Forgotten Aspects of Computer Performance

John Gustafson
Ames Lab-ISU
Ames, Iowa

This work was supported by the Applied Mathematical Sciences Program of the USDOE, contract W-7405-ENG-82

SCL

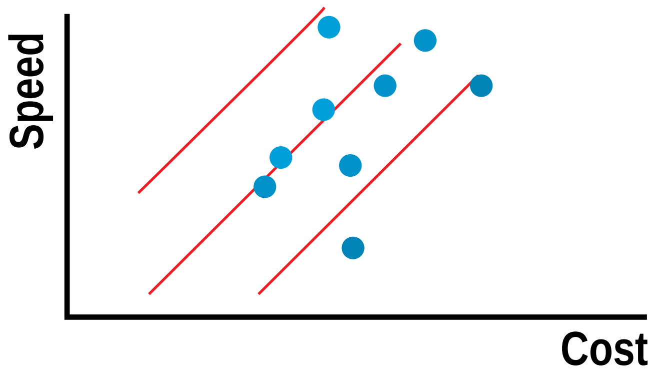
AMES LABORATORY

Thesis

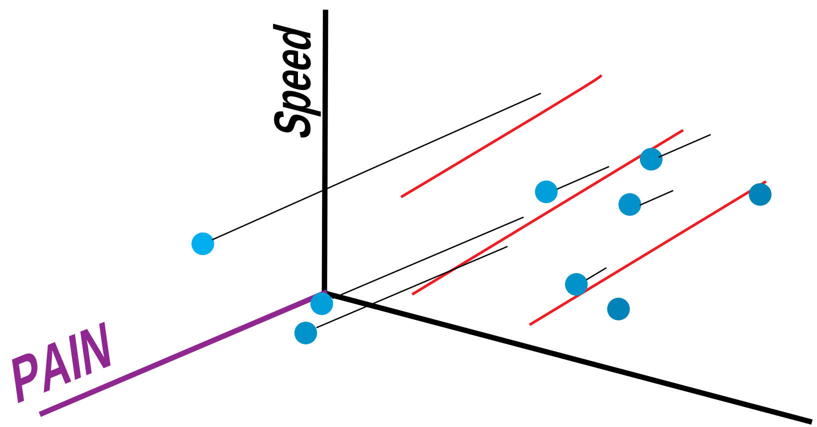
Our community sometimes equates "performance" with "teraflops" or "speedup," and forgets or takes for granted other aspects of performance.

It is time to reconsider those other aspects. If we don't, we could wind up with 100 teraflops and yet have low performance!

Typical Marketing Chart



What They Don't Tell You



High-performance computing cannot ignore conversion effort when comparing two approaches.

Ease-of-Use is
Part of Performance

What is *Speed*?

Sure, speed = work/time
But what is *work*, for a computer?
Higher speed isn't just time reduction.

From Ambrose Bierce, *The Devil's Dictionary* **:**

Logic, n. The art of thinking and reasoning in strict accordance with the limitations and incapacities of human misunderstanding. The basis of logic is the syllogism, consisting of a major and a minor premise and a conclusion—thus:

Major Premise : Sixty men can do a piece of work sixty times as quickly as one man.
Minor Premise : One man can dig a post-hole in sixty seconds.
Conclusion : Sixty men can dig a post-hole in one second.

This may be called the syllogism arithmetical, in which, by combining logic and mathematics, we obtain a double certainty, and are twice blessed.

Assessing Computational "Work"

```
      PARAMETER (N = 1000, N2 = N * N, N3 = N2 * N)
      REAL  A(N), B(N)
      INTEGER  L(N3), M(N3)

      DO 1 I = 1, N
1        A(I) = B(I) * B(I)

      DO 2 I = 1, N2
2        L(I) = N2 - I

      DO 3 I = 1, N3
3        M(I) = L(I)
      END
```

A thousand floating-point operations, or …

A million integer operations, or …

A billion memory references?

AMES LABORATORY

Can operations be standardized and counted?

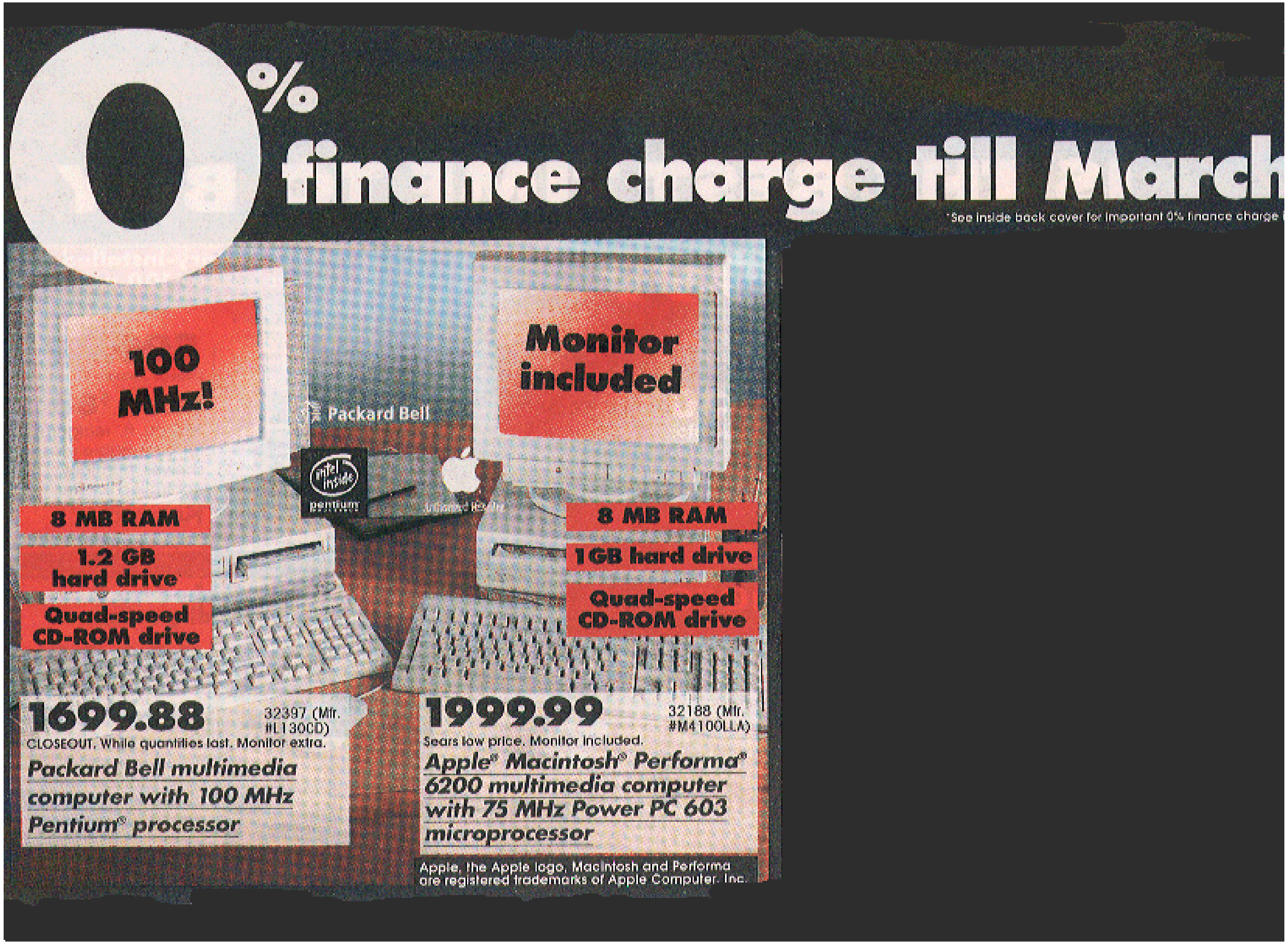
What does the "64-bit" arithmetic mean?
- *Mantissas range from 47 to 56 bits*
- *Even IEEE format doesn't prevent differences.*
- *How much work is, say* x ** y? *Or* ABS (x)?

