GROMACS in the Cloud: From traditional HPC to global parallelism

Carsten Kutzner,[1] Vytas Gapsys,[1] Christian Kniep,[2] Helmut Grubmüller[1]

[1]Max Planck Institute for Biophysical Chemistry, Theoretical and Computational Biophysics Dept., Göttingen [2]Amazon Web Services, Amazon Development Center Germany, Berlin

Introduction

- Cloud-based computing is offered by Amazon Web Services (AWS), Microsoft Azure, Google Cloud, and more providers
- We evaluate the suitability of cloud computing for biomolecular simulations by setting up a **cloud-based HPC cluster for GROMACS**[1] molecular dynamics (MD) simulations on AWS
- We compare cloud costs and performance to a typical on-premises department cluster
- Depending on the scientific questions addressed, an MD project might fall more into the realm of **high performance computing (HPC)**, or **high throughput computing (HTC)**
- In an HPC scenario, the performance of an individual (usually large) MD system needs to be maximized, often by scaling across multiple nodes (or instances in the cloud)
- In an HTC scenario one wants to minimize either the time to solution or the costs of running a large ensemble of (usually smaller) simulations
- Therefore, we benchmark i) **which instances deliver the highest GROMACS performace** for HTC, and ii) which **offer the best performance to price ratio** for HTC

Questions addressed

- How competitive is the cloud compared to an on-premises department cluster or a traditional HPC center?
- **Is an on-premises cluster still worth it?** Should we migrate our scientific workloads into the cloud?

Methods

Setting up a cloudy cluster

- We use ParallelCluster 2.10 (https://github.com/aws/aws-parallelcluster) as open source, free, cluster management tool
- We use Spack 0.15.4 as flexible package manager for HPC software (https://github.com/spack/spack.git)
- default install: `spack install gromacs` with GCC 7.3.1, FFTW 3.3.8, hwloc 1.11
- With GPU support via CUDA 10.2 and IntelMPI 2019: `spack install gromacs@2020.2 +cuda ^intel-mpi`

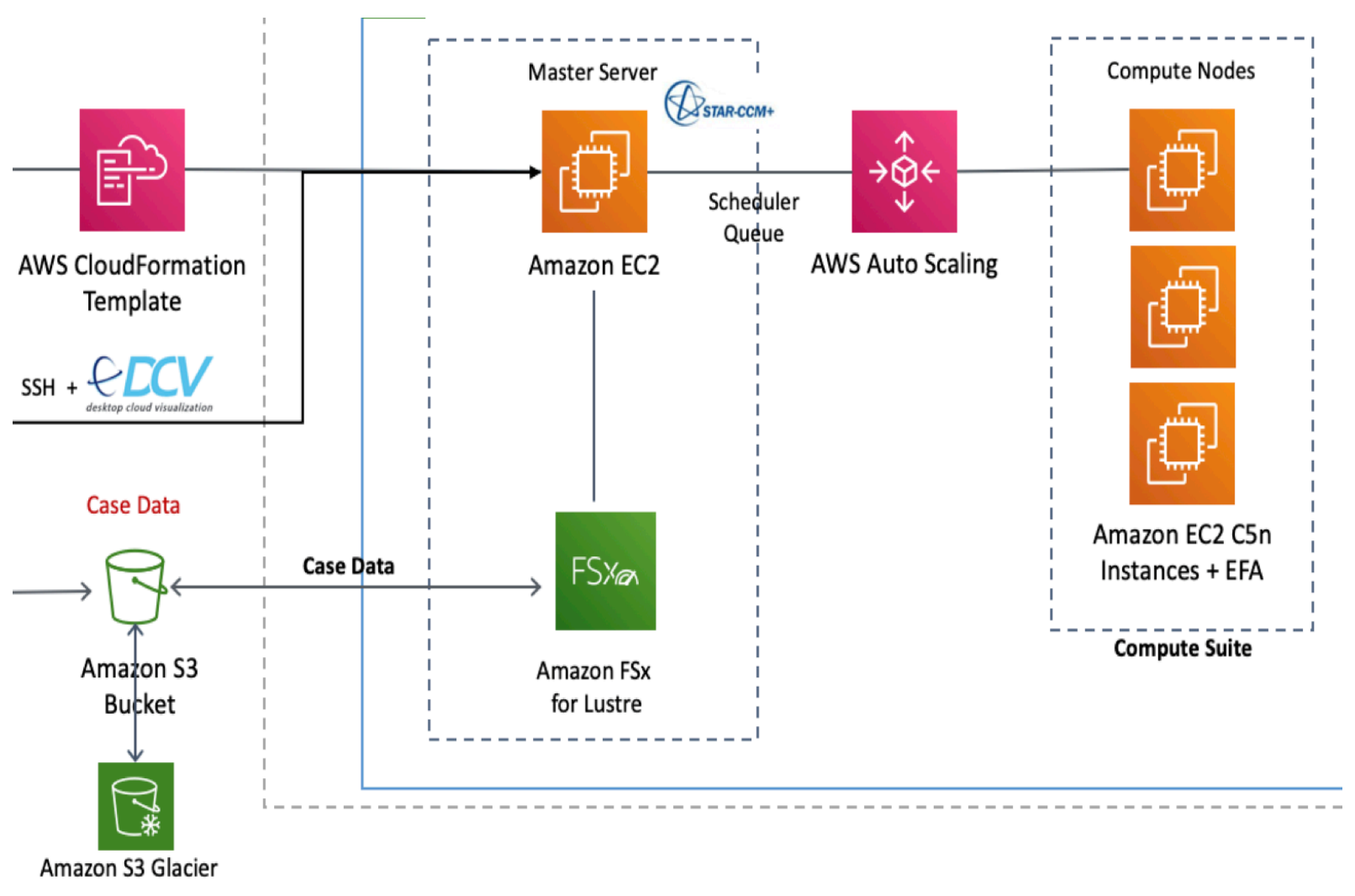
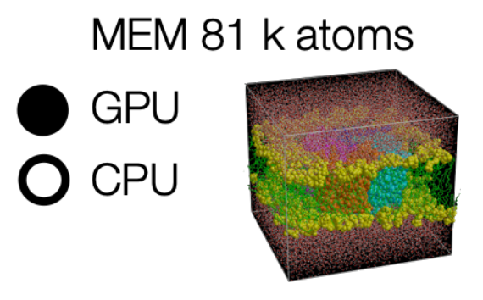


