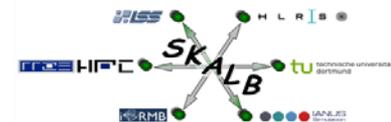


High-order Finite-Element Seismic Wave Propagation Modeling with MPI on a Large GPU Cluster

Dimitri Komatitsch, Dominik Göddeke, Gordon Erlebacher, David Michéa

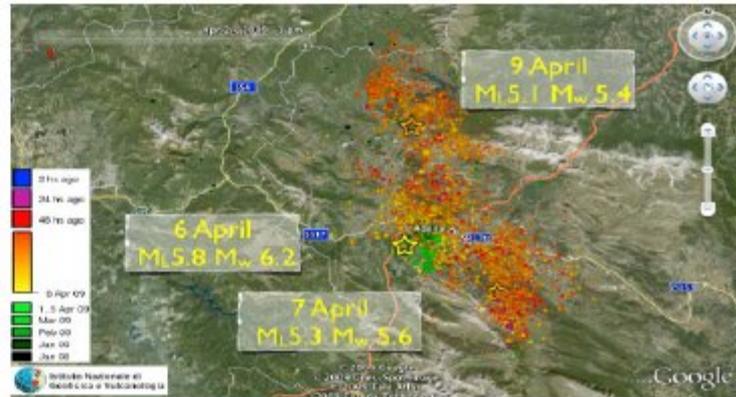
AGU Fall Meeting, IN17 High-Resolution Modeling in the Geosciences
using GPUs and Many-Core Architectures
San Francisco, USA, Dec. 5, 2011

Laboratory of Mechanics and Acoustics, CNRS, Marseille, France
Faculty of Mathematics, TU Dortmund, Germany
Department of Scientific Computing, Florida State University, Tallahassee, USA
Bureau de Recherches Géologiques et Minières, Orléans, France



- **Application domains**

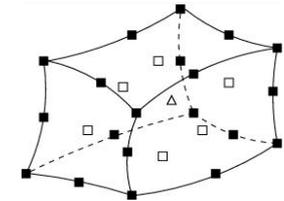
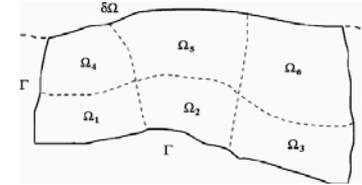
- Better understanding (imaging) of the Earth's interior: forward and adjoint/inverse problems in seismology
- Earthquakes in sedimentary basins
- At the scale of a region, a continent, or in the full Earth
- Running different aftershock scenarios after a large earthquake
- Active seismic acquisition experiments in the oil and gas industry
- Non-destructive testing, ultrasonics,...



- **Mathematical model**
 - Linear seismic wave equation for heterogeneous acoustic, elastic, viscoelastic and/or poro-elastic media
- **Main challenges:**
 - Discontinuities in the media (geological interfaces), often involving very high seismic frequencies
 - Honour topography and sedimentary layers in the model
- **Efficient numerical and (!) computational methods required**
 - Huge (really huge!) resulting discrete models

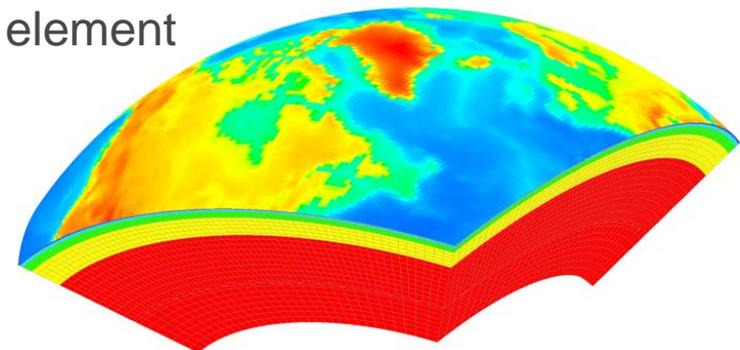
- **SEM basic principles**

- Mesh the domain with large, curvilinear hexahedral “spectral” elements to honour e.g. topography and interior discontinuities
- Approximate quantities in each element with high-order GLL interpolation
- Use GLL points to integrate weak form as well \Rightarrow strictly diagonal mass matrix, no large linear system to solve
- Single precision is always sufficient for SEM codes \Rightarrow faster, and uses twice less memory