Count fetches and stores as operations?
- *Is there error detection/correction?*
- *Out of what size address space?*
- *Word aligned? In cache? Pipelined? Interleaved? ...*

The work is not the "operation" count

SCL

AMES LABORATORY



SCL

AMES LABORATORY

The LINPACK Myth:

"If a computer has a higher rank on the LINPACK benchmark, it will run my application faster."

- LINPACK needs very little memory
- LINPACK needs little memory bandwidth
- LINPACK tolerates high latency
- Kernel: 67% Floating-Point Math, 33% Load/Store

Real application instruction frequencies:

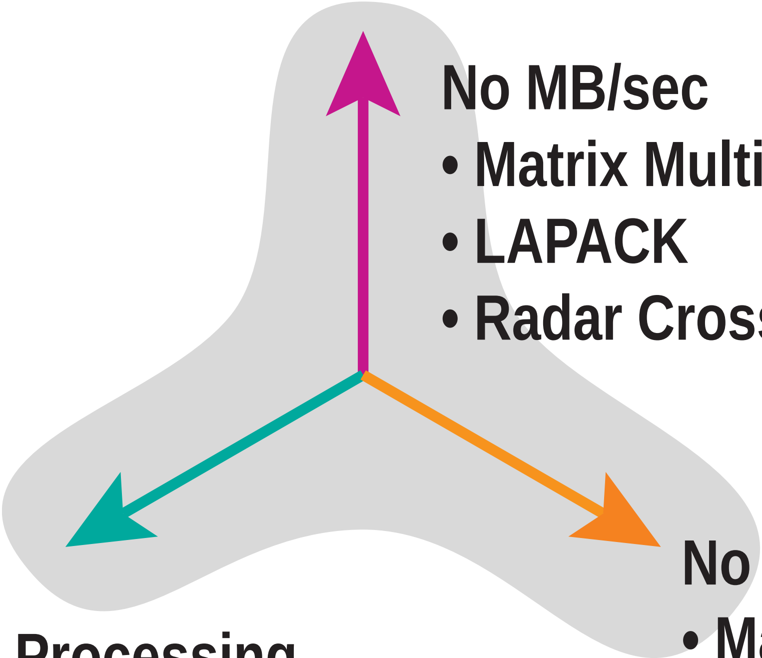
Load/Store	84%
Bit Manipulations	7%
Floating-Point Math	4%
Call/Return	3%
System Management	2%
Character Operations	<1%

Source:
"IBM RT PC Computer Technology,"
IBM Form No. SA23-1057, 1986

SCL

AMES LABORATORY

Things to Do With Unbalanced Systems

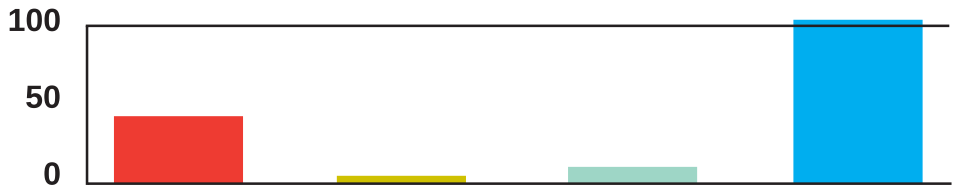


No MB/sec
- Matrix Multiply
- LAPACK
- Radar Cross-Section

No MFLOPS
- Sorting
- Transaction Processing
- Word Processing

No Communication
- Mandelbrot, Fractals
- Monte Carlo
- Ray Tracing

	TMC CM-2	Cray Y-MP/8	Intel iPSC/860	nCUBE nCUBE2
Application MFLOPS	1104	436	362	2605
"Peak" rated MFLOPS	2667	15000	7680	2409
Fraction of Peak	41%	3%	5%	106%



Peak FLOPS No Longer Predicts Actual FLOPS

Means-Based vs Ends-Based Metrics

MEANS-BASED		ENDS-BASED
Flop/s	⟷	Time to Compute Answer
Bytes of RAM	⟷	Detail, Content of Answer
Number of Processors	⟷	Feasible Problems to Attempt
Use of Commodity Parts	⟷	Cost, Availability of System
Word Size	⟷	Closeness to Actual Physics
ECC Memory	⟷	Reliability of Answer
Speedup	⟷	Product Line Range

Which Algorithm Would You Pick?

Explicit Timestepping	⟷	Implicit Timestepping
Conventional Matrix Multiply	⟷	Strassen, Winograd Methods
Cholesky Decomposition	⟷	PC Conjugate Gradient
All-to-All N-Body Methods	⟷	Barnes-Hut, Greengard
Successive Over-Relaxation	⟷	Multigrid
Time-Domain Operators	⟷	FFT's
Recompute Gaussian Integrals	⟷	Compute Once and Store
Material Property Function	⟷	Table Look-Up
HIGHER FLOP/S RATES		FASTER ANSWERS

SCL

AMES LABORATORY

Example: Dense vs Sparse Matrix Multiply

"Cray Food:
>80% Peak

```
      DO 1 K=1,1000
       DO 1 J=1,1000
        DO 1 I=1,1000
1        C(I,J)= C(I,J)+A(I,K)*B(K,J)
```

2.0 GFLOPS
1.0 Seconds

Ruins
Pipelining

```
      DO 2 K=1,1000
       DO 2 J=1,1000
        IF (B(K,J).NE.0.) THEN
         DO 1 I=1,1000
          IF (A(I,K).NE.0.)C(I,J)=…
1       END IF
2 CONTINUE
```

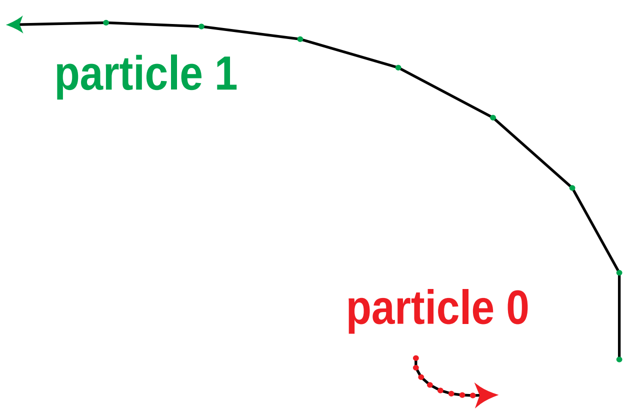
0.2 GFLOPS
0.1 Seconds

Ten Times Faster or Ten Times Slower?

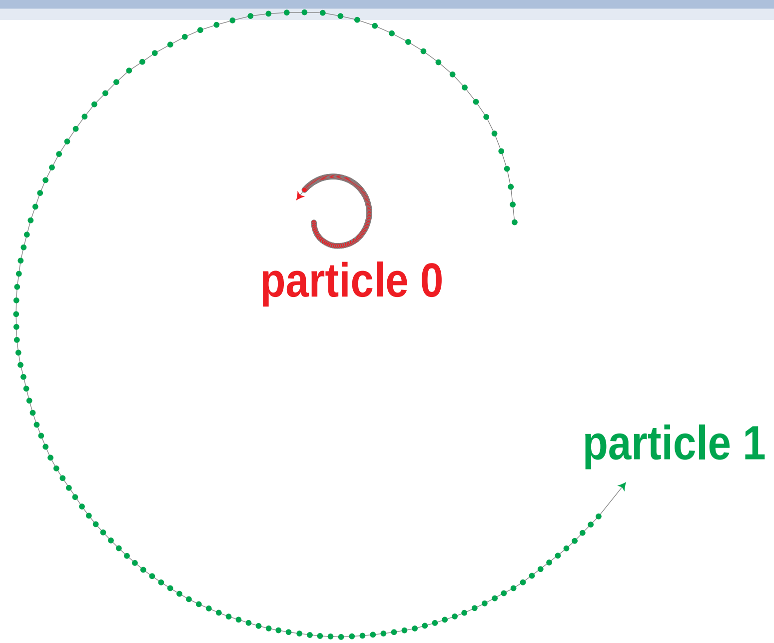
N-Body Challenges

- Planetary position was a Grand Challenge in the 1940s.
- Size of N often taken as figure of merit. “Billion-particle simulation.”
- GRAPE processor project uses all-to-all method, measures ops/sec.
- Materials science, astrophysics, and fusion all require N-body variants.
- Greengard, Barnes-Hut et al. made force calculation take O (Np^2) work

To double the physical accuracy of any N-body method appears to take at least four times as much work. This is not generally understood. Physical accuracy and N are not proportional.



