ECE 574 – Cluster Computing
Lecture 23

Vince Weaver

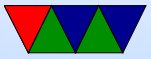
https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

18, 23, 25 April 2024

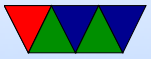
Announcements

- Remember no final exam for this course
- Still catching up on grading homeworks and 2nd exam
- Don't forget to do online student evaluations
- Remember Final Project Writeup (due 3rd May)



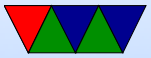
HW #7 notes

- Fine grained parallelism
- Running on the Pi-cluster
  - Test with np=7, some code failed that worked with 2 9d4b6548fa8c6ff66602ef5993aca90f common seems to be not gathering in the extra lines
  - Reading from each core rather than Bcast doesn't help anything.



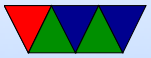
HW #7 – Pi Cluster Results

- space station input only scale to 4 cores?
- See raw data next slides
- On 4 cores, MPI smart enough to use local methods for broadcast which are much faster. Note how much slower once it has to go out over ethernet
- To get best usage of an MPI cluster you need to have a lot of CPU usage to make up for the slow network behavior



HW #7 – Input Sizes

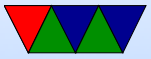
input	dimension	size
butterfinger	320x320	307kB
earth	2048x2048	13MB
spacestation	4288x2929	38MB
Jan5	10848x10848	353MB



HW #7 – Pi Cluster/Spacestation

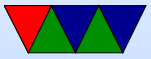
cores	load	bcast	convolve	combine	gather	tail	store	total
1	0.2	0	3.9	0.3	0.04	0	1.4	5.9
2	0.2	0.1	2.5	0.2	0.2	0	1.4	4.6
4	0.2	0.1	1.2	0.09	0.5	0	0.3	2.3
8	0.2	1.3	0.2	0.04	0.3	0	0.3	2.4
16	0.2	1.5	0.4	0.02	0.5	0	1.0	3.7
20	0.2	2.7	0.3	0.02	0.6	0	0.4	4.0

note: refuses to run with 32 threads



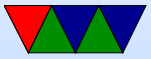
HW #7 – Pi Cluster/Jan5

cores	load	bcast	convolve	combine	gather	tail	store	total
1	2.2	0	56	3.1	0.4	0	6.4	68
2	2.2	0.4	33	1.5	1.6	0	5.5	44
4	2.2	0.4	27	0.8	3.0	0	5.6	39
8	2.2	13	2.4	0.3	4.9	0	4.7	28
16	2.2	19	1.0	0.2	3.6	0	2.9	29
20	2.2	29	0.8	0.1	4.7	0	3.6	41



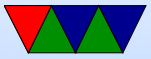
HW #7 – Historical Pi2 spacestation results

cores	load	bcast	convolve	combine	gather	tail	store	total
1	1.0	0	12.8	3.8	0.1	0	3.4	21.2
2	1.0	0.1	6.4	1.9	1.8	0	2.4	13.7
4	1.0	0.3	3.2	0.9	3.0	0	2.4	10.9
8	1.0	5.6	1.7	0.5	4.6	0	2.4	15.8
16	1.0	7.3	0.7	0.2	6.5	0	2.4	18.2
32	1.0	8.0	0.3	0.1	6.4	0	2.4	18.3
64	1.0	8.8	0.1	0.06	6.9	0	2.4	19.5



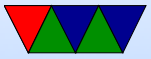
HW #8 notes

- Be careful memory copying, if copying an array of 9 ints need to copy 36 bytes (not 9)
- Also, you can pass in ints as parameters (no need to allocate space and then memcpy in. Or you could, but if you do you would use points and allocate space properly)
- Be sure you are using *unsigned char* for the image data, not signed char.
- Limits and matrix indexing



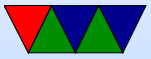
HW #8 results with jan5 image

Type	Load	Copy	Convolve	Combine	Store	Total
OMPx16	1s		0.882	0.135	0.9	3.12
MPIx16	1s	0.5+1.4	0.6	0.1	1.0	4.9
Cuda	1s	0.3	0.2	0.2	1.0	3.3
OpenCL CUDA	1s	0.2	0.4	0.4	0.9	2.9
OpenCL intel	1s	0.3	0.2	0.2	0.9	3.0
OpenCL pocl	1s	0.4	0.6	0.7	1.0	3.4



