

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- Gustafson's Law
- quality up

MCS 572 Lecture 1
Introduction to Supercomputing
Jan Verschelde, 9 January 2012

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- Gustafson's Law
- quality up

Catalog Description

Introduction to supercomputing on vector and parallel processors; architectural comparisons, parallel algorithms, vectorization techniques, parallelization techniques, actual implementation on real machines.

Prerequisites:

- 1 a working knowledge of C/C++ (mcs 360),
- 2 familiarity with algorithms at the level of introductory numerical analysis (mcs 471).

MCS 572 is one of the courses on the computational science prelim.

Content of the Course

Two recommended text books:

Barry Wilkinson and Michael Allen: *Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers*. 2nd Edition. Prentice-Hall 2005.

David B. Kirk and Wen-mei W. Hwu: *Programming Massively Parallel Processors. A Hands-on Approach*. Elsevier 2010.

Parallel programming goals:

- 1 design and analysis of parallel programs;
- 2 implementation using MPI, OpenMP, and threads;
- 3 application to scientific problems.

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- Gustafson's Law
- quality up

Organization and Expectations

Three programming parts of the course:

- 1 using Message-Passing Interface (MPI) for clusters,
- 2 for shared memory: pthreads and OpenMP,
- 3 programming Graphics Processing Units (GPUs) using CUDA (Compute Unified Device Architecture) of NVIDIA.

Activities throughout the semester:

- several homework collections,
- midterm exam could be take home,
- three computer projects.

The first two computer projects will be on prescribed topics and may be solved in pairs. The third project must be done individually and could form the basis for a project presentation at the end.

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- Gustafson's Law
- quality up

what is a supercomputer?

Supercomputing = use of a supercomputer (also called high performance computing).

Definition

A *supercomputer* is a computing system (hardware, system & application software) that provides close to the best currently achievable sustained performance on demanding computational problems.

Classification at www.top500.org.

A *flop* is a floating point operation. Performance is often measured in number of flops per second.

If two flops can be done per clock cycle, then a processor at 3GHz can theoretically perform 6 billion flops (6 gigaflops) per second.

All computers in the top 10 achieve more than 1 petaflop per second.

top 10 of November 2011

Rank	Site	Computer/Year Vendor	Cores	R _{max}	R _{peak}	Power
1	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect / 2011 Fujitsu	705024	10510.00	11280.38	12659.9
2	National Supercomputing Center in Tianjin China	NUDT YH MPP, Xeon X5670 6C 2.93 GHz, NVIDIA 2050 / 2010 NUDT	186368	2566.00	4701.00	4040.0
3	DOE/SC/Oak Ridge National Laboratory United States	Cray XT5-HE Opteron 6-core 2.6 GHz / 2009 Cray Inc.	224162	1759.00	2331.00	6950.0
4	National Supercomputing Centre in Shenzhen (NSCS) China	Dawning TC3600 Blade System, Xeon X5650 6C 2.66GHz, Infiniband QDR, NVIDIA 2050 / 2010 Dawning	120640	1271.00	2984.30	2580.0
5	GSIC Center, Tokyo Institute of Technology Japan	HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows / 2010 NEC/HP	73278	1192.00	2287.63	1398.6
6	DOE/NNSA/LANL/SNL United States	Cray XE6, Opteron 6136 8C 2.40GHz, Custom / 2011 Cray Inc.	142272	1110.00	1365.81	3980.0
7	NASA/Ames Research Center/NAS United States	SGI Altix ICE 8200EX/8400EX, Xeon HT QC 3.0/Xeon 5570/5670 2.93 Ghz, Infiniband / 2011 SGI	111104	1088.00	1315.33	4102.0
8	DOE/SC/LBNL/NERSC United States	Cray XE6, Opteron 6172 12C 2.10GHz, Custom / 2010 Cray Inc.	153408	1054.00	1288.63	2910.0
9	Commissariat a l'Energie Atomique (CEA) France	Bull bulx super-node S6010/S6030 / 2010 Bull	138368	1050.00	1254.55	4590.0
10	DOE/NNSA/LANL United States	BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz, Voltaire Infiniband / 2009 IBM	122400	1042.00	1375.78	2345.0

system terms and architectures

core for a CPU: unit capable of executing a thread,
for a GPU: a streaming multiprocessor.

R_{\max} maximal performance achieved on the LINPACK
benchmark (solving a dense linear system) for problem
size N_{\max} , measured in Gflop/s.

R_{peak} theoretical peak performance measured in Gflop/s.

Power total power consumed by the system.

Types of architectures, using

- commodity leading edge microprocessors running at their maximal clock and power limits;
- special processor chips running at less than maximal power to achieve high physical packaging densities;
- mix of chip types and accelerators (GPUs).

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- **speedup and efficiency**
- Amdahl's Law
- Gustafson's Law
- quality up

speedup and efficiency

By p we denote the number of processors.

$$\text{Speedup } S(p) = \frac{\text{sequential execution time}}{\text{parallel execution time}}.$$

Another measure for parallel performance:

$$\text{Efficiency } E(p) = \frac{\text{speedup}}{\text{\#processors}} = \frac{S(p)}{p} \times 100\%.$$

In the best case, we hope: $S(p) = p$ and $E(p) = 100\%$.

If $E = 50\%$, then on average processors are idle for half of the time.

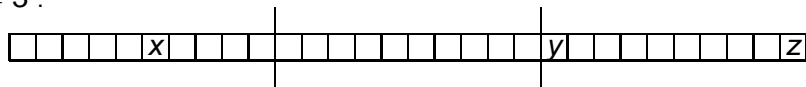
superlinear speedup

While we hope for $S(p) = p$, we may achieve $S(p) > p$.

