ATPESC 2023

ARGONNE TRAINING PROGRAM ON EXTREME-SCALE COMPUTING

Growing Up at Argonne National Laboratory

Jack Dongarra

University of Tennessee

Oak Ridge National Laboratory

University of Manchester

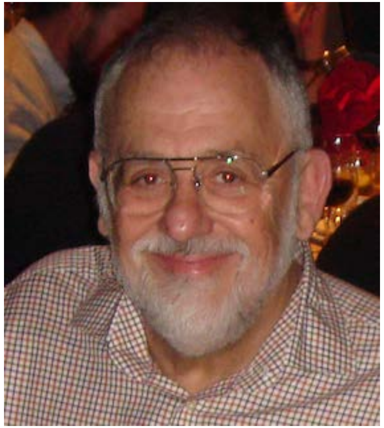
I wanted to be a science high school teacher

- Enrolled as an undergraduate at a college for teachers for the Chicago public school system
- My last semester in college my physics professor encouraged me to apply to a program to spend a semester at Argonne working with a scientist.

Brian Smith

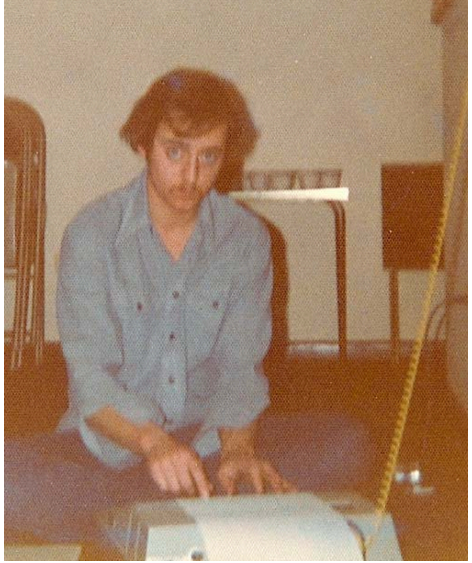
Worked on a software project called EISPACK.

Many visitors from various universities.



Cleve Moler, U of New Mexico

Late 70’s - New Mexico Days



- Encouraged to pursue PhD by many visitors.
- Cleve said he would customize a degree program at the U of New Mexico in the Math Department.
- I was detailed from Argonne to work at Los Alamos.
- Spent one semester at UNM@LANL, then 2 semesters on the UNM campus.
- Cleve was at Stanford on Sabbatical during my last year at UNM.
    - The plan was to finish my courses & exams and then join Cleve at Stanford.
- On to Stanford and Serra House.
- Then back to ANL and to finish my dissertation

1970s HPC Systems

CDC 7600 36.4 MHz (27.5 ns clock cycle)

- Primary memory 65 Kwords (60-bit words)
- Seymour Cray design
- Peak 36 Mflop/s
- Broke down at least once/day (often four or five times)

IBM 370/195 18.5 MHz (54 ns clock cycle)

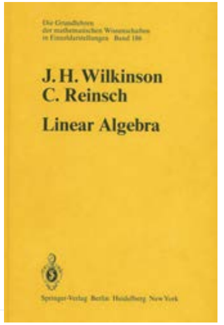
- High degree of parallelism
- Up to 7 operations at a time
- Up to 4 MB of memory

Both systems had a high degree of instruction-level pipelining and parallelism.

Over the Past 50 Years Evolving SW and Alg Tracking Hardware Developments

Features: Performance, Portability, and Accuracy		
EISPACK (1970's) (Translation of Algol to F66)	 	Rely on - Fortran, but row oriented

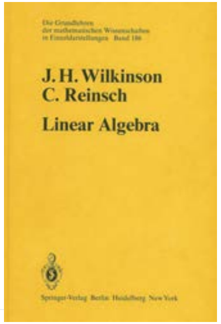
- **EISPACK** is a software library for numerical computation of eigenvalues and eigenvectors of matrices,
  - Written in FORTRAN.
  - Contains subroutines for calculating the eigenvalues of nine classes of matrices:
    - complex general, complex Hermitian, real general, real symmetric, real symmetric banded,
    - real symmetric tridiagonal, special real tridiagonal, generalized real, and
    - generalized real symmetric matrices.
  - The library drew heavily on Algol algorithms developed by Jim Wilkinson & colleagues.



Over the Past 50 Years Evolving SW and Alg Tracking Hardware Developments

Features: Performance, Portability, and Accuracy		
EISPACK (1970's) (Translation of Algol to F66)	 	Rely on - Fortran, but row oriented
Level 1 Basic Linear Algebra Subprograms (BLAS)		Standards for: Vector-Vector operations

- **EISPACK** is a software library for numerical computation of eigenvalues and eigenvectors of matrices,
  - Written in FORTRAN.
  - Contains subroutines for calculating the eigenvalues of nine classes of matrices:
    - complex general, complex Hermitian, real general, real symmetric, real symmetric banded,
    - real symmetric tridiagonal, special real tridiagonal, generalized real, and
    - generalized real symmetric matrices.
  - The library drew heavily on Algol algorithms developed by Jim Wilkinson & colleagues.



My First Paper

SOFTWARE—PRACTICE AND EXPERIENCE, VOL. 9, 219–226 (1979)

Unrolling Loops in FORTRAN*

J. J. DONGARRA AND A. R. HINDS
Argonne National Laboratory, Argonne, Illinois 60439, U.S.A.

SUMMARY

The technique of 'unrolling' to improve the performance of short program loops resorting to assembly language coding is discussed. A comparison of the benefit 'unrolling' on a variety of computers using an assortment of FORTRAN com presented.

- Reduces loop overhead
  - Level of unrolling dedicated by the instruction stack size
- Help the compiler to:
  - Facilitates pipelining
  - Increases the concurrence between independent functional units
- Provided ~15% performance improvement

TECHNIQUE

When a loop is unrolled, its contents are replicated one or more times, with appropriate adjustments to array indices and the loop increment. For instance, the DAXPY[9] sequence, which adds a multiple of one vector to a second vector:

```
      DO 10 I = 1,N
        Y(I) = Y(I)+A*X(I)
   10 CONTINUE
```

would, unrolled to a depth of four, assume the form

```
      M = N-MOD(N,4)
      DO 10 I = 1,M,4
        Y(I)   = Y(I)   +A*X(I)
        Y(I+1) = Y(I+1)+A*X(I+1)
        Y(I+2) = Y(I+2)+A*X(I+2)
        Y(I+3) = Y(I+3)+A*X(I+3)
   10 CONTINUE
```

Basic Linear Algebra Subprograms for Fortran Usage

C. L. LAWSON
Jet Propulsion Laboratory
R. J. HANSON
Sandia Laboratories
D. R. KINCAID
The University of Texas at Austin
and
F. T. KROGH
Jet Propulsion Laboratory

A package of 38 low level subprograms for many of the basic operations of numerical linear algebra is presented. The package is intended to be used with Fortran. The operations in the package include dot product, elementary vector operation, Givens transformation, vector copy and swap, vector norm,

MCS Division c. 1983

Originally called the Applied Mathematics Division until 1982

Back row: Jim Boyle (w/picture of Larry Wos), John Gabriel, Ken Dritz, Joe Cook, Bob Veroff, Hans Kaper, Paul Messina, Bernie Matkowsky, Jim Cody, James Lyness, Wayne Cowell, Burt Garbow, Ken Hillstrom, Brian Smith, LuAnn Phebus
Seated: JD, Rusty Lusk, Mike Minkoff, Gary Leaf, Jorge More', Danny Sorensen, Bruce Char, Doris Pool, Judy Beumer

The group had a culture of friendship

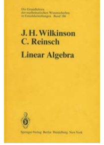
Mathematics and Computer Science Division in 1983

- Linear Algebra
  - EISPACK, LINPACK, BLAS
- Optimization
  - MINPACK
- Special Functions
  - FUNPACK
- Numerical Solution to PDEs
  - Fluid Flow
  - Sturm Liouville operators
  - Bifurcation Phenomena
- Quadrature
- IEEE Floating Point Arithmetic
- Theorem Proving
  - Aura
- Symbolic Computation
- Parallel Programming Methodologies & Tools
  - Monitors/macros
- Program Languages
- Program Development Aids and Automatic Transformations
  - TAMPR
- Fortran Standards Committee

Building things that worked

Things like P4, PVM, MPI, MPICH where just a glimmer in our eyes at this stage.