- **Good trade-off between conflicting goals**

- “Hybrid” approach: Combines accuracy of pseudo-spectral methods with the geometric flexibility of finite element methods
- Alleviates their respective difficulties
- Relatively easy to parallelize



- **Resource requirements**

- SEM codes for seismic wave propagation easily fill up *any* given machine
- Want high resolution, good accuracy, high seismic frequencies,...
- $O(1K-10K)$ processors, $O(10-100)$ terabyte, $O(1-10)$ hours to complete
- Just for forward simulations (this talk), even more true for adjoint/inverse problems (current collaboration with Princeton, Basel and Zürich)
- Classical strong scaling via more nodes and bigger machines not a very useful option (not much left of the machine to scale with 😊)

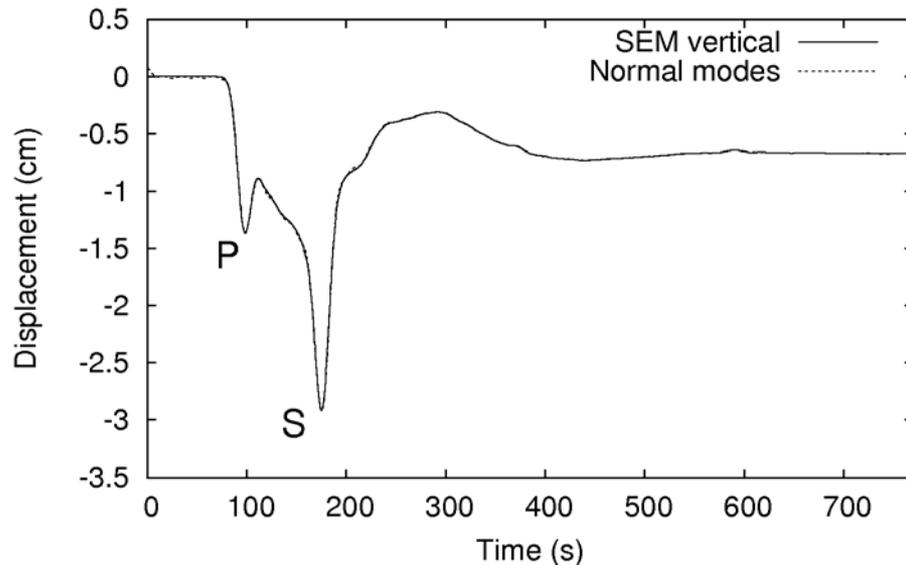
- **Enter the world of GPUs and accelerators**

- Offer strong scaling (=faster computations) within each node
- Up to 20x faster with “moderate” programming effort (CUDA,...)
- “Green computing”: FLOPS per watt is what will matter most in the next decade (in the race to Exaflops)

- **Pure CPU + MPI programming model will soon be over**
 - Higher performance and better energy efficiency only by increasing concurrency in hardware (multi-core, many-core)
 - Road to Exascale: Accelerator-based systems one of two possible avenues (see for instance the “International Exascale Software Report” and roadmap @ www.exascale.org)
 - Algorithmic and methodological perspective: Need to adapt to fine-grained massive parallelism, “MPI+X”
- **GPUs are current-generation representatives of this trend**
 - Lots of GPU-accelerated systems deployed
 - 3 of the 5 fastest supercomputers in the world (TOP500 list), TIER-1 and Tier-2 systems worldwide
 - Lots of improvements in usability: Proper tool-chains, debuggers and profilers, MPI directly between GPU memories, ...
 - Soon: OpenMP-like programming for all cores and GPUs in one node

