bioexcel
Centre of Excellence for Computational Biomolecular Research

Horizon 2020
European Union Funding
for Research & Innovation

BioExcel Webinar Series #37

More bang for your buck: Improved use of GPU Nodes for GROMACS 2018

Presenter: Carsten Kutzner (Max Planck Institute)

Host: Rossen Apostolov (KTH Royal Institute of Technology)

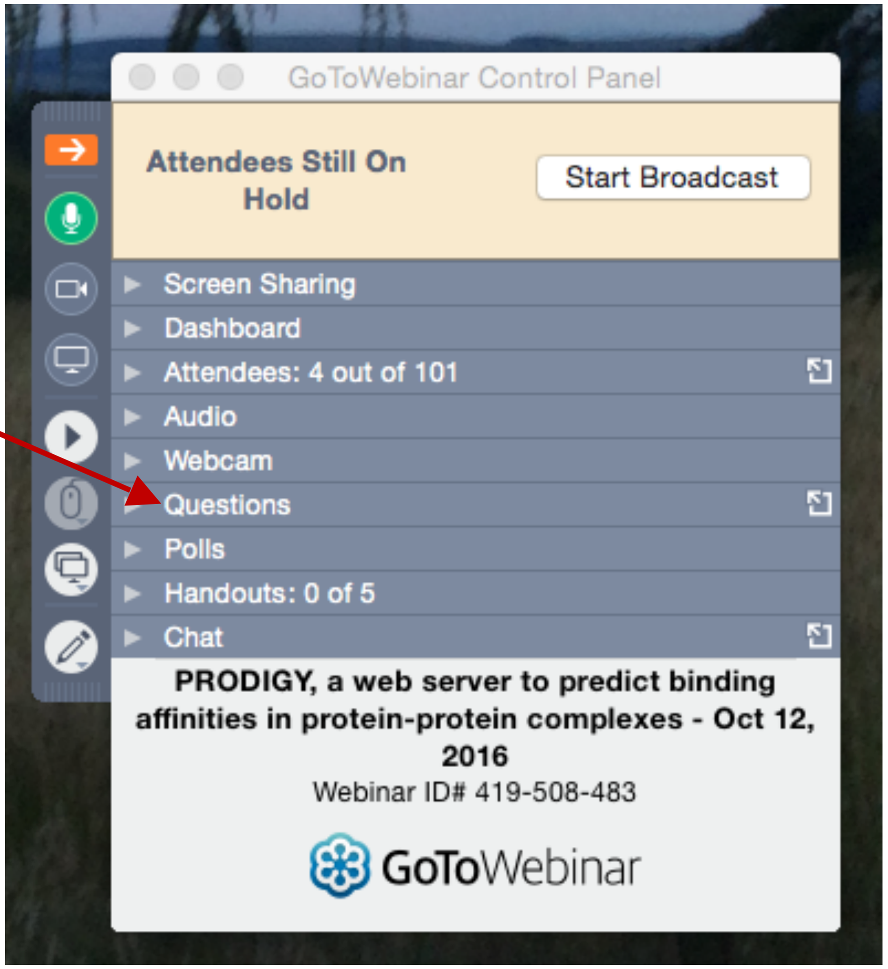
● REC

This webinar is being recorded

Audience Q&A session

- Please use the Questions function in GoToWebinar application

- Any other questions or points to discuss after the live webinar? Join the discussions at http://ask.bioexcel.eu.



PRODIGY, a web server to predict binding affinities in protein-protein complexes - Oct 12, 2016

Webinar ID# 419-508-483

Today’s Presenter

Carsten Kutzner

Max Planck Institute for Biophysical Chemistry

Carsten studied physics at the University of Göttingen. For his PhD he focused on numerical simulations of Earth’s magnetic field, which brought him in contact with high performance and parallel computing. After a stay at the MPI for Solar System Research he moved to computational biophysics. Since 2004 he has been working at the Max Planck Institute for Biophysical Chemistry in the lab of Helmut Grubmüller. His is interested in method development, high performance computing, and atomistic biomolecular simulations.

Twitter:

@kutznercarsten https://twitter.com/kutznercarsten

@CompBioPhys https://twitter.com/CompBioPhys

Homepage:

https://www.mpibpc.mpg.de/grubmueller/kutzner

Carsten Kutzner[1], Szilárd Páll[2], Martin Fechner[1], Ansgar Esztermann[1], Bert de Groot[1], Helmut Grubmüller[1]

MORE

FOR YOUR BUCK!

Improved use of GPU Nodes for GROMACS 2018

1 Theoretical & Computational Biophysics, MPI for Biophysical Chemistry, Göttingen
2 Center for High Performance Computing, KTH Royal Institute of Technology, Stockholm

Motivation

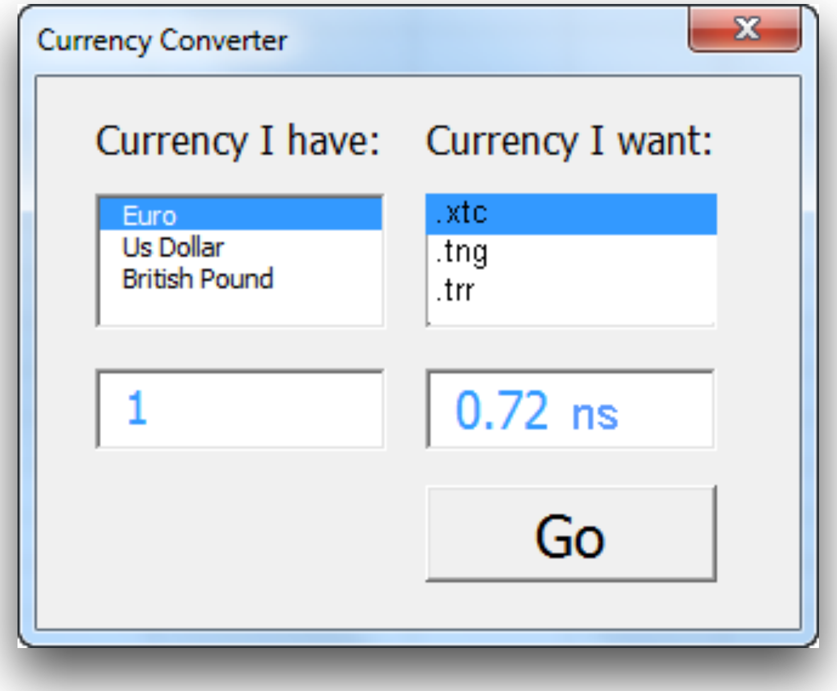
- Many MD groups buy small compute clusters from a **fixed budget**

- How to optimally make use of that?

  - We run mostly **GROMACS** MD,
    ➔ tailor nodes for GROMACS,
    maximise cost-efficiency by specialisation

  - queue is always full ➔ optimise for **throughput / single-node performance**

  - (scaling ➔ HPC centres)



Currency Converter

Currency I have: Euro

Currency I want: .xtc

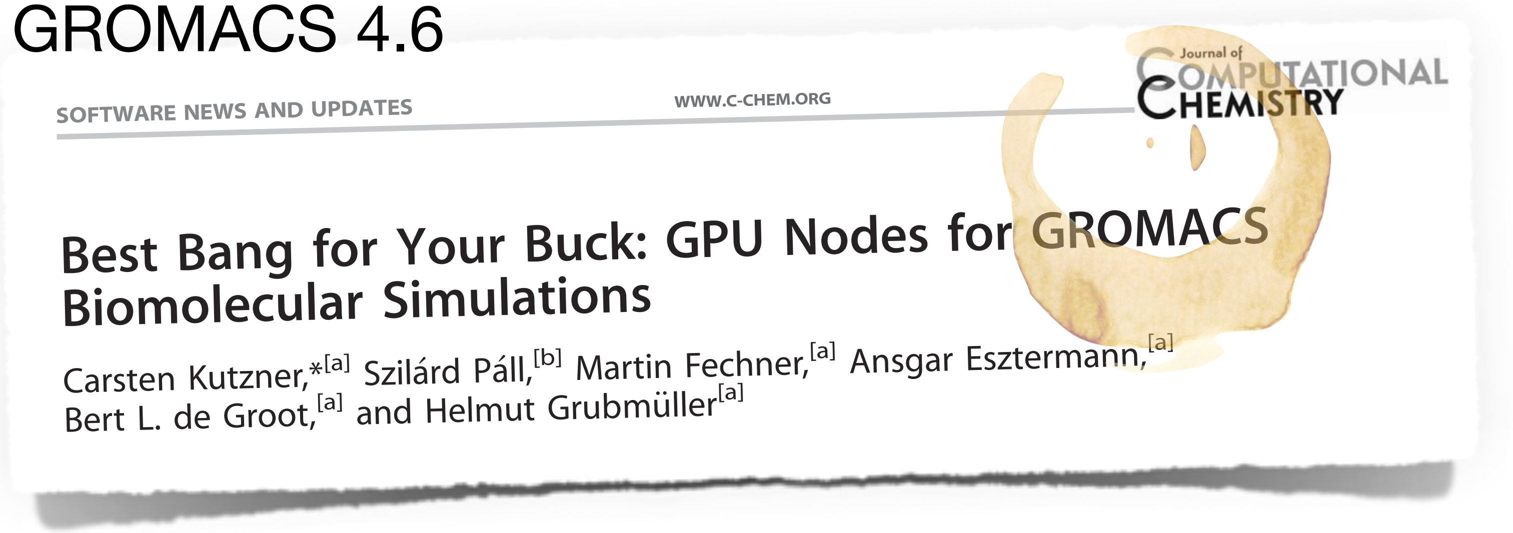
1

0.72 ns

- **Given a fixed budget,
  how can we produce as much MD trajectory as possible?**