Over the Past 50 Years Evolving SW and Alg Tracking Hardware Developments



Features: Performance, Portability, and Accuracy		
EISPACK (1970's) (Translation of Algol to F66)	 	Rely on - Fortran, but row oriented
Level 1 Basic Linear Algebra Subprograms (BLAS)		Standards for: Vector-Vector operations
LINPACK (1980's) (Vector operations)	 	Rely on - Level-1 BLAS operations - Column oriented

1984 -1992

Argonne’s Parallel Menagerie

Rusty Lusk and I were the Directors of the ARCF

ACTIVITIES AND OPERATIONS OF THE

ADVANCED COMPUTING RESEARCH FACILITY

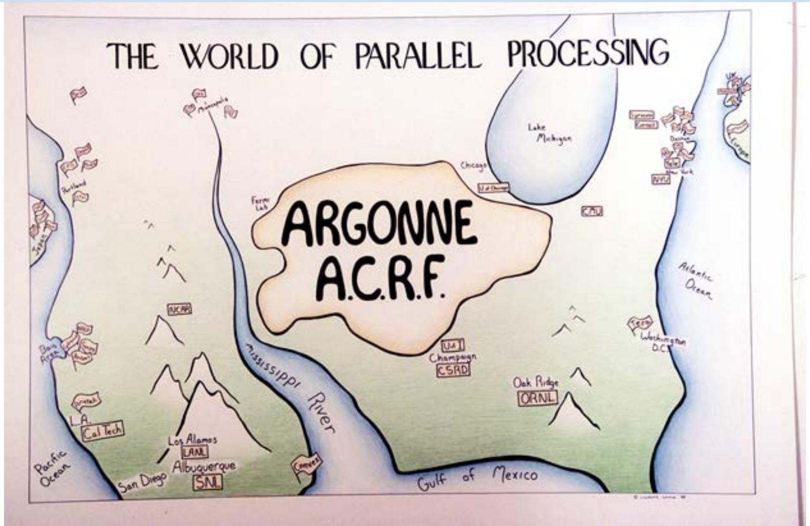
ACRF

r the U. S. DEPARTMENT OF EN

under Contract W-31-109-Eng-

Several radically different parallel architectures, from shared to distributed memory; from vector to dataflow to bit parallel processors

- Thinking Machines CM-2, w/16,384 procs.
- Active Memory Technology DAP-510, w/1024 procs.
- BBN TC 2000 (Butterfly II), w/32 procs.
- Cydrom Cydra 5, VLIW architecture
- Denelcor HEP, w/4 PEMs
- Intel iPSC/d5 hypercube w/32 procs.
- Sequent Balance 21000, w/24 procs.
- Encore Multimax, w/20 procs.
- Intel iPSC/d4 hypercube, w/16 vector procs.
- Alliant FX/8, w/8 vector procs.
- Ardent Titan graphics supercomputer, w/4 vector procs.



An Accidental Benchmarker

LINPACK was an NSF Project w/ ANL, UNM, UM, & UCSD
We worked independently and came to Argonne in the summers

Top 23 List from 1977
Performance of solving *Ax=b* using LINPACK software

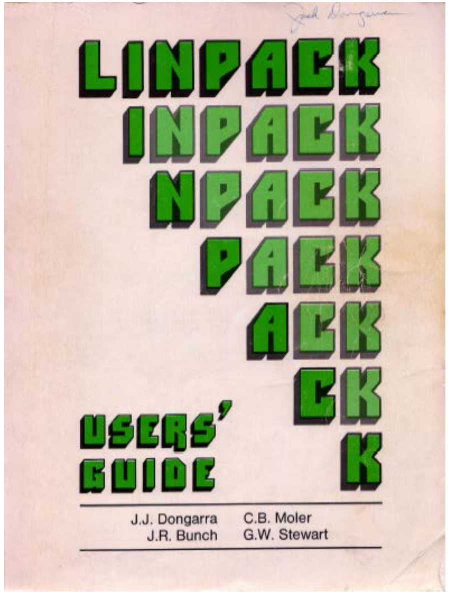
$\frac{2}{3}n^3 + 2n^2$ ops / time

UNIT = 10**6 TIME/(1/3 100**3 + 100**2)

Facility		TIME N=100 secs.	UNIT micro-secs.	Computer	Type	Compiler
NCAR	14.0	.049	0.14	CRAY-1	S	CFT, Assembly BLAS
LASL	4.64	.148	0.43	CDC 7600	S	FTN, Assembly BLAS
NCAR	3.58	.192	0.56	CRAY-1	S	CFT
LASL	3.27	.210	0.61	CDC 7600	S	FTN
Argonne	2.31	.297	0.86	IBM 370/195	D	H
NCAR	1.91	.359	1.05	CDC 7600	S	Local
Argonne	1.77	.388	1.33	IBM 3033	D	H
NASA Langley	1.40	.489	1.42	CDC Cyber 175	S	FTN
U. Ill. Urbana	1.36	.506	1.47	CDC Cyber 175	S	Ext. 4.6
LLL	1.24	.554	1.61	CDC 7600	S	CHAT, No optimize
SLAC	1.19	.579	1.69	IBM 370/168	D	H Ext., Fast mult.
Michigan	1.09	.631	1.84	Amdahl 470/V6	D	H
Toronto	.772	.890	2.59	IBM 370/165	D	H Ext., Fast mult.
Northwestern	.477	1.44	4.20	CDC 6600	S	FTN
Texas	.356	1.93*	5.63	CDC 6600	S	RUN
China Lake	.352	1.95*	5.69	Univac 1110	S	V
Yale	.265	2.59	7.53	DEC KL-20	S	F20
Bell Labs	.197	3.46	10.1	Honeywell 6080	S	Y
Wisconsin	.187	3.49	10.1	Univac 1110	S	V
Iowa State	.194	3.54	10.2	Itel AS/5 mod3	D	H
U. Ill. Chicago	[illegible]	4.10	11.9	IBM 370/158	D	G1

Appendix B of the Linpack Users' Guide
- Designed to help users estimate the run time for solving systems of equation using the Linpack software.

First benchmark report from 1977;
- Cray 1 to DEC PDP-10



Performance of Various Computers Using Standard Linear Equations Software in a Fortran Environment

Jack J. Dongarra

Mathematics and Computer Science Division
Argonne National Laboratory
Argonne, Illinois 60439

October 24, 1983

Abstract - In this note we compare a number of different computer systems for solving dense systems of linear equations using the LINPACK software in a Fortran environment. There are about 50 computers compared, ranging from a Cray X-MP to the 68000 based systems such as the Apollo and SUN Workstations.

The timing information presented here should in no way be used to judge the overall performance of a computer system. The results reflect only one problem area: solving dense systems of equations using the LINPACK [1] programs in a Fortran environment.

The LINPACK programs can be characterized as having a high percentage of floating point arithmetic operations. The routines involved in this timing study, SGEFA and SGESL, use algorithms which are column oriented. By column orientation we mean the programs usually references array elements sequentially down a column, not across a row. Column orientation is important in increasing efficiency in a Fortran environment because of the way in which arrays are stored. Most of the floating point operations in LINPACK actually take place in a set of subprograms called the Basic Linear Algebra Subprograms (BLAS) [2]. These routines are called by the LINPACK routines repeatedly throughout the calculation. The BLAS reference one-dimension arrays, rather than two-dimensional arrays.

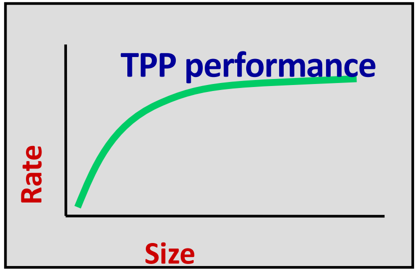
Note that these numbers are for a problem of order 100. The execution speeds on some machines, particularly the vector computers, may not have reached their asymptotic rates or the algorithms used may not fully utilize the features of certain machines. (See the appendix for a specific comparison of large scientific computers in Fortran which better reflects their performance.)