Some Results

For more details, see our 2010 paper in „Journal of Computational Physics“



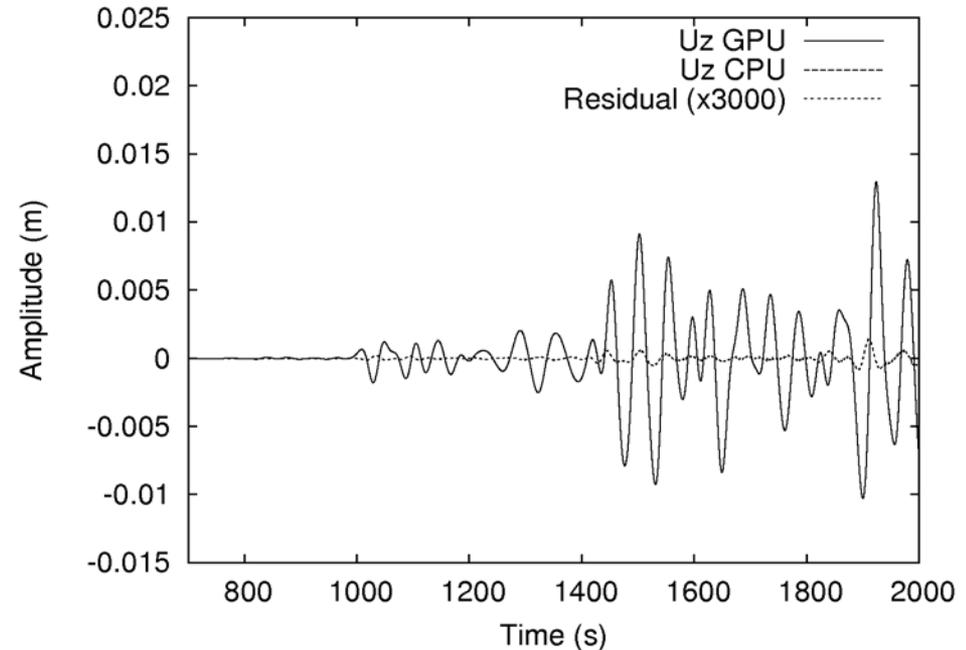
Bolivia, June 1994, Mw = 8.2

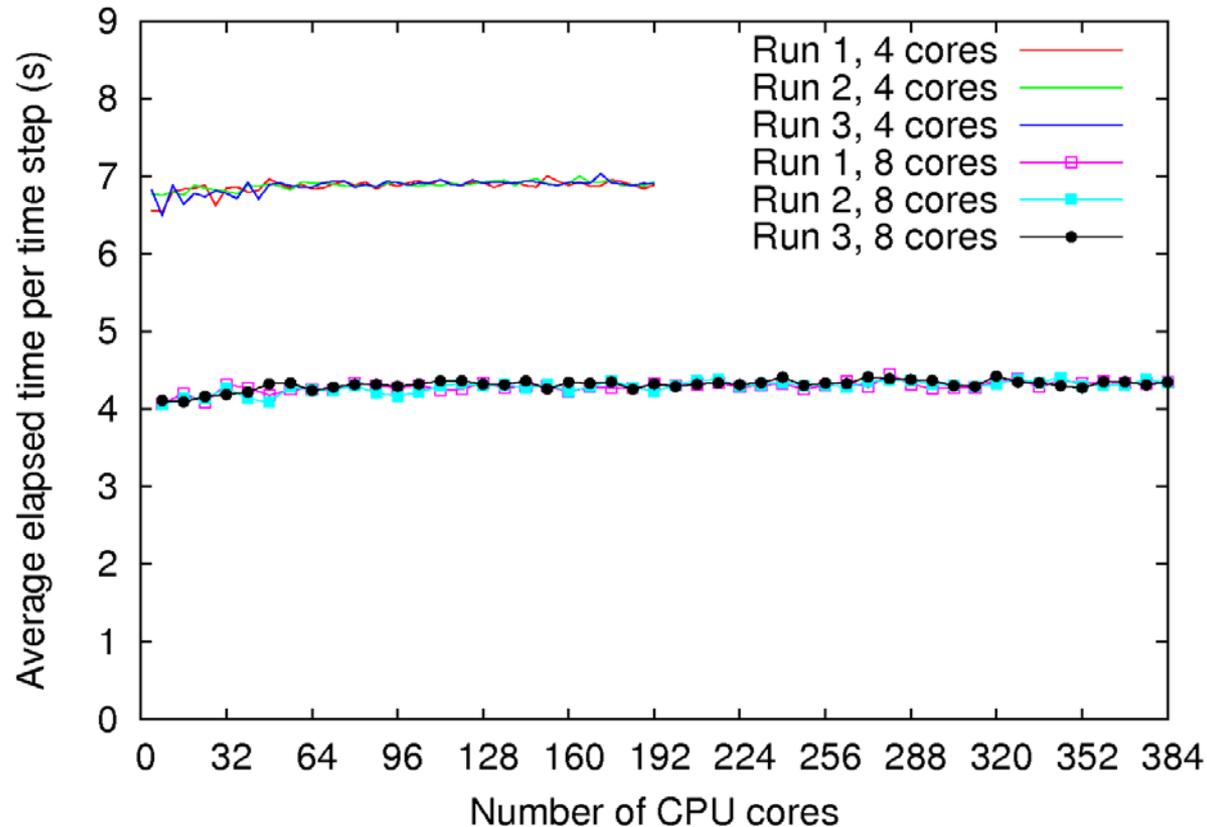
Quasi-analytical solution computed via summation of normal modes.