Outline

- 2014 - GROMACS 4.6



SOFTWARE NEWS AND UPDATES WWW.C-CHEM.ORG Journal of COMPUTATIONAL CHEMISTRY

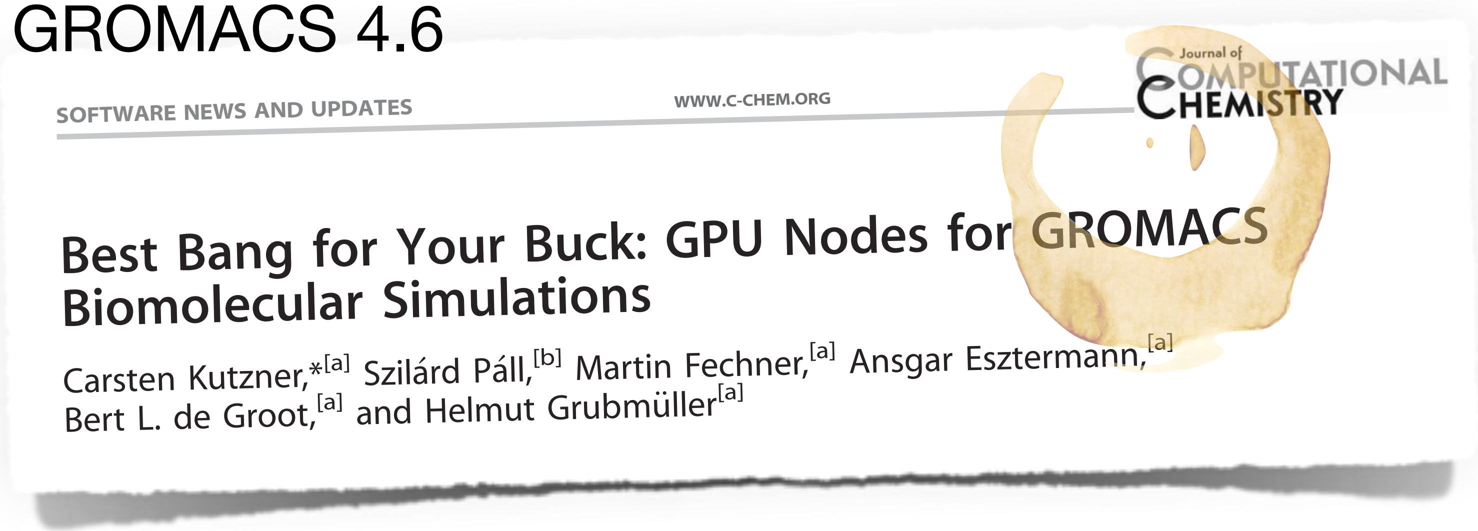
Best Bang for Your Buck: GPU Nodes for GROMACS Biomolecular Simulations

Carsten Kutzner,*[a] Szilárd Páll,[b] Martin Fechner,[a] Ansgar Esztermann,[a] Bert L. de Groot,[a] and Helmut Grubmüller[a]

- recap: what were our conclusions in 2014/15?
- hardware & software developments and their impact
- update

Outline

- 2014 - GROMACS 4.6

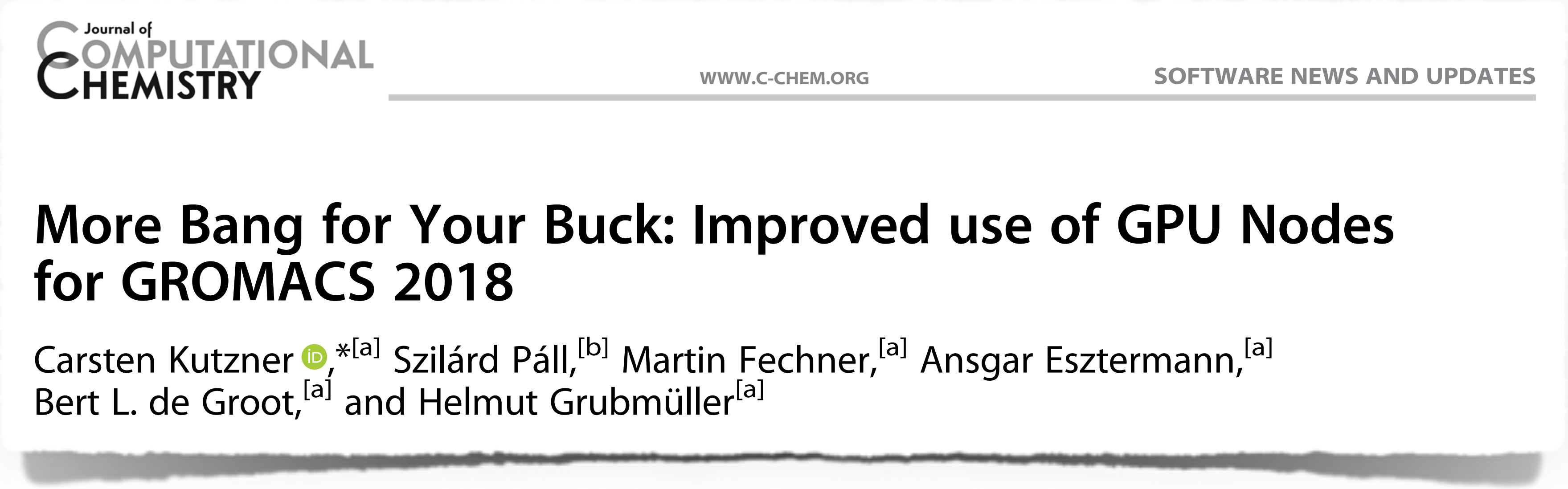


SOFTWARE NEWS AND UPDATES WWW.C-CHEM.ORG Journal of COMPUTATIONAL CHEMISTRY

Best Bang for Your Buck: GPU Nodes for GROMACS Biomolecular Simulations

Carsten Kutzner,*[a] Szilárd Páll,[b] Martin Fechner,[a] Ansgar Esztermann,[a] Bert L. de Groot,[a] and Helmut Grubmüller[a]

- recap: what were our conclusions in 2014/15?
- hardware & software developments and their impact
- update



Journal of COMPUTATIONAL CHEMISTRY WWW.C-CHEM.ORG SOFTWARE NEWS AND UPDATES

More Bang for Your Buck: Improved use of GPU Nodes for GROMACS 2018

Carsten Kutzner,*[a] Szilárd Páll,[b] Martin Fechner,[a] Ansgar Esztermann,[a] Bert L. de Groot,[a] and Helmut Grubmüller[a]

Approach

- from ~10 CPU types + ~10 GPU models we **assemble** and **benchmark** various compute nodes
  - CPU nodes
  - GPU nodes with 1, 2, 3, and 4 GPUs
  - **consumer** and **professional** GPUs
- determine **performance-to-price (P/P) ratio**

consumer GPUs (GeForce)

GTX 980
GTX 1070
GTX 1070Ti
GTX 1080
GTX 1080Ti
RTX 2070
RTX 2080
RTX 2080Ti

professional GPUs (Tesla)

Quadro P6000
Tesla V100

CPUs

Ryzen (16 core)
Epyc (24 core)
Core i7 (4 core)
Xeon (4, 6, 8, 10, and 20 core)

Approach

- from ~10 CPU types + ~10 GPU models we **assemble** and **benchmark** various compute nodes
  - CPU nodes
  - GPU nodes with 1, 2, 3, and 4 GPUs
  - **consumer** and **professional** GPUs
- determine **performance-to-price (P/P) ratio**

- no comprehensive evaluation of currently available hardware!
  - but aim to uncover HW with good P/P ratio
- no strong scaling!

consumer GPUs (GeForce)

GTX 980
GTX 1070
GTX 1070Ti
GTX 1080
GTX 1080Ti
RTX 2070
RTX 2080
RTX 2080Ti

professional GPUs (Tesla)

Quadro P6000
Tesla V100

CPUs

Ryzen (16 core)
Epyc (24 core)
Core i7 (4 core)
Xeon (4, 6, 8, 10, and 20 core)

Approach

- from ~10 CPU types + ~10 GPU models we **assemble** and **benchmark** various compute nodes
    - CPU nodes
    - GPU nodes with 1, 2, 3, and 4 GPUs
    - **consumer** and **professional** GPUs
- determine **performance-to-price (P/P) ratio**

- no comprehensive evaluation of currently available hardware!
    - but aim to uncover HW with good P/P ratio
- no strong scaling!

- benchmark MD systems:

consumer GPUs (GeForce)

GTX 980
GTX 1070
GTX 1070Ti
GTX 1080
GTX 1080Ti
RTX 2070
RTX 2080
RTX 2080Ti

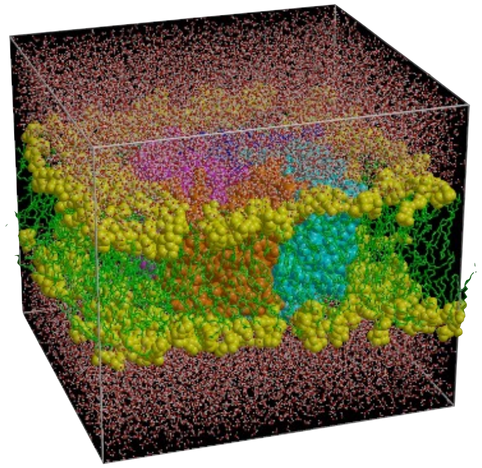
professional GPUs (Tesla)

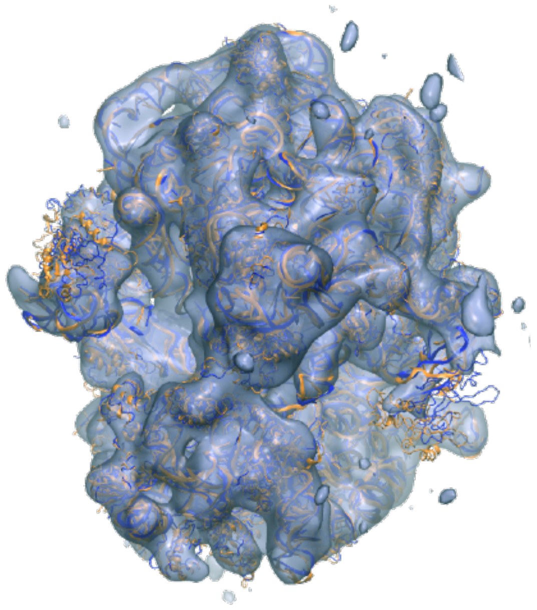
Quadro P6000
Tesla V100

CPUs

Ryzen (16 core)
Epyc (24 core)
Core i7 (4 core)
Xeon (4, 6, 8, 10, and 20 core)

80k atom MEM benchmark
channel in membrane + water + ions, PME, 2 fs time step

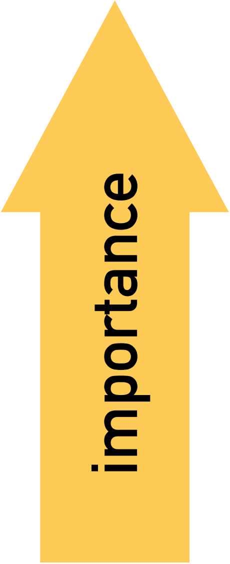




2M atoms RIB benchmark
ribosome in solution, PME, 4 fs time step

What do we really want?