Hmm… looks suspicious



Ah. Much better. Looks right.

The "Eyeball Metric" Only Works for Graphics

AMES LABORATORY

History of Concern for Validity

1940	Ignored. Just use lots of decimals.
1950	Monte Carlo debated; roundoff studied.
1960	Wilkinson proves validity of linear algebra.
1970	First 60-bit, 64-bit computer architectures.
1980	PASCAL-SC, ACRITH, ULTRITH.
1990	Ignored. Just use IEEE arithmetic.

Comparison with physical experiments is getting rarer.
Accuracy is neglected; speedup, FLOPS emphasized.
Different answers for parallel methods cause surprise.

SCL

AMES LABORATORY

Example: LINPACK Residuals

```
Value for n (maximum =  1160):
Please send the results of this run to:

Jack J. Dongarra
Computer Science Department
University of Tennessee
Knoxville, Tennessee 37996-1300

Fax: 615-974-8296

Internet: dongarra@cs.utk.edu

    norm. resid      resid           machep         x(1)          x(n)
 1.33497627E+01  2.96423996E-12  2.22044605E-16  1.00000000E+00  1.00000000E+00


   times are reported for matrices of order  1000
     factor     solve      total     mflops      unit      ratio
times for array with leading dimension of 580
 2.415E+01  1.844E+00  2.600E+01  2.572E+01  7.776E-02  4.642E+02
```

Which of these gets the most attention?

AMES LABORATORY

Measuring Answer Quality

If F is the answer, bound it rigorously by F^+ and F^-.

Define total error as

$$E = \int\int\int\int (F^+ - F^-)\, dx\, dy\, dz\, dt$$

and define the answer quality Q = 1/E.

This has several desirable consequences:

- Removes need for flops/second or instructions/second metrics
- Allows fair comparison of different algorithms and computer architectures
- Permits clear and rigorous statement of goal for Grand Challenges

SCL

AMES LABORATORY

One Approach: Integral Equations

One way to find F^+ and F^- is to restate the PDE as an integral equation, if possible. Integral equations of the Second Kind are usually tractable.

$$f(x) - \int K(x, s)\, f(s)\, ds = g(x)$$

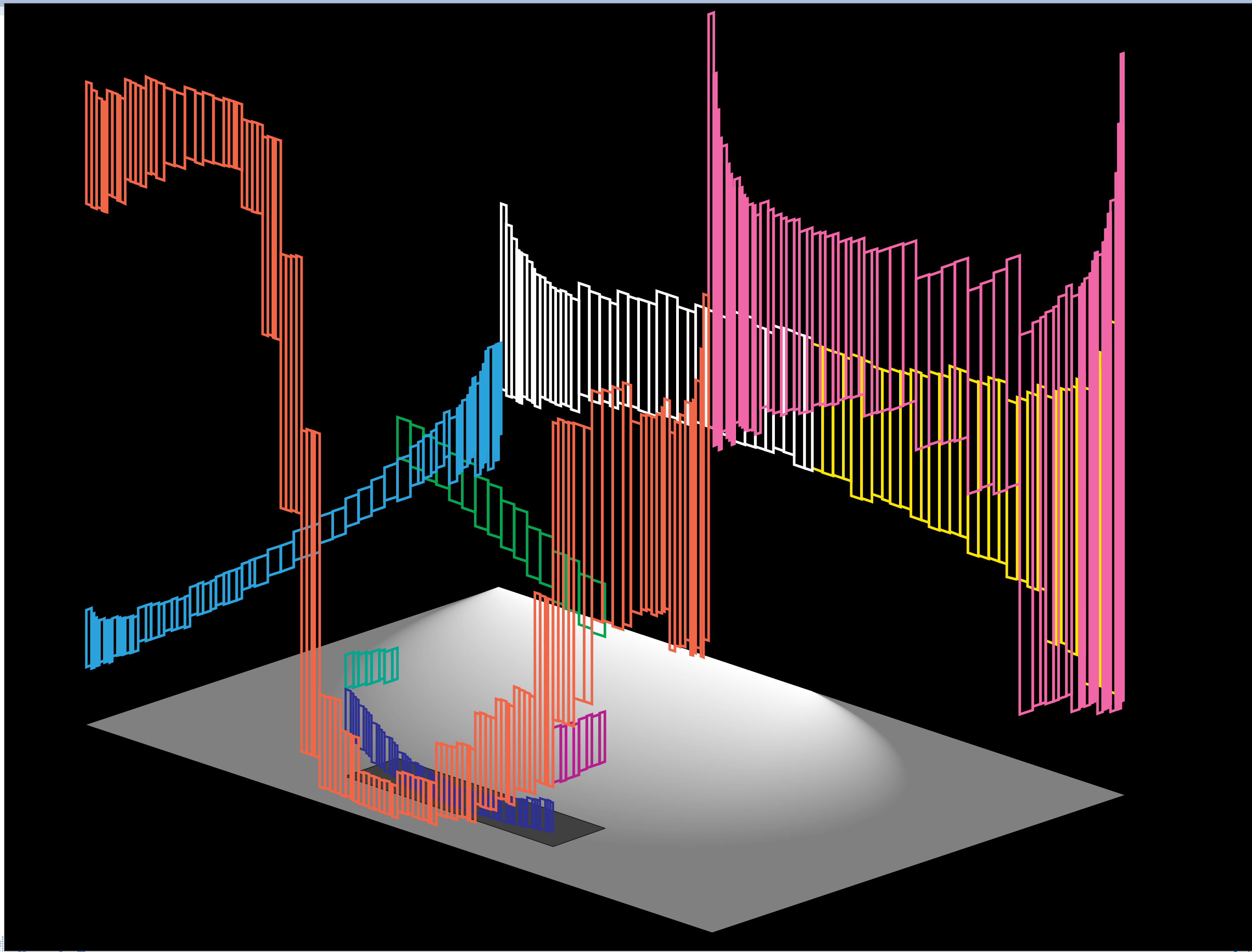
One can bound each variable on a discretization, and bound the integral.
Physical reasoning may be needed to get an initial bound.

We have found quality definitions and corresponding algorithms for

- Nonlinear ODEs
- Heat transfer problems
- The N-body problem
- Laplace's equation