Figure 1. A cloudy HPC cluster. Just like a regular cluster a cloud-based cluster has a master to submit compute jobs via a SLURM queue to a fleet of (possibly inhomogeneous) compute instances. With AutoScaling, nodes are started and stopped depending on whether there are jobs in the queue, reducing job waiting times in the queue to essentially zero.

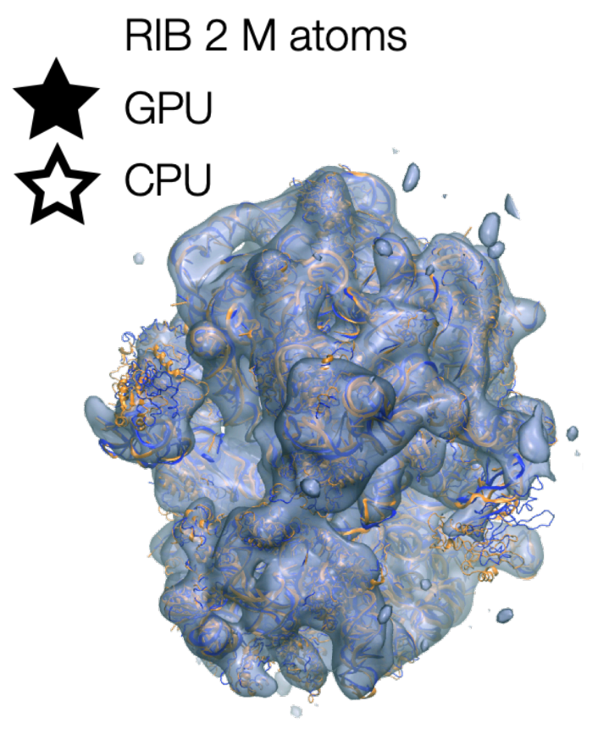
Benchmark MD systems

- We use the following benchmark systems to evaluate GROMACS 2020 performance:

- **MEM**
  81k atoms Aquaporin tetramer embedded in lipid membrane surrounded by ions and water, PME electrostatics, 2 fs time step, NPT



- **RIB**
  2M atoms, Ribosome in water, PME electrostatics, 4 fs time step, NPT



Benchmarking procedure

- Typical benchmark run command[2]

```
mpirun -n $mpi gmx_mpi mdrun -s MEM.tpr
       -ntomp $nt -nsteps 10000 -resethway
       -npme 0 -pin on -cpt 1440
```

