SINTEF

GRAPHICS CARDS AND CONSERVATION LAWS

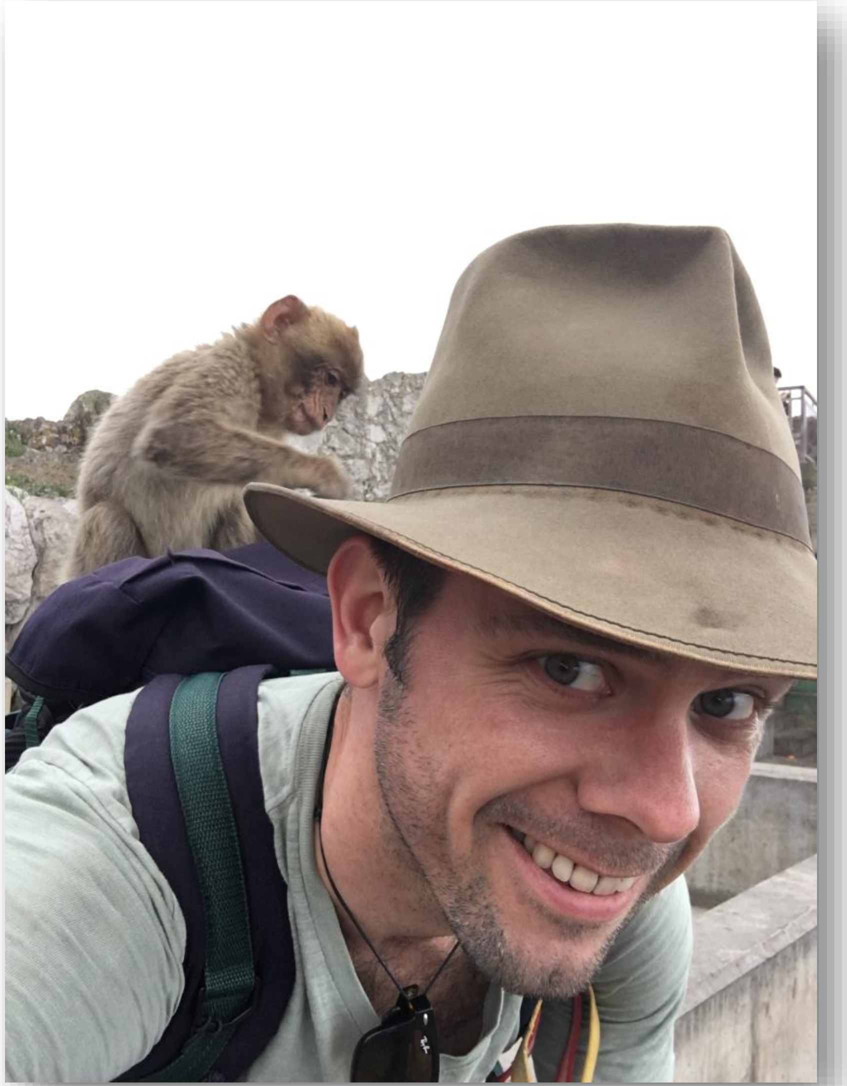
André R. Brodtkorb, Researcher
Department of Mathematics and Cybernetics
SINTEF Digital

My background

- PhD from University of Oslo (2010)
- Associate professor (Førsteamanuensis) at OsloMet
- Researcher at SINTEF (currently on 80% leave)

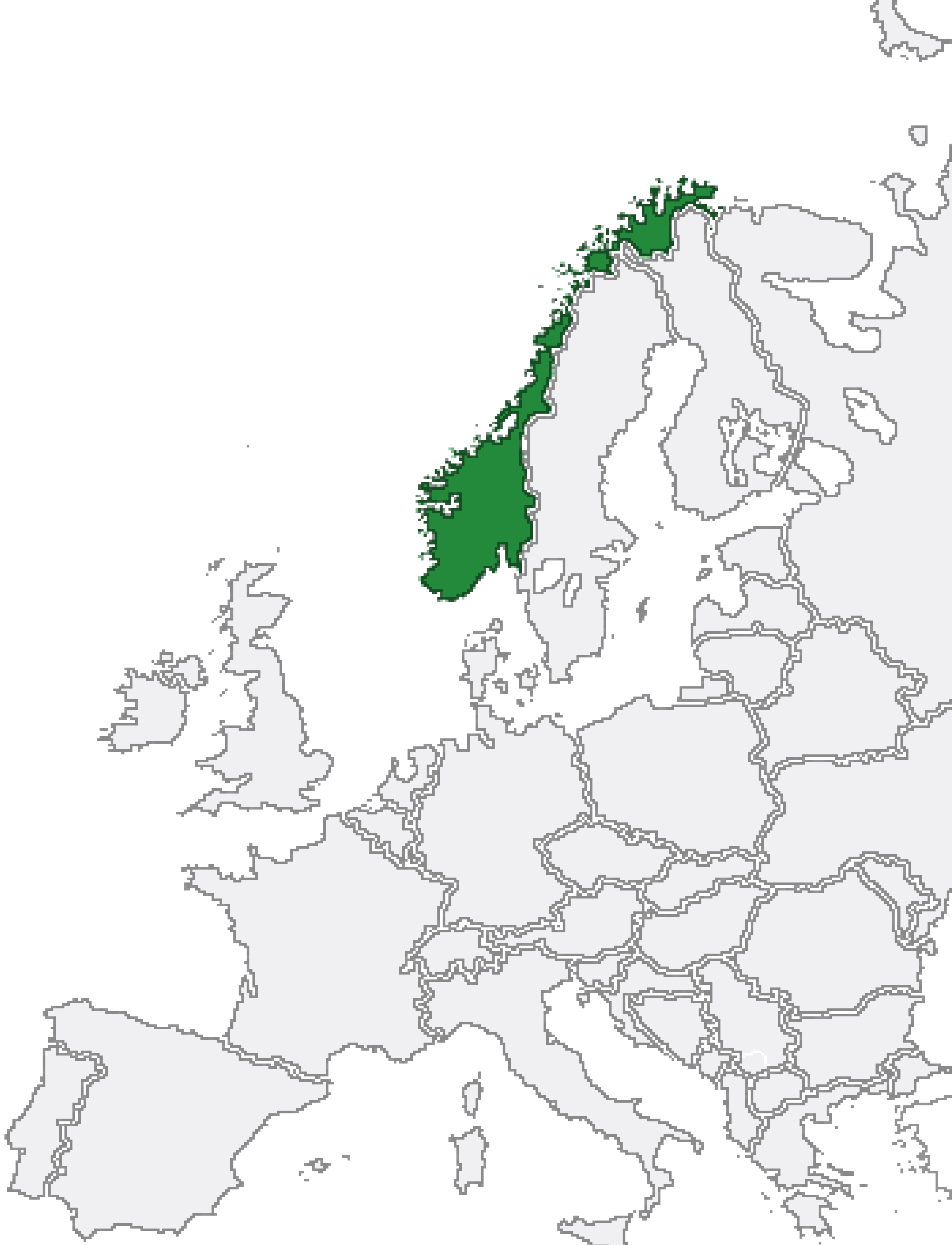
- Previously lectured at NITH and UiO.

- Have taught computer graphics and geometry, multivariate calculus, algorithms and data structures, GPU computing, …



SINTEF

- Established 1950 by the Norwegian Institute of Technology.
- The largest independent research organisation in Scandinavia.
- A non-profit organisation.
- Motto: “Technology for a better society”.
- Key Figures*
  - 2100 Employees from 70 different countries.
  - 73% of employees are researchers.
  - 3 billion NOK in turnover
    (about 360 million EUR / 490 million USD).
  - 9000 projects for 3000 customers.
  - Offices in Norway, USA, Brazil,
    Chile, and Denmark.



