Fundamentals of a numerical cloud computing for applied sciences – Preparing cloud computing for 'Simulation as a Service'

Markus Geveler, Stefan Turek

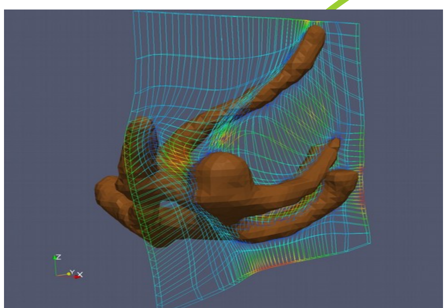
Cloud consultation workshop, Brussels, November 7th 2016

markus.geveler@math.tu-dortmund.de

Simulation as a service (SIMaaS) : our vision

Special purpose cloud meets special purpose software

→ simulation 'problem'



User: SME, scientist

→ problem-oriented
allocation: HW, SW
→ options?
→ target metrics:
→ time
→ cost (money, energy)

SIMaaS special purpose cloud
→ accelerators! We need their on-chip memory bandwidth
→ maybe @ higher unit cost but with low cost to solution
→ toolchain focus on end-user problems, not generality

→ vertical integration into scientific workflow

Example compute node: **AWS P2 instance:**

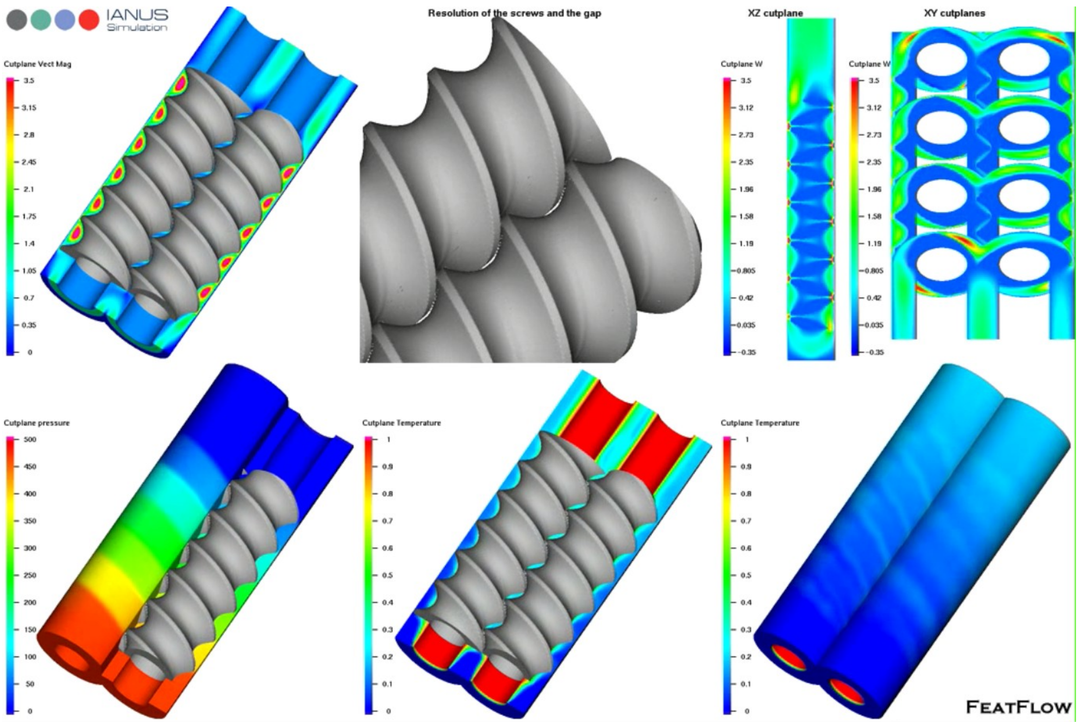
→ **2x Broadwell Xeon E5v4, 16 cores each**
→ **8x NVIDIA Tesla K80 dual GPU**
+high-end ethernet interconnect
This is already 'reality'...
Other IaaS providers with GPUs: **Microsoft, Penguin, Nimbix, …**