Pressure and shear waves are accurately computed, static offsets are reproduced.

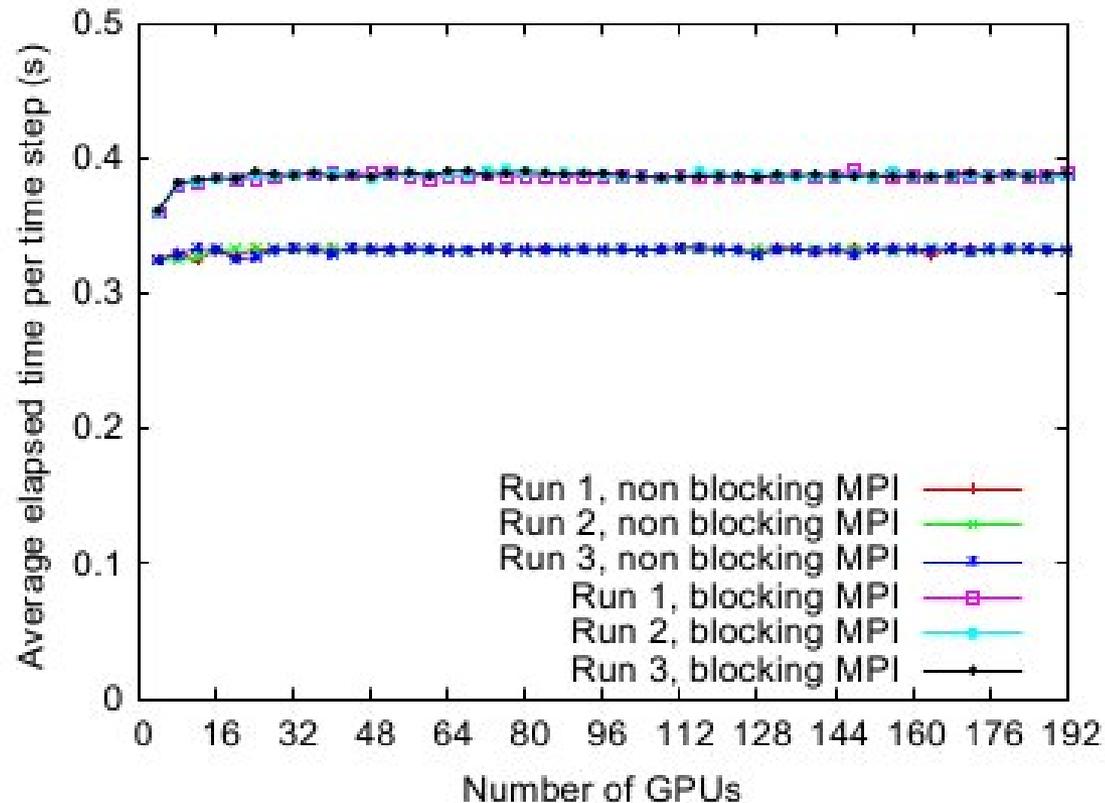
No visible difference between CPU and GPU solutions

Close-up shows that the only difference is tiny floating point noise (similar to switching from a compiler to another on a CPU, or using -O3 instead of -O2)