- We vary the number of MPI ranks vs. OpenMP threads on the instances and also check whether separate PME nodes improve performance
- Report average performance (ns/d) over two runs for the optimal parameters each for various AWS instances:

- **Intel Platinum** `c5` (and `c5n` with fast EFA network)
  2–96 vCPUs (2 vCPUs = 2 hardware threads = 1 core)
- **High frequency** `m5zn` **instances**
  8–48 vCPUs
- **AMD EPYC** `c5a`
  2–96 vCPUs
- **ARM Graviton2** `c6g`
  4–64 vCPUs
- **V100 GPUs** `p3`
  8–96 vCPUs with up to 1–8 V100 GPUs
- **A100 GPUs** `p4d`
  96 vCPUs with 8 A100 GPUs
- **T4 GPUs** `g4dn`
  4–96 vCPUs with 1–4 NVIDIA T4 GPUs

Results: Cloud vs. on-prem cluster

How fast ist cloud computing?

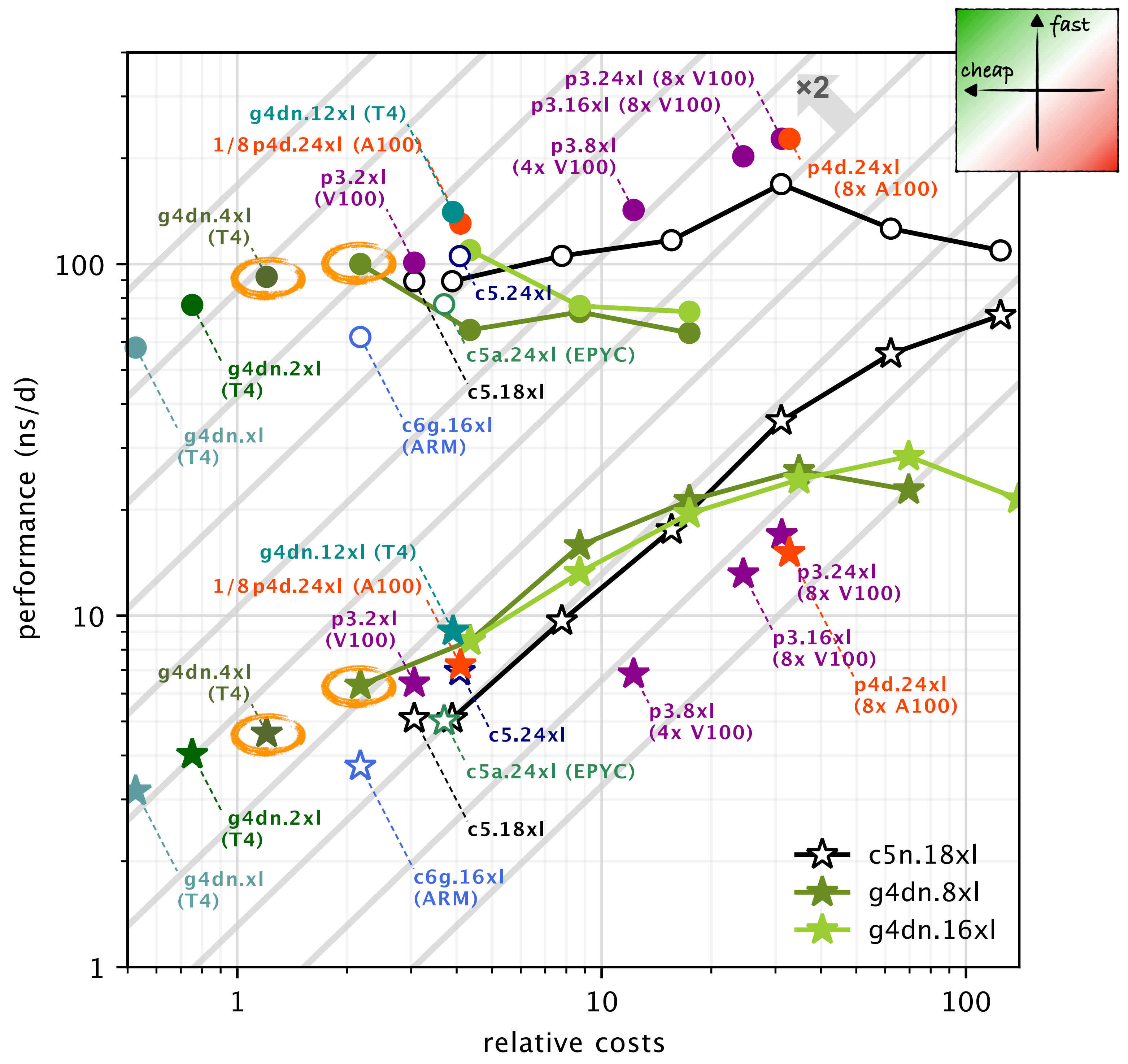


Figure 2. GROMACS performance as a function of instance costs for the MEM (circles) and RIB (stars) benchmark on CPU (open symbols) and GPU instances (filled symbols). The tilted grey lines are isolines of equal performance to price ratio with better configurations to the upper left. g4dn GPU instances offer the best performance to price ratio, therfore we used g4dn.4xl and g4dn.8xl instances **(highlighted in orange)** for the cost comparison (Fig. 3.)

How competitive is cloud computing?

- For a fair comparison, we consider all costs that arise for operating a typical, 500 node department cluster over three years.
- Compute costs only, we are not considering costs for storage of trajectories
- Net costs to produce one μs of RIB trajectory