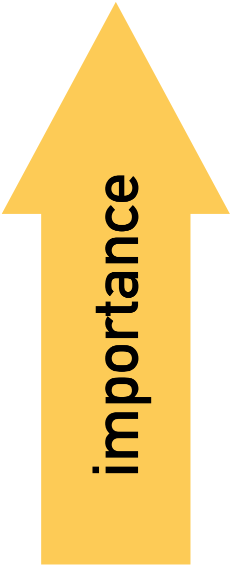
Hardware requirements:


importance

1. high performance-to-price (P/P) ratio
2. low energy consumption
3. low rack space requirements
   packing density at least 1 GPU per U
4. reasonably high performance of a single simulation
   ➔ one simulation per GPU on GPU nodes,
   one simulation per node on CPU nodes

What do we really want?

Hardware requirements:



1. high performance-to-price (P/P) ratio

2. low energy consumption

3. low rack space requirements
   packing density at least 1 GPU per U

4. reasonably high performance of a single simulation
   ➔ one simulation per GPU on GPU nodes,
   one simulation per node on CPU nodes

best resembles how the cluster is actually used

Details for the hardware comparison benchmarks

- **GROMACS 2018**

- GCC 6.4 + CUDA 9.1
- GCC 5.4 + CUDA 8.0
  (~2.5% slower, taken into account)

- AVX2_128 SIMD for AMD CPUs
- AVX2_256 SIMD for recent Intel CPUs
  - (AVX_256 SIMD for old Intel CPUs)

- OpenMP enabled

- Nodes with 2, 3, or 4 GPUs:
  - using Intel MPI 2017

- **Nodes booted from a common software image** (Scientific Linux 7.4)

Details for the hardware comparison benchmarks

- **GROMACS 2018**

- GCC 6.4 + CUDA 9.1
- GCC 5.4 + CUDA 8.0
  (~2.5% slower, taken into account)

- AVX2_128 SIMD for AMD CPUs
- AVX2_256 SIMD for recent Intel CPUs
  - (AVX_256 SIMD for old Intel CPUs)

- OpenMP enabled

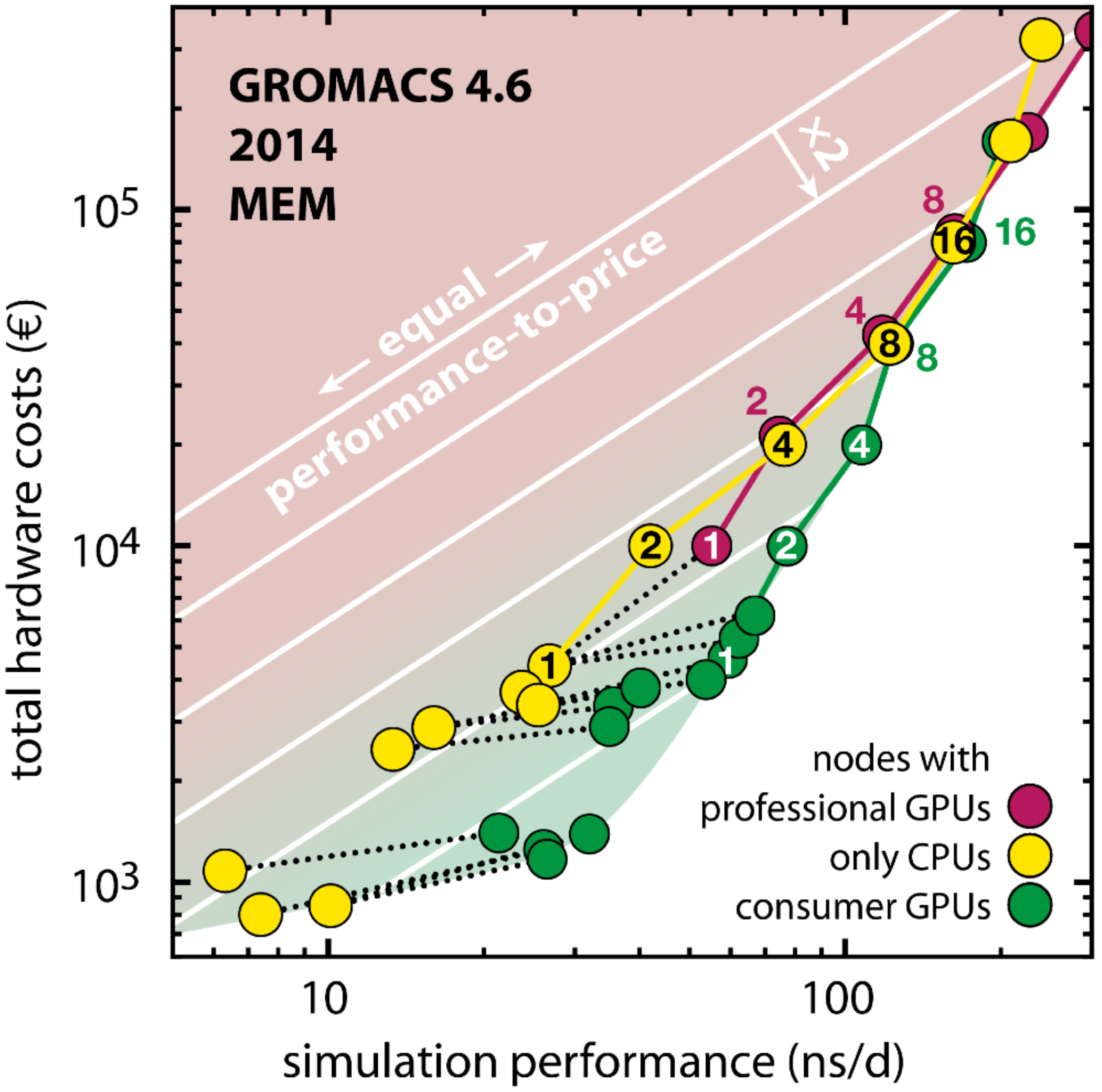
- Nodes with 2, 3, or 4 GPUs:
  - using Intel MPI 2017

- **Nodes booted from a common software image** (Scientific Linux 7.4)

- benchmarks
  - average of two runs
  - MEM: 20,000 steps, average over last 5,000
  - RIB: run for 10,000 steps, average over last 2,000

- on multi-GPU nodes, benchmarks use 1 simulation per GPU (via -multidir),
  - reported node performance (ns/d) is sum of the performances of the individual simulations
    (**“aggregate” performance**)

2014: First Comprehensive Hardware Evaluation



- Main 2014 result:

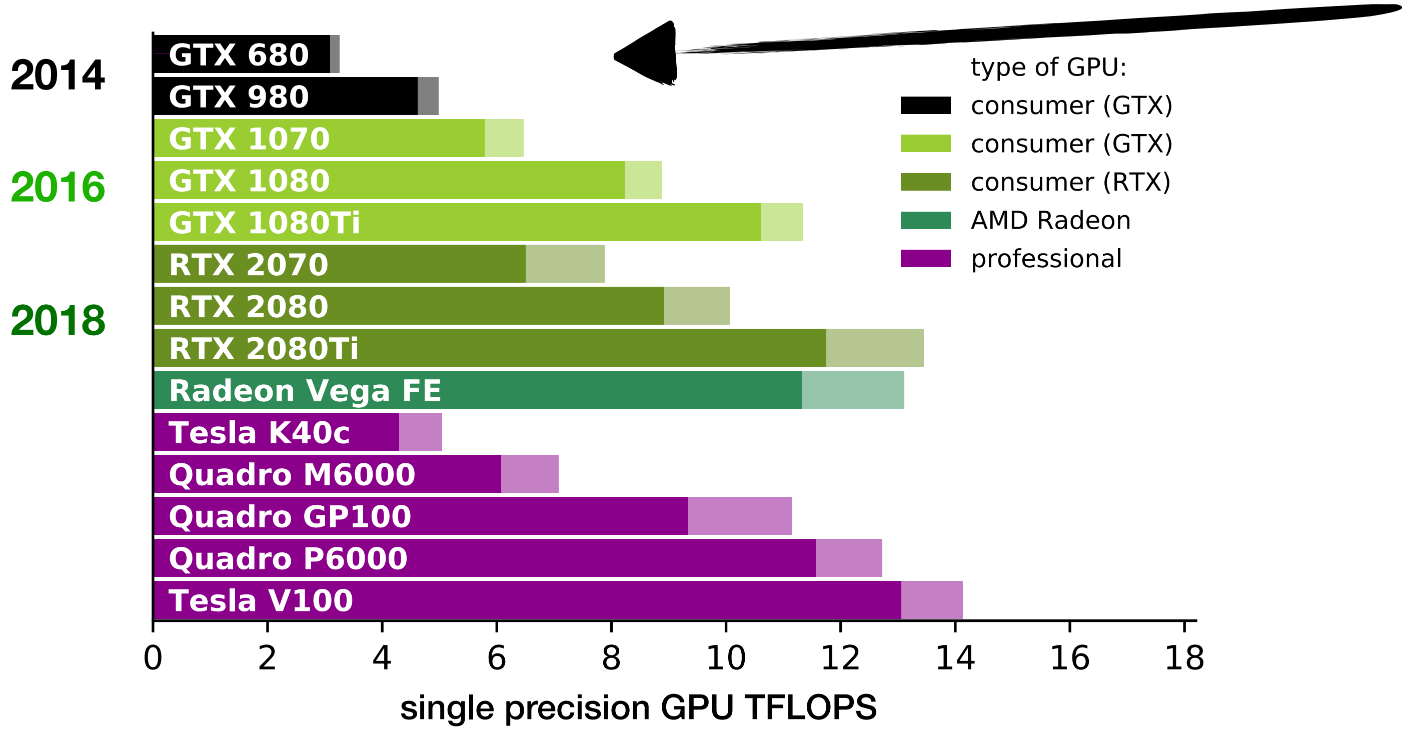
  ● nodes with GeForce consumer GPUs

  produce **2–3x** as much MD trajectory per invested € as

  ● CPU nodes

C Kutzner, S Páll, M Fechner, A Esztermann, BL de Groot, H Grubmüller.
Best bang for your buck: GPU nodes for GROMACS biomolecular simulations.
JCC 36 (26), pp. 1990 - 2008 (2015)