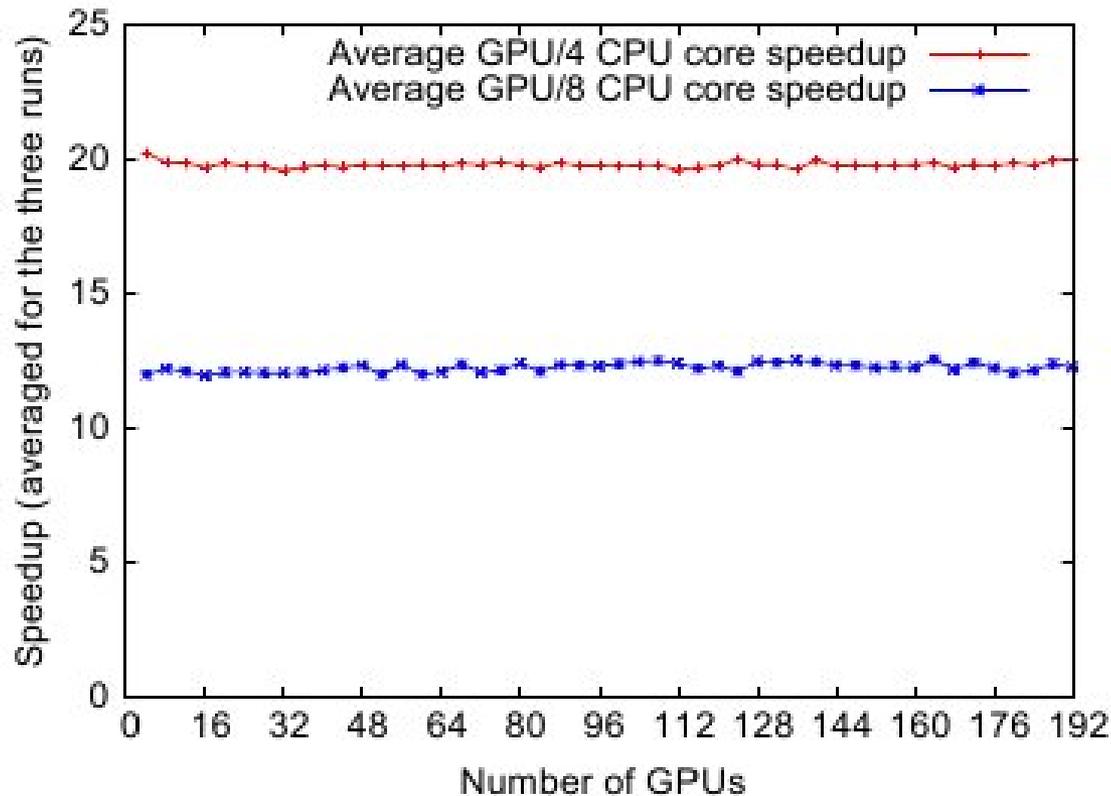




- Constant (and very large) problem size per node (4*3.6 GB, 8*1.8 GB)
- Weak scaling excellent at least up to 17 billion unknowns
- 4-core case uses 2+2 Nehalem cores per node
- Strong scaling only 60% gain due to memory bus and network contention

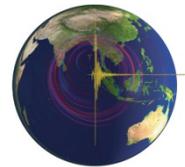


- Constant problem size of 3.6 GB per GPU (90% of GPU memory)
- Weak scaling excellent up to at least 17 billion unknowns
- Using blocking MPI results in 20% slowdown; non-blocking MPI allows us to overlap communications with calculations



- Fair comparison: CPU reference code extremely optimised
- Average speedup vs. 2 quad-core Nehalem CPUs is 13x
- Average speedup vs. same number of nodes in the cluster is 21x

- Seismic modeling and tomography (using a spectral-element method) can greatly benefit from GPU computing, with **significant acceleration**
- Porting the SPECFEM3D code to such GPUs has required a significant but reasonable amount of work; but porting to GPUs is becoming easier these days (more flexible tools)
- Application to adjoint/inverse problems is underway in the context of a collaboration with Princeton, Basel and Zürich:
 - **Today, 4-6 pm, room 3022 (Moscone West)**
 - **DI14A: Advances in Computational Modelling in Geoscience**
 - **Daniel Peter**, Max Riethmann, Dimitri Komatitsch and Jeroen Tromp: «*Advances in high-performance spectral-element solvers for seismic tomography*»
- We will release these tools to the community: **geodynamics.org**
 - Next open-source versions of SPECFEM3D and SPECFEM3D_GLOBE will have full GPU support (and also PML absorbing conditions, but that is another story)
- We will test e.g. the **future MIC / Knights Corner hardware from Intel** as soon as it is available