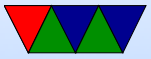
HW #9 notes

- Most issues were going out of bounds due to bad border calculation
- On C on CPU this will give you segfault
- With an accelerator, this still is illegal, but the asynchronous error reporting might not report the issue until later (which makes it hard to debug what's going on)



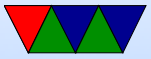
Myths and Legends in High-Performance Computing

- Paper by Matsuoka, Domke, Wahib, Drozd, and Hoefler



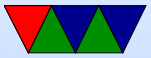
Myth1: Quantum Computing

- Unclear how quickly will progress
- Potentially limited to only certain workloads
- Seems unlikely would replace HPC



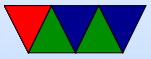
Myth2: Deep Learning

- Concerns about accuracy
- Depends what kind of HPC output you are looking for, and whether you need exact numerical accuracy in results



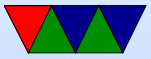
Myth3: Extreme Specialization

- Like cellphones, where each device has special hardware function unit
- Custom accelerators?
- The only really successful accelerator has been GPUs, and that's mostly due to most workloads being memory bound
- Three reasons specialization will fail
  - Most accelerators help with strong scaling (CPU bound) but that's not the current problem



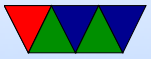
Also splitting up problem is complex, only works on modern systems as the accelerators are homogeneous

- The old idea that "transistors are free" is going down as Moore's Law stalls. Mostly unused silicon becomes expensive
- Software for accelerators is hard to write, so the benefit must outweigh this (especially for relatively small number of supercomputers, vs the millions of devices regular companies sell)



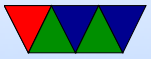
Myth4: Everything will run on Accelerators

- By accelerator they mostly mean GPU? CPUs not needed?
- Workloads are often compute-bound, memory bandwidth bound, or memory latency bound
- Historically things were compute and bandwidth bound. GPUs can handle those cases. But now things becoming latency bound, which CPUs are better at?
- There are some strong scaling workloads where FPGA or CPUs can still outclass GPU



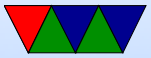
Myth5: FPGAs (Reconfigurable Hardware)

- Promises of 100x speedup?
- Intel and AMD had both bought FPGA companies without much result
- FPGAs less dense and less power efficient than CPUs
- Could be better if "hard" FPU blocks included on board FPGA
- Though if its a large number of FPUs with some glue logic, you're essentially back to designing GPUs



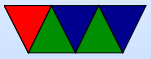
Myth6: Zettascale

- Intel claims Zettaflop by end of decade
- history
  - 1 teraflop 1997 (Asci red) 0.85MW
  - 1 petaflop 2008 (roadrunner) 2.3MW
  - 1 exaflop 2021 (China OceanLight?) 35MW, 2022 (Frontier) 21MW
- Is it possible? Technically yes, unless Linpack stops scaling
- More likely might be 2038 at 50MW



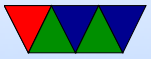
Myth7: Memory/Core Ratio

- People concentrated on flops
- Led to a data movement crisis
- Rent's rule? (?) (1960s IBM, correlation between logic blocks and number of I/O pins needed to access it)
- Can work on optimizing this



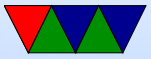
Myth8: Diassgregation

- What is Memory disaggregation? (sort of like network attached storage, but for RAM rather than disks)
- Silicon Photonics
- Two issues: low cost manufacture, optical switching
- Circuit switching (crossbar?) vs Packet Switching
- Hard to buffer light or process in-flight
- Compute Express Link (CXL)
- Speed of Light Issues



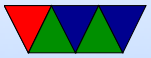
Myth9: Applications are Improving

- Three ways to avoid end of Moore's Law (hardware)
  - new architectures
  - new materials (i.e., move on from CMOS)
  - abandon von neumann computers, move to quantum or something else
- Is there an Algorithmic Moore's Law?
  - is there a limit to software optimization too?



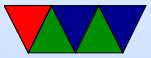
Myth10: End of Fortran

- What is the proper lay of abstraction for optimization?
  - low-level virtual machine?
  - C/C++/assembly?
  - dataflow representation?
- Domain specific languages?
- Languages like Python hard to optimize
  but subsets like Numpy can be?