[Map CC-BY-SA 3.0 based on work by Hayden120 and NuclearVacuum, Wikipedia]

Todays "popular science" seminar

- Desktop parallel computing: What are GPUs, and why care about them?
- Things to consider when designing parallell code
- (Maybe) Videos
- Live coding session – Python and GPUs for the wave equation

SINTEF

Heterogeneous Computing Group

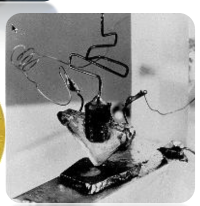
http://hetcomp.com

Desktop parallell computing

History lesson: development of the microprocessor 1/2

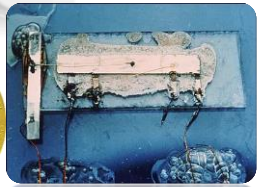
1942: Digital Electric Computer
(Atanasoff and Berry)

1956

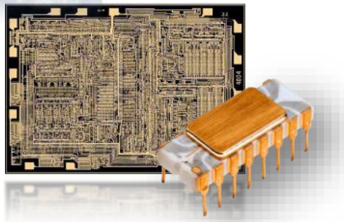


1947: Transistor
(Shockley, Bardeen, and Brattain)

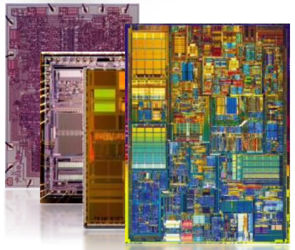
2000



1958: Integrated Circuit
(Kilby)



1971: Microprocessor
(Hoff, Faggin, Mazor)



1971- Exponential growth
(Moore, 1965)

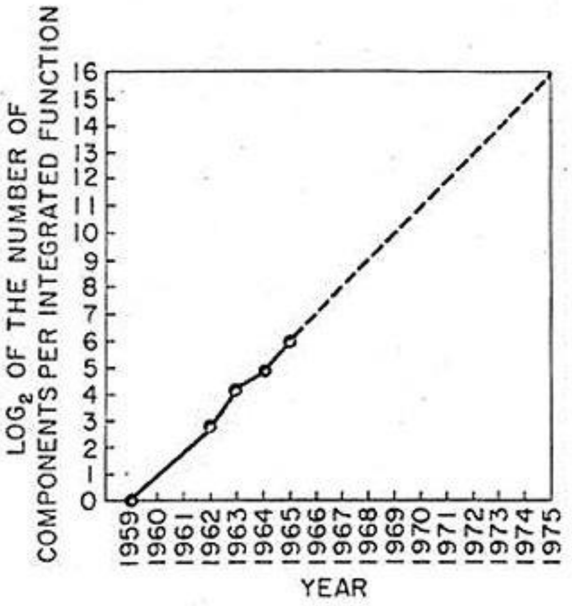
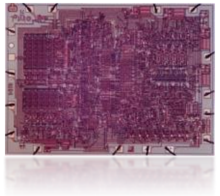


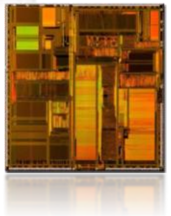
Fig. 2 Number of components per integrated function for minimum cost per component extrapolated vs time.

History lesson: development of the microprocessor 2/2

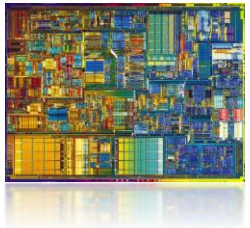


1971: 4004,
2300 trans, 740 KHz

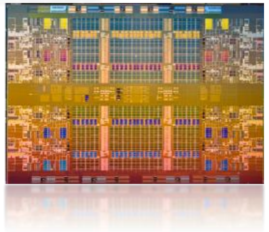
1982: 80286,
134 thousand trans, 8 MHz



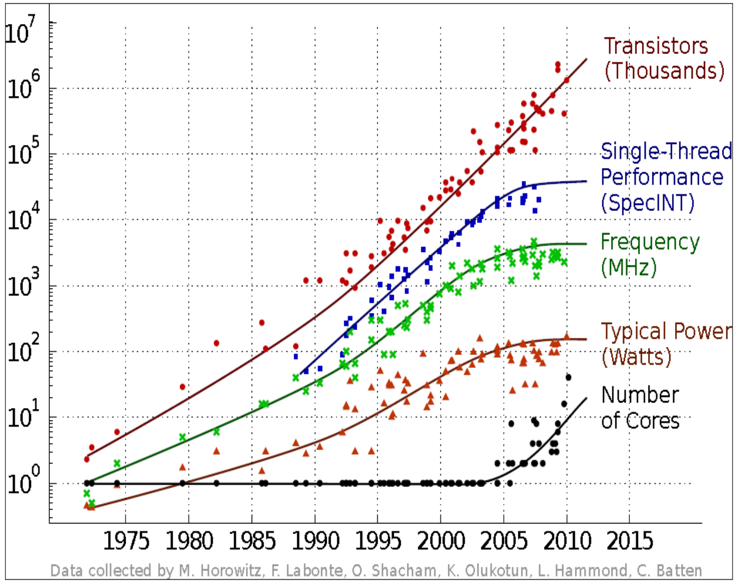
1993: Pentium P5,
1.18 mill. trans, 66 MHz



2000: Pentium 4,
42 mill. trans, 1.5 GHz



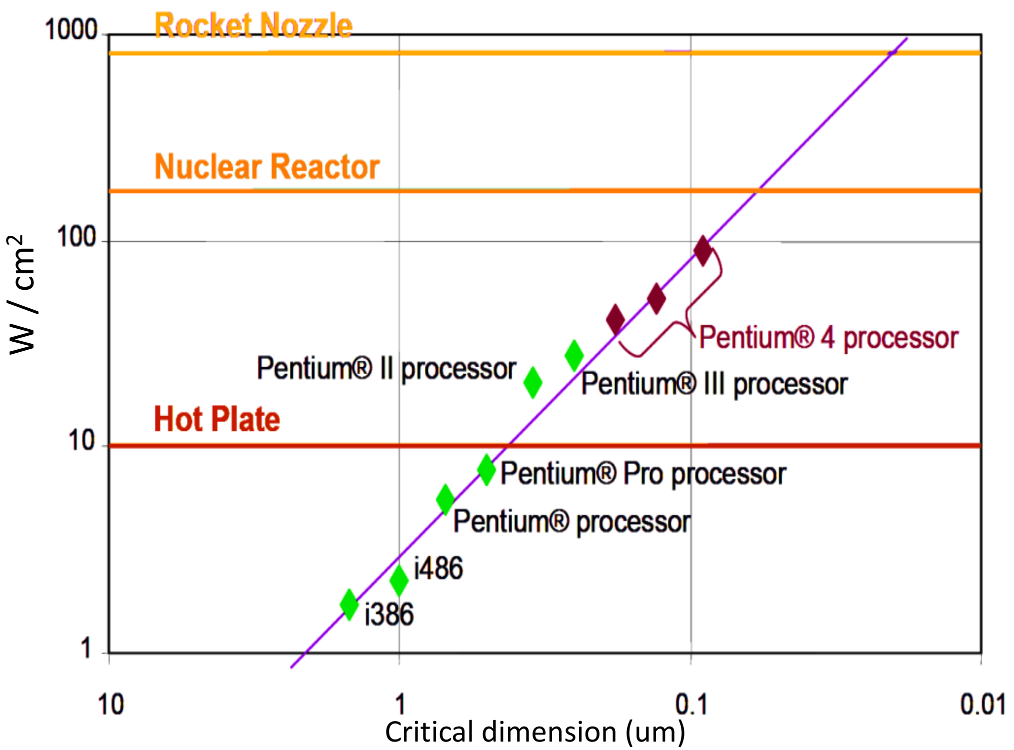
2010: Nehalem
2.3 bill. Trans, 8 cores, 2.66 GHz



SINTEF

Power density in CPUs

- Heat density approaching that of nuclear reactor core: Power wall
- Traditional cooling solutions (heat sink + fan) insufficient
- Industry solution: multi-core and parallelism!

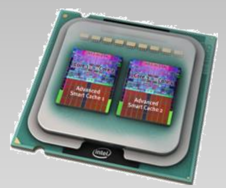


Graph taken from G. Taylor, “Energy Efficient Circuit Design and the Future of Power Delivery” EPEPS’09

Why Parallelism?