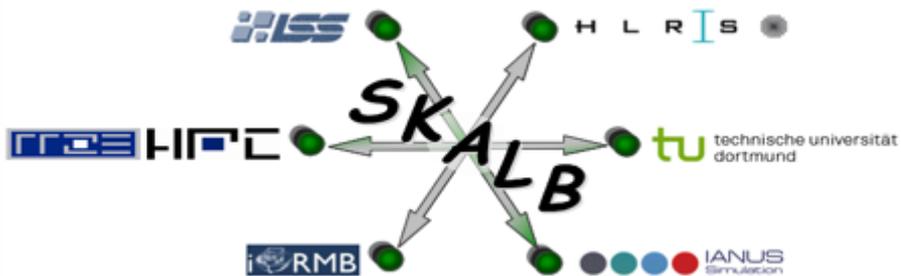


- **Germany**

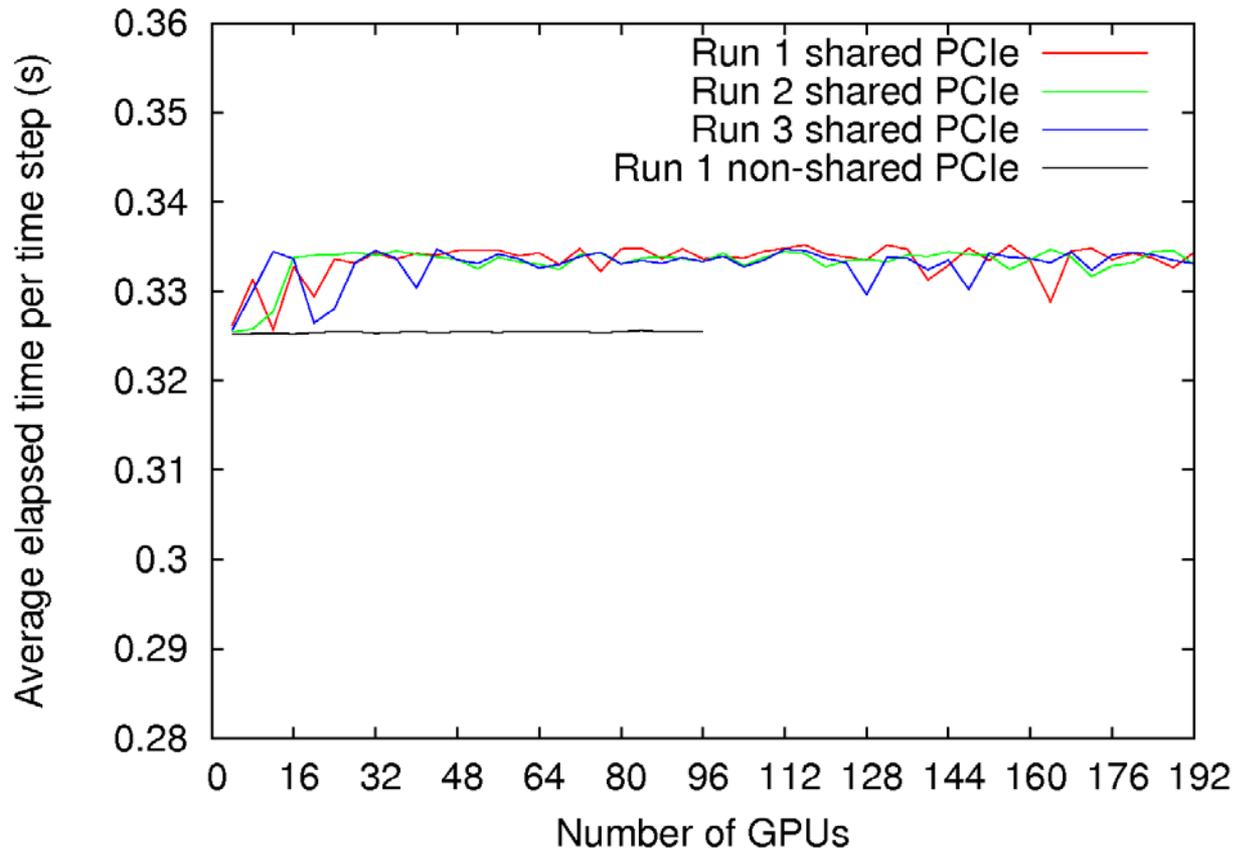
- Bundesministerium für Bildung und Forschung (BMBF), call „HPC Software für skalierbare Parallelrechner“, SKALB project (01IH08003D)