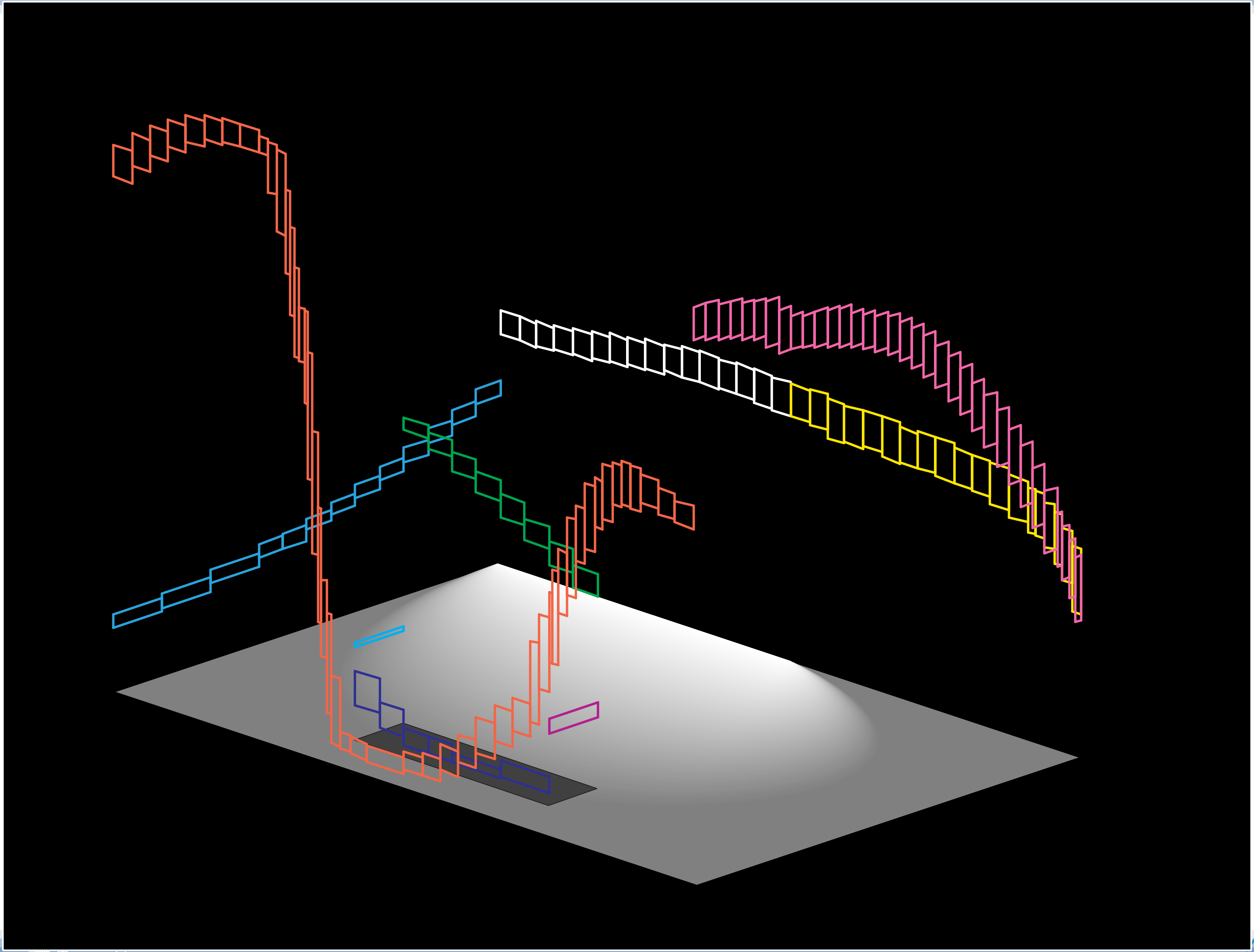
SCL

AMES LABORATORY



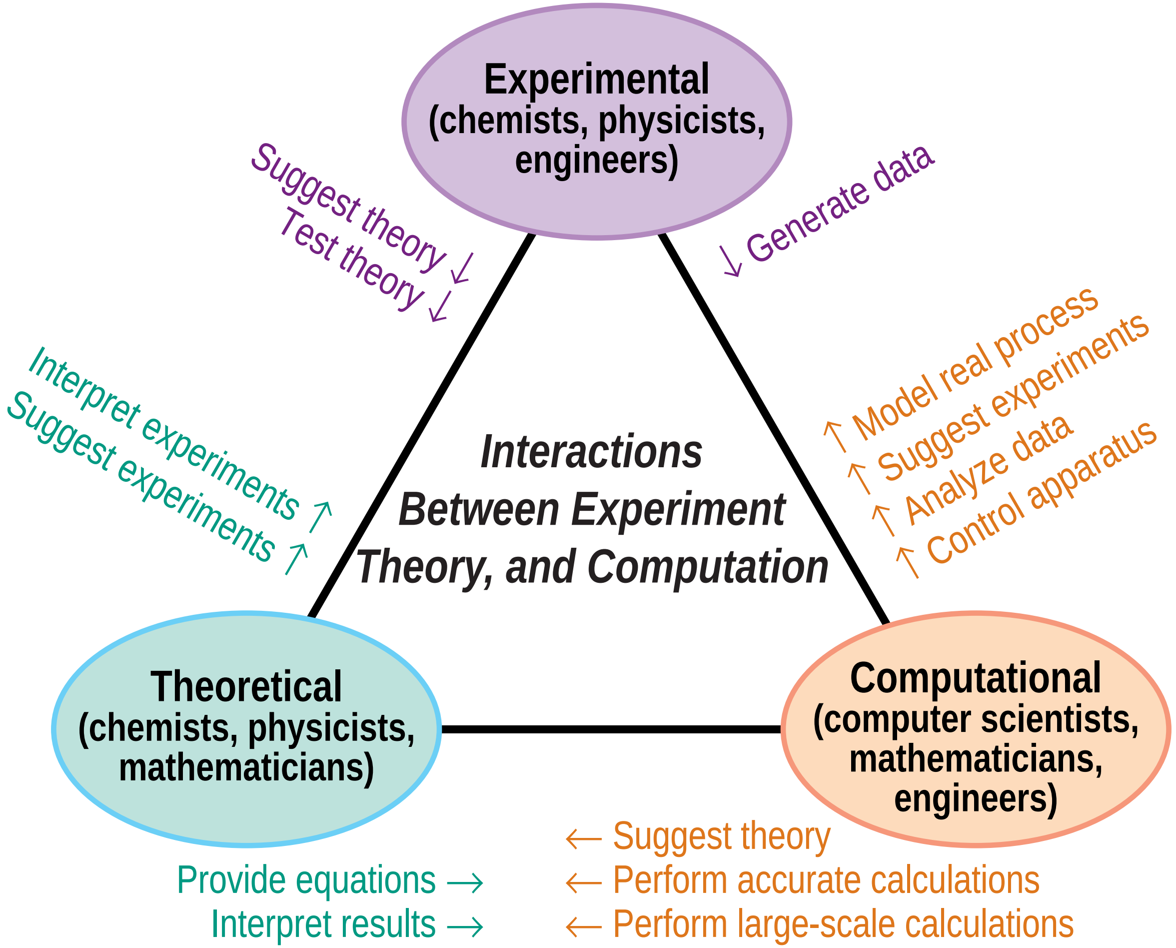
SCL

AMES LABORATORY



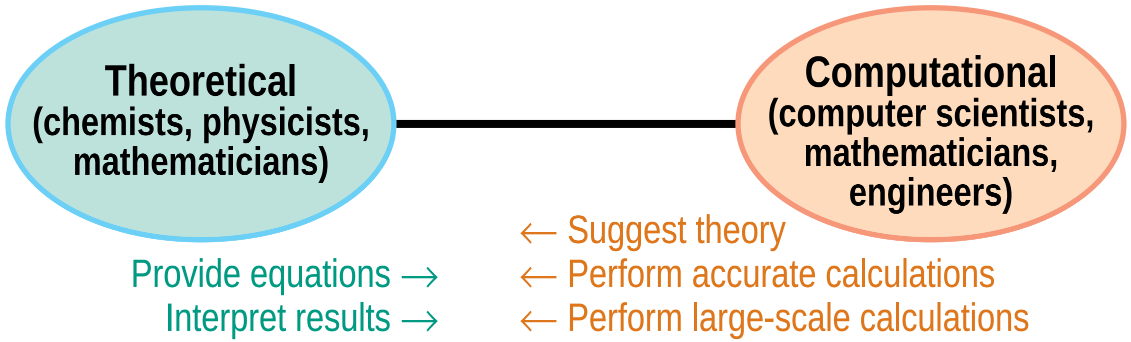
AMES LABORATORY

Computational Science as a Complement to Traditional Science



AMES LABORATORY

What Happens If There is No Experimental Component?



"More teraflops" isn't the answer.

ASCI Scenario 1:
(cost of about $100M)

Parallel computer rated at 3 TFLOPS sustains 1 TFLOPS, 70% parallel efficiency modeling nuclear weapon test.
No proof of correctness, no accuracy goal.

ASCI Scenario 2:
(cost of about $0.01M)

Computer rated at 0.0002 TFLOPS sustains unknown TFLOPS and efficiency modeling nuclear weapon test.
Answers have 95% confidence, match prior physical experiments.

If the Answer if Wrong, the Performance is ZERO.