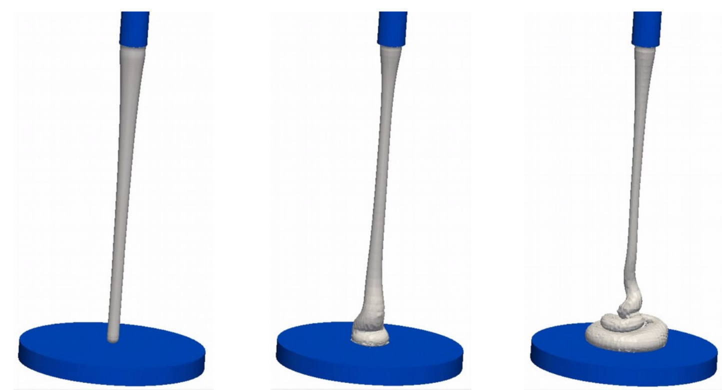
→ **Hardware turn-over dynamics: very fast,**
→ **vendor investment-prone**

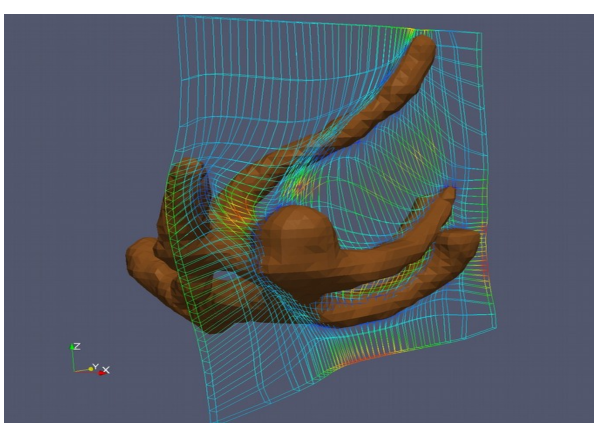
SIMaaS

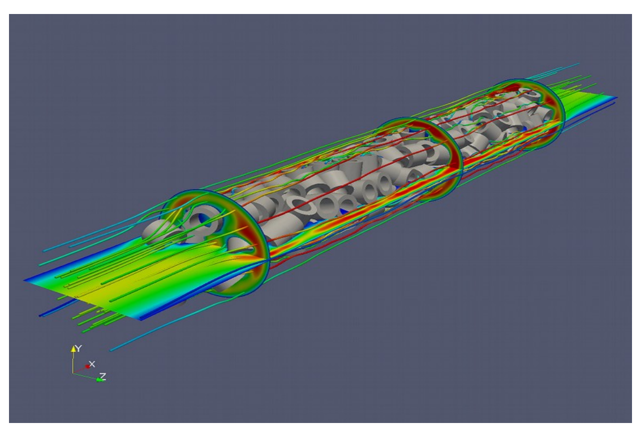
Simulation of technical flows

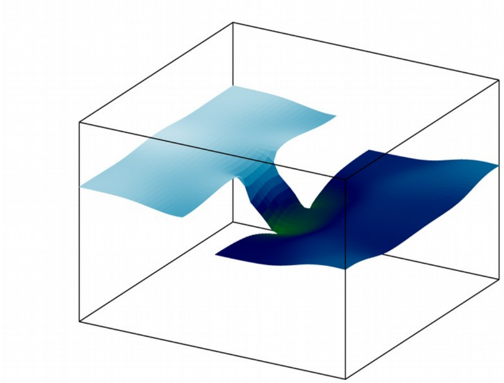
→ billions of globally coupled equations
→ 'big compute' not big data
→ but our 'compute' load is memory bandwidth-bound