The table was compiled over a period of time. Subsequent software and hardware changes to a computer system may affect the timing to some extent.

Top500 Since 1993

- Since 1978 I maintained a LINPACK Benchmark list.
- Hans Meuer and Erich Strohmaier had a list of fastest computers ranked by peak performance.
- Listing of the 500 most powerful computers in the World.
- Yardstick: Performance for

Ax=b, dense problem



Maintained and updated twice a year:

SC'xy in the States in November

Meeting in Germany in June

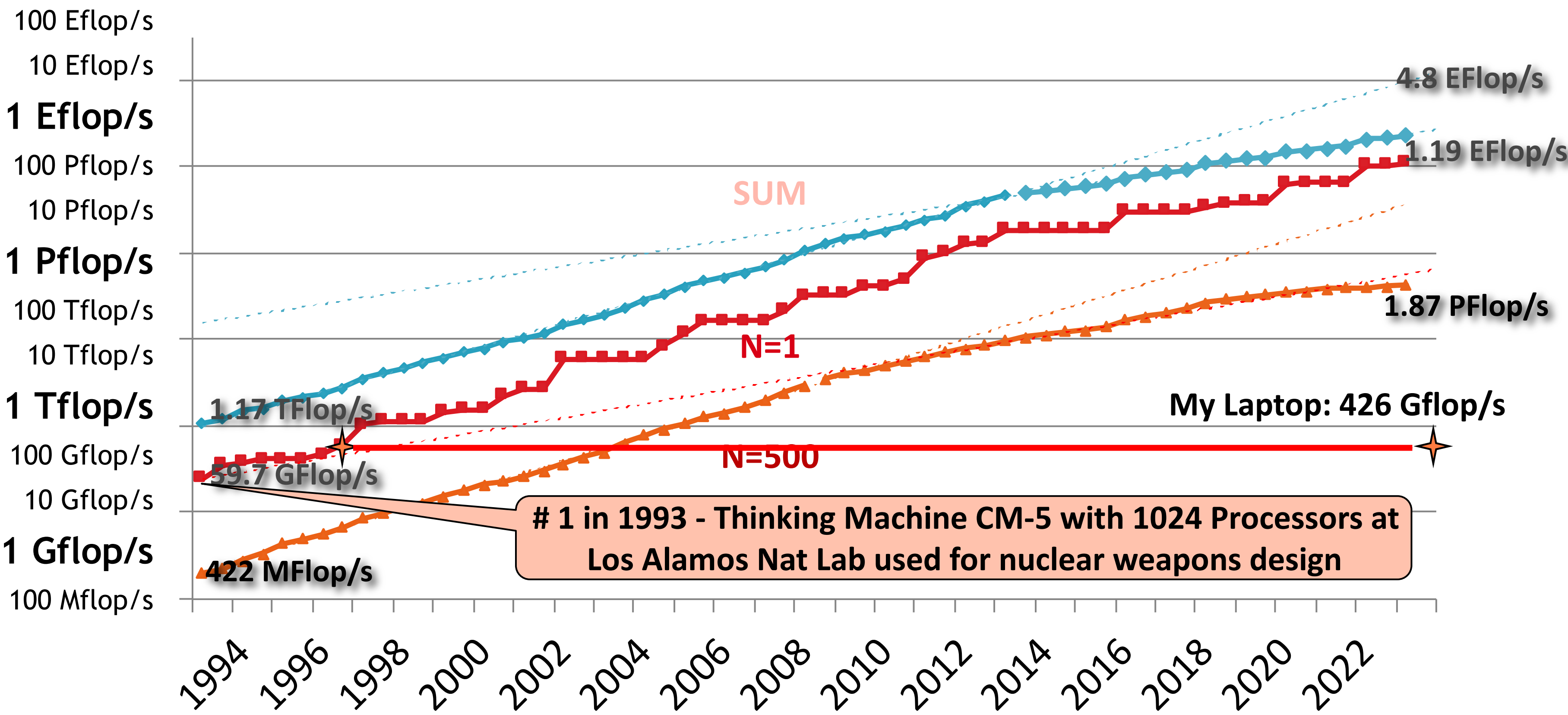
#1 Systems on the Top500 Over the Past 30 Years

Top500 List (# of times)	Computer	HPL r_{max} (Tflop/s)	Procs/Cores	Matrix Size	Hours To BM	MW
6/93 (1)	TMC CM-5/1024 (DOE LANL)	.060	1,024	52,224	0.4	
11/93 (1)	Fujitsu Numerical Wind Tunnel (Nat. Aerospace Lab of Japan)	.124	140	31,920	0.1	1.
6/94 (1)	Intel XP/S140 (DOE SNL)	.143	3,680	55,700	0.2	
11/94–11/95 (3)	Fujitsu Numerical Wind Tunnel (Nat. Aerospace Lab of Japan)	.170	140	42,000	0.1	1.
6/96 (1)	Hitachi SR2201/1024 (Univ. of Tokyo)	.220	1,024	138,240	2.2	
11/96 (1)	Hitachi CP-PACS/2048 (Univ of Tsukuba)	.368	2,048	103,680	0.6	
6/97–6/00 (7)	Intel ASCI Red (DOE SNL)	2.38	9,632	362,880	3.7	.85
11/00–11/01 (3)	IBM ASCI White, SP Power3 375 MHz (DOE LLNL)	7.23	8,192	518,096	3.6	
6/02–6/04 (5)	NEC Earth-Simulator (JAMSTEC)	35.9	5,120	1,000,000	5.2	6.4
11/04–11/07 (7)	IBM BlueGene/L (DOE LLNL)	478.	212,992	1,000,000	0.4	1.4
6/08–6/09 (3)	IBM Roadrunner –PowerXCell 8i 3.2 Ghz (DOE LANL)	1,105.	129,600	2,329,599	2.1	2.3
11/09–6/10 (2)	Cray Jaguar - XT5-HE 2.6 GHz (DOE ORNL)	1,759.	224,162	5,474,272	17	6.9
11/10 (1)	NUDT Tianhe-1A, X5670 2.93Ghz NVIDIA (NSC Tianjin)	2,566.	186,368	3,600,000	3.4	4.0
6/11–11/11 (2)	Fujitsu K computer, SPARC64 VIIIfx (RIKEN)	10,510.	705,024	11,870,208	29	9.9
6/12 (1)	IBM Sequoia BlueGene/Q (DOE LLNL)	16,324.	1,572,864	12,681,215	23	7.9
11/12 (1)	Cray XK7 Titan AMD + NVIDIA Kepler (DOE ORNL)	17,590.	560,640	4,423,680	0.9	8.2
6/13–11/15 (6)	NUDT Tianhe-2 Intel IvyBridge + Xeon Phi (NSCC Guangzhou)	33,862.	3,120,000	9,960,000	5.4	17.8
6/16–11/17 (4)	Sunway TaihuLight System (NSCC Wuxi)	93,014.	10,549,600	12,288,000	3.7	15.4
6/18–11/19 (4)	IBM Summit Power9 + Nvidia Volta (DOE ORNL)	148,600	2,414,592	16,473,600	3.3	10.1
6/20–11/22 (4)	Fujitsu Fugaku ARM A64FX (RIKEN)	442,010	7,630,828	21,288,960	4.4	29.9
6/22 - ? (1)	HPE Frontier AMD + AMD (DOE ORNL)	1,102,000	7,733,248	24,440,832	2.5	21.1

US	11
Japan	7
China	3

DOE
LANL: 2
SNL: 2
LLNL: 3
ORNL: 4

Performance Development of HPC over the Last 30 Years from the Top500



June 2023: The TOP 10 Systems (52% of the Total Performance of Top500)