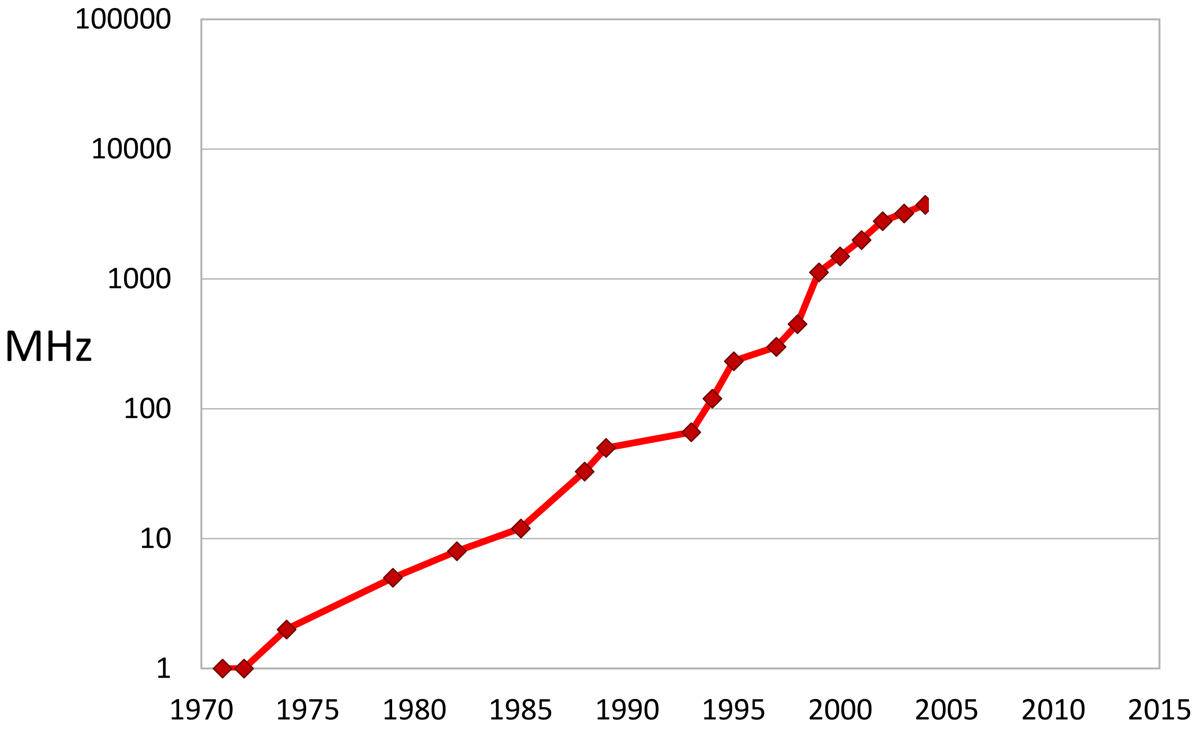
The power density of microprocessors
is proportional to the clock frequency cubed:[1]

$$P_d \propto f^3$$

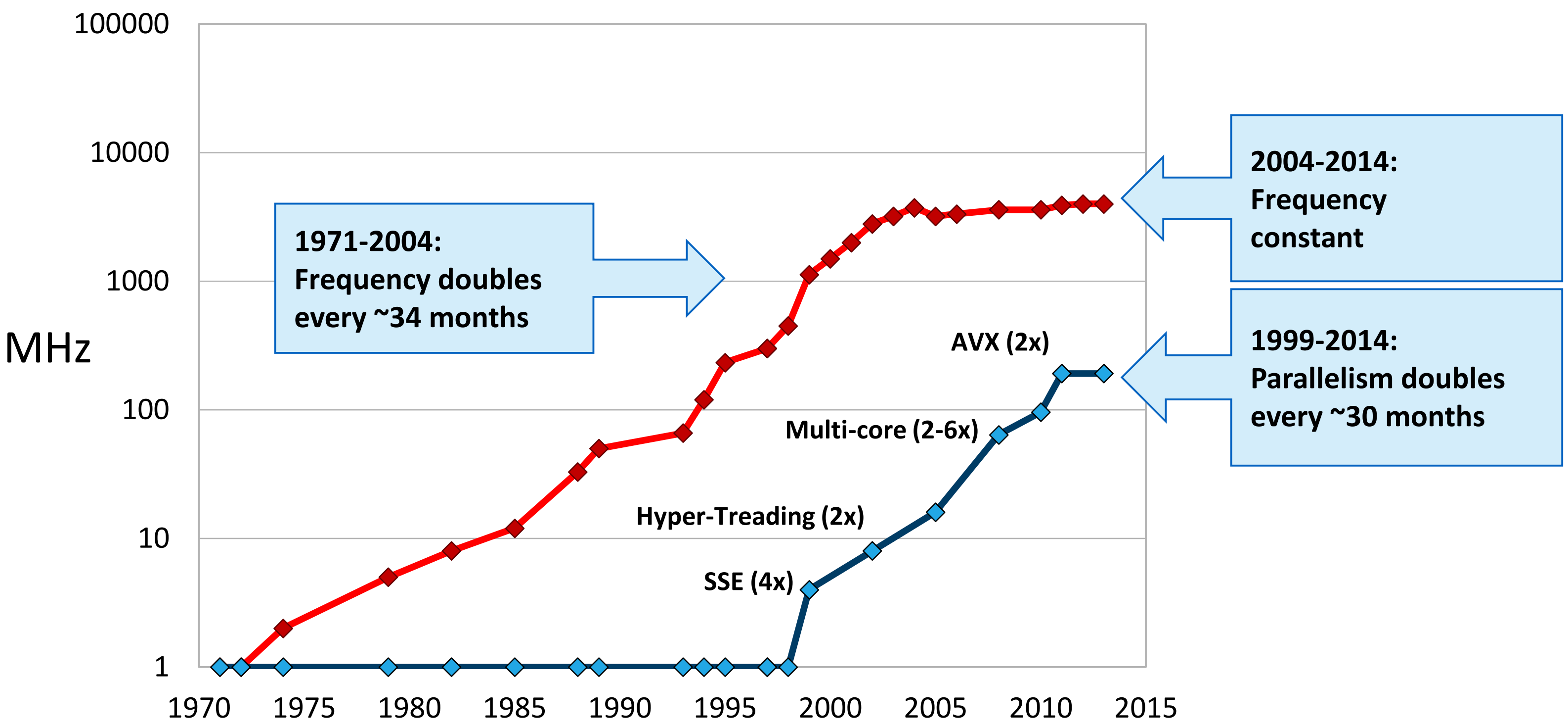
	Single-core	Dual-core
		
Frequency	100%	85%
Performance	100%	90% 90%
Power	100%	100%

[1] Brodtkorb et al. State-of-the-art in heterogeneous computing, 2010

Frequency scaling



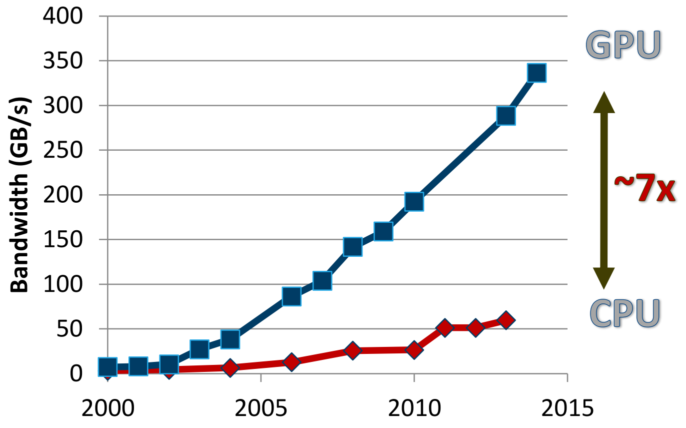
Frequency scaling

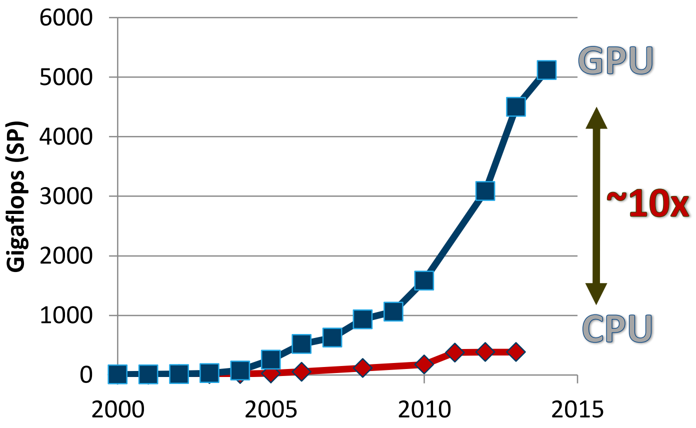


- 1970-2004: Frequency doubles every 34 months (Moore's law for performance)
- 1999-2014: Parallelism doubles every 30 months

Massive Parallelism: The Graphics Processing Unit

- Up-to <u>5760</u> floating point operations in parallel!
- 5-10 times as power efficient as CPUs!