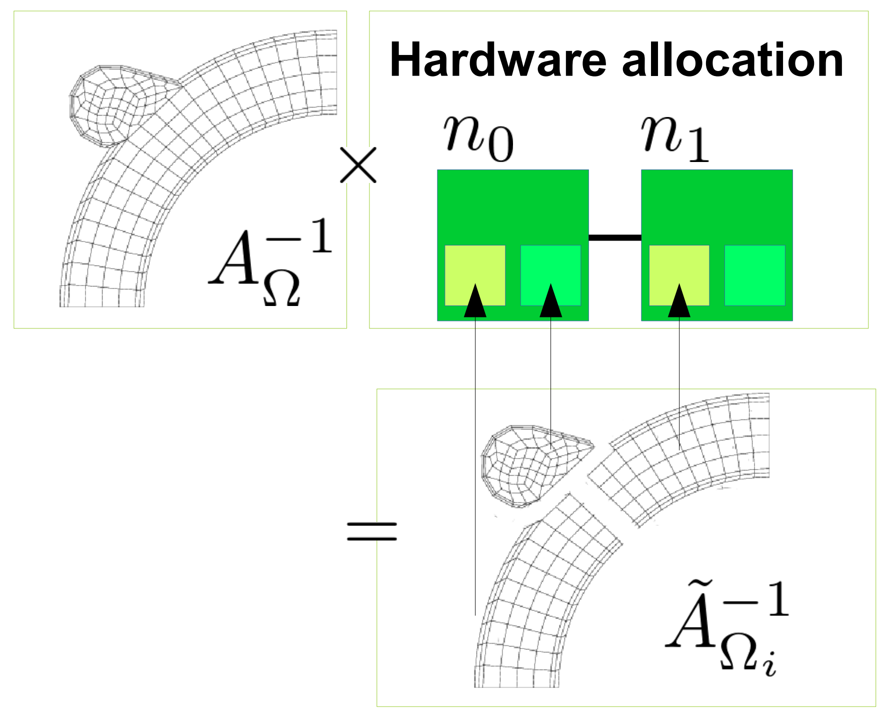
Baseline hypothesis

Numerical cloud computing imposes fundamental mathematical problems

→ numerically scalable asynchronous parallel solution of large systems of globally coupled equations **on 'a priori' unknown heterogenous hardware** platforms

→ challenges close to the 'exascale challenges' but not exactly the same

- → **domain decomposition**
- → **asynchronicity**
- → **loadbalancing**
- → **scheduling, mapping**
- → **autotuning**
- → **resilience, fault tolerance**



→ when the provider changes the hardware:
will you re-develop?

$$A_\Omega^{-1} \leftarrow \tilde{\sum_i} (A_{\Omega_i}^{-1} = R_i A_\Omega^{-1})$$

Current state of SIMaaS (in the EU)