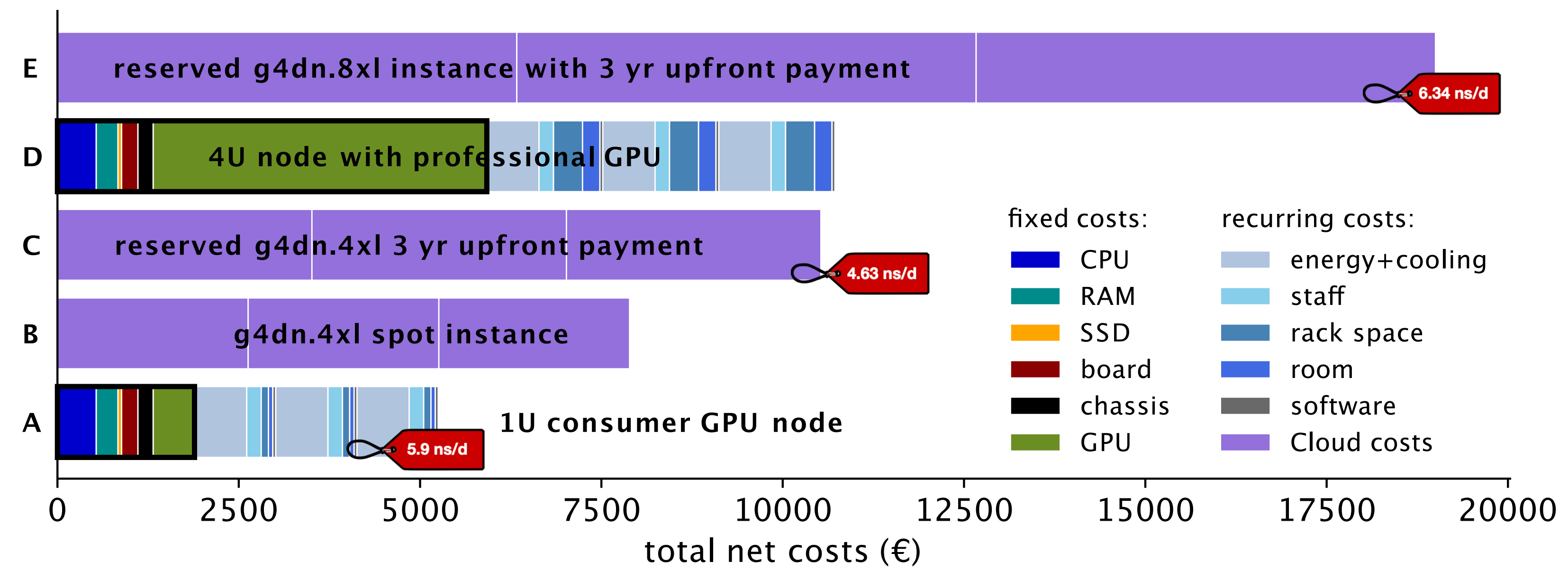


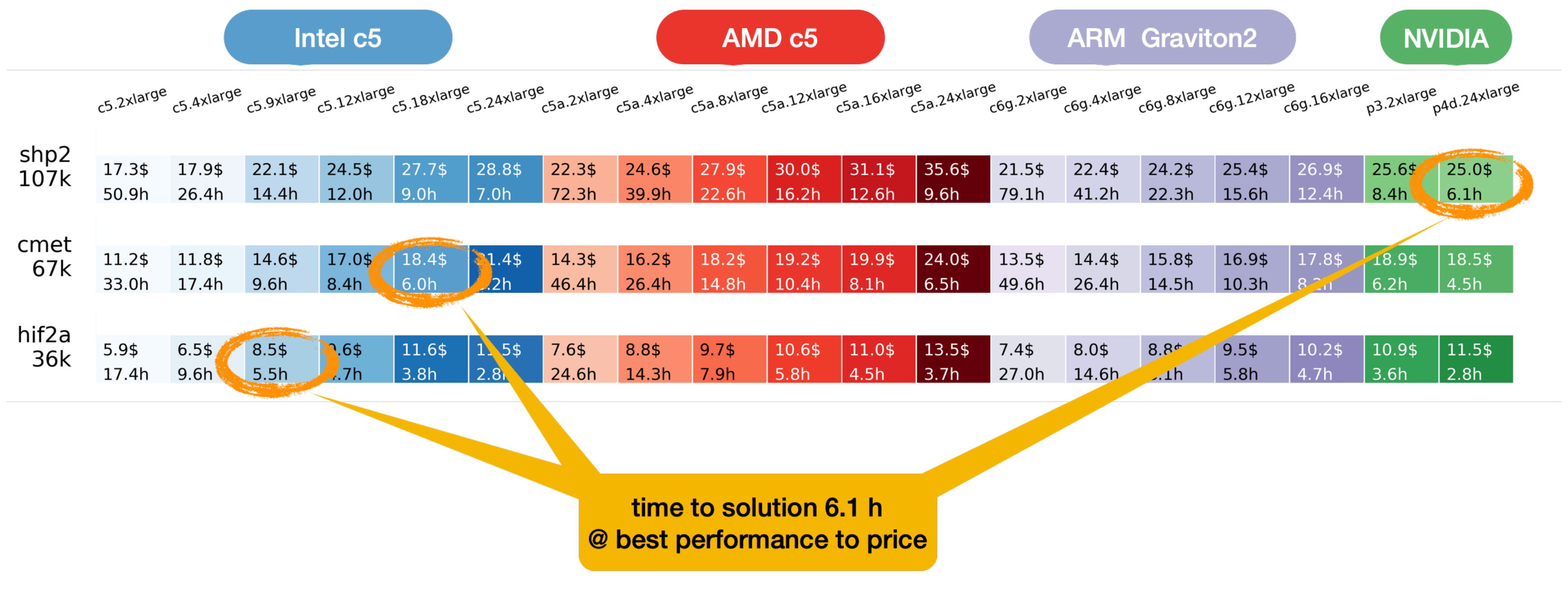
Figure 3. Total net costs for 3 years of operation of a node in an on-premises cluster compared to cloud instances with similar performance. Violet bars show costs of cloud instances selected for high performance to price ratio with GROMACS (compare Fig. 1), in blocks of one year.

A Cost of a consumer GPU node[2] tailored to GROMACS with yearly recurring costs (mainly energy/cooling) for 3 years.
B Average costs for renting a g4dn.4xl instance on the spot market.
C Costs for a g4dn.4xl instance for a 3 years reservation with upfront payment.
D Same as A, but for a 4 U node with a professional GPU (Quadro P6000).
E Costs for a g4dn.8xl instance for a 3 years reservation with upfront payment.

Outlook: HTC in the cloud

Speed up computational drug design with global parallelism!

- Aim: Compute ensemble of 20,000 MD systems (5k – 100k atoms) as fast as possible
- Approach: run all systems at the same time, run each system on a separate spot instance, wherever there is capacity globally
- Challenges: orchestrate simulations globally over many regions, real-time monitoring of actual costs
- Time to solution can not be smaller than longest individual run time, therefore start large systems on powerful instances and mall systems on cheap instances (Fig. 6)