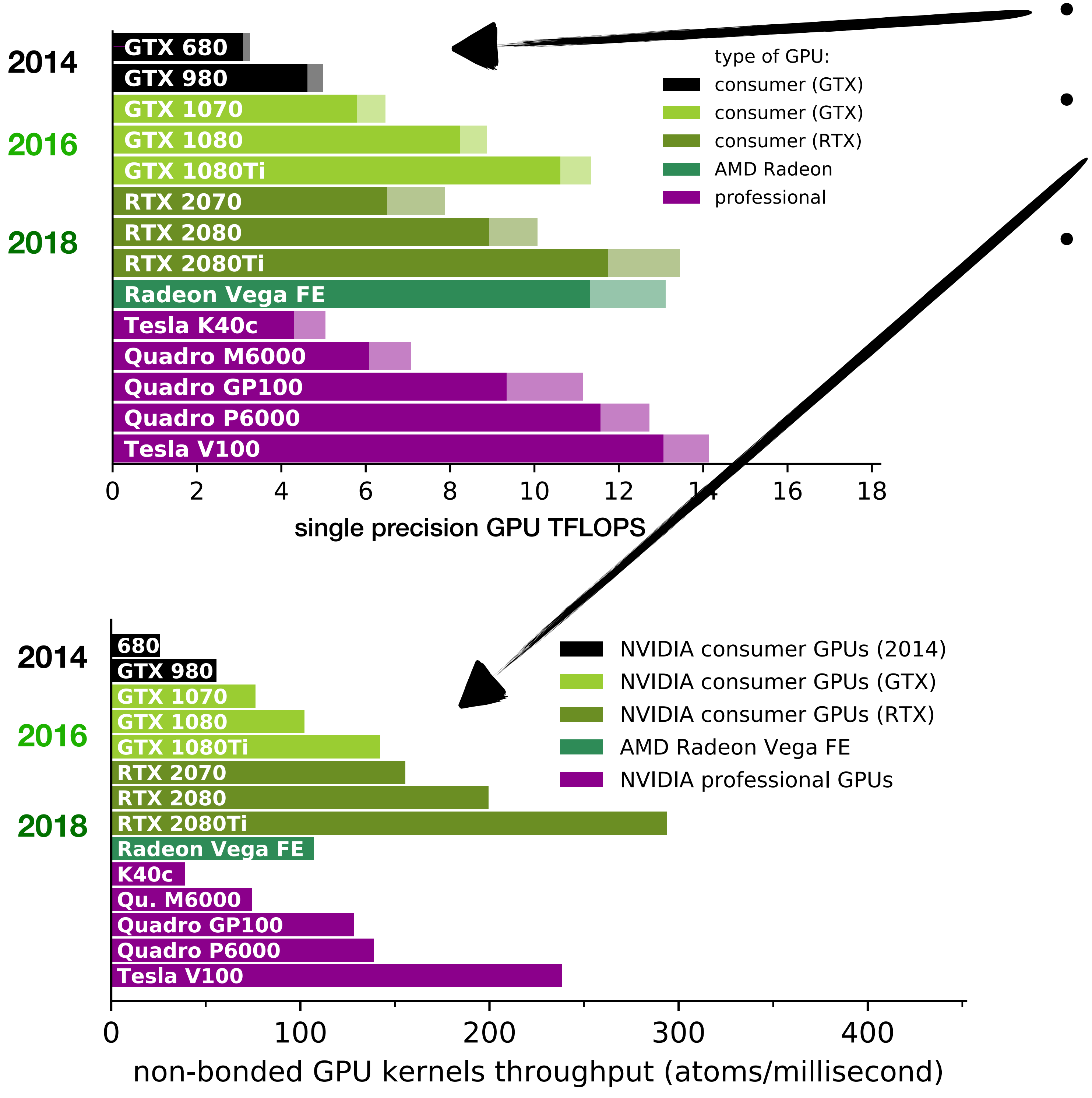
Hardware Developments Since 2014



- FLOP-based GPU processing power x3!

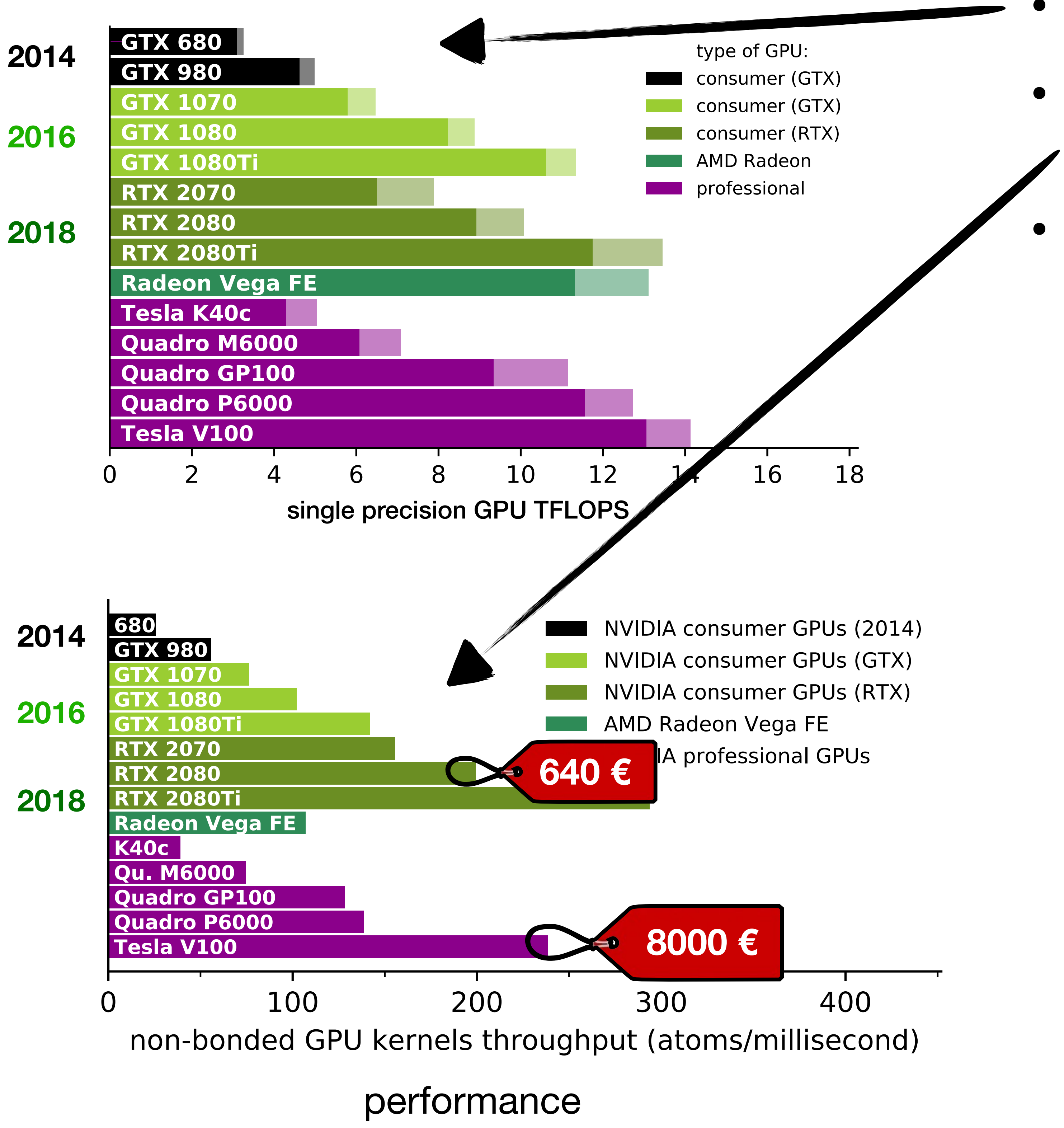
Hardware Developments Since 2014

- FLOP-based GPU processing power x3!
- + microarchitectural improvements: up to **6x performance increase** in GPU kernels
- CPU performance: only modest gains