- **France**

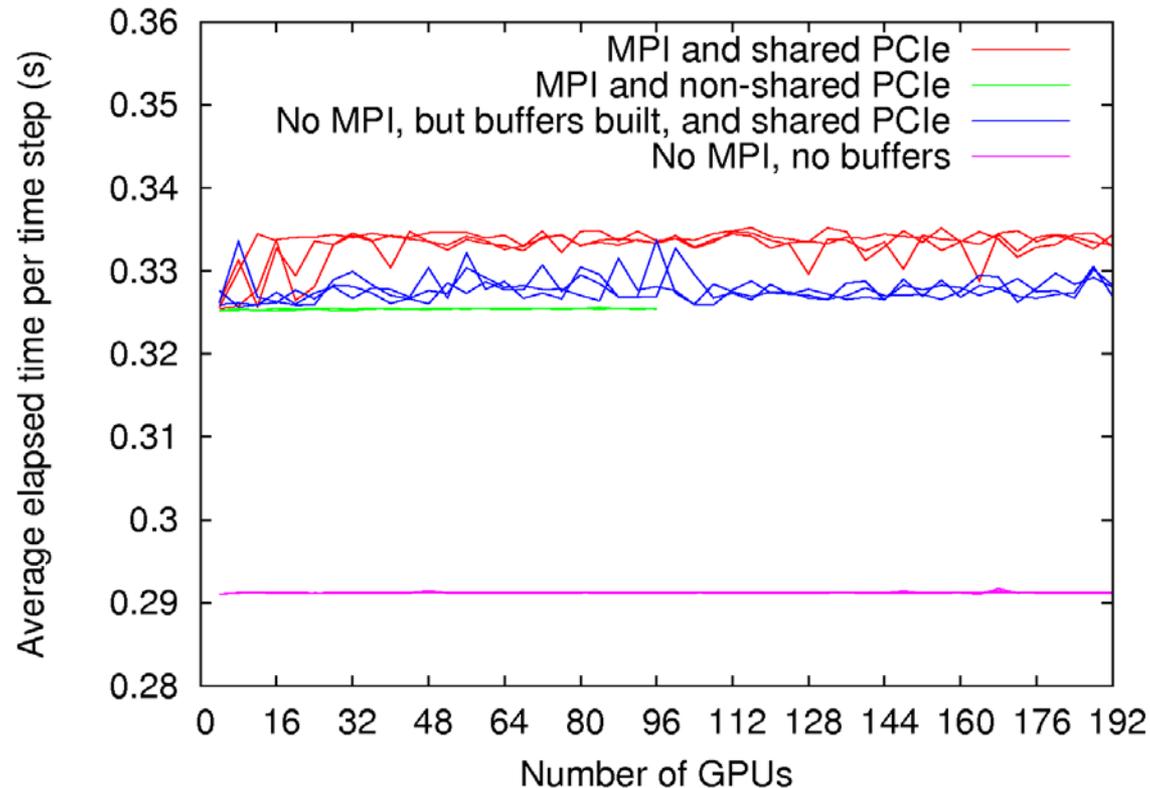
- GENCI / CEA-CCRT / ANR / INRIA



Slides for questions



- 2 GPUs share one PCIe bus in the Tesla S1070 architecture
- This could potentially be a huge bottleneck!
- But in practice it is not: bus sharing introduces fluctuations between runs and a slowdown $\leq 3\%$



- Effect of overlapping (no MPI = replace send/receive with zeroing)
- Red vs. blue curve: Difference $\leq 2.8\%$, i.e., very good overlap
- Green vs. magenta: Total overhead cost of running this problem on a cluster is $\leq 12\%$ (for building, processing and transmitting buffers)