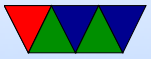
Myth11: Low or Mixed Precision

- fp64 expensive
- Can break application
- Abandon IEEE-754?
- AI had interesting issue, fp16 faster than fp32, but did not always converge. bfloat16 had wider range, but still issue. So then 19-bit (tensorfloat-32) still faster than fp32 but better behaved



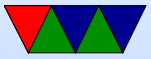
Myth12: The Cloud

- Cloudification of Supercomputers
  many supercomputers have cloud-like features
- Supercomputerification of Clouds



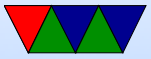
Related Supercomputer Fugaku Note

- Partnership with Amazon to create runtime identical to Fugaku but in amazon cloud (ARM processor)
- During peak times if SC busy, can run in cloud instead
- Can also develop on cloud before moving to SC



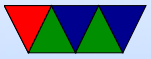
Edge Computing

- Trend?
- Fog computing
- Funny how things swing back and forth from edge to cloud and back



Upcoming Exaflop Systems

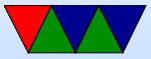
- Aurora – see below
- El Capitan – LLNL (see below)
- Jupiter (in EU) – NVIDIA Grace Hopper GH200, NVIDIA ARM CPUs. Rhea chips (also ARM). Infiniband NDR Dragonfly+, 21PB flash storage, 700PB tape backup



More Exascale

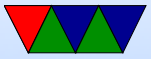
- Exascale Day – 10/18
- Frontier (OLCF-5)
  ORNL
  1.5 Exaflops, $600 million
  AMD Epyc and Radeon Instinct GPUs, 30MW, 100 racks

- Aurora
  DOE (Argonne)
  Intel and Cray (now HPE)



originally supposed to be 2018 with Xeon Phi
1 exaFLOP
$500 million
general scientific community. low carbon tech, subatomic particles, cancer, cosmology, solar cells
over 9000 nodes, each two Intel Sapphire Rapids CPUs, Golden Cove, 10nm, DDR5 RAM
6 Xe ponte vecchio GPUs, chiplets

- El Capitan
  LLNL (2023), NNSA
  $600 million



Replace Sierra (IBM Power9 + Nvidia)

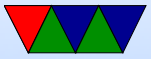
2 Exaflops

Zen 4

less than 40MW

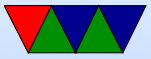
Infinity Fabric

Connecting nodes is AMD slingshot fabric, 200Gb/s, one port per CPU



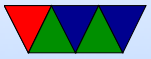
Zettascale

- Eurolab- 4 -HPC Long-Term Vision on High-Performance Computing Editors: Theo Ungerer, Paul Carpenter
- zettascale by 2030?
- convergence with big data?
- deep neural networks?
- die-stacking
- non-volatile memory
- resistive computing?
- neuromorphic computing? – try to replicate nerves in



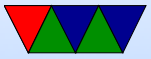
silicon

- quantum computing?
- nanotubes?
- graphene/diamond based transistors?
- optical networks on die / Terahertz communication
- HP Labs "the Machine"



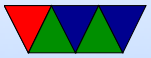
Zettascale

- Challenges
- Lines of code. 10-100 Euro per line?
- Approximate Computing
- auto-tuning
- debugging and profiling
- extreme data
- cloud, big data
  modern data centers 20MW cover 17 football fields
- exabytes of data, merge with cloud

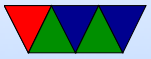


Disruptive Tech

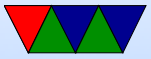
- Moores Law continues? 1.5nm by 2030?
- DRAM to 7.7nm in 2028, 32GB/chip? Scaling DRAM below 20nm hard
- Might be stuck at 32GB unless something new happens
- DUV argon Flouride excimer lasers, 193nm
  (deep ultraviolet) excited dimer, noble gas plus reactive gas
- die stacking, chiplets
- non-volatile memory



- spintronics
- memristors
- Photonics, 15ps/mm in silicon, 5ps/mm in waveguides
  stacked chips can have photonic layer
- mode-division multiplexing, free-air propagation, plasmonics
- photonic non-volatile mem, photonic computing
- memristive computing
- neuromorphic
- quantum computing
  d-wave, qubits, mili-Kelvin, new algos



- nanowires
- graphene, how you make it, 100GHz transistor
- diamond transistors



Last Notes

- Near memory / in memory computing
- power
- analog computing
- end of von neuman (memory hierarchies)
- Green computing, liquid nitrogen temps (memory story)
- System software, programming languages