Rank	Site	Computer	Country	Cores	Rmax [Pflops]	% of Peak	Power [MW]	GFlops/ Watt
1	DOE / OS Oak Ridge Nat Lab	Frontier, HPE Cray Ex235a, AMD 3rd EPYC 64C, 2 GHz, AMD Instinct MI250X, Slingshot 10	USA	8,699,904	1,194	71	22.7	52.6
2	RIKEN Center for Computational Science	Fugaku, ARM A64FX (48C, 2.2 GHz), Tofu D Interconnect	Japan	7,299,072	442.	82	29.9	14.8
3	EuroHPC /CSC	LUMI, HPE Cray EX235a, AMD 3rd EPYC 64C, 2 GHz, AMD Instinct MI250X, Slingshot 10	Finland	1,268,736	304.	72	2.94	52.3
4	EuroHPC/CINECA	BullSequana XH2000, Xeon Platinum 8358 32C 2.6GHz, NVIDIA A100 (108C), Quad-rail NVIDIA HDR100	Italy	1,824,768	239.	78	7.4	32.2
5	DOE / OS Oak Ridge Nat Lab	Summit, IBM Power 9 (22C, 3.0 GHz), NVIDIA GV100 (80C), Mellonox EDR	USA	2,397,824	149.	74	10.1	14.7
6	DOE / NNSA L Livermore Nat Lab	Sierra, IBM Power 9 (22C, 3.1 GHz), NVIDIA GV100 (80C), Mellonox EDR	USA	1,572,480	94.6	75	7.44	12.7
7	National Super Computer Center in Wuxi	Sunway TaihuLight, SW26010 (260C), Custom Interconnect	China	10,649,000	93.0	74	15.4	6.05
8	DOE / OS NERSC - LBNL	Perlmutter HPE Cray EX235n, AMD EPYC 64C 2.45GHz, NVIDIA A100, Slingshot 10	USA	706,304	64.6	71	2.59	27.4
9	NVIDIA Corporation	Selene NVIDIA DGX A100, AMD EPYC 7742 (64C, 2.25GHz), NVIDIA A100 (108C), Mellanox HDR	USA	555,520	63.4	80	2.64	23.9
10	National Super Computer Center in Guangzhou	Tianhe-2A NUDT, Xeon (12C) , MATRIX-2000 (128C) + Custom Interconnect	China	4,981,760	61.4	61	18.5	3.32

Current #1 System Overview

System Performance

- Peak performance of 2 Eflop/s for modeling & simulation
- Power: 20+ MW
- Peak performance of 11.2 Eflop/s for 16 bit floating point used in for data analytics, ML, and artificial intelligence

Each node has

- 1-AMD EPYC 7A53 CPU w/64 cores (2 Tflop/s)
  - < 1% performance of the system
- 4-AMD Instinct MI250X GPUs
  - Each w/220 cores (4*53 Tflop/s)
  - 99% performance of the system
- 730 GB of fast memory
- 2 TB of NVMe memory

The system includes

- 9408 nodes
  - 37,632 GPUs
  - 8.88M Cores
- Cray Slingshot interconnect
  - 4 end points per node
- 706 PB Memory
  - (695 PB Disk + 11 PB SSD)

System Overview (Based on public data)

System Performance

- Peak performance of 3.34 Eflop/s for modeling & simulation @ 64 bit float pt
  - At 1.6 GHz (nominal, may be lower)
- Facility Power capacity 60 MW
- Peak performance of 53.5 Eflop/s for 16 bit floating point used in for data analytics, ML, and artificial intelligence

Each node has

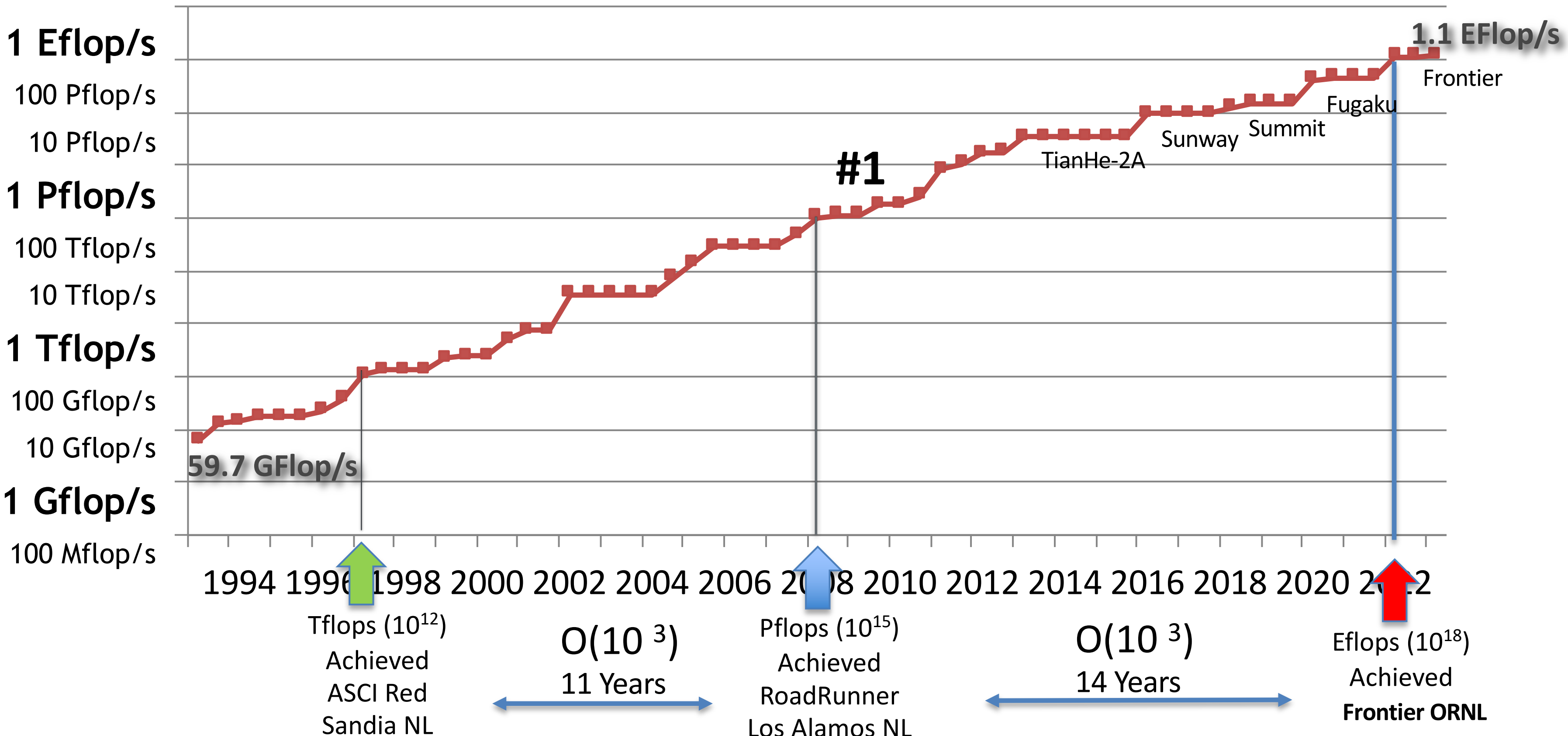
- 2 - Intel Sapphire Rapids CPU processors; w/52 cores (5.3 Tflop/s)
  - < 2% performance of the system
- 6 - Intel Xe Ponte Vecchio GPUs
- (6*52.4 Tflop/s = 314 Tflop/s)
  - 98% performance of the system
- 896 GB of HBM memory; Plus 1.02 TB DDR5 on the CPUs