Reliability Scenario 1:

Parallel supercomputer rated at 1 TFLOPS Has Mean Time Between Failure of 4 hours.

Mean Job Size < 10^{16} f.p. operations.

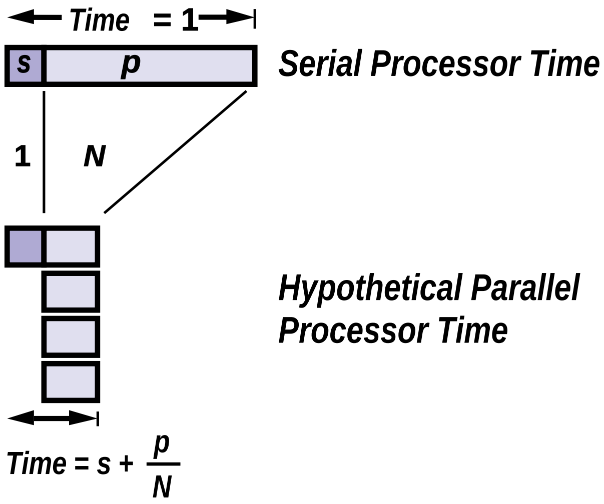
Reliability Scenario 2:

Parallel fault-tolerant sytem rated at 0.01 TFLOPS has MTBF of 10 months.

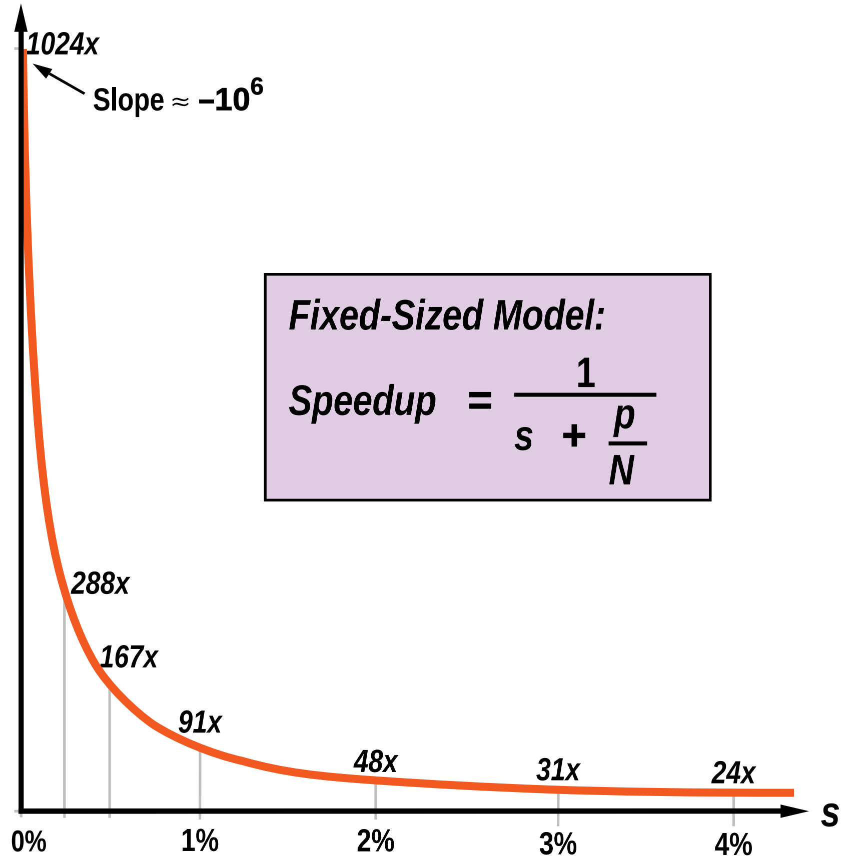
Mean Job Size <10^{17} f.p. operations.

If the Answer is Wrong, the Performance is ZERO.

Amdahl's Law, 10 Years Ago



Even with infinitely many processors, speedup can never exceed 1/s .

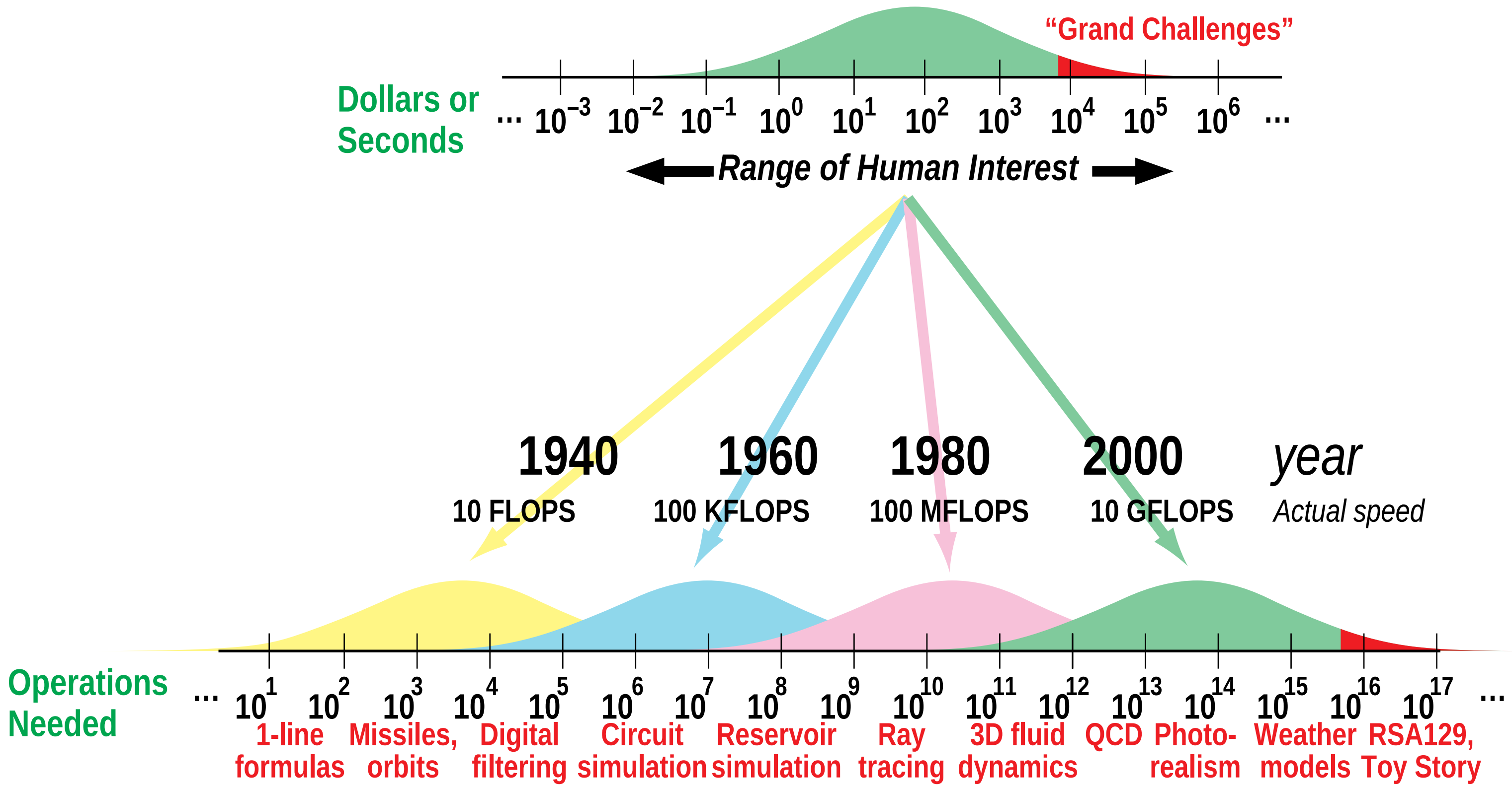


Fixed-Sized Model:

$$\text{Speedup} = \frac{1}{s + \frac{p}{N}}$$

Why I Hate Amdahl’s Law

- Problems grow as computer power increases; the time people are willing to wait for an answer is relatively constant.
- Uniprocessor problems use only 1/N of the memory on N distributed memory processors.
- Memory gets closer with more processors.
- Computers don’t double in cost when the number of processors doubles.
- The parallel array can be shared.