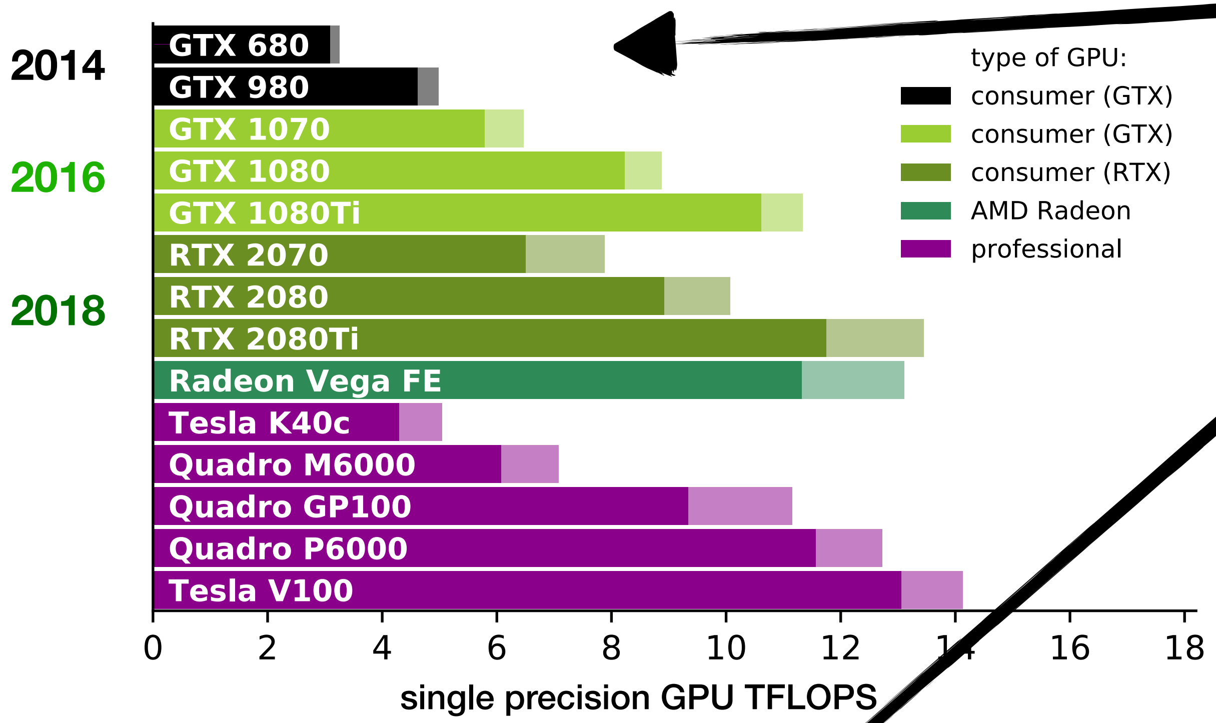
performance

Hardware Developments Since 2014

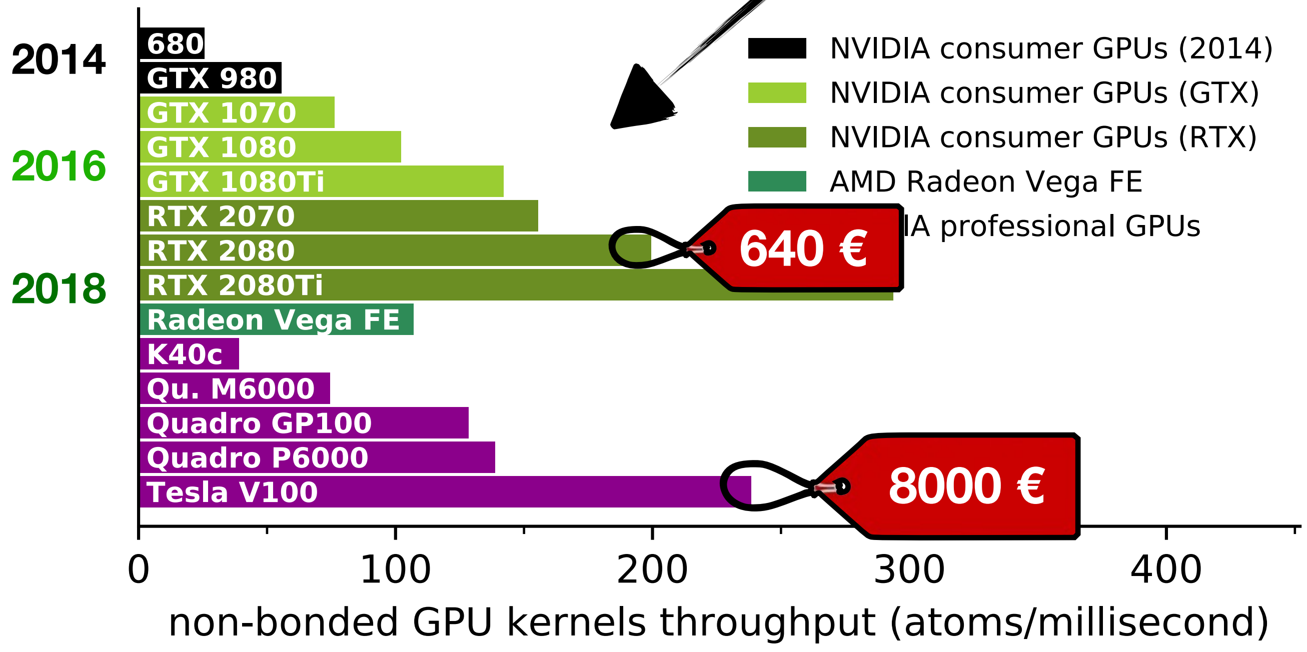


- FLOP-based GPU processing power x3!
- + microarchitectural improvements: up to **6x performance increase** in GPU kernels
- CPU performance: only modest gains

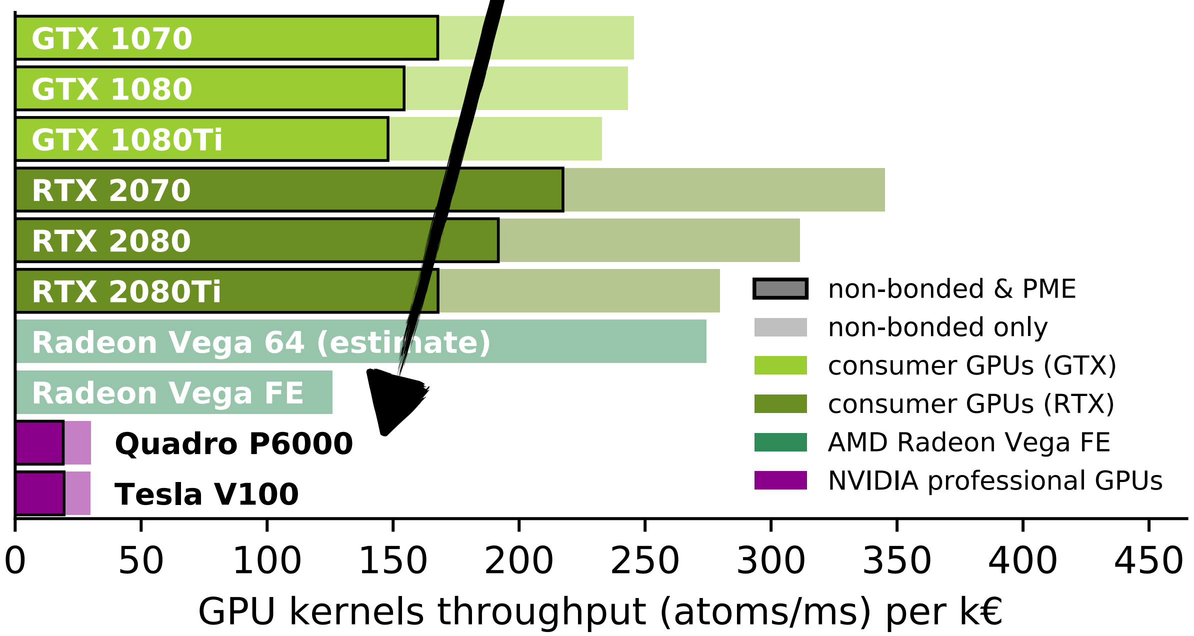
Hardware Developments Since 2014



- FLOP-based GPU processing power x3!
- + microarchitectural improvements: up to **6x performance increase** in GPU kernels
- CPU performance: only modest gains
- **professional Tesla GPUs** compete with **consumer GPUs** in terms of performance, but are lagging far behind in terms of P/P