SIMaaS research funding in the EU: Fortissimo 1 + 2

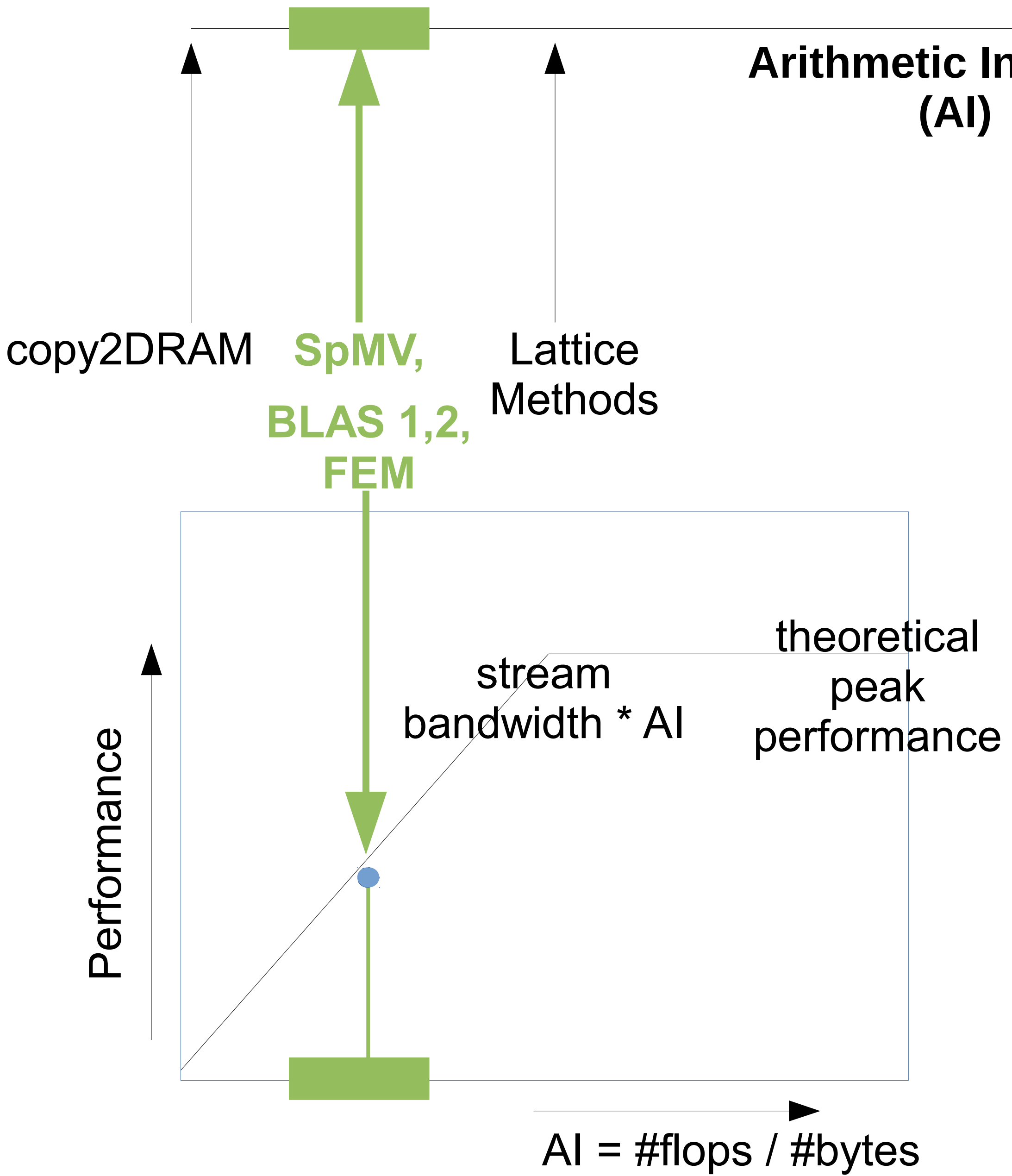
→ SME-driven 'showcases'
→ +domain specialist
→ +HPC center
→ also covering technical flow simulation
→ promising, **but:**

→ not covered: fundamental research (exascale challenges under the special constraints of cloud computing)

→ if we start to push forward like this, important aspects are not covered
→ SIMaaS will happen, but at **suboptimal efficiency, welfare**

(Numerical) cloud computing should be dealt with
→ from a theoretical point of view as well as
→ from a feasibility perspective

What we can expect from hardware



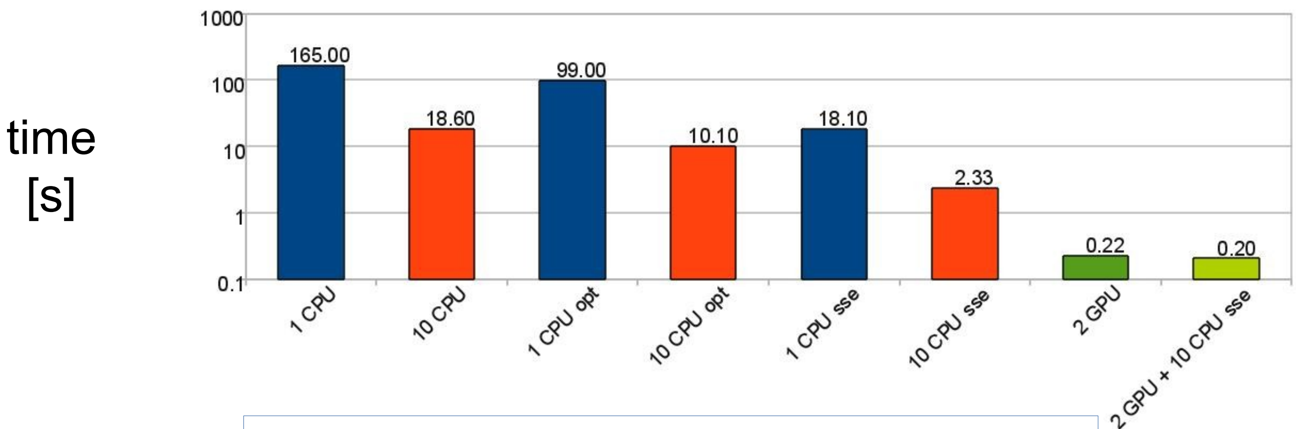
→ **high on-chip memory bandwidth needed**