	c5.2xlarge	c5.4xlarge	c5.9xlarge	c5.12xlarge	c5.18xlarge	c5.24xlarge	c5a.2xlarge	c5a.4xlarge	c5a.8xlarge	c5a.12xlarge	c5a.16xlarge	c5a.24xlarge	c6g.2xlarge	c6g.4xlarge	c6g.8xlarge	c6g.12xlarge	c6g.16xlarge	p3.2xlarge	p4d.24xlarge
shp2 107k	17.3$ 50.9h	17.9$ 26.4h	22.1$ 14.4h	24.5$ 12.0h	27.7$ 9.0h	28.8$ 7.0h	22.3$ 72.3h	24.6$ 39.9h	27.9$ 22.6h	30.0$ 16.2h	31.1$ 12.6h	35.6$ 9.6h	21.5$ 79.1h	22.4$ 41.2h	24.2$ 22.3h	25.4$ 15.6h	26.9$ 12.4h	25.6$ 8.4h	25.0$ 6.1h
cmet 67k	11.2$ 33.0h	11.8$ 17.4h	14.6$ 9.6h	17.0$ 8.4h	18.4$ 6.0h	[illegible] [illegible]	14.3$ 46.4h	16.2$ 26.4h	18.2$ 14.8h	19.2$ 10.4h	19.9$ 8.1h	24.0$ 6.5h	13.5$ 49.6h	14.4$ 26.4h	15.8$ 14.5h	16.9$ 10.3h	17.8$ [illegible]	18.9$ 6.2h	18.5$ 4.5h
hif2a 36k	5.9$ 17.4h	6.5$ 9.6h	8.5$ 5.5h	[illegible] [illegible]	11.6$ 3.8h	[illegible] [illegible]	7.6$ 24.6h	8.8$ 14.3h	9.7$ 7.9h	10.6$ 5.8h	11.0$ 4.5h	13.5$ 3.7h	7.4$ 27.0h	8.0$ 14.6h	[illegible] [illegible]	9.5$ 5.8h	10.2$ 4.7h	10.9$ 3.6h	11.5$ 2.8h

Figure 6. Exemplary mapping of MD systems to instances to minimize the time to solution while keeping costs low. For three exemplary benchmark systems shp2, cmet and hif2a (left column), the colored bars show the estimated run time of the simulation in hours (lower number) and the estimated costs for the run in US dollars (upper number) for various AWS instances (top row, c5.2xlarge, c5.4xlarge, ...)

Conclusions

- AWS offers a **wide range of instances** (2–96 core Intel / AMD / ARM instances with and without GPUs, see grey box in lower left corner of poster) so that the one best suited for the job can be picked
- **Cloud-based HPC is feasible**, e.g. on c5n instances with fast EFA network. However parallel scaling not (yet) as good as in a traditional HPC center (Fig. 5)
- Compared to instances with the highest performance to price ratio (g4dn), **a department cluster aggressively optimized for GROMACS simulations can produce 2–3× as much trajectory per €** (see cost comparison). However, this factor is likely not achieved for HPC centers that need to serve many different applications
- Time to solution for large simulation ensembles, as e.g. used for computational drug design, could decrease from weeks on a traditional cluster to **overnight in the global cloud**

Cost comparison

- We tune our department cluster aggressively for throughput with GROMACS, by **using GeForce consumer GPUs**, reasonably priced CPUs, omitting HPC interconnects, a small amount of RAM, and dense packing (~1 GPU per U)
- This leads to trajectory costs of only 1/3× the trajectory costs of CPU nodes or nodes with Tesla GPUs[2] (as often are used in a traditional HPC center)
- Our **exemplary 1U consumer GPU node** (Fig. 3A, 20 hardware threads + RTX 2080 GPU) has total net costs of 5250 € for three years of operation and produces 5.9 ns of RIB trajectory per day, i.e. 5.9 ns/d × 3 y × 365 d/y = 6.46 μs traj. in total. This leads to **trajectory costs of 810 €/μs**
- AWS **cloud g4dn.8xl instances** (Fig. 3E, 32 vCPUs + T4 GPU) offer a RIB performance of 6.34 ns/d at a good performance to price ratio (Fig. 2 lower left). Over 3 years an instance costs 19000 € and produces 6.94 μs traj., leading to **trajectory costs of 2740 €/μs**
- **g4dn.4xl instances** (16 vCPUs + T4 GPU) offer an even higher performance to price ratio, albeit at a slightly lower RIB performance (4.63 ns/d, Fig. 2 lower left). Over 3 years an instance costs 10530 € and produces 5.07 μs traj., yielding **trajectory costs of 2080 €/μs**
- A still cheaper way would be to use g4dn.4xl spot instances at ~30% of the on-demand price. This would cost 7900 € and would yield **trajectory costs of 1560 €/μs**
- However, an on-premises **cluster specialized for GROMACS** can produce MD trajectory at 0.3 – 0.5 × the price of cloud instances

