THE UNIVERSITY of York

High Performance Computing - The Future

Prof Matt Probert

http://www-users.york.ac.uk/~mijp1

Autumn Term 2022

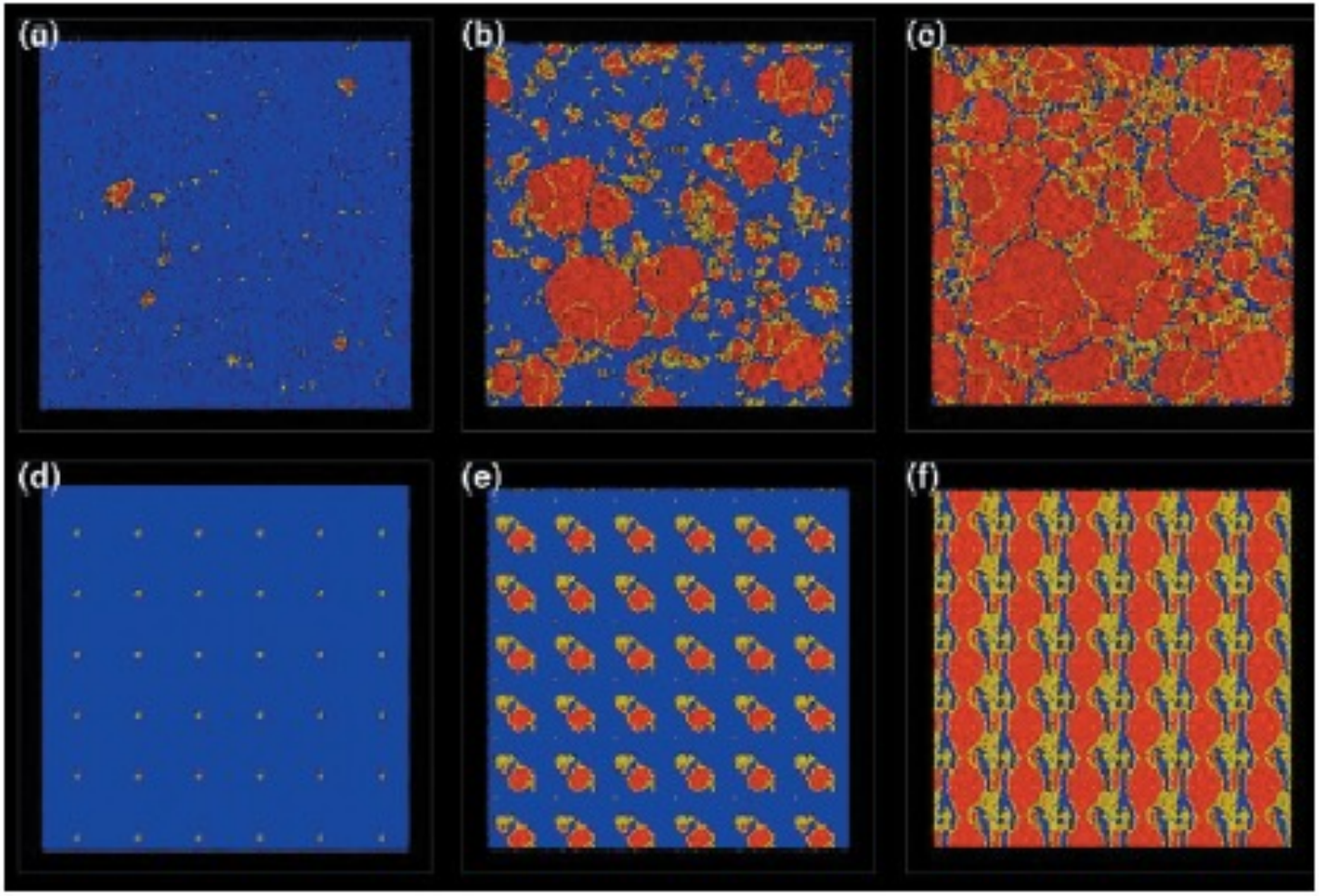
Overview

- Big Computing / Big Data
- HPC Languages
- GPU programming
- New CPUs

Big Computing

Why do we need Big Computing?