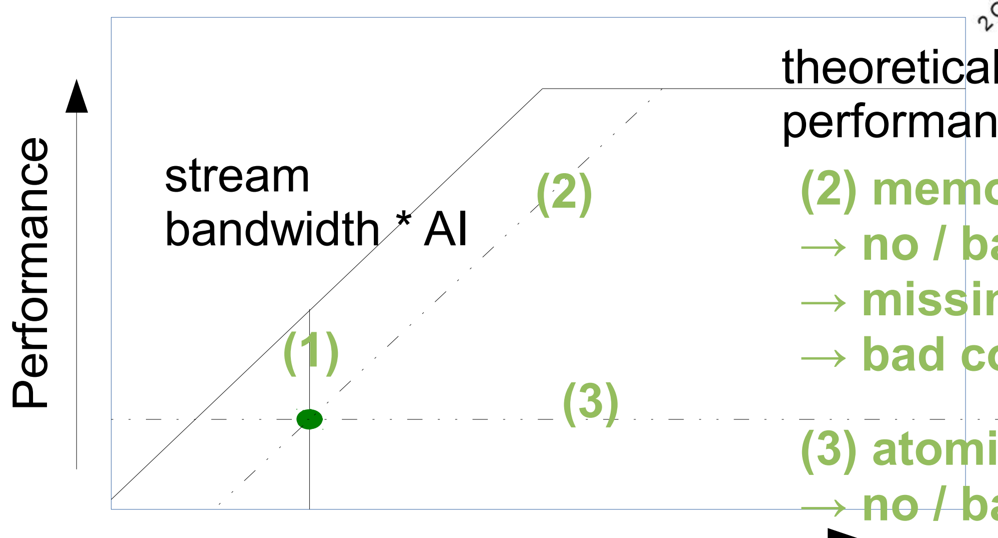
→ **low latency, high bandwidth interconnects needed**