Example. Sequential search in unsorted list.

A parallel search by p processors divides the list evenly in p sublists.

$p = 3$:



The sequential search time depends on position in list.

The parallel search time depends on position in sublist.

⇒ huge speedup if at first element of last sublist.

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

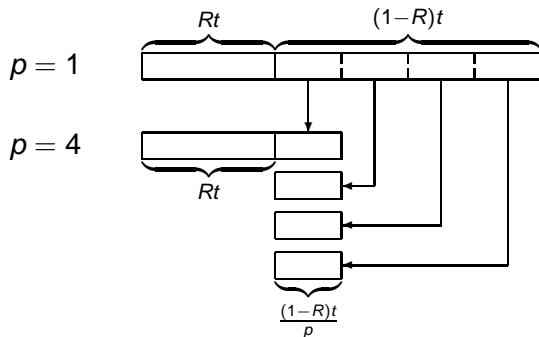
3 Measuring Performance

- speedup and efficiency
- **Amdahl's Law**
- Gustafson's Law
- quality up

predicting speedup

Consider a job that takes time t on one processor.

Let R be the fraction of t that must be done sequentially, $R \in [0, 1]$.



$$\text{Speedup on } p \text{ processors } S(p) \leq \frac{t}{Rt + \frac{(1-R)t}{p}} = \frac{1}{R + \frac{1-R}{p}} \leq \frac{1}{R}.$$

Amdahl's Law

Theorem (Amdahl (1967))

Let R be the fraction of the operations which cannot be done in parallel. The speedup with p processors is bounded by $\frac{1}{R + \frac{1-R}{p}}$.

Corollary. $S(p) \leq \frac{1}{R}$ as $p \rightarrow \infty$.

Example. Suppose 90% of the operations in an algorithm can be executed in parallel. What is the best speedup with 8 processors? What is the best speedup with an unlimited amount of processors?

$$p = 8: \frac{1}{\frac{1}{10} + (1 - \frac{1}{10}) \frac{1}{8}} = \frac{80}{17} \approx 4.7 \qquad p = \infty: \frac{1}{1/10} = 10$$

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

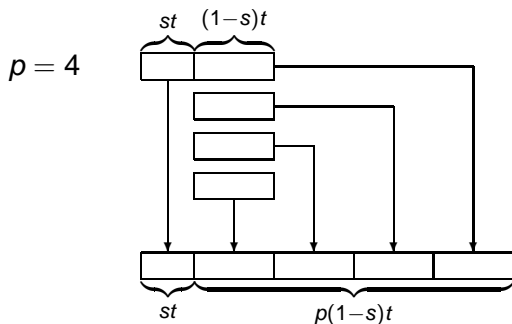
- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- **Gustafson's Law**
- quality up

scaled speedup

Consider a job that took time t on p processors.
Let s be the fraction of t that is done sequentially.



$$\text{Scaled speedup } S_s(p) \leq \frac{st + p(1-s)t}{t} = s + p(1-s) = p + (1-p)s.$$

Gustafson's Law

The problem size scales with the number of processors!

Theorem (Gustafson's Law (1988))

If s is the fraction of serial operations in a parallel program run on p processors, then the scaled speedup is bounded by $p + (1 - p)s$.

Example. Suppose benchmarking reveals that 5% of time on a 64-processor machine is spent on one single processor (e.g.: root node working while all other processors are idle). Compute the scaled speedup.

$$p = 64, s = 0.05: S_s(p) \leq 64 + (1 - 64)0.05 = 64 - 3.15 = 60.85.$$

Welcome to MCS 572

1 About the Course

- content and organization
- expectations of the course

2 Supercomputing

- definition and classification

3 Measuring Performance

- speedup and efficiency
- Amdahl's Law
- Gustafson's Law
- **quality up**

quality up

More processing power often leads to better results.

- finer granularity of a grid
e.g.: discretization of space and/or time in a differential equation
- greater confidence of estimates
e.g.: enlarged number of samples in a simulation
- compute with larger numbers (multiprecision arithmetic)
e.g.: solve an ill-conditioned linear system

$$\text{quality up } Q(p) = \frac{\text{quality on } p \text{ processors}}{\text{quality on 1 processor}}$$

$Q(p)$ measures improvement in quality using p procesors, keeping the computational time fixed.

summary and recommended reading

We defined supercomputing, speedup, and efficiency.
Gustafson's Law reevaluates Amdahl's Law.

Available to UIC via the ACM digital library:

- Jeannette M. Wing: **computational thinking.**
Communications of the ACM 49(3):33-35, 2006.
- Peter M. Kogge and Timothy J. Dysart: **Using the TOP500 to trace and project technology and architecture trends.**
In *SC'11 Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis.*
ACM 2011.
- John L. Gustafson: **Reevaluating Amdahl's Law.**
Communications of the ACM 31(5):532-533, 1988.

Exercises

Homework will be collected at a to be announced date.

Exercises:

- 1 How many processors whose clock speed runs at 3.0GHz does one need to build a supercomputer which achieves a theoretical peak performance of at least 4 Tera Flops? Justify your answer.
- 2 Suppose we have a program where 2% of the operations must be executed sequentially. According to Amdahl's law, what is the maximum speedup which can be achieved using 64 processors? Assuming we have an unlimited number of processors, what is the maximal speedup possible?
- 3 Benchmarking of a program running on a 64-processor machine shows that 2% of the operations are done sequentially, i.e.: that 2% of the time only one single processor is working while the rest is idle. Use Gustafson's law to compute the scaled speedup.