Human Factors Limit Problem Size.

SCI

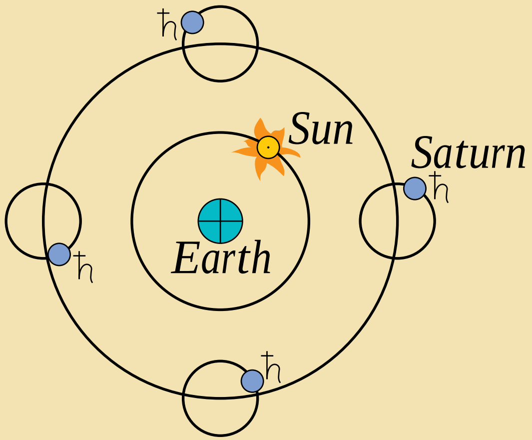
AMES LABORATORY

Financial Limits on Performance

SCL

AMES LABORATORY

"Ptolemaic" Speedup Theory



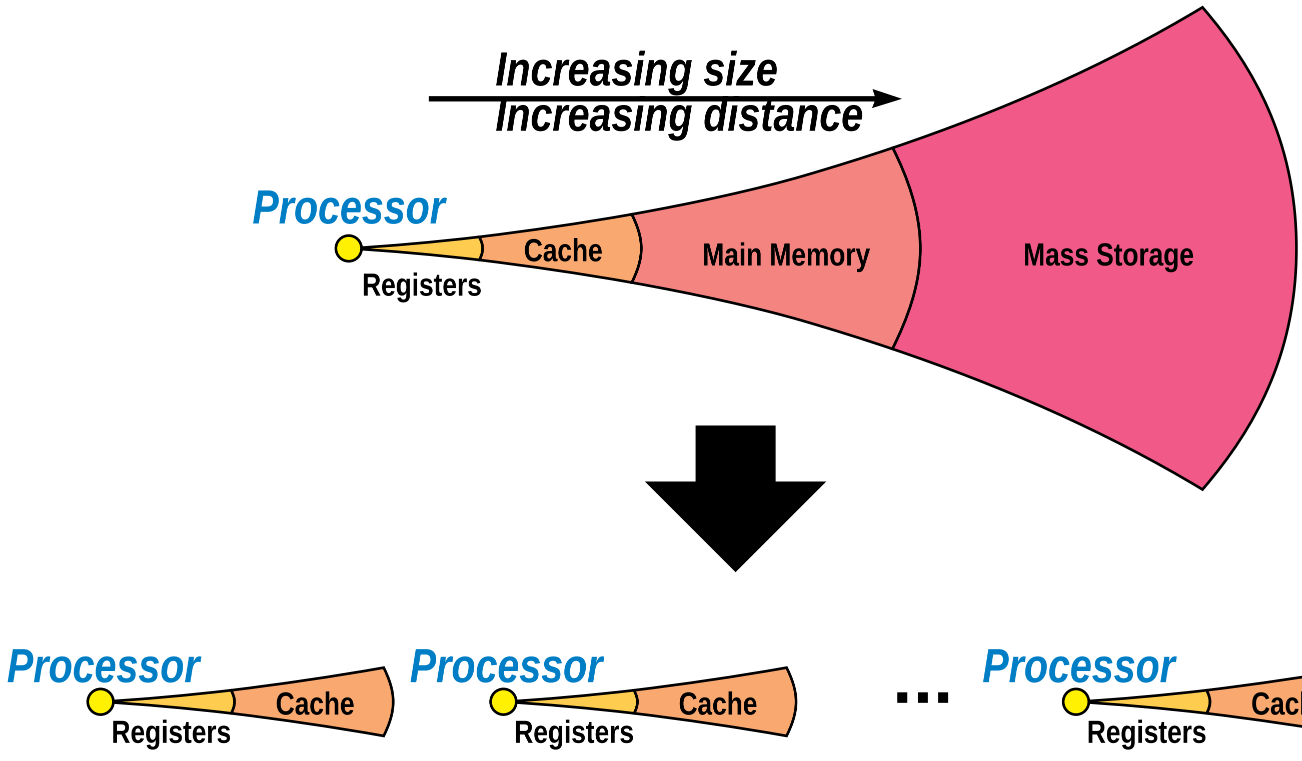
The retrograde motion of Saturn can be explained by building circles on circles, preserving the theory that the Earth is fixed at the center of the Universe.

$$\text{Speedup} = \frac{1}{\left(\frac{s'}{s'+p'N}\right) + \frac{\left(1 - \frac{s'}{s'+p'N}\right)}{N}}$$

$$= N + (1-N)s'$$

Scaled speedup can be explained by making s a function of N, preserving Amdahl's theory.

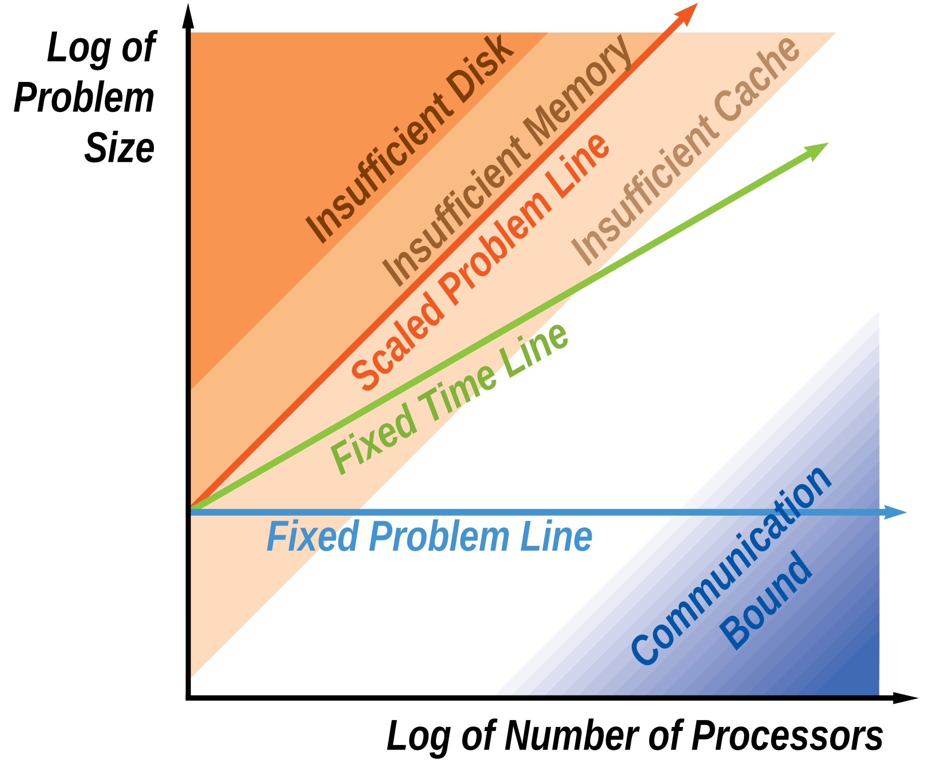
AMES LABORATORY



Memory is Never Flat.