Costs of on-prem cluster

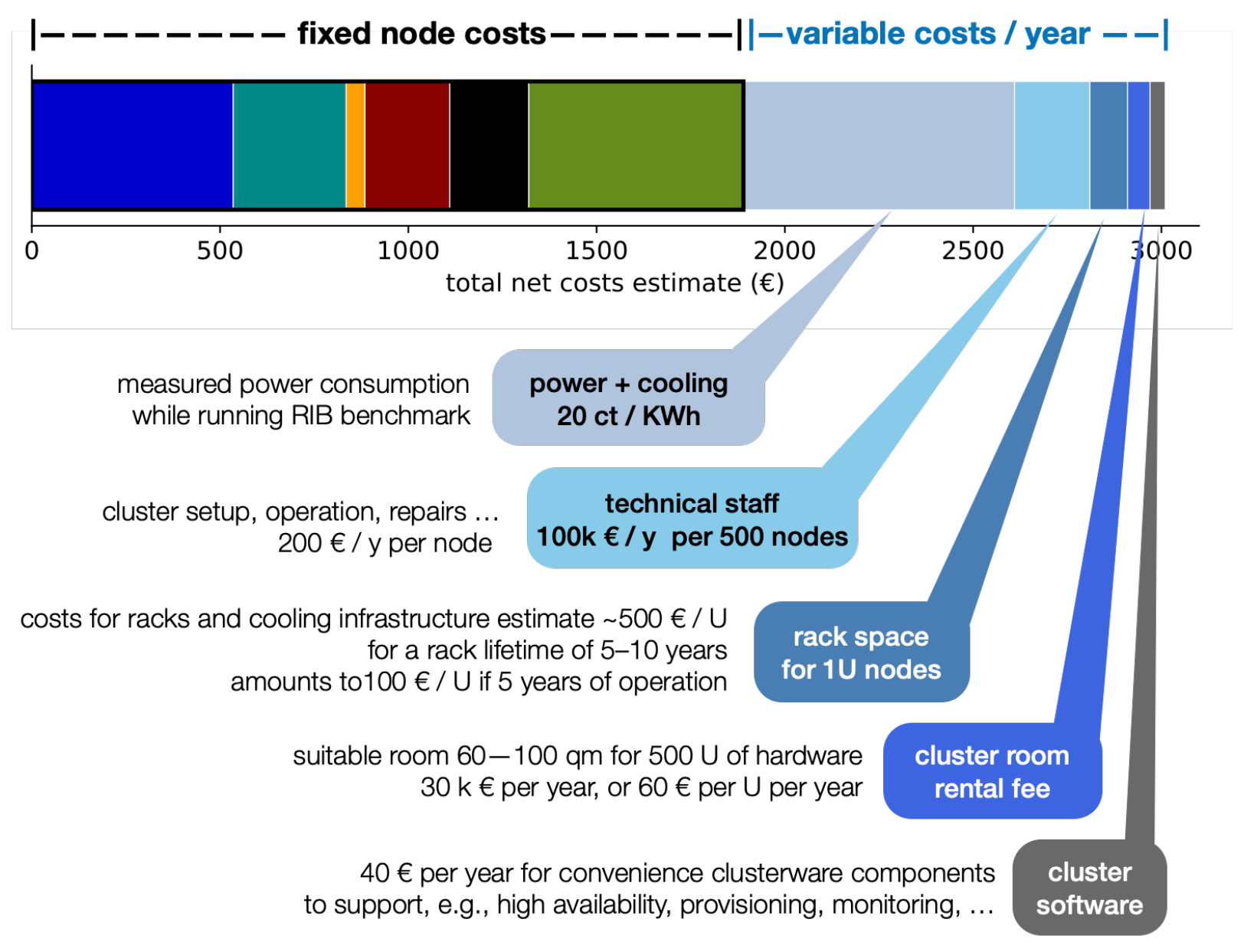


Figure 4. Breakdown of total node costs for on-premises cluster for the first year of operation. See also Fig. 3 for legend.