- Domain decomposed MD with dynamic domains



Snapshots from simulations of solidification in tantalum. The top sequence displays nucleation (a) and growth (b) occurring in a 16,372,000-atom simulation, resulting in a realistic distribution of grains and grain boundaries (c). The same process modeled using 64,000 atoms (d–f) produced the artificial final structure shown in (f).

Beowulf

- Beowulf designs are cheap and popular
  - Hardly use this name anymore – become so ubiquitous
  - Rapid growth since mid 1990s – large part of Top500
  - Enabled by powerful and cheap CPUs *and* developments in network technology (Infinband, etc.)
- Typically “fast compute, slow interconnect”
  - Challenges to large-scale parallelism
  - Need lots of latency hiding to get good scaling
- Hence interest in slow/low-power CPUs
  - E.g. Intel Atom, ARM and IBM Power PC CPUs
  - High packing density, lower running costs,
  - And easier to get codes to scale!

ExaScale Computing

- ExaScale = 10^{18} FLOPs
  - Original plan was 2018 according to Moore's Law but only reached in 2022 –delayed
  - Frontier is first true Exa with 1.1 ExaFLOPs
  - UK target is 2024?
  - Power/cooling limitations
  - Programming methods
  - Component reliability - MTBF
  - Parallel scaling challenges
- What science will become capable? How to manage the data generated?

Compare Fugaku …

- Fujitsu with ARM CPUs & no accelerators
- A64FX 48core 2.2GHz CPU
  - 7,630,848 cores in 158,976 nodes
- Performance:
  - LINPACK 442 TFLOPs
  - Peak = 537 TFLOPs
  - i.e. 83% peak at 29.9 MW
- Cost ~$1b

… with Frontier

- HPE Cray with ARM CPUs & GPUs
  - AMD 3rd gen EPYC 64core 2GHz CPU
  - AMD INSTINCT MI250X GPU
  - 8,730,112 cores in 9,472 nodes
- Performance:
  - LINPACK 1.102 EFLOPs
  - Peak = 1.685 EFLOPs
  - i.e. 65% peak at 17.8 MW
- Cost ~$0.6b
- Cheaper, less power but harder to program
  - Need hybrid MPI + HIP or OpenMP
- Many of the Top500 are hybrid machines
  - Not a trend that is going away soon
  - But we desperately need open standards to get portability and longevity of codes

Big Data

- LHC at CERN generates huge datasets – 35 TB/day – which need to be stored and analysed
  - Hence CERN is at the forefront of implementing distributed computing – cannot store & process such large amounts of data – needs to be able to distribute it around the world to get local storage and analysis
  - Large strain on networks – dedicated 10 Gbit/s fibre optic links to 11 “Tier 1” institutions
  - Called ‘LHC Computing Grid’ – a practical way of managing the volumes of data to be generated

Distributed Data

- Who else?
  - Large data sets requiring distributed processing
  - Issue with trust – not standard PC types.

- Square Kilometre Array (SKA)
  - Sites in South Africa & Australia
  - Building started in 2018, treaty signed in 2019, full size system to be online by 2027
  - Will have total x50 sensitivity and 10,000x faster than any other radio telescope array
  - On-site computer power of 10^8 PCs using Chinese CPUs
  - Generate one exabyte of data/day
  - Phase 1 (underway) = £2b & Phase 2 (future) = ?? £20b ??

- Distributed computing?