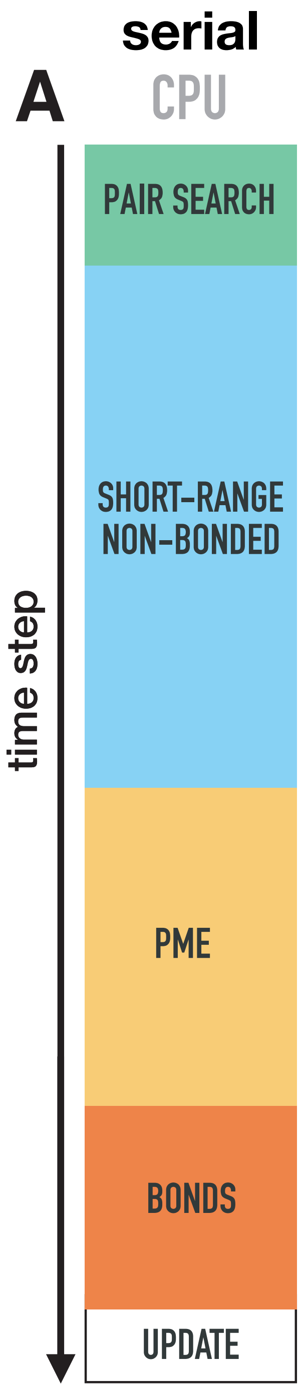


performance

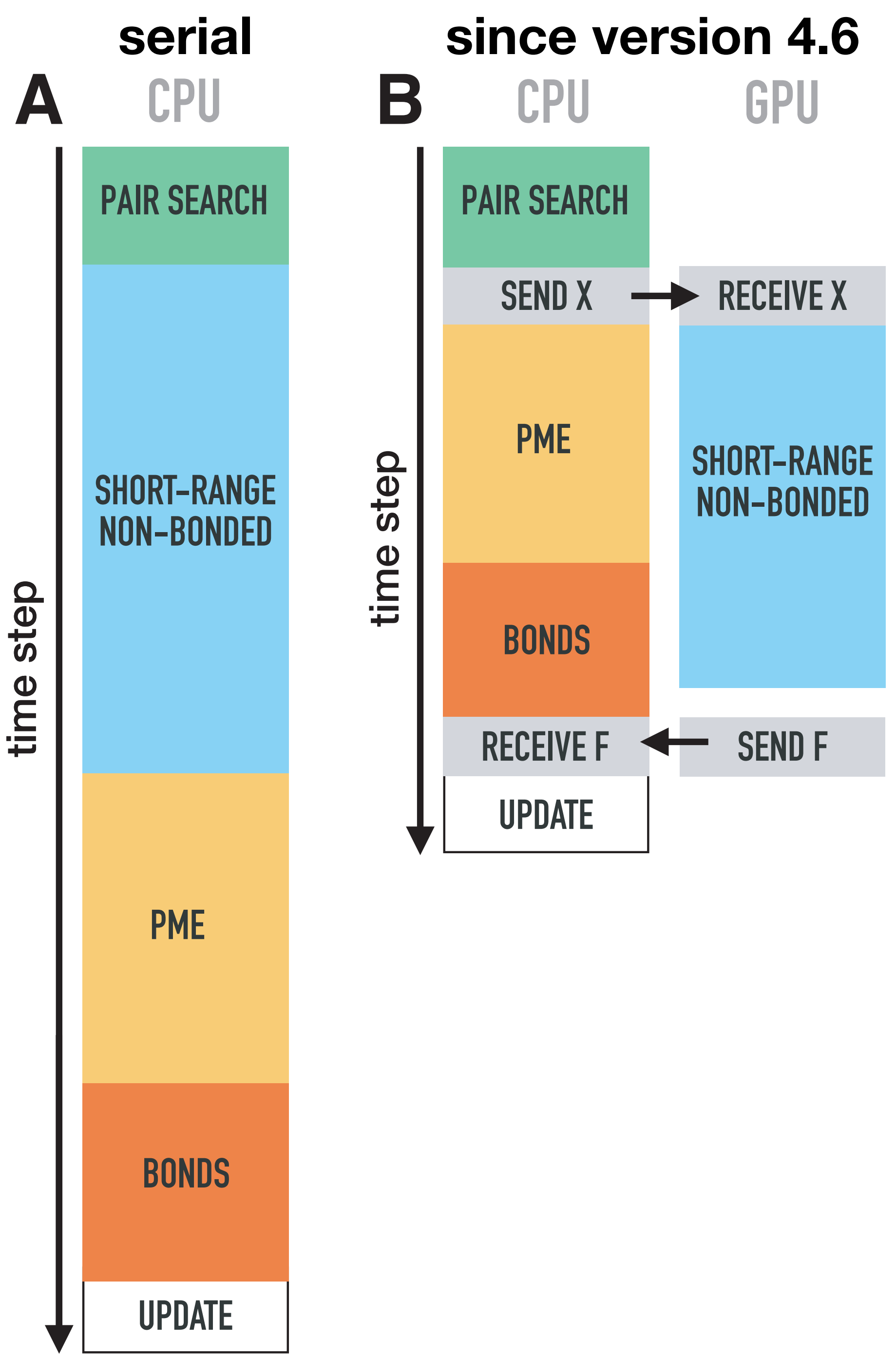


performance / price ratio

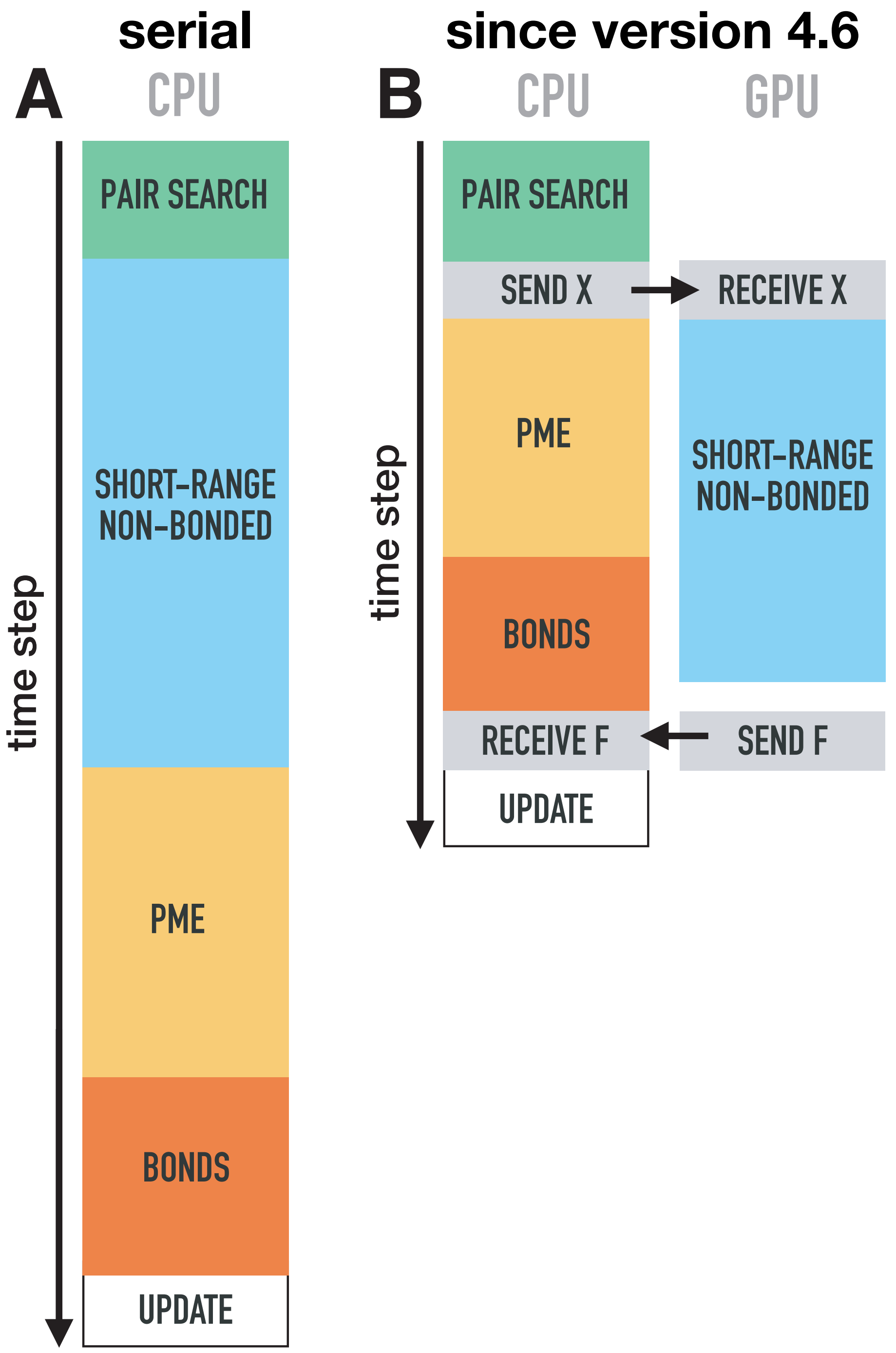
Software Developments



Software Developments



Software Developments



serial

A

CPU

time step

PAIR SEARCH

SHORT-RANGE NON-BONDED

PME

BONDS

UPDATE

since version 4.6

B

CPU

GPU

time step

PAIR SEARCH

SEND X

RECEIVE X

PME

SHORT-RANGE NON-BONDED

BONDS

RECEIVE F

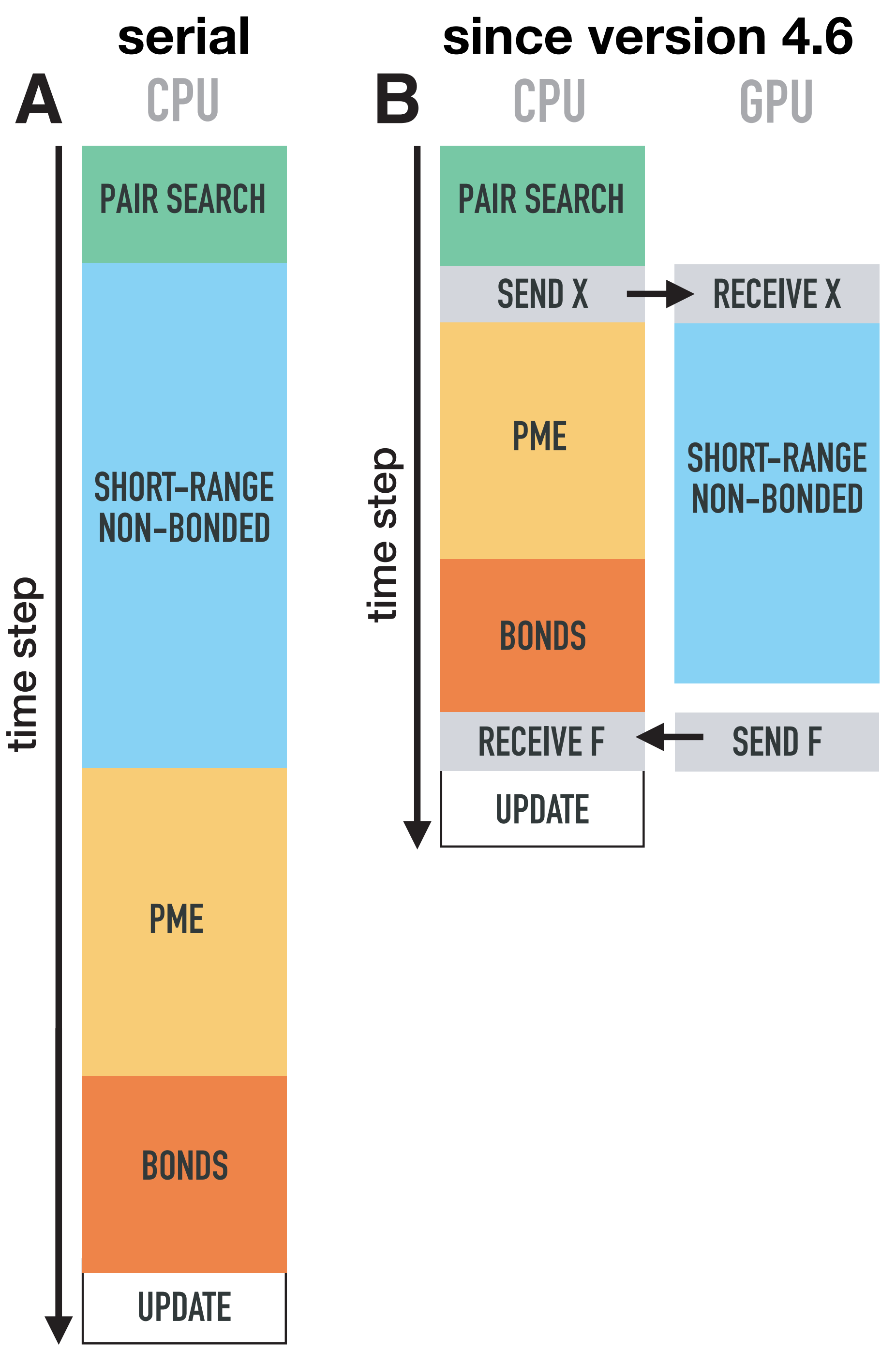
SEND F

UPDATE

since version 2018

1. dual pair lists with dynamic pruning
2. PME offloading

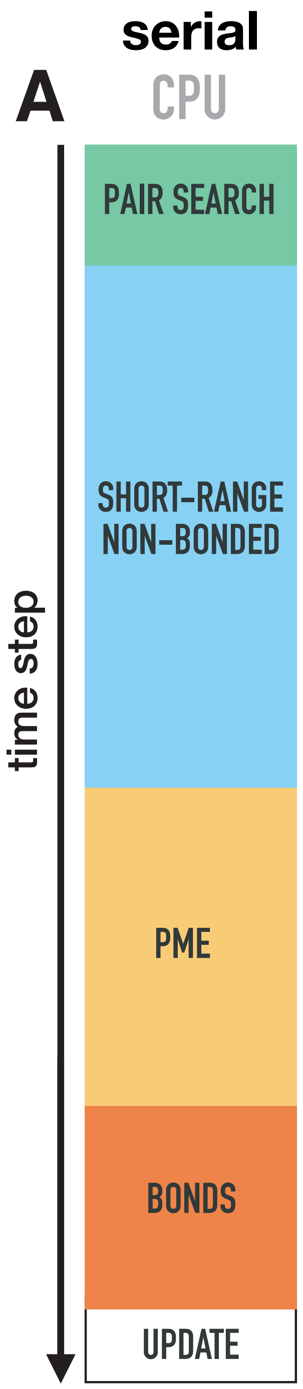
Software Developments

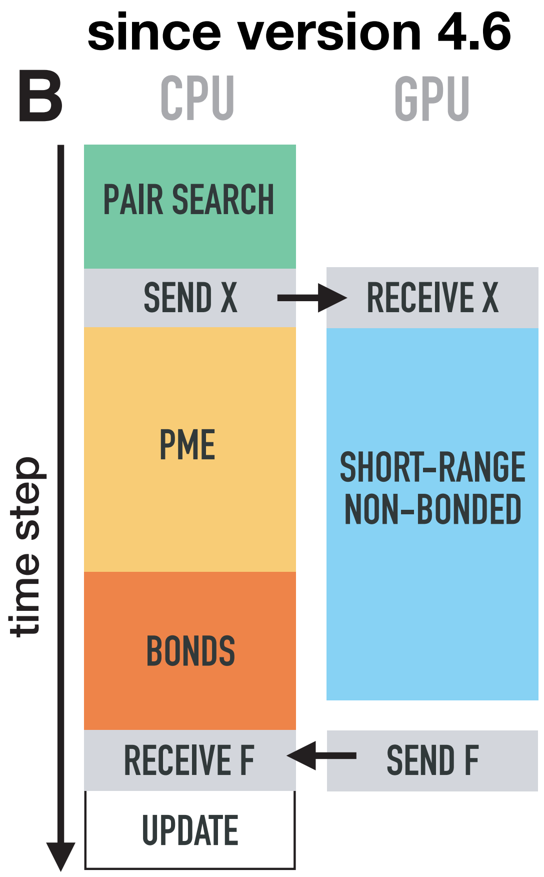


since version 2018

1. **dual pair lists with dynamic pruning**
2. PME offloading

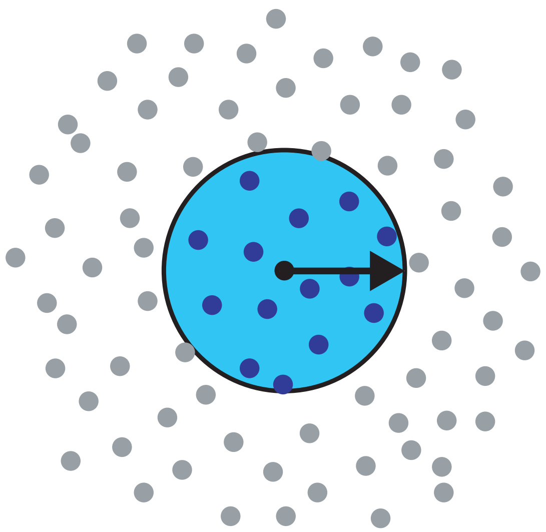
Software Developments





simple pair list