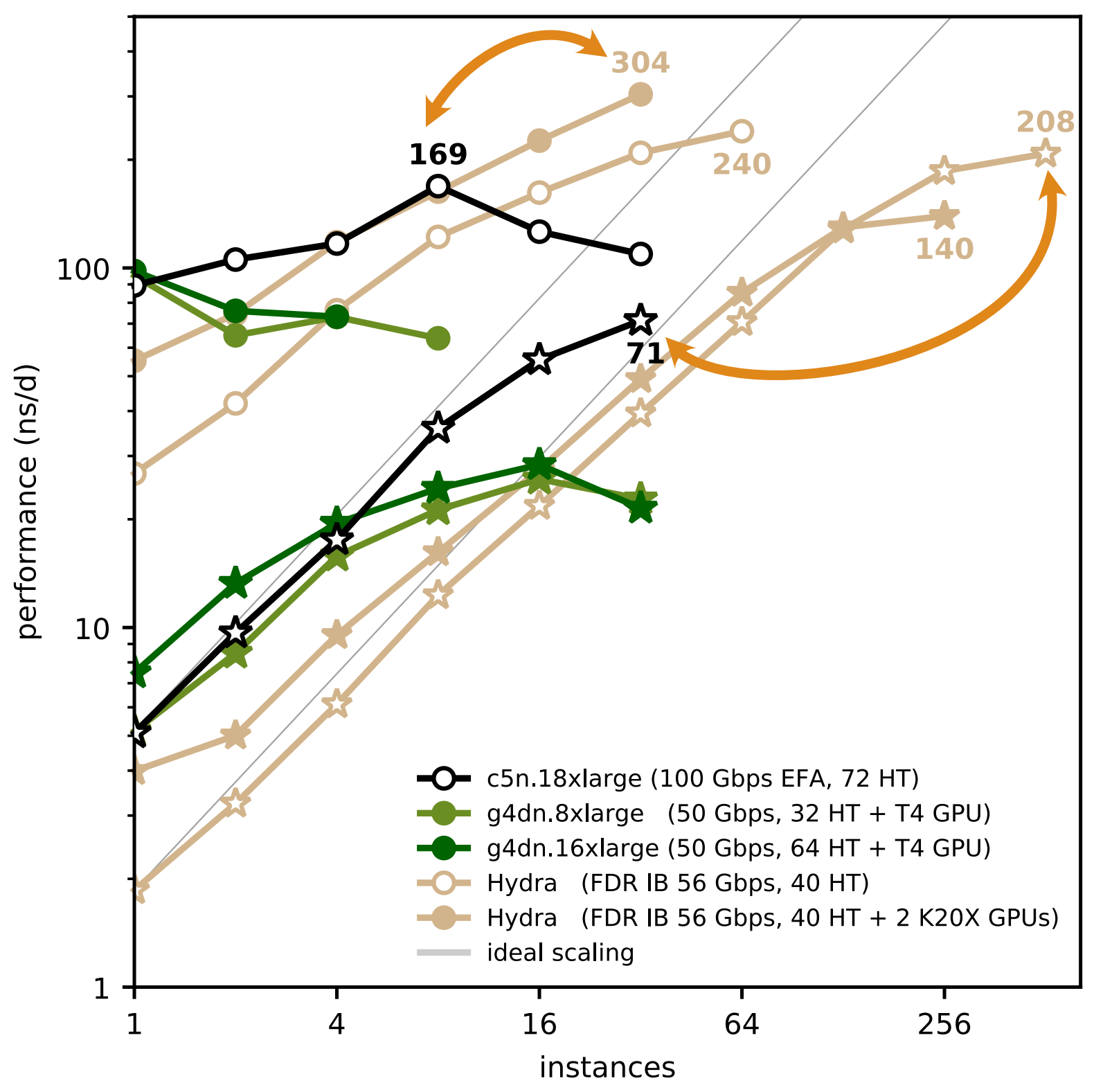


Figure 5. Parallel scaling of GROMACS in the cloud compared to a traditional HPC center for the MEM (circles) and RIB (stars) benchmark on CPU (open symbols) and GPU instances (filled symbols), as in Fig. 2. In a HPC center **(brown curves)**, both the scaling as well as the total absolute performances **(arrows)** are higher.[1]

Acknowledgments

Compute time to carry out this project was generously provided by AWS. Many thanks to the AWS team (Torsten Bloth, Stephen Sachs, Johannes Schulz, and Evan Bollig) for advice and helpful discussions. The total costs of rack space (Fig. 4) were estimated by A. Esztermann.

References

1. S Páll et al. *Tackling exascale software challenges in molecular dynamics simulations with GROMACS.* EASC 2014, Stockholm, pp. 3–27 (Eds. Markidis, S.; Laure, E.) Springer, Cham (2015)
2. Kutzner, C, et al. *More Bang for Your Buck: Improved use of GPU Nodes for GROMACS 2018.* JCC 40, 27 (2019): pp 2418–2431