Distributed Computing Projects

- Folding@Home
  - Focus on SARS-CoV-2 & proteins that interact with it as part of the Moonshot Collaboration to find treatments
  - 1st computing project ever to sustain 1 PFLOP (Sept07)
  - Now at 2.4 ExaFLOPs for COVID research!
    - Using 700,000 home PCs
  - Can run on CPU/GPU and Win/Mac/Linux and …
    - Was also PS3 until Sony 2012 stopped support
  - Data generated has produced over 225 papers so far
  - MPI parallel since 2006, threads and OpenCL (not CUDA) for both nVidia & AMD GPUs in 2010
- Lots of other “@home” projects including SETI@home (the first) …

Cloud Computing

- An increasingly popular (commercial) approach to providing compute cycles
- Replace ‘capital’ by ‘recurrent’ costs
- Buy cycles from large compute farms,
  - e.g. Amazon (AWS since 2006) or Google Cloud Platform or Microsoft Azure or …
- Used to be generic VM offering
- Then went to ‘Software as a Service’ with dedicated images for particular packages
- Now providing dedicated HPC offerings

HPC on Cloud

- Advantages
  - Access to additional hardware on demand
  - Now includes GPU, Xeon-Phi, FPGA, etc
  - Useful for occasional users or to try-out tech
- Disadvantages
  - High cost for regular users
  - Typically 6p/core-hour vs 2p for own hardware
  - Amazon EC2 allows for spot pricing
  - Limited in system size available
  - Non-local geography for parts of system
  - Data security? Storage costs?

HPC Languages

Dedicated Language?

- Alternative to using a common language (e.g. C/C++ or Fortran) + libraries or directives
  - Tried in early days but lost out
- DARPA started High Productivity Computing Systems program in 2004 to build peta-scale
  - IBM Roadrunner 2008 at Los Alamos USA
  - Develop hardware + languages + o/s + file system
- Languages included
  - Fortress (Sun), Chapel (Cray), X10 (IBM)
  - All examples of PGAS (partitioned global address space) languages
- Or improve traditional languages?

Fortress (Sun)

- In mid-2000s there was a lot of effort to rebrand Java as a HPC language (Java Grande) but:
  - No IEEE 754 support + few intrinsic math functions
  - Not in MPI or OpenMP standards
  - Slow unless converted to native code
  - Java Grande Forum died – brief revival when Java went GPL (end 06) but then nothing …
- SUN created a new HPC language – “Fortress”
  - Designed to be a secure Fortran that was intrinsically parallel and type-safe with pseudocode syntax
  - Started in 2005, open source in 2007, but then problems with JVM licensing meant Oracle decided to drop the project in 2012 so now looks dead …

Chapel (Cray)

- Designed to separate algorithms from data representation
- Multi-threaded parallelism for data + task + nested parallelism
  - Based upon HPF ideas
- With support for OOP
- Started in 2009, open source, still being actively developed

X10 (IBM)

- Focus on concurrency and distribution with OOP like Java or C#
- Asynchronous PGAS
- Uses parent + child to handle locks/race conditions
  - Parent can wait for child but not v.v.
- Can use JVM or compile to native code
- Started in 2004, open source, still being actively developed

UPC/UPC++ (Unified Parallel C)

- Based upon C99 with SPMD model
- Can handle either shared or distributed memory machines
  - An explicitly parallel execution model
  - Appears as shared address space to programmer
    - any variable can be r/w from any processor but physically associated with a single processor
  - Synchronization primitives and a memory consistency model
  - Memory management primitives

Fortran 2008 – Co-Arrays

- Allows SPMD within Fortran
  - easier to use than MPI
  - designed for data decomposition
- Example

```
REAL, DIMENSION(N)[*] :: X,Y
X(:) = Y(:)[Q]
```

  - Additional [ ] shows that this item is a co-array and is distributed
  - Second line shows how to copy values from one "memory image" to another (c.f. MPI_Send/Recv)
- Full support in gfortran since v5.0
  - Most F2008 in gfortran 6.0, few minor bits not in 9.0
- Idea copied by Cray for Coarray C++

Future Fortran?