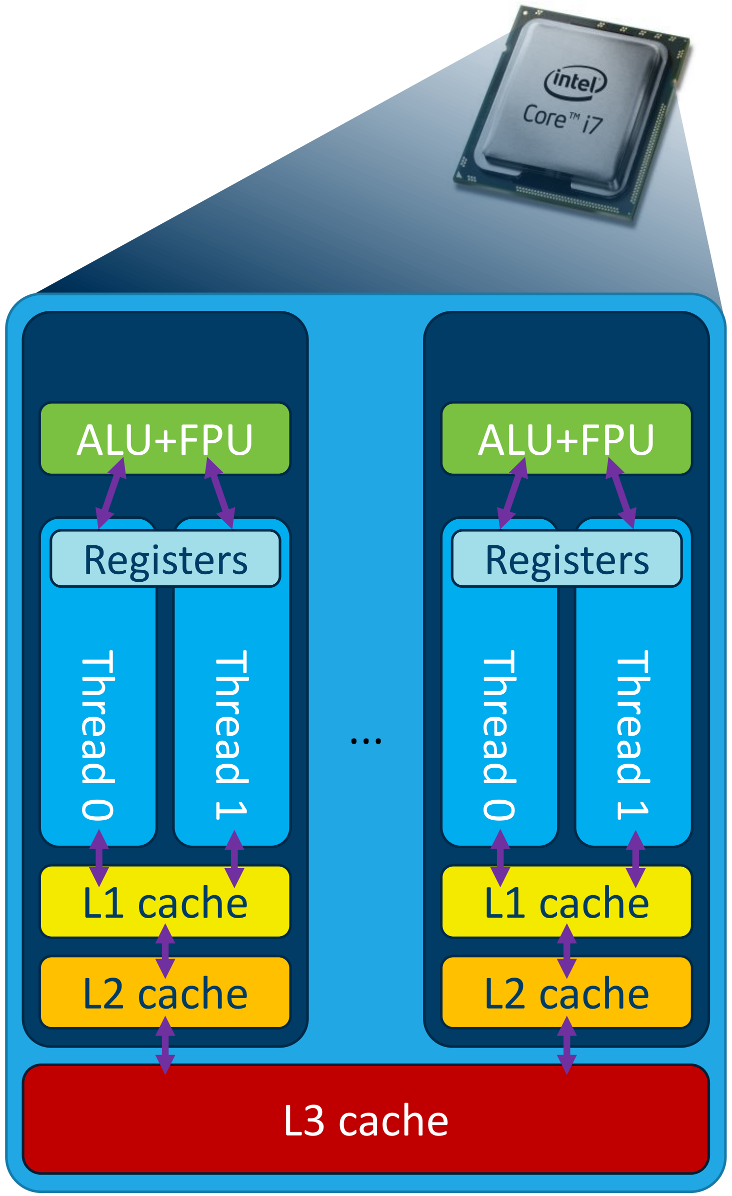




Multi-core CPU architecture

- A single core
  - L1 and L2 caches
  - 8-wide SIMD units (AVX, single precision)
  - 2-way Hyper-threading (<u>hardware</u> threads)
    When thread 0 is waiting for data,
    thread 1 is given access to SIMD units
  - Most transistors used for cache and logic

- Optimal number of FLOPS per clock cycle:
  - 8x: 8-way SIMD
  - 6x: 6 cores
  - 2x: Dual issue (fused mul-add / two ports)
  - <u>Sum: 96!</u>

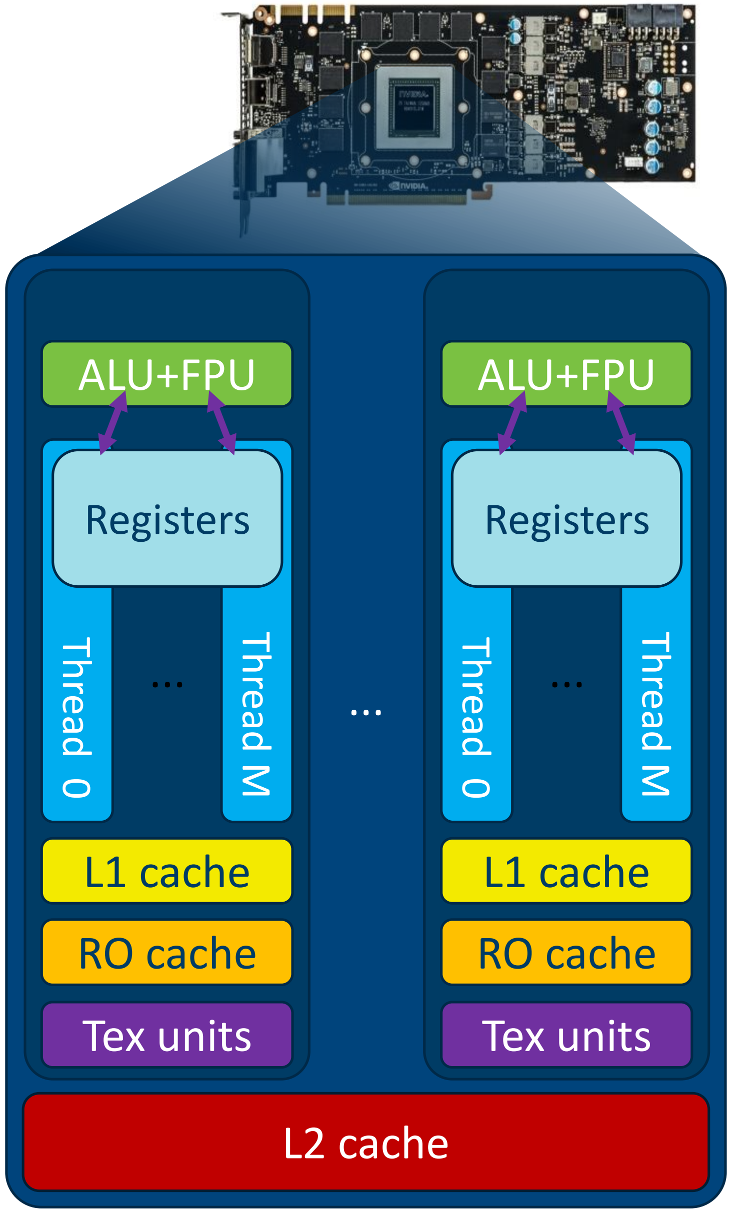


Simplified schematic of CPU design

Many-core GPU architecture

- A single core (Called streaming multiprocessor, SMX)
  - L1 cache, Read only cache, texture units
  - Six 32-wide SIMD units (192 total, single precision)
  - Up-to 64 warps simultaneously (hardware warps)
    Like hyper-threading, but a warp is 32-wide SIMD
  - Most transistors used for floating point operations

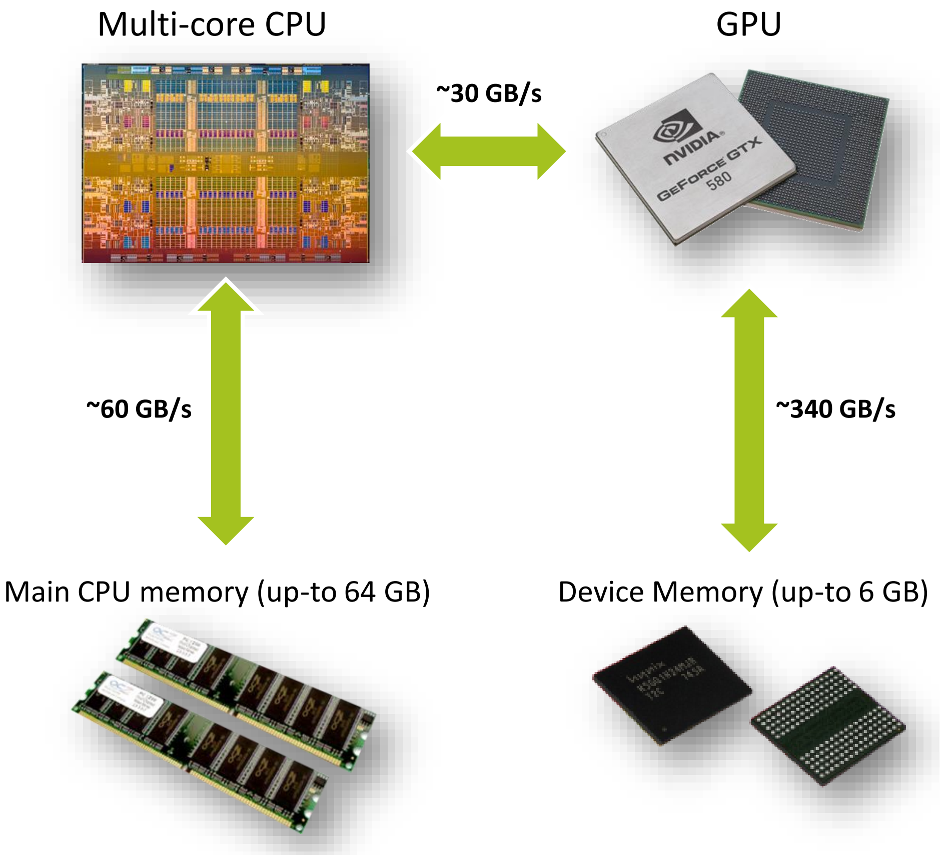
- Optimal number of FLOPS per clock cycle:
  - 32x: 32-way SIMD
  - 2x: Fused multiply add
  - 6x: Six SIMD units per core
  - 15x: 15 cores
  - Sum: 5760!