Hardware-oriented numerics

Hardware efficiency: apply performance models until optimal

Hardware-centric perspective





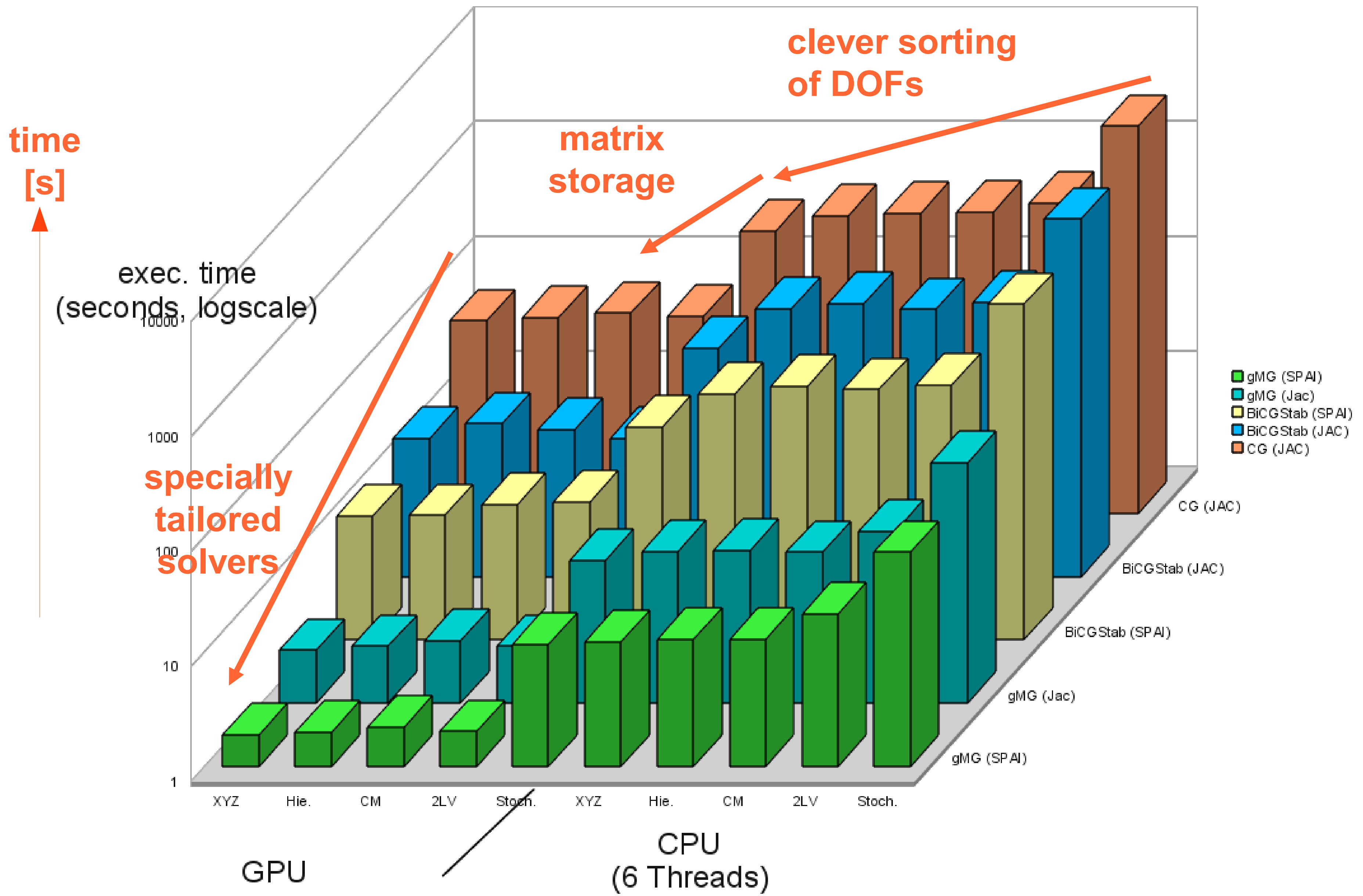
(2) memory access patterns:
→ no / bad software prefetching
→ missing NUMA adjustments
→ bad coalescence