- Fortran 2018 standard published (Nov 2018)
  - Minor revision of F2008 & originally called F2015
- Two major additions:
  - TS29113 (Further Interoperability with C), was approved in 2012 and available in Intel ifc v16
  - TS18508 (Additional Parallel Features in Fortran), extends coarrays and SELECT RANK (gfortran v10). Extends DO CONCURRENT with data locality (not yet)
- Plus minor improvements
  - to environment variables, STOP commands, etc.
- Some features already implemented

GPU Programming

How to program a GPU

- nVidia has CUDA for its GPUs
  - Vendor lock-in with nvcc and hardware but cross-platform (Windows, Mac and Linux)
  - SDK supports PathScale Open64 C compiler + third-party wrappers available for Python, .Net, etc.
  - v8 (Sept 2016) *aka* Pascal has unified memory model and direct access to main RAM
    - Removes key performance bottleneck but only on non-x86 architectures (hence SUMMIT & Bede using Power9 CPUs)
- AMD has HIP as high-level and ROCm as low-level alternatives for Radeon GPUs
- Intel promotes OneAPI for its GPU & FPGAs
- Also OpenCL, OpenACC and OpenMP v4 for non-vendor specific approaches!

New CPUs

Low Power HPC

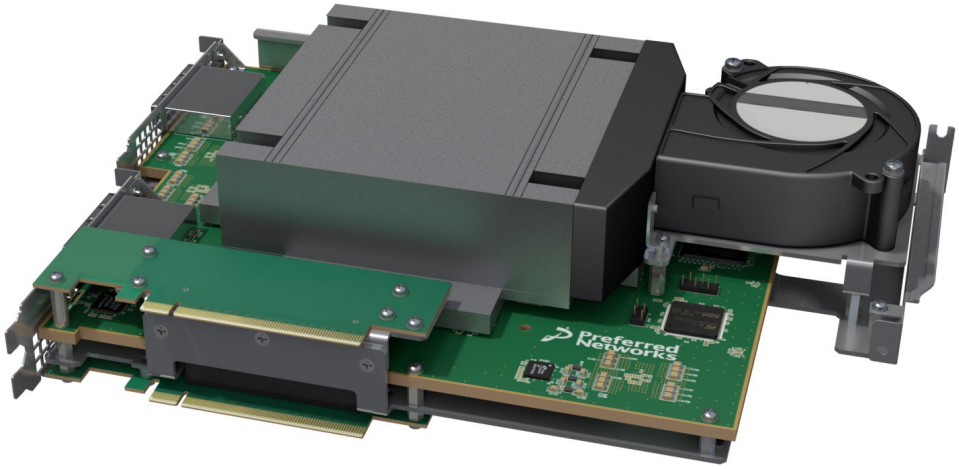
- Green500 focuses on power-per-Watt
  - #1 machine (2022) uses Intel Xeon Platinum + nVidia H100
    - Linpack = 2.0 TFLOPs vs peak = 5.4 TFLOPs at 31 kW
  - 7 in top #10 are AMD EPYCC based & 3 are Xeon Platinum
  - 6 use AMD Instinct accelerators, 3 nVidia , 1 with MN-Core
  - Lots of different CPUs in the list inc. ARM and novel designs
- Exascale?
  - If could scale #1 machine to Exascale it would need 16 MW
  - When Green500 was launched in 2007 it was projected to take 3000 MW => almost 200x better!
  - Frontier is #6 and IS 1st Exascale & takes only 21 MW

- Also look at SWaP (space, wattage and performance) = performance/(space*power)

MN-Core

- Japanese accelerator developed by Preferred Networks
- Designed for training phase of deep learning with focus on matrix math
  - 1 TFLOP/W in ½ precision tensor core
  - Massive SIMD
  - Minimal instruction set – no if-branches!
  - NOT a general-purpose accelerator

Designed to be used within pyTorch …



AMD

- Now at 7 nm process – Intel still at 14 nm until 2023!
- Radeon Vega/Navi GPGPU line
  - Use HIP not CUDA, or OpenMP not OpenACC
- Ryzen desktop CPU
  - Ryzen 7 ~ Intel i7 etc
  - Threadripper up to 64 cores, 128 threads
- EPYC HPC CPU
  - Server grade CPU to compete with Intel Scalable Xeon
  - Based on Zen architecture (new in 2017)
  - ARCHER2 has 2nd gen EPYC Rome (64 core)
  - Frontier (Oak Ridge) has EPYC + Radeon and 1st to sustain 1 EFLOPS
- APU line (Accelerated Processing Unit)
  - Fusion of multiple CPU + GPU cores in a single package with flat memory
  - Used in PS4 & Xbox One – ‘system on a chip’
  - Ryzen 4000 (2020) = Zen2 CPU + Vega GPU

