ECE 574 – Cluster Computing
Lecture 2

Vince Weaver

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

vincent.weaver@maine.edu

19 January 2017

Announcements

- Put your name on HW#1 before turning in!
- Homework #2 will be posted. Articles to read.
- I'll be in Arizona on Tuesday for ISPASS Program Committee meeting. I'll (hopefully) be back Thursday.

Top500 List – November 2016

#	Name	Country	Arch	Proc	Cores	Max/Peak TFLOPS	Accel	Power kW
1	TaihuLight	China	Sunway	Sunway	10,649,600	93014/125435	?	15,000
2	Tianhe-2	China	x86	IVB	3,120,000	33862/54902	xeon-phi	18,000
3	Titan	USA/ORNL	x86	Opteron	560,640	17590/27112	NVD K20	8,209
4	Sequoia	USA/LLNL	Power	BG/Q	1,572,864	17173/20132	?	7,890
5	Cori	USA/LBNL	x86	Cray	622,336	14014/27880	Xeon Phi	3,939
6	Oakforest	Japan	x86	??	556,104	13554/24913	Xeon Phi	2,719
7	K computer	Japan/RIKEN	SPARC	VIIIfx	705,024	10510/11280	?	12,660
8	Piz Daint	Switzerland	x86	Intel	206,720	9779/15998	NVD Tesla	1,312
9	Mira	USA/Argonne	Power	BG/Q	786,432	8586/10066	?	3,945
10	Trinity	USA/LANL	x86	??	301,056	8100/11078	?	4,233
11	UK Met	UK	x86	Cray	241,920	6765/8128	?	?
12	Marconi	Italy	x86	?	241,808	6223/10833	Xeon Phi	?
13	Pleiades	US/NASA	x86	SGI	241,108	5951/7107	?	4407
14	Hazel Hen	Germany	x86	?	185,088	5640/7403	?	3615
15	Shaheen II	Saudi Arabia	x86	SNB-EP	196,608	5537/7235	?	2,834
16	Pangea	France	x86	?	220,800	5283/6712	?	4150
17	Stampede	USA/TACC	x86	SNB-EP	462,462	5168/8520	XeonPhi	4,510
18	Theta	USA/Argonne	x86	?	207,360	5095/8626	XeonPhi	1087
19	Juqeen	DE/Julich	Power	BG/Q	458,752	5008/5872	?	2,301
20	Cheyenne	USA/NCAR	x86	??	144,900	4788/5332	?	1727
21	Vulcan	USA/LLNL	Power	BG/Q	393,216	4293/5033	?	1,972
22	Abel	USA/Geo	x86	??	145920	4042/5369	?	1800

How long does it take to run LINPACK? How much money does it cost to run LINPACK?

How much RAM? How much cooling?

Turnover since last time I taught the class?

What goes into a top supercomputer?

- Commodity or custom
- Architecture: x86? SPARC? Power? ARM embedded vs high-speed?
- Memory
- Storage

  How much?

  Large hadron collider one petabyte of data every day

  Shared? If each node wants same data, do you need to replicate it, have a network filesystem, copy it around with jobs, etc? Cluster filesystems?

- Reliability. How long can it stay up without crashing?
  Can you checkpoint/restart jobs?
  Sequoia MTBF 1 day.
  Blue Waters 2 nodes failure per day.
  Titan MTBF less than 1 day
- Power / Cooling
  Big river nearby?
- Accelerator cards / Heterogeneous Systems
- Network
  How fast? Latency? Interconnect? (torus, cube,

hypercube, etc)
Ethernet? Infiniband? Custom?

- Operating System
  Linux? Custom? If just doing FP, do you need overhead of an OS? Job submission software, Authentication
- Software – how to program?
  Too hard to program can doom you. A lot of interest in the Cell processor. Great performance if programmed well, but hard to do.
- Tools – software that can help you find performance problems

Introduction to Performance Analysis

What is Performance?

- Getting results as quickly as possible?
- Getting *correct* results as quickly as possible?
- What about Budget?
- What about Development Time?
- What about Hardware Usage?
- What about Power Consumption?