Simplified schematic of GPU design

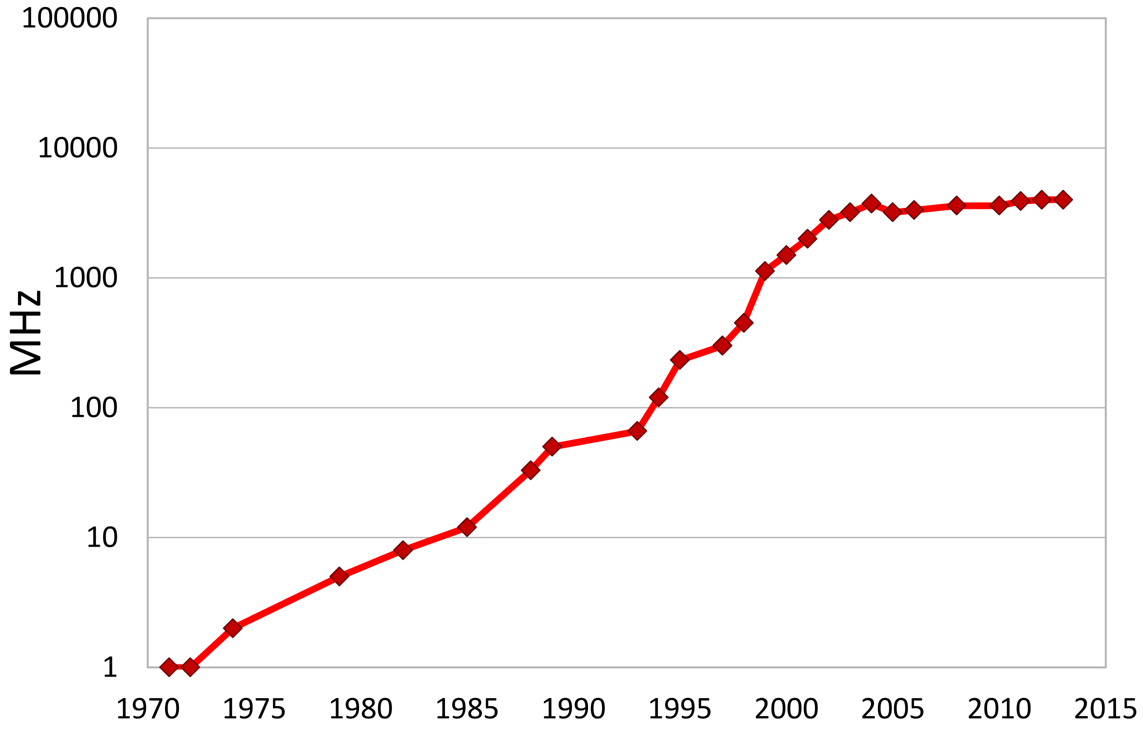
Heterogeneous Architectures

- Discrete GPUs are connected to the CPU via the PCI-express bus
  - Slow: 15.75 GB/s each direction
  - On-chip GPUs use main memory as graphics memory
- Device memory is limited but fast
  - Typically up-to 6 GB
  - Up-to 340 GB/s!
  - Fixed size, and cannot be expanded with new dimm's (like CPUs)



Parallell algorithm design

The beach law



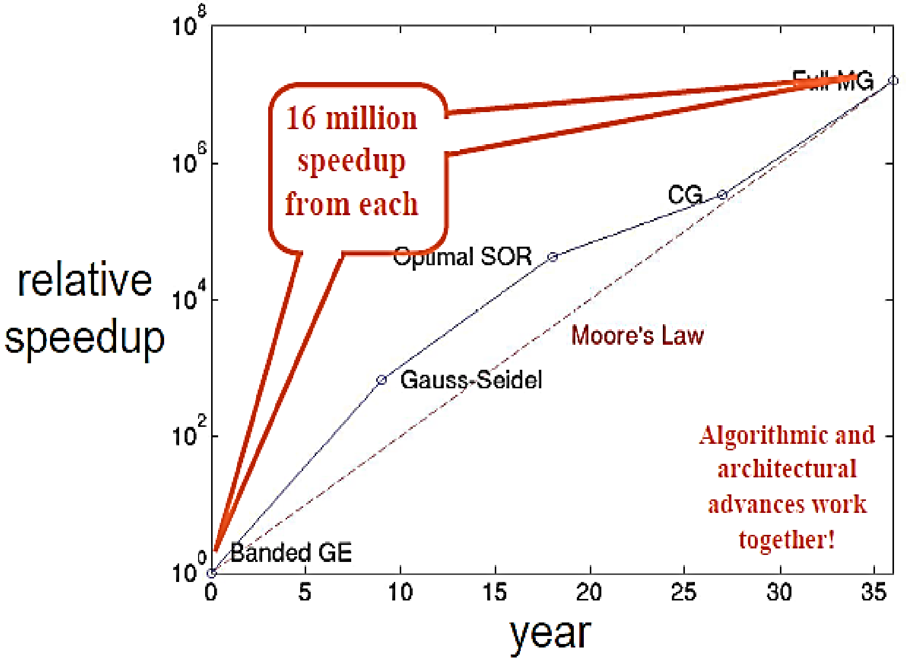
Performance doubles every 18 months

– go to the beach and relax

SINTEF

Algorithm design and computer architecture

- The key to increasing performance, is to consider the full algorithm and architecture interaction.
- A good knowledge of <u>both</u> the algorithm <u>and</u> the computer architecture is required.

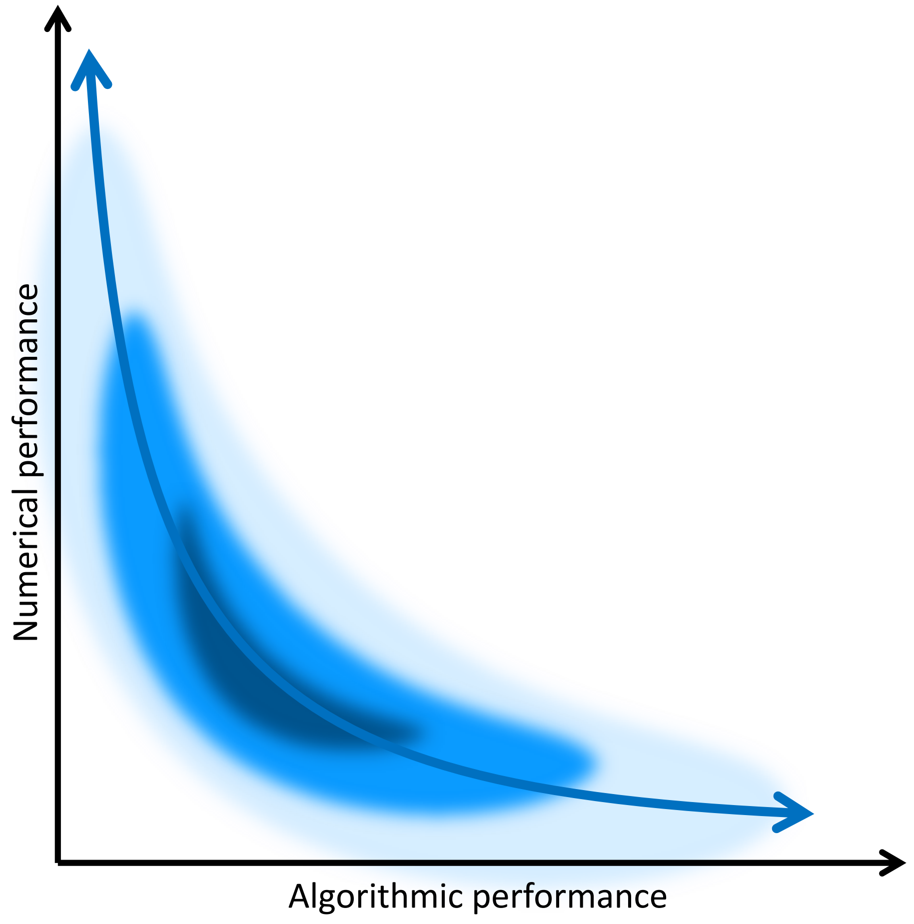


Graph from David Keyes, Scientific Discovery through Advanced Computing, Geilo Winter School, 2008