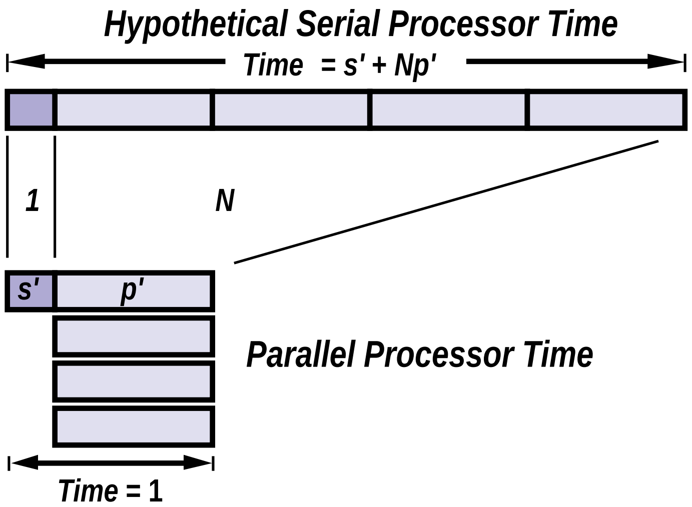
SCL

AMES LABORATORY

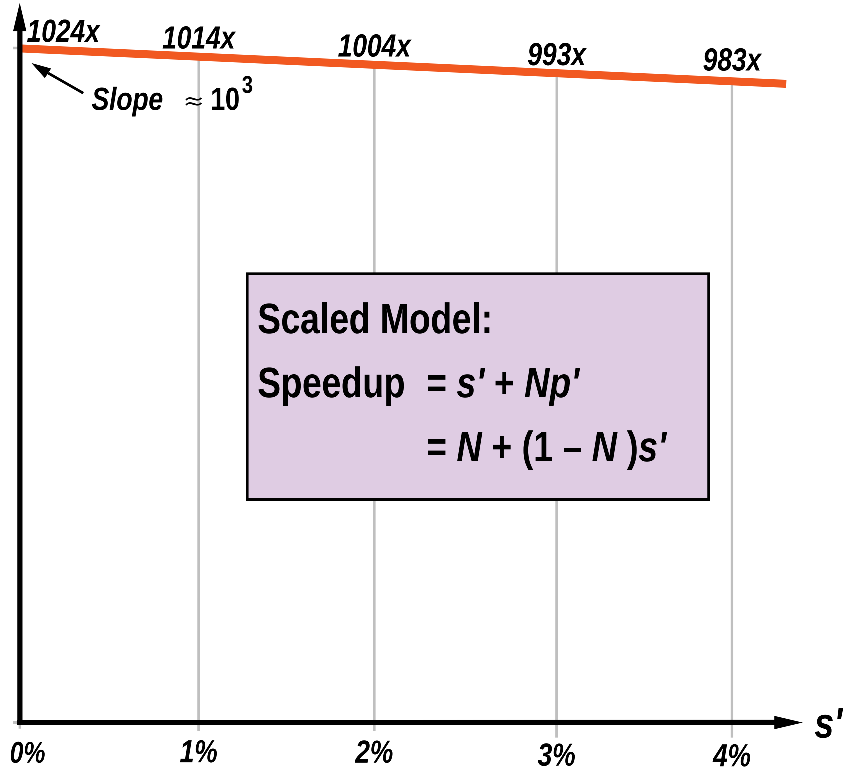


Speed gets used to Increase Problem Size, Not to Reduce Execution Time.

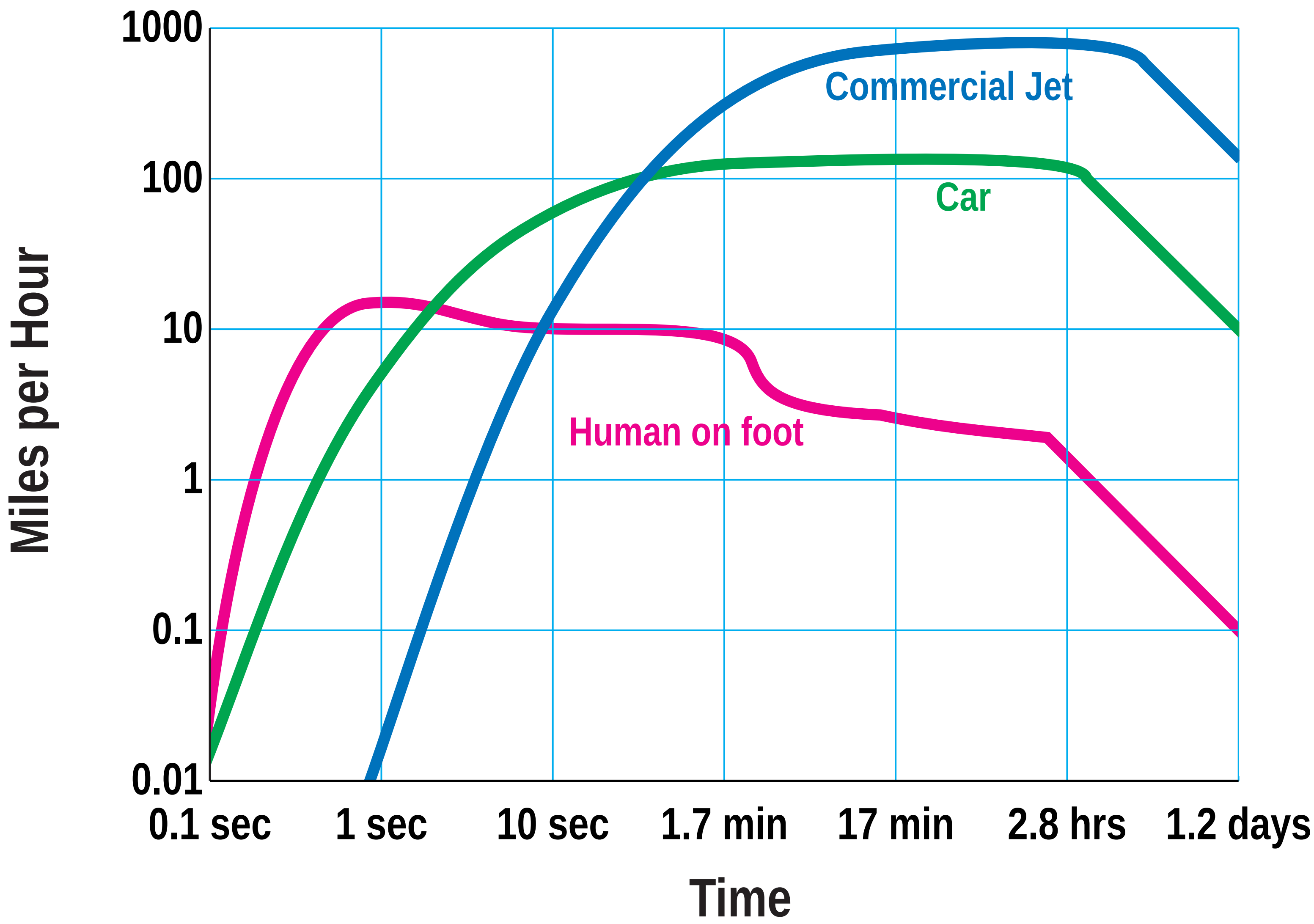
10 Years of “Scaled Speedup”



When the problem can scale
to match available power,
there is no limit to speedup.