Motivation for HPC Optimization

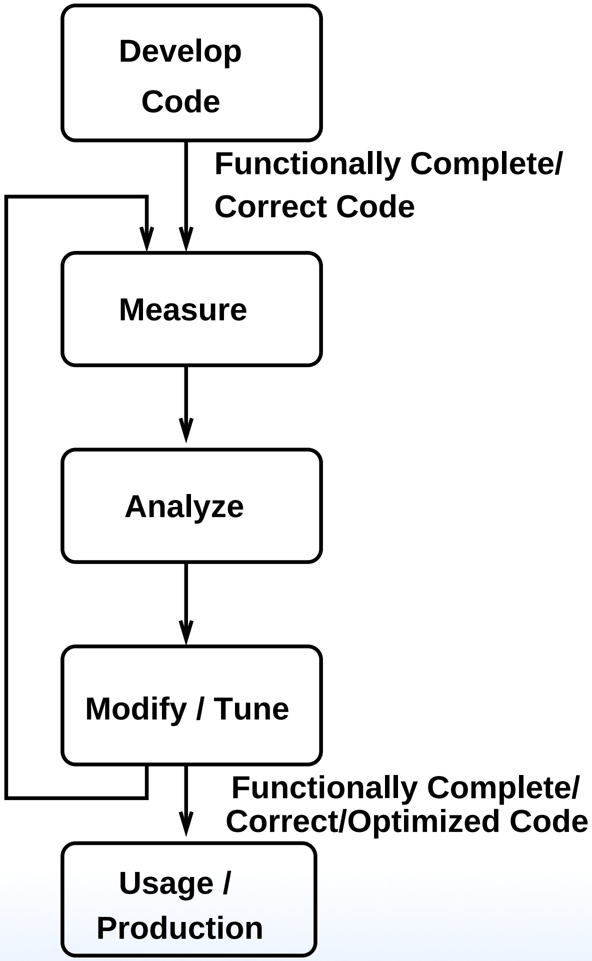
HPC environments are expensive:

- Procurement costs: $\sim$\$40 million
- Operational costs: $\sim$\$5 million/year
- Electricity costs: 1 MW / year $\sim$\$1 million
- Air Conditioning costs: ??

Know Your Limitation

- CPU Constrained
- Memory Constrained (Memory Wall)
- I/O Constrained
- Thermal Constrained
- Energy Constrained

Performance Optimization Cycle


Develop Code
Functionally Complete/ Correct Code
Measure
Analyze
Modify / Tune
Functionally Complete/ Correct/Optimized Code
Usage / Production