SINTEF

Algorithmic and numerical performance

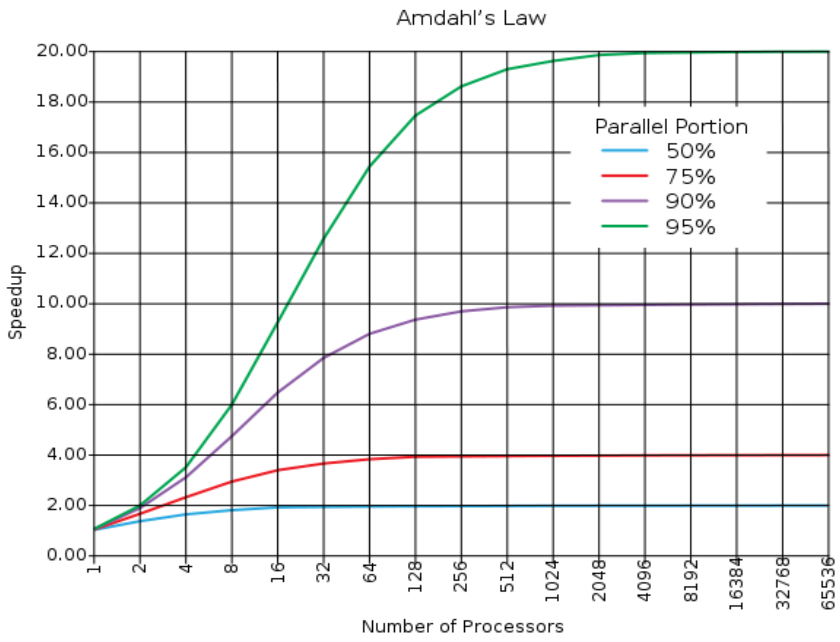
- Total performance is the product of algorithmic **and** numerical performance
- Your mileage may vary: algorithmic performance is highly problem dependent
- Many algorithms have low numerical performance
- Need to consider both the algorithm and the architecture for maximum performance



Limits on performance 1/4

- Most algorithms contains a mixture of work-loads:
  - Some serial parts
  - Some task and / or data parallel parts

- <u>Amdahl's law:</u>
  - There is a limit to speedup offered by parallelism
  - Serial parts become the bottleneck for a massively parallel architecture!
  - Example: 5% of code is serial: maximum speedup is 20 times!



$$S(N) = \frac{1}{(1-P) + \frac{P}{N}}$$

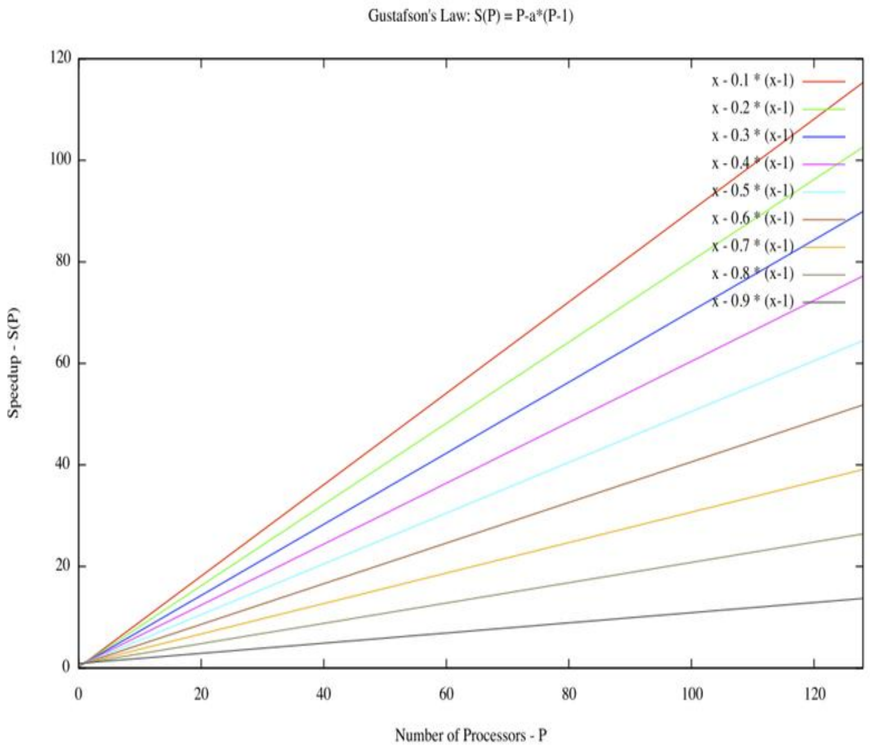
S: Speedup
P: Parallel portion of code
N: Number of processors

Graph from Wikipedia, user Daniels220, CC-BY-SA 3.0

SINTEF

Limits on performance 2/4

- Gustafson's law:
  - If you cannot reduce serial parts of algorithm, make the parallel portion dominate the execution time
  - Essentially: solve a bigger problem!



$$S(P) = P - \alpha \cdot (P - 1).$$

S: Speedup
P: Number of processors
α: Serial portion of code

Graph from Wikipedia, user Peahihawaii, CC-BY-SA 3.0

Limits on performance 3/4

- Moving data has become the major bottleneck in computing.
- Downloading 1GB from Japan to Switzerland consumes roughly the energy of 1 charcoal briquette[1].
- A FLOP costs less than moving one byte[2].
- Key insight: flops are free, moving data is expensive

[1] Energy content charcoal: 10 MJ / kg, kWh per GB: 0.2 (Coroama et al., 2013), Weight charcoal briquette: ~25 grams

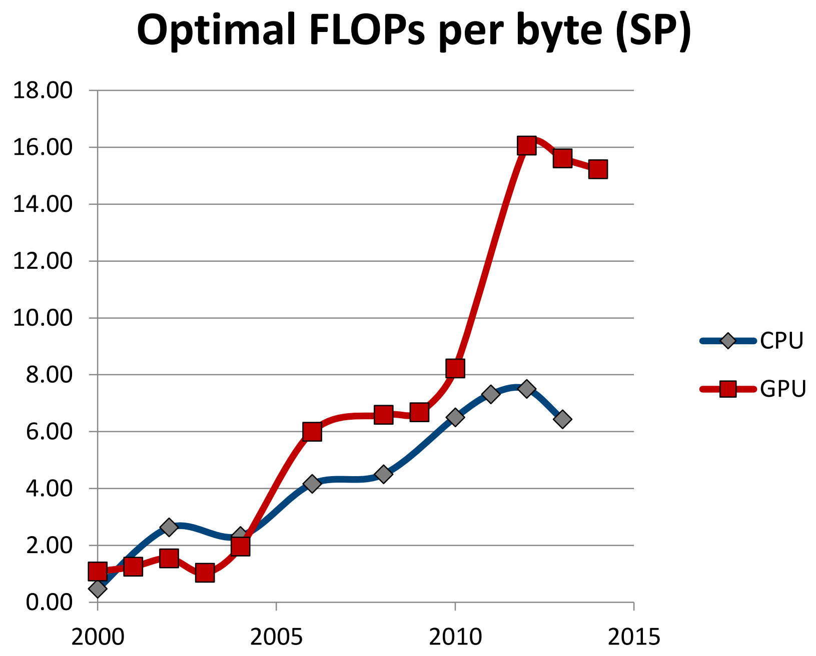
[2]Simon Horst, Why we need Exascale, and why we won't get there by 2020, 2014

Limits on performance 4/4

- A single precision number is four bytes
  - You must perform <u>over 60 operations</u> for each float read on a GPU!
  - Over 25 operations on a CPU!