The system includes

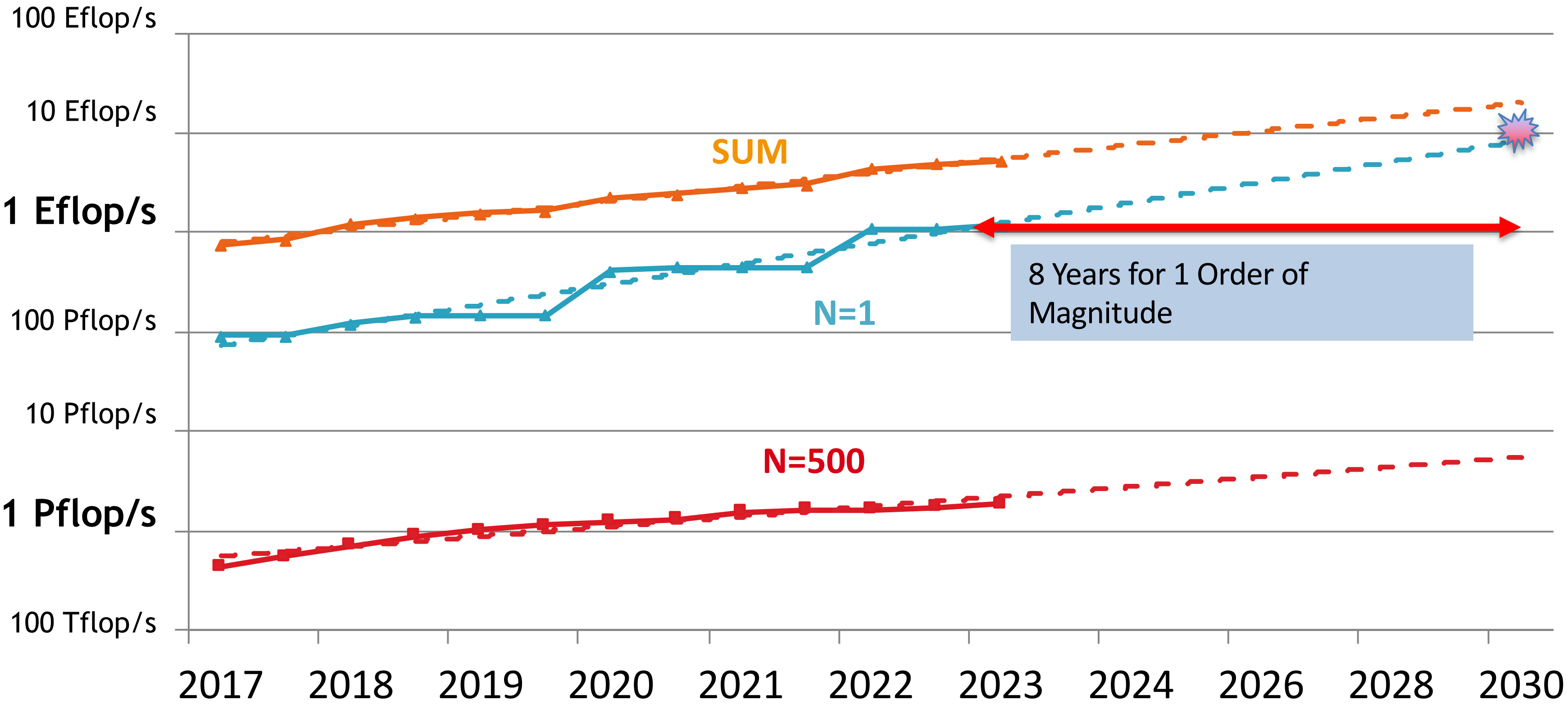
- 10,624 nodes
  - 63,744 GPUs
  - 1.1M Cores
- Cray Slingshot interconnect
  - 8 end points per node
- 10.9 PB DDR Memory
- 9.52 PB HBM
  - (230 PB Intel Optane)