(3) atomic operations, most likely:
→ no / bad SIMD
→ no FMA

(1) bad AI due to 'wrong' numerics or implementation

Hardware-oriented numerics

Numerical efficiency

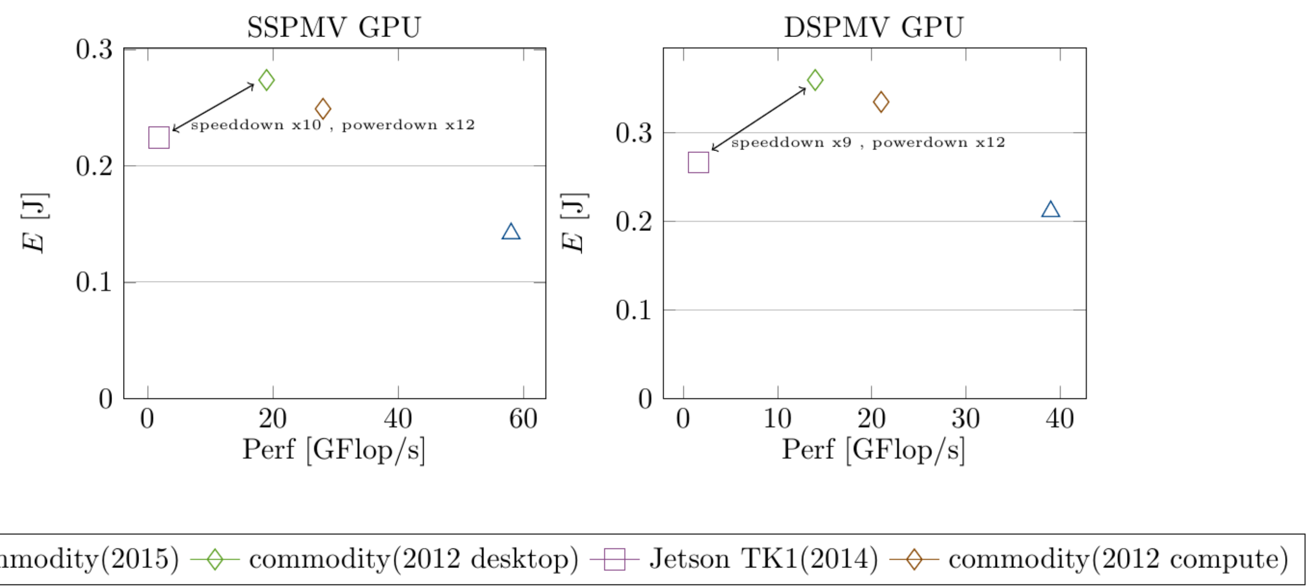


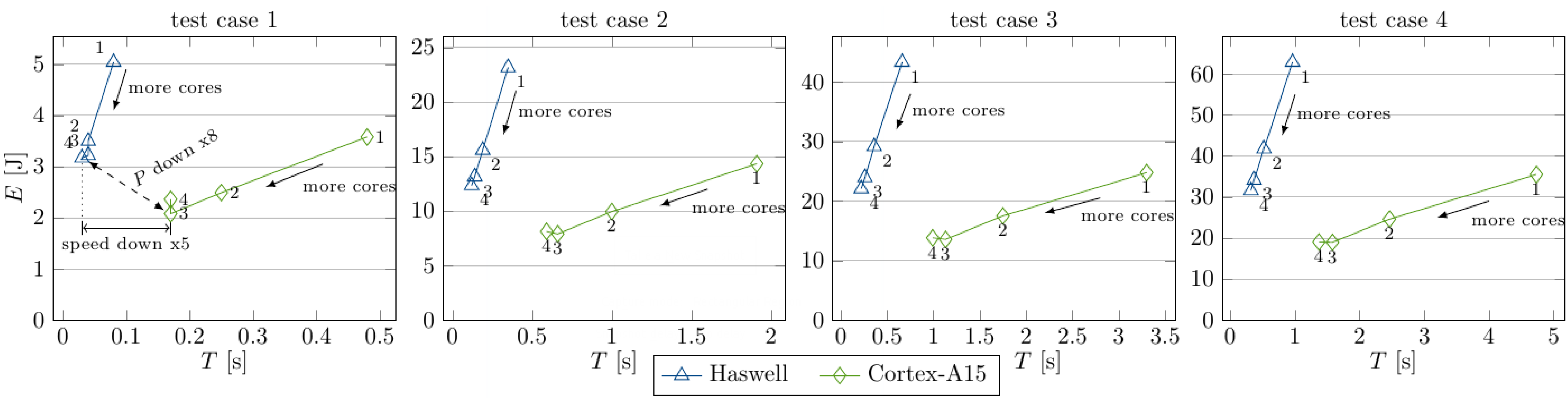
Hardware-oriented numerics

Energy efficiency

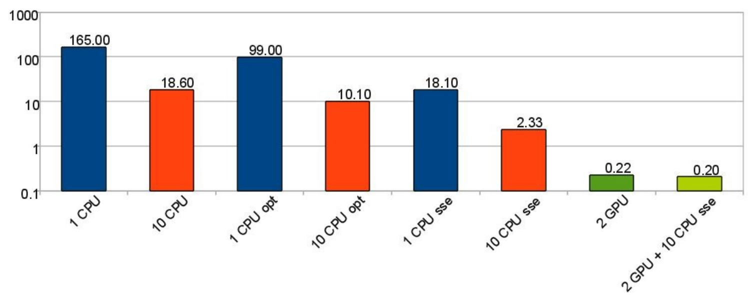
→ this is about the true cost of simulation