Intel

- Falling behind AMD & ARM
  - Still 14nm. 10nm was a disaster & 7nm now due in 2023.
  - Cancelled Xeon Phi in 2017 so no 3rd gen.
    - 2nd gen had 72 Atom cores with 4 threads/core so 2.8 TFLOP dp@200 W. c.f. Intel ASCI Red (1997) was 1st TFLOP supercomputer with 10,000 Pentiums - cost $55 m!
  - Focus now on Xeon Scalable (but Ryzen cheaper)
  - Aurora (Argonne) was to be based on 3rd Gen Phi for pre-Exa in 2018. Change to Scalable chips in 2021. Due now …
  - Frontier 1st to Exa with AMD - DoE keen on ‘made in USA’.
  - 3rd Gen Xeon Scalable (Platinum) up to 40 cores (80 threads) + support for Optane ‘persistent memory’ for fastest I/O
    - Optane up to 4.5 TB/socket as replacement HD
    - Ponte Vecchio GPU at 7nm in 2022

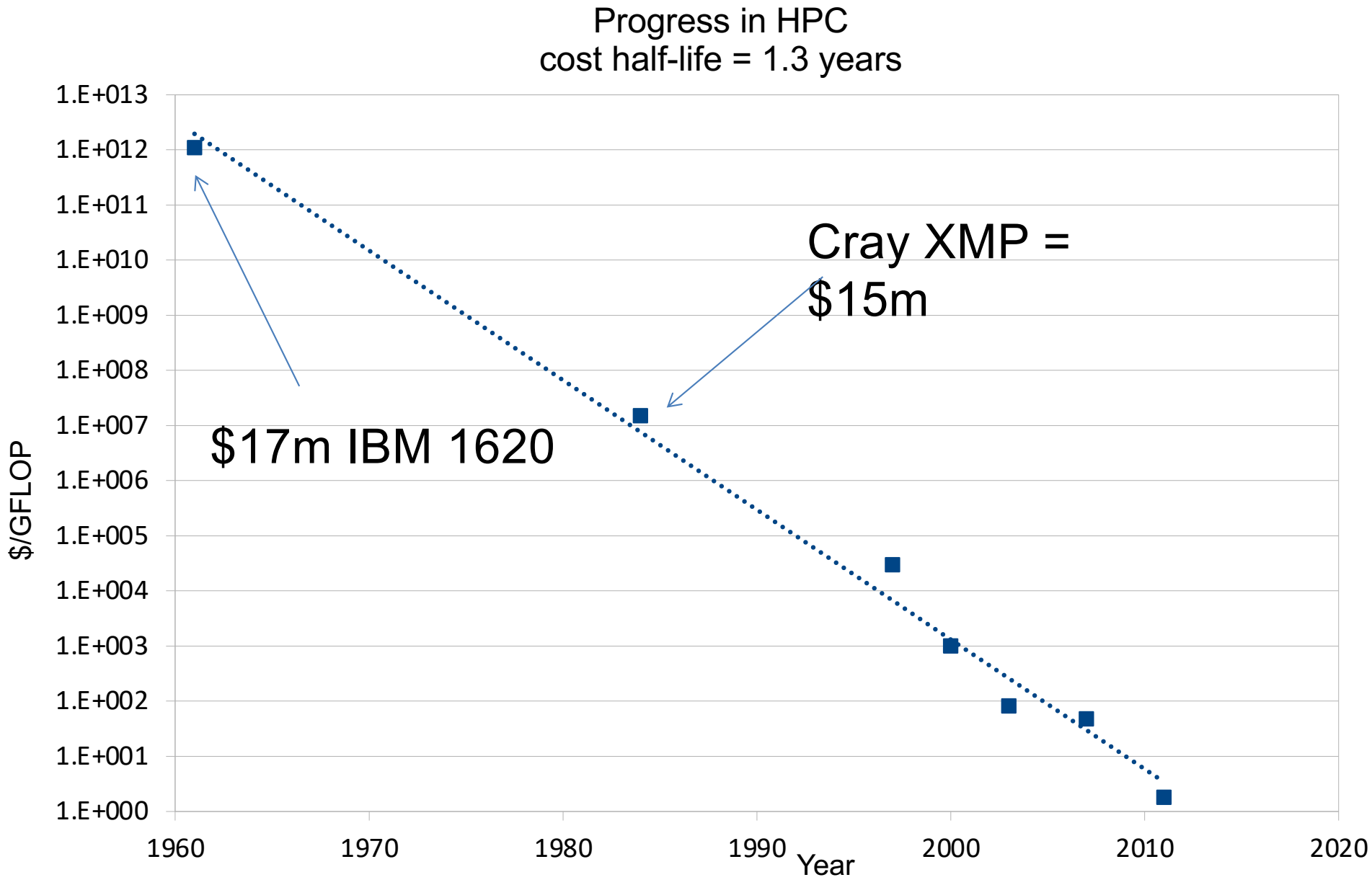
ARM

- Suddenly big in HPC space
  - Isambard (Tier-2) in 2016 at Bristol
  - Was #1 in Top500 in 2020 (Fugaku) no GPUs
  - 7nm process at Taiwan fab
  - Came from Acorn in Cambridge for BBC-B
    - Dominant in mobile/embedded as low-power
    - Bought by Japanese ‘SoftBank Group’ in 2016 for £24b. nVidia was buying for $40b but blocked.
  - Apple M1 Macs have ARM CPU (2020+)
  - EU developing own chip from ARM design for the European Processor Initiative

Affordable Supercomputing?

- Cost/FLOP has been going down for many years
- Recent developments in supercomputing include Beowulf / GPU / MIC etc
- New trends include ARM and custom CPU designs
- New dedicated ML/AI accelerators including MN-Core, Cerebras, Graphcore and nVidia Tensor Cores
  - Designed for SGEMM etc
- New dedicated DPUs including nVidia BlueField (bought Mellanox in 2020)
- Often new design is ARM or FPGA based ….