230 PB of NVMe memory total (DAOS servers)

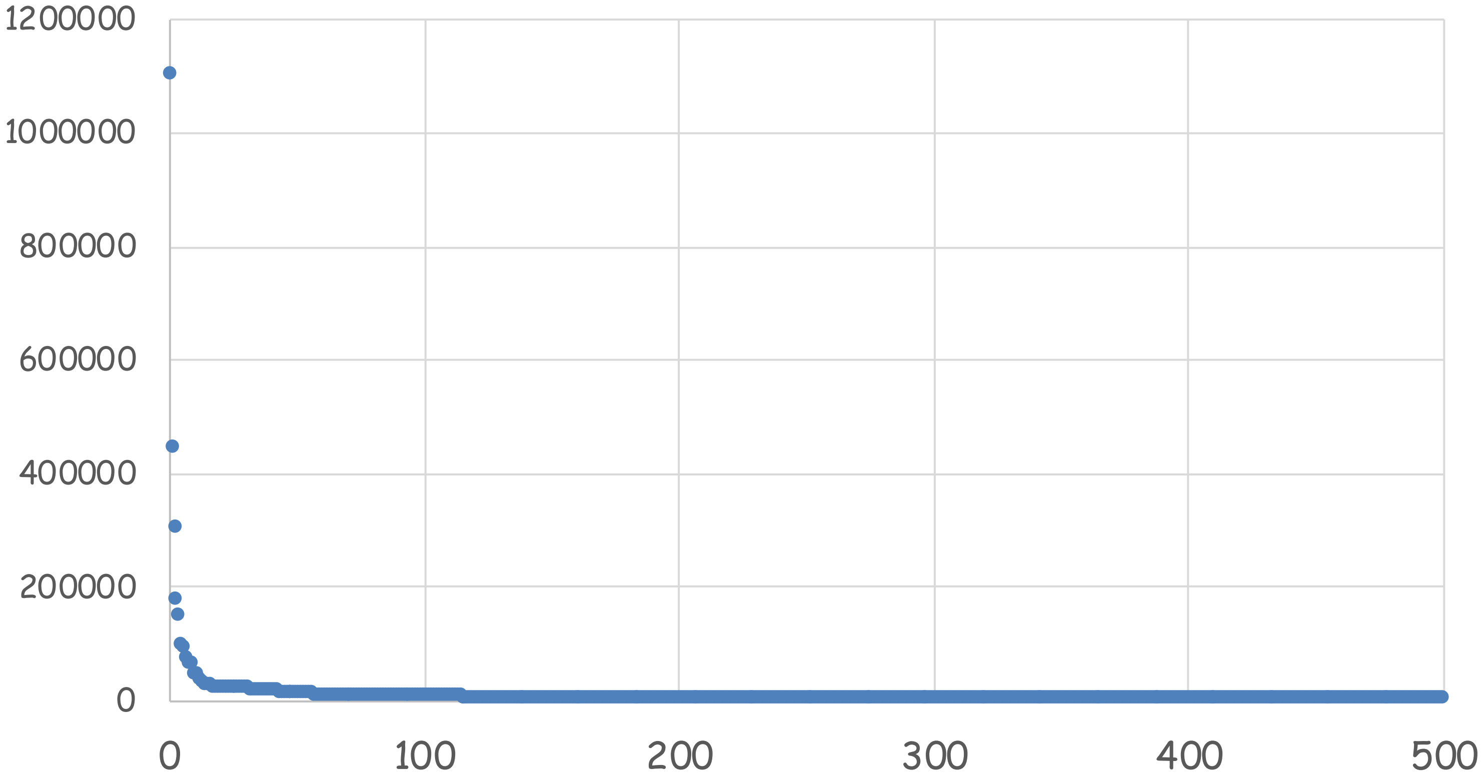
PERFORMANCE DEVELOPMENT



PROJECTED PERFORMANCE DEVELOPMENT

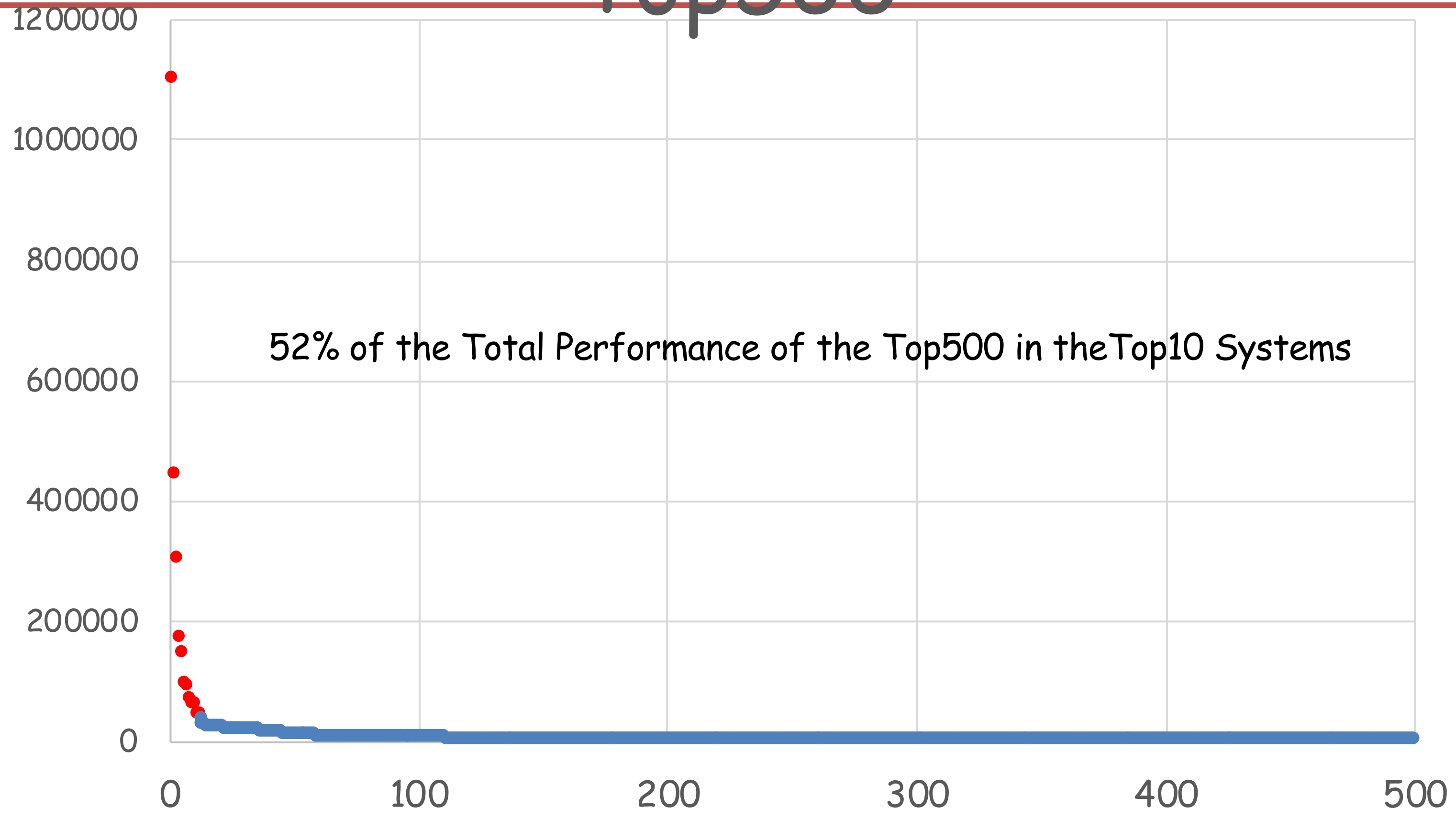


Top500

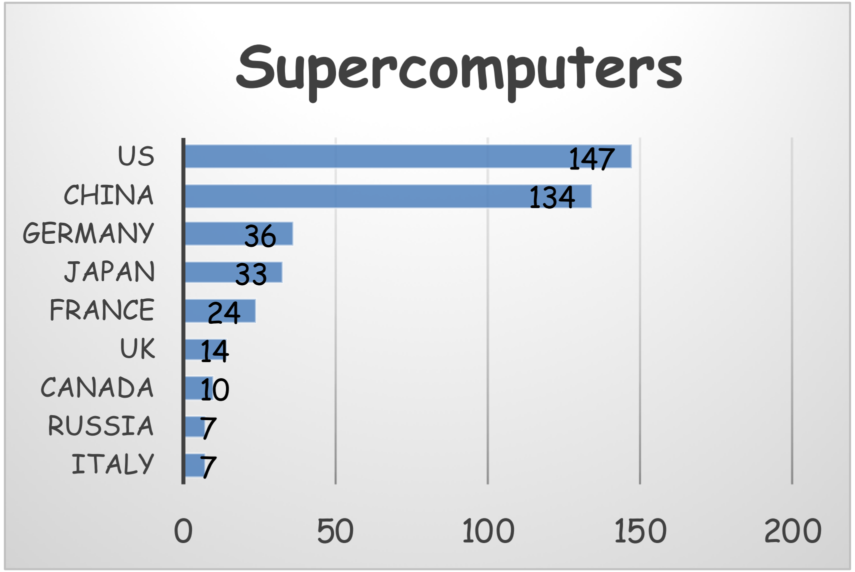


1200000
1000000
800000
600000
400000
200000
0
0
100
200
300
400
500

Top500

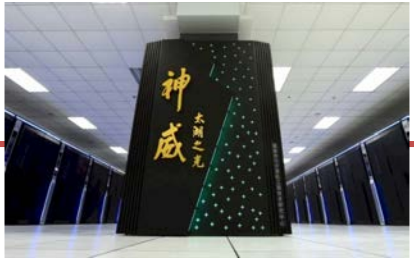

1200000
1000000
800000
600000
400000
200000
0
0
100
200
300
400
500
52% of the Total Performance of the Top500 in theTop10 Systems

China



China: Top producer overall.

5 main manufactures of HPC in China:
Lenovo(168), Inspur(43), Sugon(23),
NUDT(3), Huawei(2) with 250 systems total.



Rumored 2 Exascale Systems in Chinese

- Qingdao Marine Sunway Pro "OceanLight" (Shandong Prov)
  - Completed March 2021, 1.3 EFlops Rpeak and 1.05 EFlops Linpack
  - ShenWei post-Alpha CPU ISA architecture with large & small core structure
  - Est 96 cabinets x 1024 SW39010 390-core 35MW
  - Science on this machine won Gordon Bell Prize in 2021

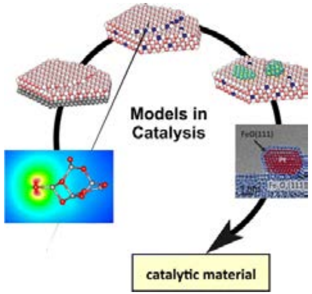
- NSCC Tianjin Tianhe-3
  - Dual-chip FeiTeng ARM and Matrix accelerator node architecture
  - Est -1.7 EFlops Rpeak

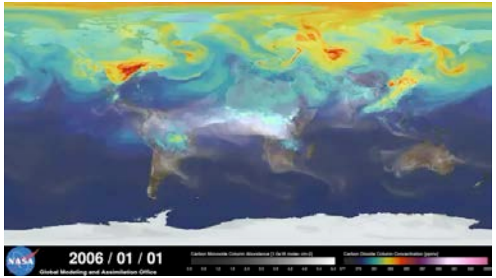
Performance and Benchmarking Evaluation Tools

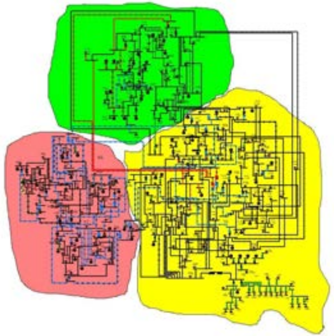
- Linpack Benchmark - Longstanding benchmark started in 1979
  - Lots of positive features; easy to understand and run; shows trends
- However, much has changed since 1979
  - Arithmetic was expensive then and today it is over-provisioned and inexpensive
- Linpack performance of computer systems is no longer strongly correlated to real application performance
  - Linpack benchmark based on dense matrix multiplication
- Designing a system for good Linpack performance can lead to design choices that are wrong for today's applications

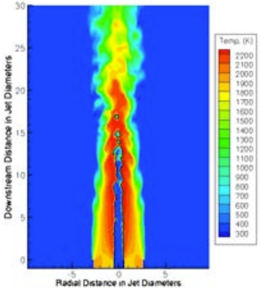
Today’s Top HPC Systems Used to do Simulations

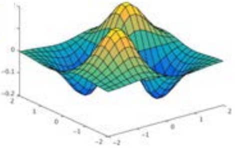
- *Climate*
- *Combustion*
- *Nuclear Reactors*
- *Catalysis*
- *Electric Grid*
- *Fusion*
- *Stockpile*
- *Supernovae*
- *Materials*
- *Digital Twins*
- *Accelerators*
- …