- This groups algorithms into two classes:
  - Memory bound
    Example: Matrix multiplication
  - Compute bound
    Example: Computing π

- The third limiting factor is latencies
  - Waiting for data
  - Waiting for floating point units
  - Waiting for ...



Conservation laws

Conservation laws

- Conserve – synonyms: keep up, maintain, preserve, save
- Examples: preserve amount of water, preserve total energy in a system, …
- Can be formulated as partial differential equations

SINTEF

Example: The linear wave equation

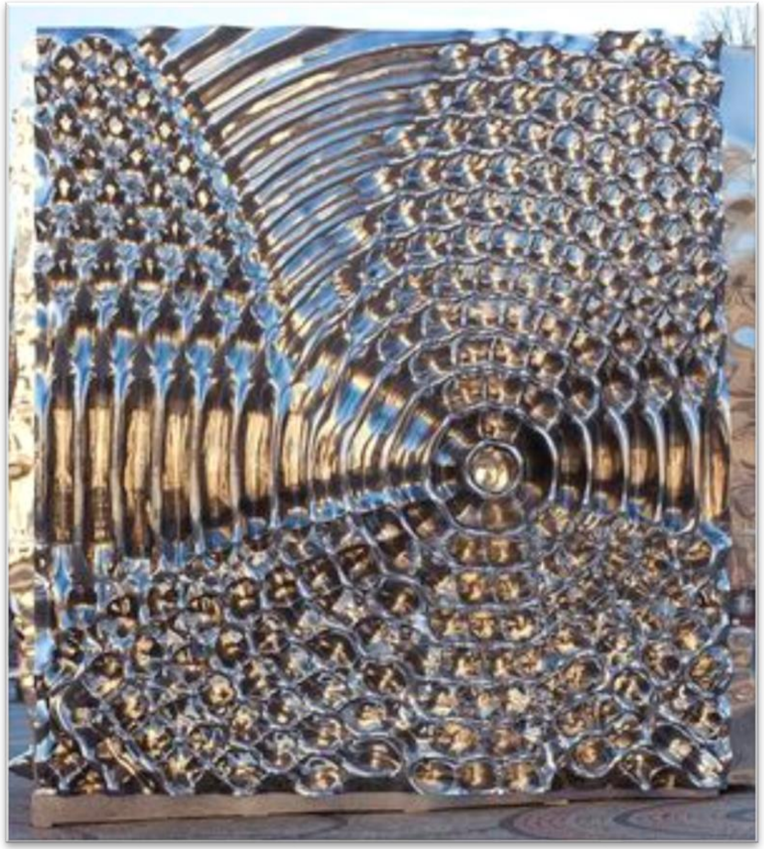
$$\frac{\partial^2 u}{\partial t^2} = c^2 \nabla^2 u$$

- Can describes vibration of string (in 1D)

- u is the deflection of the string
- c is a material property (related to wave propagation speed)

Violin image CC0

Example: The 2D wave equation



$$\frac{\partial^2 u}{\partial t^2} = c\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right]$$

$$\frac{1}{\Delta t^2}(u_{i,j}^{n+1} - 2u_{i,j}^{n} + u_{i,j}^{n-1})$$
$$= \frac{c}{\Delta x^2}(u_{i-1,j}^{n} - 2u_{i,j}^{n} + u_{i+1,j}^{n}) + \frac{c}{\Delta y^2}(u_{i,j-1}^{n} - 2u_{i,j}^{n} + u_{i,j+1}^{n})$$

```
u2[center] = 2.0f*u1[center] - u0[center]
    + kappa*dt/(dx*dx) * (u1[west] - 2*u1[center] + u1[east])
    + kappa*dt/(dy*dy) * (u1[south] - 2*u1[center] + u1[north]);
```