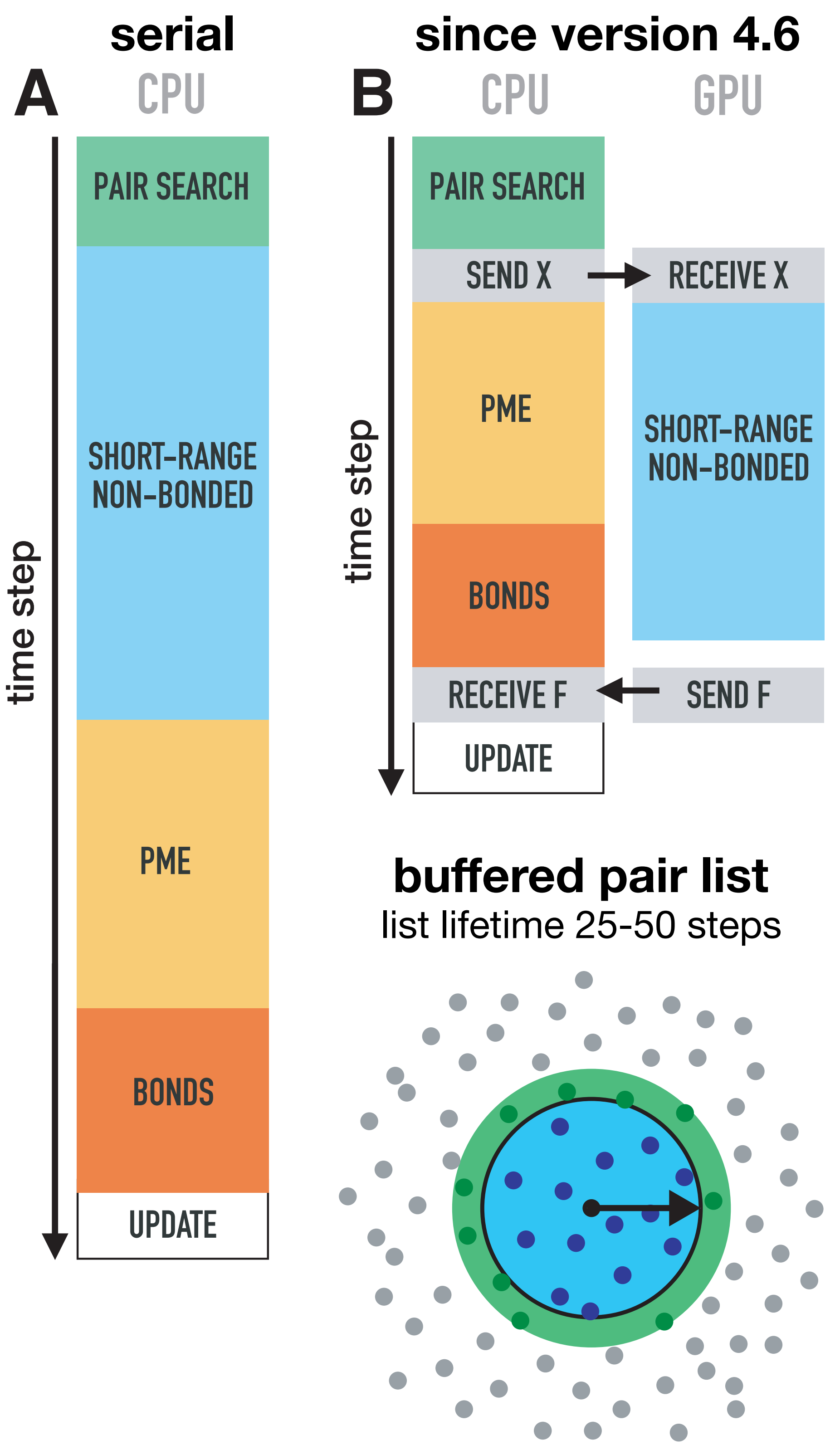
list lifetime 1 step



since version 2018

1. **dual pair lists with dynamic pruning**
2. PME offloading

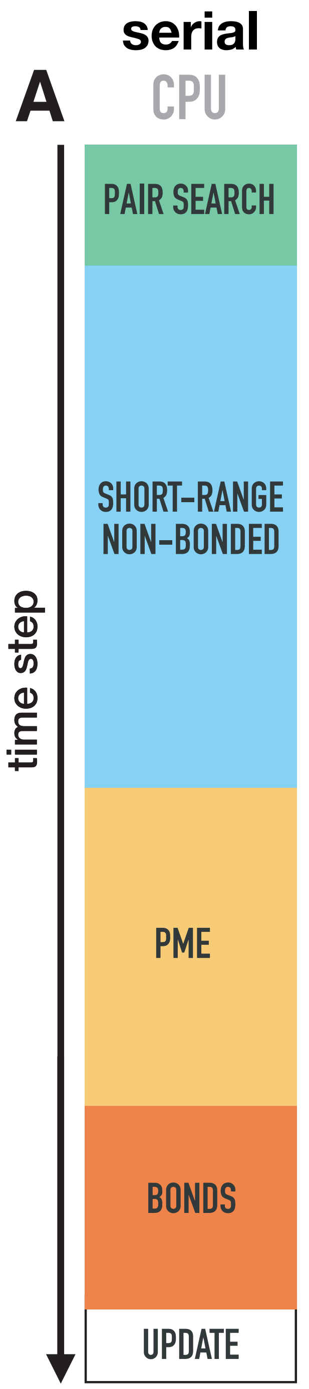
Software Developments



since version 2018

1. **dual pair lists with dynamic pruning**
2. PME offloading

Software Developments



serial

A

CPU

PAIR SEARCH

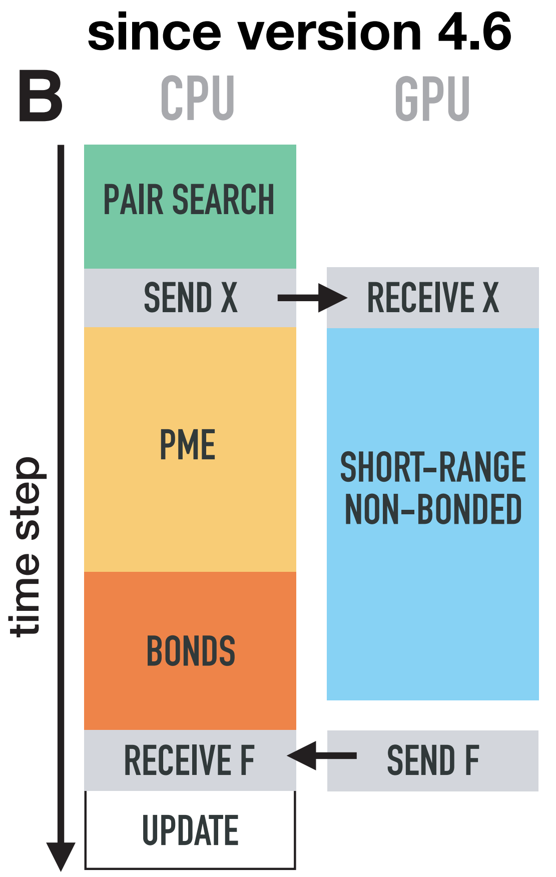
SHORT-RANGE NON-BONDED

PME

BONDS

UPDATE

time step



since version 4.6

B

CPU

GPU

PAIR SEARCH

SEND X → RECEIVE X

PME

SHORT-RANGE NON-BONDED

BONDS

RECEIVE F ← SEND F

UPDATE

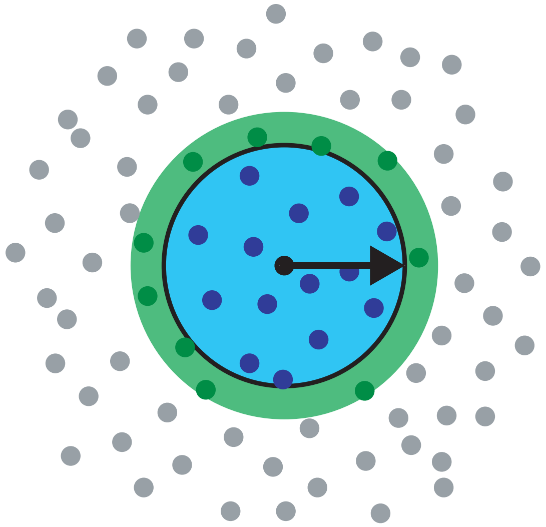
time step

since version 2018

1. **dual pair lists with dynamic pruning**
2. PME offloading

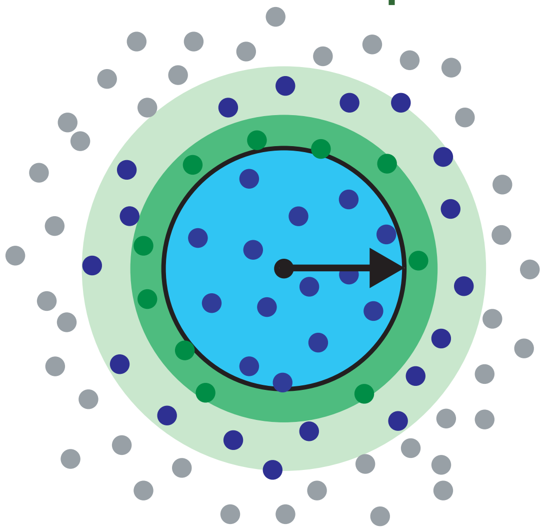
buffered pair list

list lifetime 25-50 steps

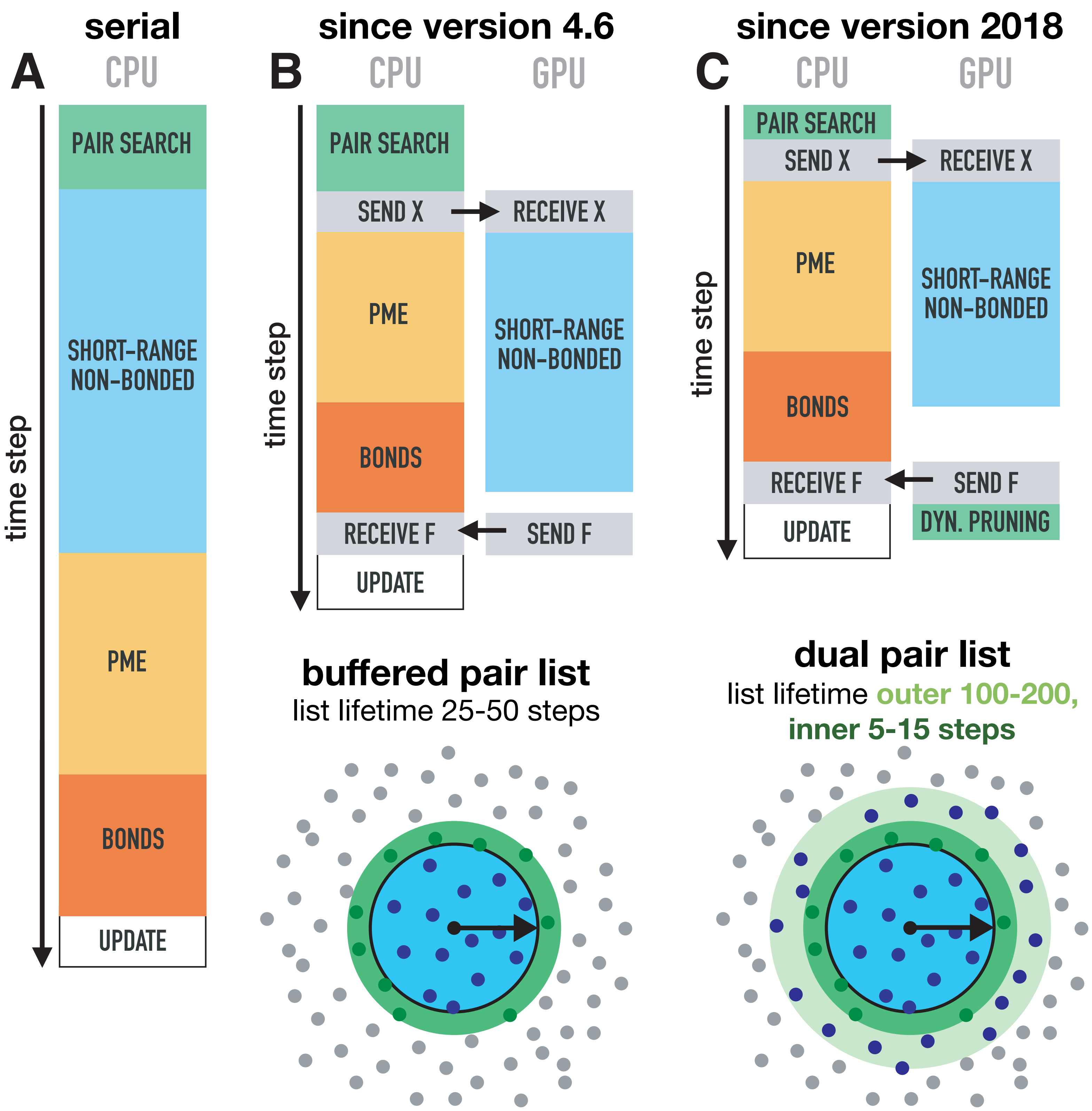


