the same chip)
The GPU Architecture

- SIMT – single instruction, multiple thread; a GPU has many SIMT cores

- A large data-parallel operation is partitioned into many thread blocks (one per SIMT core); a thread block is partitioned into many warps (one warp running at a time in the SIMT core); a warp is partitioned across many in-order pipelines (each is called a SIMD lane)

- A SIMT core can have multiple active warps at a time, i.e., the SIMT core stores the registers for each warp; warps can be context-switched at low cost; a warp scheduler keeps track of runnable warps and schedules a new warp if the currently running warp stalls
The GPU Architecture
Architecture Features

• Simple in-order pipelines that rely on thread-level parallelism to hide long latencies

• Many registers (~1K) per in-order pipeline (lane) to support many active warps

• When a branch is encountered, some of the lanes proceed along the “then” case depending on their data values; later, the other lanes evaluate the “else” case; a branch cuts the data-level parallelism by half (branch divergence)

• When a load/store is encountered, the requests from all lanes are coalesced into a few 128B cache line requests; each request may return at a different time (mem divergence)
GPU Memory Hierarchy

• Each SIMT core has a private L1 cache (shared by the warps on that core)

• A large L2 is shared by all SIMT cores; each L2 bank services a subset of all addresses

• Each L2 partition is connected to its own memory controller and memory channel

• The GDDR5 memory system runs at higher frequencies, and uses chips with more banks, wide IO, and better power delivery networks
Why the recent emphasis on accelerators?

- Stagnant single- and multi-thread performance with general-purpose cores
  - Dark silicon (emphasis on power-efficient throughput)
  - End of scaling
  - No low-hanging fruit

- Emergence of deep neural networks
Commercial Hardware

Machine Learning accelerators

Google
- Google TPU (inference and training)
- Recent NVIDIA chips (Volta, NVDLA)

NVIDIA

Microsoft
- Microsoft Brainwave, Catapult
- Intel Loihi and Nervana
- Cambricon
- Graphcore (training)
- Cerebras (training)
- Groq (inference)
- Tesla FSD (inference)
Machine Learning Workloads

- Dominated by dot-product computations
- Deep neural networks: convolutional and fully-connected layers
- Convolutions exhibit high data reuse
- Fully-connected layers have high memory-to-compute ratio
Google TPU

- **Version 1:** 15-month effort, basic design, only for inference, 92 TOPs peak, 15x faster than GPU, 40 W 28nm 300 mm² chip
- **Version 2:** designed for training, a pod is a collection of v2 chips connected with a torus topology
- **Version 3:** 8x higher throughput, liquid cooled

Ref: Google
Weights are pre-loaded during previous phase and inputs flow left to right.
Tesla FSD

• Tesla’s custom accelerator chip, shipping in cars since April 2019
• FSD sits behind the glovebox, consumes 72W
• 18 months for first design, next generation out in 2 years

Image Source: Tesla
NN Accelerator Chip (NNA)

• Goals: under 100 W (2% impact on driving range, cooling, etc.), 50 TOPs, batch size of 1 for low latency, GPU support as well, security/safety.

• Security: all code must be attested by Tesla

• Safety: two completely independent systems on the board that verify every output

• The FSD 2.5 design (GPU based) consumes 57 W, the 3.0 design consumes 72 W, but is 21x faster (72 TOPs)

• 20% saving in cost by designing their own chip
Detection and tracking are two of the heavy-hitters and are DNN based.
NNA Pipeline

- On-chip network moves inputs to LPDDR4: 128b@4.2 Gb/s = 68GB/s
- Includes: video encoder, image signal processor, 600 Gflop GPU, and 12-core 2.2 GHz CPU, hardware for ReLU and pooling layers
- Most importantly: 2 NN accelerator cores, each with 96x96 grid of MACs and 32MB SRAM, 2 GHz, 36 TOPs per core
NVIDIA Volta GPU

- 640 tensor cores
- Each tensor core performs a MAC on 4x4 tensors
- Throughput: 128 FLOPs x 640 x 1.5 GHz = 125 Tflops
- FP16 multiply operations
- 12x better than Pascal on training and 6x better on inference
- Basic matrix multiply unit – 32 inputs being fed to 64 parallel multipliers; 64 parallel add operations

\[ D = \begin{pmatrix}
    A_{0,0} & A_{0,1} & A_{0,2} & A_{0,3} \\
    A_{1,0} & A_{1,1} & A_{1,2} & A_{1,3} \\
    A_{2,0} & A_{2,1} & A_{2,2} & A_{2,3} \\
    A_{3,0} & A_{3,1} & A_{3,2} & A_{3,3}
\end{pmatrix}_{\text{FP16 or FP32}} \quad \begin{pmatrix}
    B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\
    B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \\
    B_{2,0} & B_{2,1} & B_{2,2} & B_{2,3} \\
    B_{3,0} & B_{3,1} & B_{3,2} & B_{3,3}
\end{pmatrix}_{\text{FP16}} \quad + \quad \begin{pmatrix}
    C_{0,0} & C_{0,1} & C_{0,2} & C_{0,3} \\
    C_{1,0} & C_{1,1} & C_{1,2} & C_{1,3} \\
    C_{2,0} & C_{2,1} & C_{2,2} & C_{2,3} \\
    C_{3,0} & C_{3,1} & C_{3,2} & C_{3,3}
\end{pmatrix}_{\text{FP16 or FP32}} \]