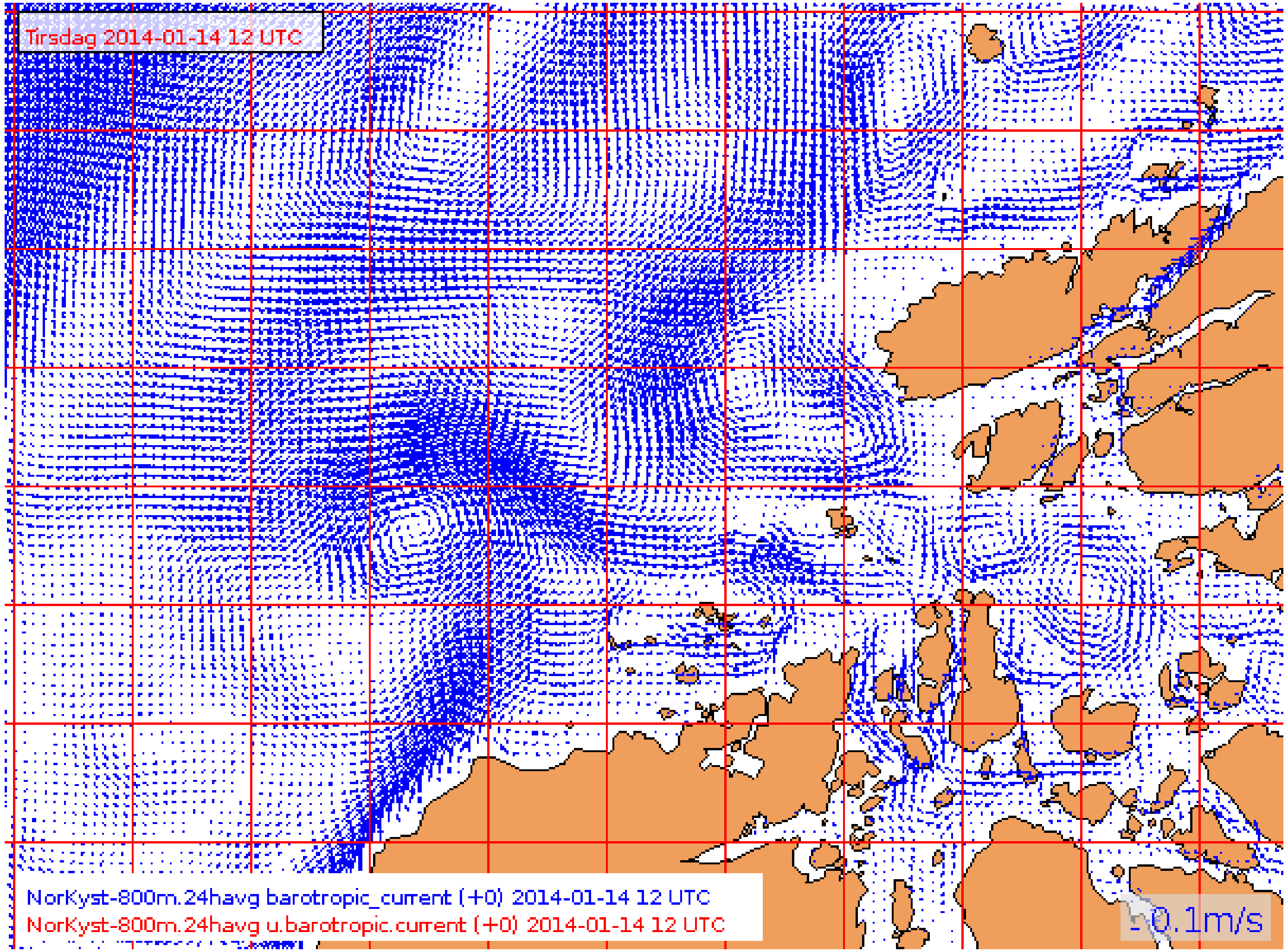
SINTEF

Høy C, ©Bård Breivik/BONO. Foto: Terje Heiestad. UiO

Using conservation laws in real life

Th

Problem statement



Solution strategy

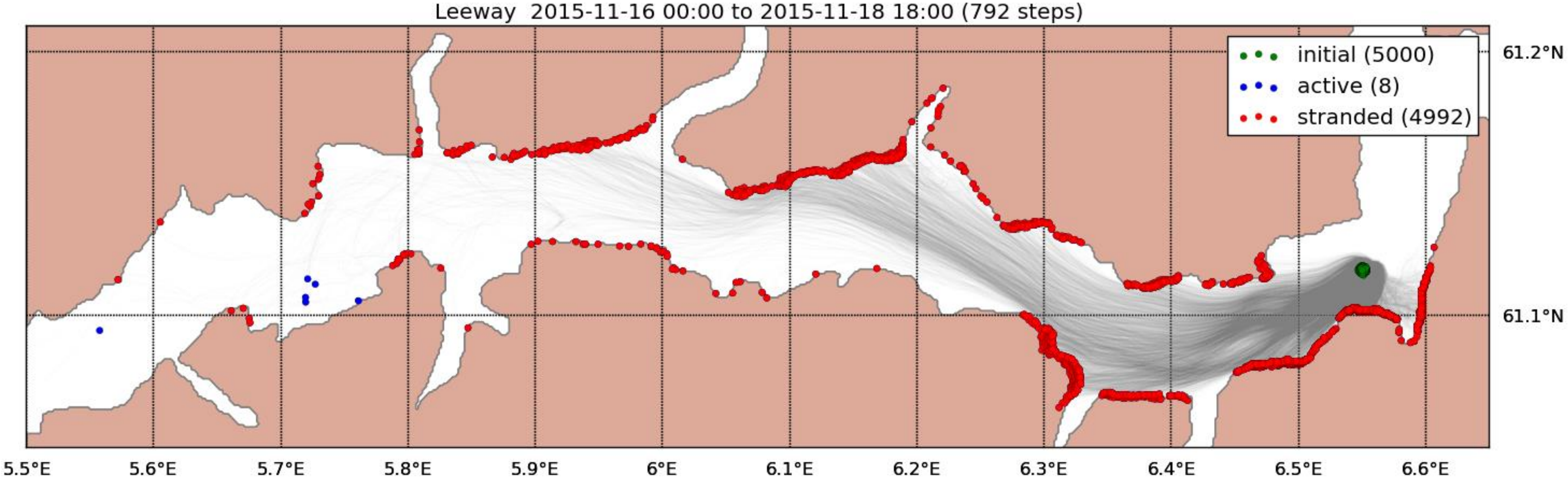
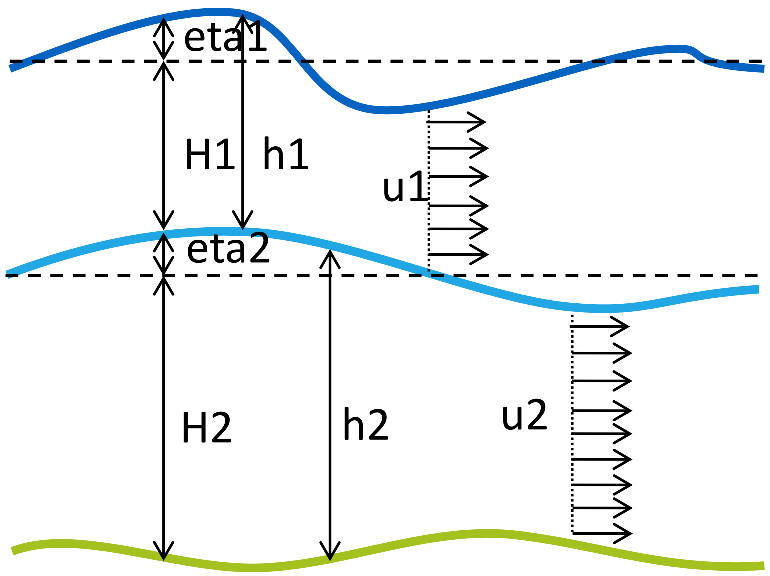


Image from Opendrift, Norwegian Meteorological Institute, Knut-Frode Dagestad.

SINTEF

2-layer non-linear scheme



- 1 layer model extendible to more layers
  - Ocean can be modeled as a stratisfied medium with multiple homogeneous layers
- Multiple layers enables baroclinic response from model

1 layer scheme, non-linear FD

$$\eta_{jk}^{n+1} = \eta_{jk}^{n-1} - \frac{2\Delta t}{\Delta x}\left(U_{jk}^{n} - U_{j-1k}^{n}\right) - \frac{2\Delta t}{\Delta y}\left(V_{jk}^{n} - V_{jk-1}^{n}\right),$$

$$V_{jk}^{n+1} = \frac{1}{C_{jk}^{y}}\left[V_{jk}^{n-1} + 2\Delta t\left(-f\overline{U}_{jk}^{n} + \frac{N_{jk}^{y}}{\Delta y} + \frac{P_{jk}^{y} + \hat{P}_{jk}^{y}}{\Delta y} + Y_{jk}^{n+1} + AE_{jk}^{y}\right)\right]$$

$$C_{jk}^{y} = 1 + \frac{2R\Delta t}{H_{jk}^{y}} + \frac{2A\Delta t(\Delta x^2 + \Delta y^2)}{\Delta x^2 \Delta y^2},$$

$$N_{jk}^{y} = \frac{1}{4}\left\{\frac{\left(V_{jk+1}^{n} + V_{jk}^{n}\right)^2}{H_{jk+1}^{n} + \eta_{jk+1}^{n}} - \frac{\left(V_{jk}^{n} + V_{jk-1}^{n}\right)^2}{H_{jk}^{n} + \eta_{jk}^{n}} + \frac{\Delta y}{\Delta x}\left[\frac{(U_{jk+1}^{n} + U_{jk}^{n})(V_{j+1k}^{n} + V_{jk}^{n})}{\overline{H}_{jk}^{n} + \overline{\eta}_{jk}^{n}} - \frac{(U_{j-1k+1}^{n} + U_{j-1k}^{n})(V_{jk}^{n} + V_{j-1k}^{n})}{\overline{H}_{jk-1}^{n} + \overline{\eta}_{jk-1}^{n}}\right]\right\},$$

$$P_{jk}^{y} = gH_{jk}^{y}\left(\eta_{jk+1}^{n} - \eta_{jk}^{n}\right), \quad \hat{P}_{jk}^{y} = \frac{1}{2}\left[(\eta_{jk+1}^{n})^2 - (\eta_{jk}^{n})^2\right],$$

$$E_{jk}^{y} = \frac{1}{\Delta x^2}\left(V_{j+1k}^{n} - V_{jk}^{n-1} + V_{j-1k}^{n}\right) + \frac{1}{\Delta y^2}\left(V_{jk+1}^{n} - V_{jk}^{n-1} + V_{jk-1}^{n}\right).$$

$$U_{jk}^{n+1} = \frac{1}{C_{jk}^{x}}\left[U_{jk}^{n-1} + 2\Delta t\left(f\overline{V}_{jk}^{n} + \frac{N_{jk}^{x}}{\Delta x} + \frac{P_{jk}^{x} + \hat{P}_{jk}^{x}}{\Delta x} + X_{jk}^{n+1} + AE_{jk}^{x}\right)\right],$$

$$C_{jk}^{x} = 1 + \frac{2R\Delta t}{H_{jk}^{x}} + \frac{2A\Delta t(\Delta x^2 + \Delta y^2)}{\Delta x^2 \Delta y^2},$$

$$N_{jk}^{x} = \frac{1}{4}\left\{\frac{\left(U_{j+1k}^{n} + U_{jk}^{n}\right)^2}{H_{j+1k} + \eta_{j+1k}^{n}} - \frac{\left(U_{jk}^{n} + U_{j-1k}^{n}\right)^2}{H_{jk} + \eta_{jk}^{n}} + \frac{\Delta x}{\Delta y}\left[\frac{(U_{jk+1}^{n} + U_{jk}^{n})(V_{j+1k}^{n} + V_{jk}^{n})}{\overline{H}_{jk}^{n} + \overline{\eta}_{jk}^{n}} - \frac{(U_{jk}^{n} + U_{jk-1}^{n})(V_{j+1k-1}^{n} + V_{jk-1}^{n})}{\overline{H}_{jk-1}^{n} + \overline{\eta}_{jk-1}^{n}}\right]\right\}, \quad (23)$$

$$P_{jk}^{x} = gH_{jk}^{x}\left(\eta_{j+1k}^{n} - \eta_{jk}^{n}\right), \quad \hat{P}_{jk}^{x} = \frac{1}{2}\left[(\eta_{j+1k}^{n})^2 - (\eta_{jk}^{n})^2\right],$$

$$E_{jk}^{x} = \frac{1}{\Delta x^2}\left(U_{j+1k}^{n} - U_{jk}^{n-1} + U_{j-1k}^{n}\right) + \frac{1}{\Delta y^2}\left(U_{jk+1}^{n} - U_{jk}^{n-1} + U_{jk-1}^{n}\right),$$

SINTEF