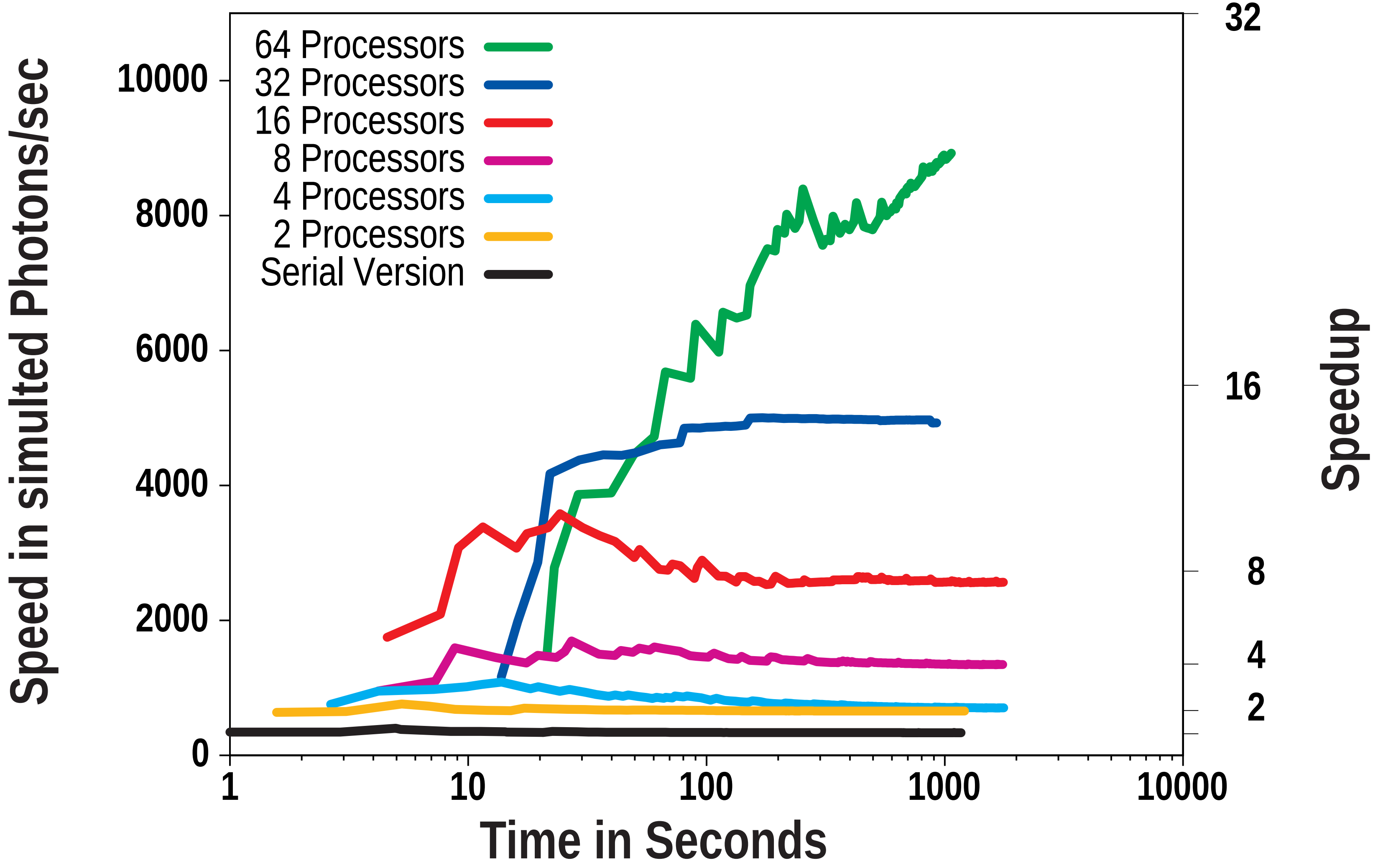


Visualizing “Speed”



AMES LABORATORY

A New Way to Graph “Speedup”



SCL

AMES LABORATORY

When Power and Problem Size Don’t Match

Problem: What is 7 times 13?

Computer	*Time to Answer*
Human	2 seconds
Pocket Calculator	30 seconds
RISC Workstation	5 minutes
Cray C90	4 hours

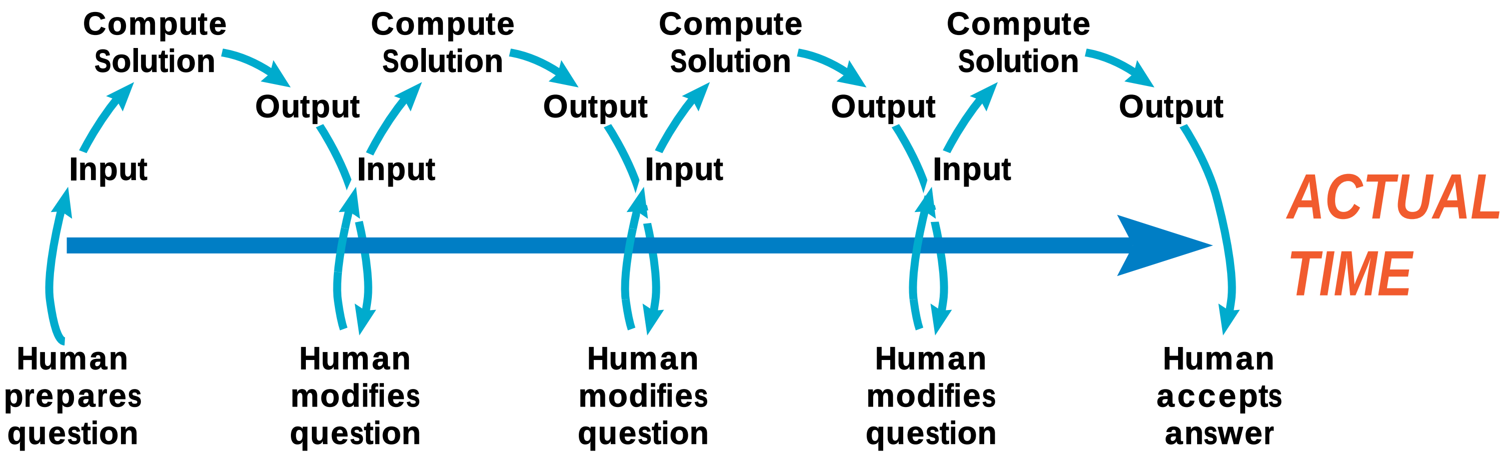
If you use too big a computer, you'll take more time. Not less.

Big Computers are for Big Problems.

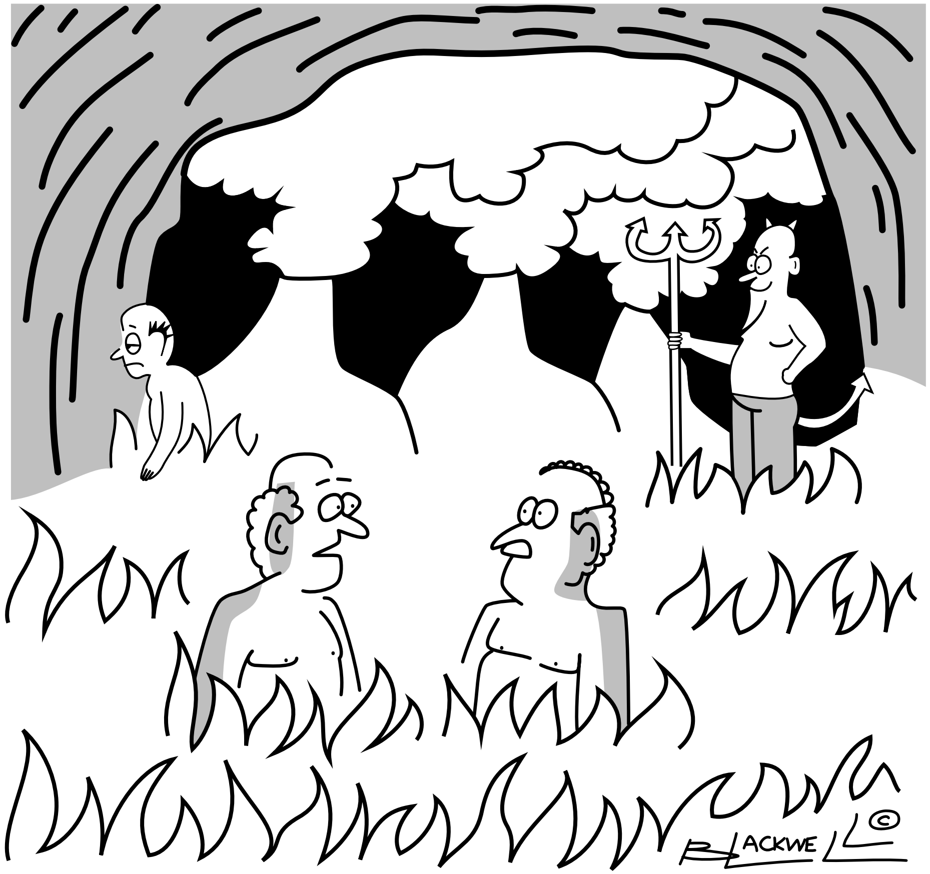
SCL

AMES LABORATORY

The Human Element in Performance Evaluation



Concluding Thought



"I was a computer performance measurement specialist. What are you here for?"

SCL

AMES LABORATORY