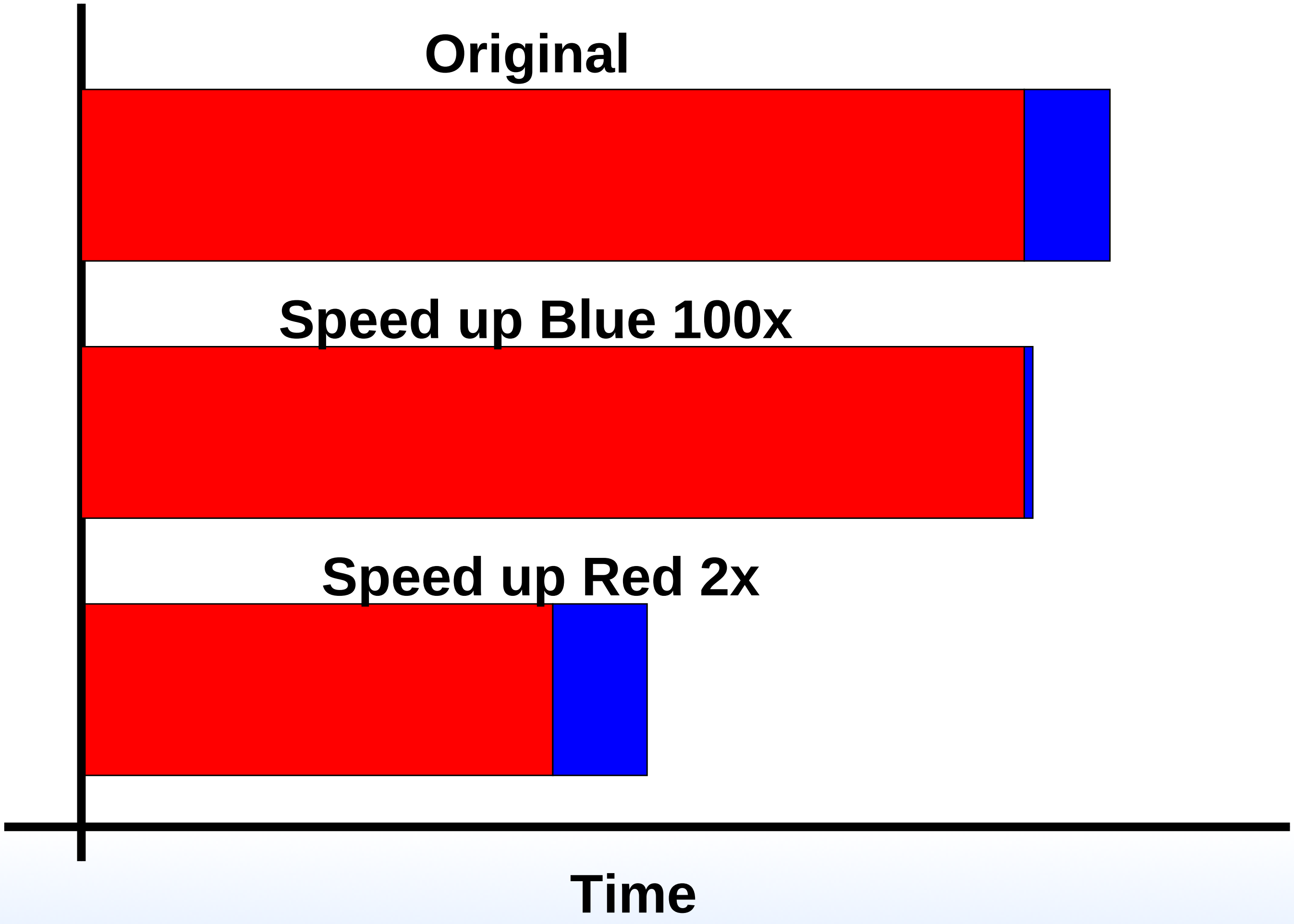
Wisdom from Knuth

“We should forget about small efficiencies, say about 97% of the time:

premature optimization is the root of all evil.

Yet we should not pass up our opportunities in that critical 3%. A good programmer will not be lulled into complacency by such reasoning, he will be wise to look carefully at the critical code; but only after that code has been identified” — Donald Knuth

Amdahl’s Law



Speedup

- Speedup is the improvement in latency (time to run)

$$S = \frac{t_{old}}{t_{new}}$$

So if originally took 10s, new took 5s, then speedup=2.

Scalability

- How a workload behaves as more processors are added
- Parallel efficiency: $E_p = \frac{S_p}{p} = \frac{T_1}{pT_p}$
- Linear scaling, ideal: $S_p = p$
- Super-linear scaling – possible but unusual

Strong vs Weak Scaling

- Strong Scaling –for fixed program size, how does adding more processors help

- Weak Scaling – how does adding processors help with the same per-processor workload

Strong Scaling

- Have a problem of a certain size, want it to get done faster.
- Ideally with problem size N, with 2 cores it runs twice as fast as with 1 core.
- Often processor bound; adding more processing helps, as communication doesn't dominate
- Hard to achieve for large number of nodes, as many

algorithms communication costs get larger the more nodes involved

Weak Scaling

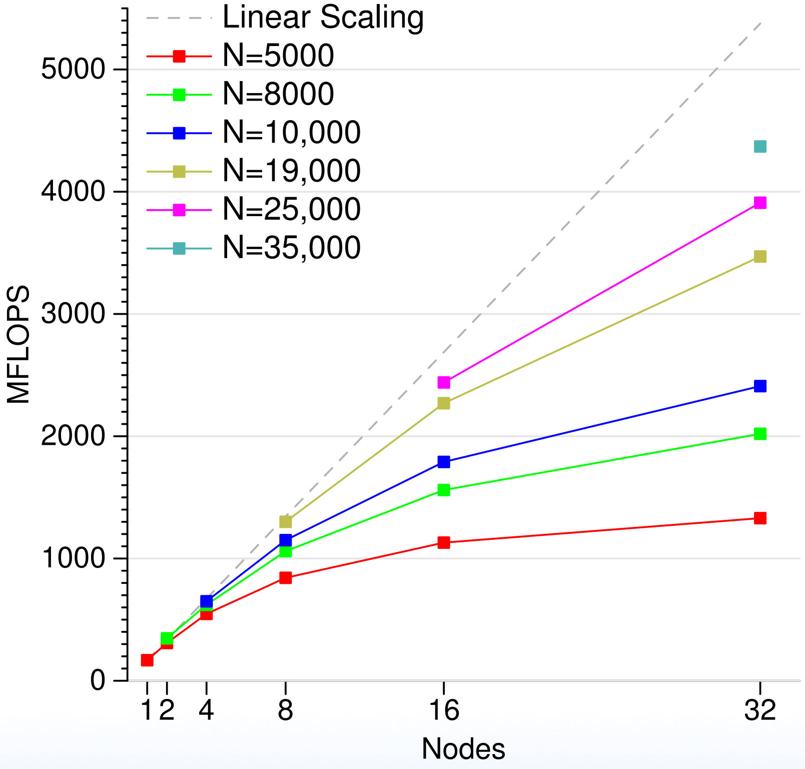
- Have a problem, want to increase problem size without slowing down.

- Ideally with problem size N with 1 core, a problem of size 2*n just as fast with 2 cores.

- Often memory or communication bound.

Scaling Example

LINPACK on Rasp-pi cluster. What kind of scaling is here?



Weak scaling. To get linear speedup need to increase problem size.
If it were strong scaling, the individual colored lines would increase rather than dropping off.