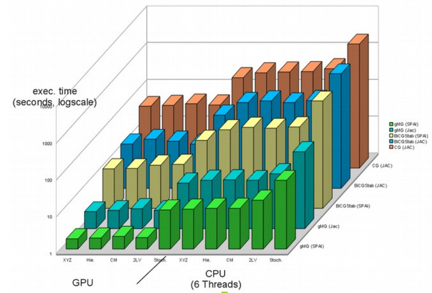
→ **for cloud providers!**

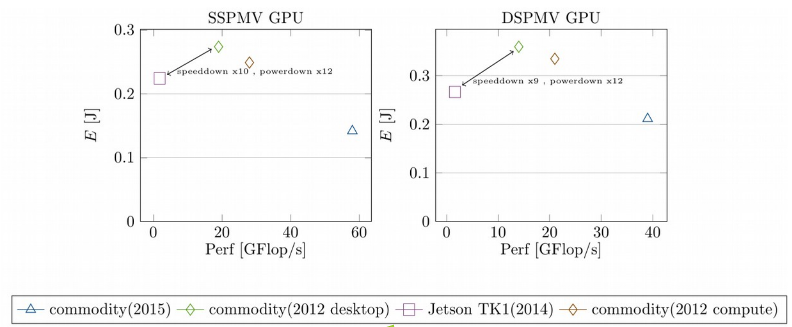




Hardware-oriented numerics







Hardware-oriented numerics :=

Enhance hardware- and
numerical efficiency simultaneously, use (the most) energy-efficient
hardware(-settings) available!

Attention: codependencies!

In-house HPC

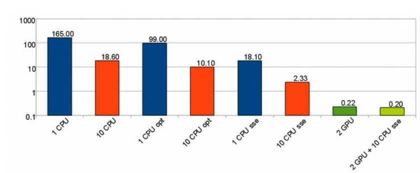
Performance is a function of hardware and code

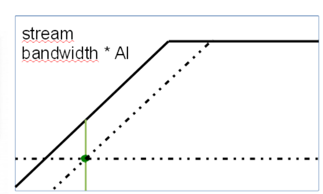
→ quality of hardware-oriented numerics decides production **efficiency**

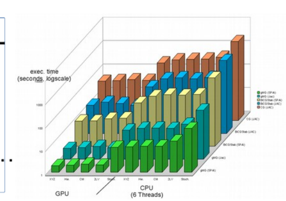
→ speedups of 1000 possible, most of it due to knowledge of HW details, 'in-house control'

in-house hardware

in-house code







Performance engineering

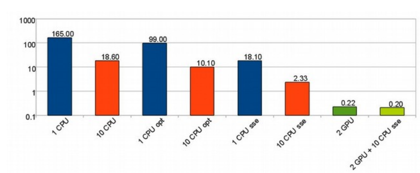
SIMaaS means hardware abstraction

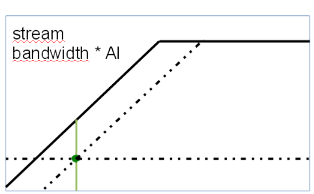
With cloud computing?

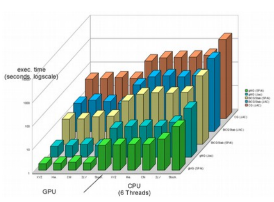
→ hardware no longer known 'a priori', heavily dependent on vendor investment
→ how integrate?
  → arbitration?
  → automation?
→ who makes this 'middleware'
→ **how can this even be done without knowledge about numerics?**

SIMaaS ?

in-house code







Performance Engineering

First steps into numerical cloud computing

Vertical integration into scientific workflow

Core

→ mathematical fundamentals: asynchronous domain decomposition type schemes
→ hardware-oriented development of numerical components
→ selection, scheduling, mapping of numerical components for a given SIMaaS allocation
→ autotuning simulation code parameters for selected components
→ loadbalancing for a maximum of node exploitation

Others

→ simulation end user-oriented development toolchain, 'sandbox mode'
→ security, licensing
→ reproducibility? Will the vendor provide the hardware again? Maybe years later?
→ 'build versus buy': at what scale should a university kick in? → performance/cost modelling!
→ ...

Proposal: from feasibility to efficiency

Solution: heavy investment in interdisciplinary research

Providers (aka cloud computing specialists) and
simulation end-users (aka applied sciences software specialists) have to learn

→ performance/cost modelling to gain control of what is going on
→ what can be automated?
→ what has to be (re-)developed, crafted from scratch (AVX kernel)?
→ how to integrate automation mechanisms into software (end user side)

Thank you

This work has been supported in part by the German Research Foundation (DFG)
through the Priority Program 1648
'Software for Exascale Computing'
(grant TU 102/48).