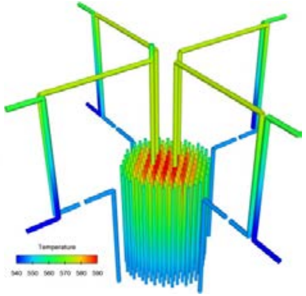


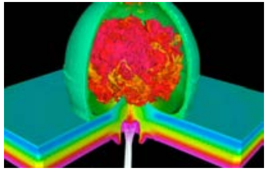


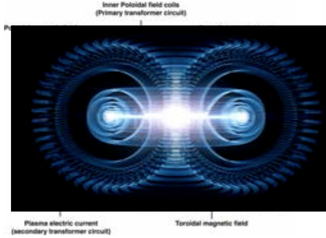






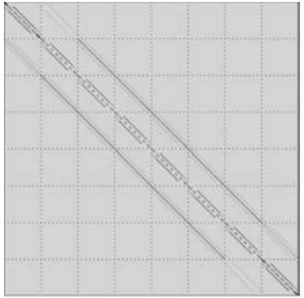






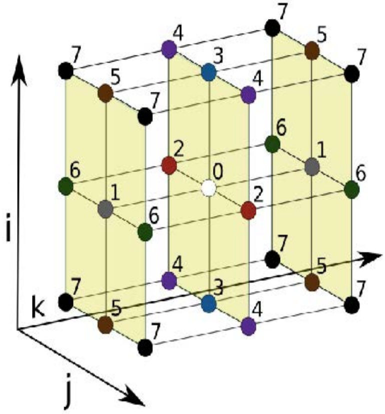
- Usually 3-D PDE’s
  - Sparse matrix computations, not dense

HPCG Results; The Other Benchmark



- High Performance Conjugate Gradients (HPCG).
- Solves *Ax=b, A* large, sparse, *b* known, *x* computed.
- An optimized implementation of PCG contains essential computational and communication patterns that are prevalent in a variety of methods for discretization and numerical solution of PDEs

- Patterns:
  - Dense and sparse computations.
  - Dense and sparse collectives.
  - Multi-scale execution of kernels via MG (truncated) V cycle.
  - Data-driven parallelism (unstructured sparse triangular solves).
- Strong verification (via spectral properties of PCG).



27-point stencil operator

HPCG Top 10, June 2023

Rank	Site	Computer	Cores	HPL Rmax (Pflop/s)	TOP500 Rank	HPCG (Pflop/s)	Fraction of Peak
1	RIKEN Center for Computational Science **Japan**	**Fugaku**, Fujitsu A64FX 48C 2.2GHz, Tofu D, Fujitsu	7,630,848	442	2	16.0	**3.0%**
2	DOE/SC/ORNL **USA**	**Frontier,** HPE Cray Ex235a, AMD 3rd EPYC 64C, 2 GHz, AMD Instinct MI250X, Slingshot 10	8,699,904	1,194	1	14.1	**0.8%**
3	EuroHPC/CSC **Finland**	**LUMI**, HPE Cray EX235a, AMD Zen-3 (Milan) 64C 2GHz, AMD MI250X, Slingshot-11	2,220,288	309	3	3.41	**0.8%**
4	[illegible]	**Leonardo**, BullSequana XH2000, Xeon Platinum 8358 [illegible]	[illegible]	[illegible]	[illegible]	3.11	[illegible]
5	DOE/SC/ORNL **USA**	**Summit**, AC922, IBM POWER9 22C 3.7GHz, Dual-rail Mellanox FDR, NVIDIA Volta V100, IBM	2,414,592	149	5	2.93	**1.5%**
6	DOE/SC/LBNL **USA**	**Perlmutter**, HPE Cray EX235n, AMD EPYC 7763 64C 2.45GHz, NVIDIA A100 SXM4 40 GB, Slingshot-10	761,856	70.9	8	1.91	**2.0%**
7	DOE/NNSA/LLNL **USA**	**Sierra**, S922LC, IBM POWER9 20C 3.1 GHz, Mellanox EDR, NVIDIA Volta V100, IBM	1,572,480	94.6	6	1.80	**1.4%**
8	NVIDIA **USA**	**Selene**, DGX SuperPOD, AMD EPYC 7742 64C 2.25 GHz, Mellanox HDR, NVIDIA Ampere A100	555,520	63.5	9	1.62	**2.0%**
9	Forschungszentrum Juelich (FZJ) **Germany**	**JUWELS Booster Module**, Bull Sequana XH2000 , AMD EPYC 7402 24C 2.8GHz, Mellanox HDR InfiniBand, NVIDIA Ampere A100, Atos	449,280	44.1	12	1.28	**1.8%**
10	Saudi Aramco **Saudi Arabia**	**Dammam-7**, Cray CS-Storm, Xeon Gold 6248 20C 2.5GHz, InfiniBand HDR 100, NVIDIA Volta V100, HPE	672,520	22.4	20	0.88	**1.6%**

Think of a race car that has the potential of 250 KPH but only goes 2 KPH!

Recently we have seen AI & ML take off

- AI and ML have been around for a long time as research efforts.
- Why Now?
  - Flood of available data (especially with the Internet)
  - Increasing computational power
  - Growing progress in available algorithms and theory developed by researchers.
  - Increasing support from industries.

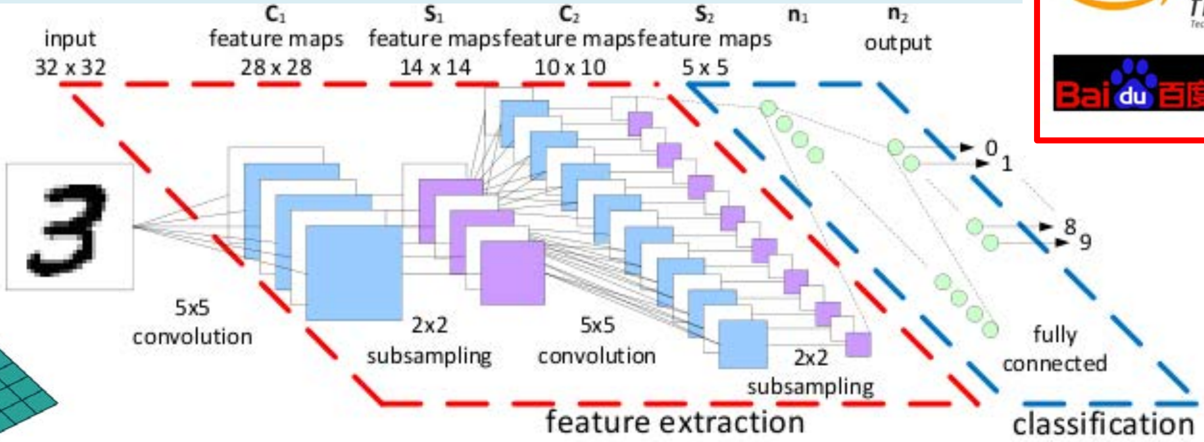
Deep Learning Needs Small Matrix Operations

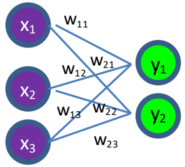
Matrix Multiply is the time-consuming part.

Convolution Layers and Fully Connected Layers require matrix multiply

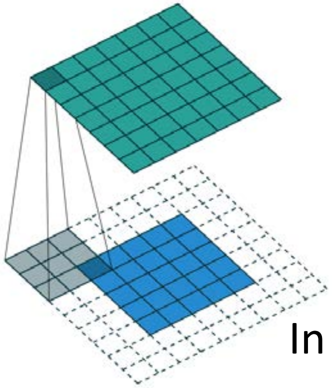
There are many GEMM's of small matrices, perfectly parallel, can get by with 16-bit floating point