Moore's Law for HPC Cost



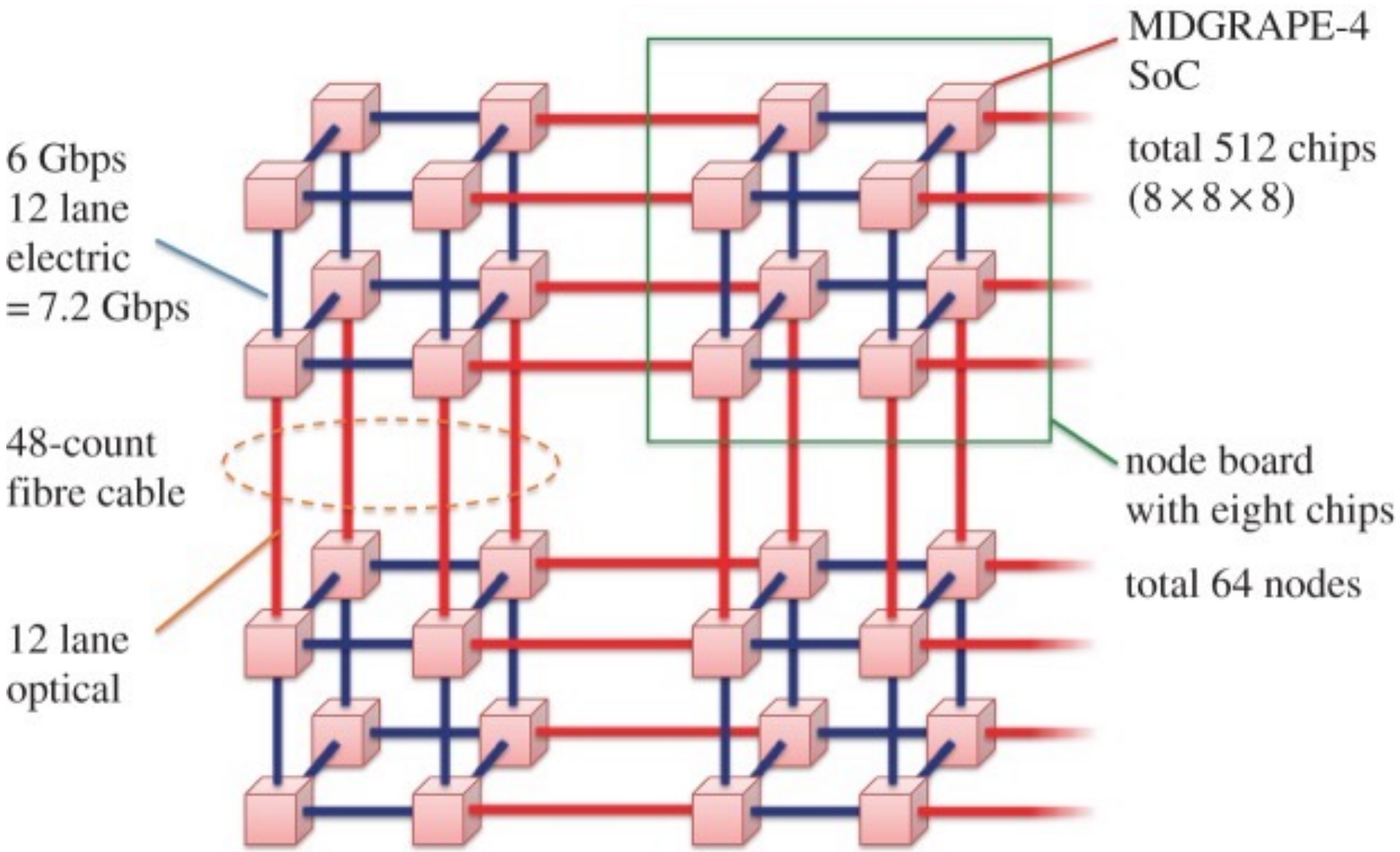
FPGA?

- Field-Programmable Gate Array
- A chip where the interconnects between logic blocks can be decided by the user ‘in the field’ using Hardware Description Lang
  - NOT a general purpose computer but can implement key functions in hardware
  - Popular in lattice-QCD
- Traditionally much slower and more expensive and lower volumes than ASIC
- Being developed by IBM & Intel & …

MD-GRAPE

- Special purpose computer for protein MD calcs
- Dedicated hardware for particular classes of force calculation etc
- System-on-Chip design: combines dp and sp cores + memory + 3D torus interconnect
- V3 had 1 PFLOP with 4080 CPUs in 2006 and cost $9m c.f. BlueGene/L at same time had 131,072 cores for 0.28 PFLOP and $250m
- V4 in beta since Feb 2016 … stalled?
- Lead designer now with PFN working on MN-Core

MDGRAPE-4


MDGRAPE-4
SoC
total 512 chips
(8×8×8)
6 Gbps
12 lane
electric
= 7.2 Gbps
48-count
fibre cable
12 lane
optical
node board
with eight chips
total 64 nodes

Further Reading

- Folding@Home at http://foldingathome.org/

- Fortress at https://github.com/stokito/fortress-lang
- Chapel at https://chapel-lang.org
- X10 at http://x10-lang.org
- UPC at http://upc-lang.org
- Fortran2018 at http://fortranwiki.org/fortran/show/Fortran+2018

- Green500 at http://www.top500.org/list/green5000
- MN-Core at https://projects.preferred.jp/mn-core/en/
- European Processor Initiative at https://www.european-processor-initiative.eu/
- Programming Aurora at https://www.iwocl.org/wp-content/uploads/iwocl-syclcon-2020-finkel-keynote-slides.pdf
- FPGA for Lattice-QCD at https://en.wikipedia.org/wiki/QPACE
- MDGRAPE at https://en.wikipedia.org/wiki/RIKEN_MDGRAPE-3