Emergence of AI-Specific Hardware Ecosystem





$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$



Convolution Step
In this case 3x3 GEMM

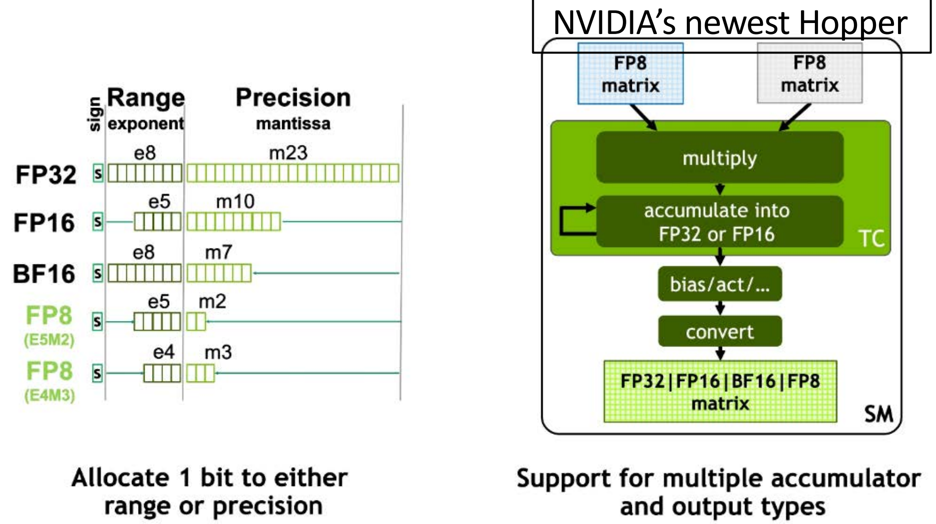
Fully Connected
Classification

Floating Point Representations

	Range	Accurac
IEEE FP 128 (Quad)	$\pm 10^{-4932}$ to 10^{4932}	
IEEE FP64 (Double)	$\pm 10^{-308}$ to 10^{308}	
IEEE FP32 (Single)	$\pm 10^{-38}$ to 10^{38}	
IEEE FP16 (Half)	$\pm 10^{-5}$ to 65504	
Google BFloat16	$\pm 10^{-38}$ to 10^{38}	

Sign bit | Exponent bits | Fraction bits (Mantissa)

NVIDIA's newest Hopper



Hopper FP8 Precisions – 2x throughput

Seeing cloud vendors now designing their own processors:
Google TPU, AWS Graviton, Apple M1

WHY MIXED PRECISION? (Less is Faster)

- There are many reasons to consider mixed precision in our algorithms…
  - Less Communication
    - Reduce memory traffic
    - Reduce network traffic
  - Reduce memory footprint
  - More Flop per second
    - Reduced energy consumption
    - Reduced time to compute

IBM Cell Broadband Engine	Apple ARM Cortex-A9	NVIDIA Kepler K10, K20, K40, K80	NVIDIA Volta/Turing	NVIDIA Volta/Turing
14x	7x	3x	2x	16x
32 bits / 64 bits	32 bits / 64 bits	32 bits / 64 bits	32 bits / 64 bits	16 bits / 64 bits

  - Accelerated hardware in current architecture.
  - Suitable numerical properties for some algorithms & problems.

J. Langou, J. Langou, P. Luszczek, J. Kurzak, A. Buttari, and J. J. Dongarra. Exploiting the performance of 32 bit floating point arithmetic in obtaining 64 bit accuracy. In *Proceedings of the 2006 ACM/IEEE Conference on Supercomputing*, 2006.

Can We Take Advantage of the Hardware?
Basically, There are Three Approaches with Mixed Precision

1. Use a mathematical technique
   - Get an approximation in lower precision then use something like Newton's method to enhance accuracy.
2. Transfer less bytes, data transfer is expensive
   - Store data in primary storage in full precision.
   - Transfer the data in short precision.
     - Could also use data compression techniques
   - Compute using full precision.
3. Use a combination of 1. & 2.

HPL-MxP Top 10 for June 2023

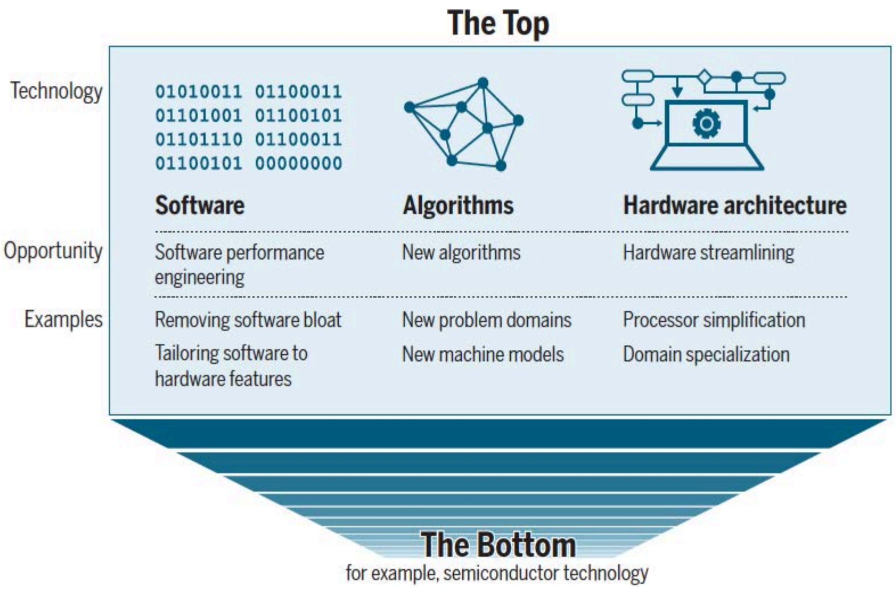
Rank	Site	Computer	Cores	HPL Rmax (Eflop/s)	TOP500 Rank	HPL-MxP (Eflop/s)	Speedup
1	DOE/SC/ORNL **USA**	**Frontier**, HPE Cray EX235a, AMD Zen-3 (Milan) 64C 2GHz, AMD MI250X, Slingshot-10	8,699,904	1.194	1	9.95	8.3
2	EuroHPC/CSC **Finland**	**LUMI**, HPE Cray EX235a, AMD Zen-3 (Milan) 64C 2GHz, AMD MI250X, Slingshot-11	2,220,288	0.309	3	3.41	11
3	RIKEN Center for Computational Science, **Japan**	**Fugaku**, Fujitsu A64FX, Tofu D	7,630,848	0.442	2	2.0	4.5
4	EuroHPC/CINECA **Italy**	**Leonardo**, BullSequana XH2000, Xeon Platinum 8358 32C 2.6GHz, NVIDIA A100 SXM4 40 GB, Quad-rail NVIDIA HDR100 Infiniband	1,824,768	0.239	4	3.11	13
5	DOE/SC/ORNL **USA**	**Summit**, AC922 IBM POWER9, IB Dual-rail FDR, NVIDIA V100	2,414,592	0.149	5	1.4	9.5
6	NVIDIA **USA**	**Selene**, DGX SuperPOD, AMD EPYC 7742 64C 2.25 GHz, Mellanox HDR, NVIDIA A100	555,520	0.063	9	0.63	9.9
7	DOE/SC/LBNL/NERSC **USA**	**Perlmutter**, HPE Cray EX235n, AMD EPYC 7763 64C 2.45 GHz, Slingshot-10, NVIDIA A100	761,856	0.071	8	0.59	8.3
8	Forschungszentrum Juelich (FZJ) **Germany**	**JUWELS Booster Module**, Bull Sequana XH2000 , AMD EPYC 7402 24C 2.8GHz, Mellanox HDR InfiniBand, NVIDIA A100, Atos	449,280	0.044	13	0.47	10
9	University of Florida **USA**	**HiPerGator**, NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, Infiniband HDR	138,880	0.017	40	0.17	9.9
10	SberCloud **Russia**	**Christofari Neo**, NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100 80GB, Infiniband	98,208	0.012	55	0.12	10.3

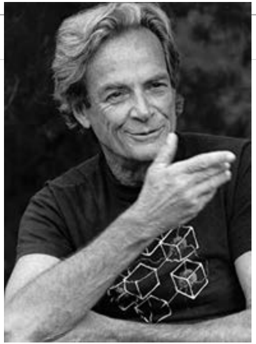
The Take Away

- HPC Hardware is Constantly Changing
  - Scalar
  - Vector
  - Distributed
  - Accelerated
  - Mixed precision
- Three computer revolutions
  - High performance computing
  - Deep learning
  - Edge & AI
- Algorithm / Software advances follows hardware.
  - And there is “plenty of room at the top”

“There’s plenty of room at the Top: What will drive computer performance after Moore’s law?”

Leiserson *et al.*, *Science* **368**, 1079 (2020) 5 June 2020





Feynman’s 1959 Lecture @ CalTech