dual pair list

list lifetime outer 100-200, inner 5-15 steps



Software Developments



A serial

CPU

PAIR SEARCH

SHORT-RANGE NON-BONDED

PME

BONDS

UPDATE

time step

B since version 4.6

CPU | GPU

PAIR SEARCH

SEND X → RECEIVE X

PME | SHORT-RANGE NON-BONDED

BONDS

RECEIVE F ← SEND F

UPDATE

time step

C since version 2018

CPU | GPU

PAIR SEARCH

SEND X → RECEIVE X

PME | SHORT-RANGE NON-BONDED

BONDS

RECEIVE F ← SEND F

UPDATE | DYN. PRUNING

time step

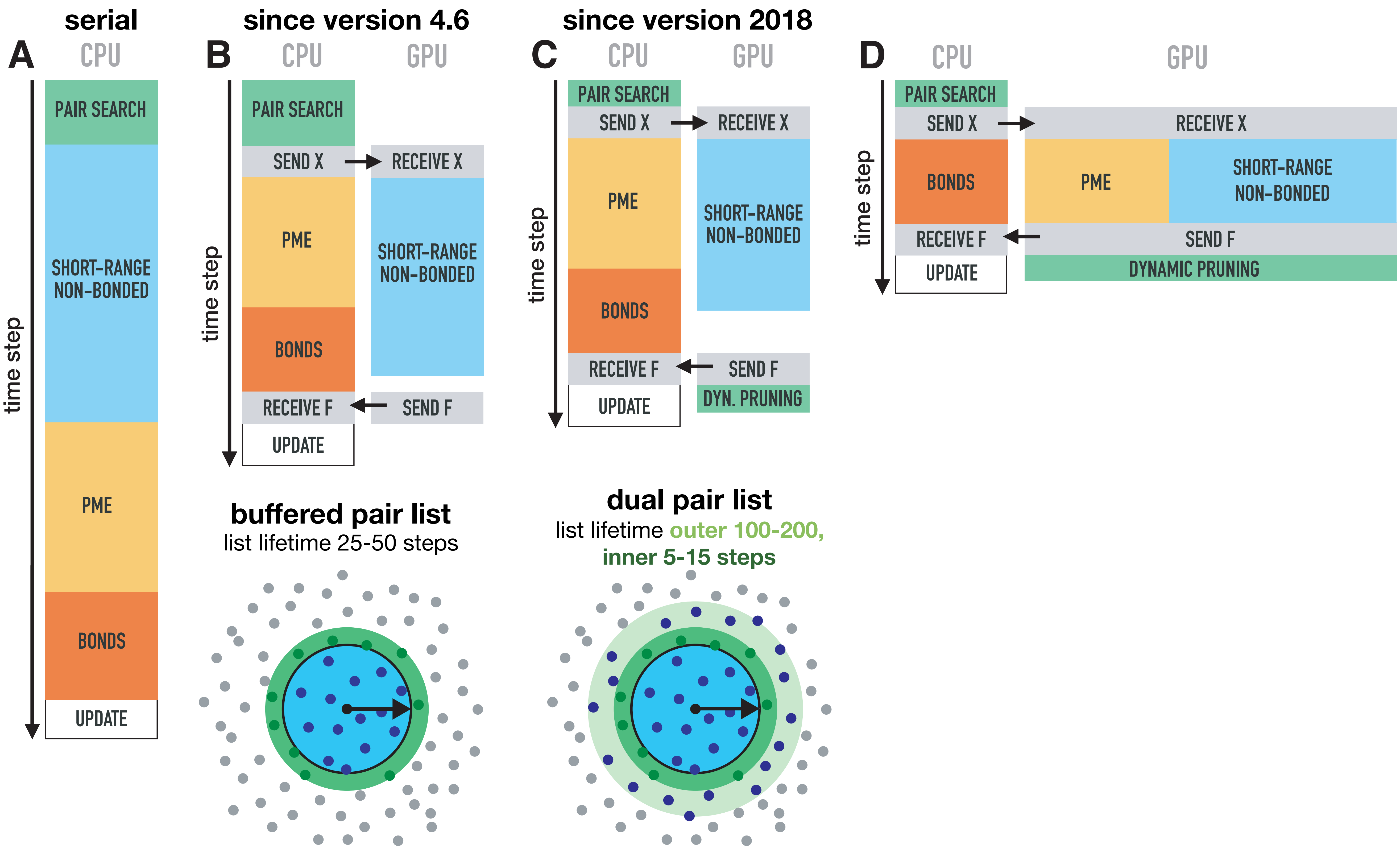
buffered pair list

list lifetime 25-50 steps

dual pair list

list lifetime outer 100-200, inner 5-15 steps

Software Developments



serial

A

CPU

PAIR SEARCH

SHORT-RANGE NON-BONDED

PME

BONDS

UPDATE

time step

since version 4.6

B

CPU

GPU

PAIR SEARCH

SEND X

RECEIVE X

PME

SHORT-RANGE NON-BONDED

BONDS

RECEIVE F

SEND F

UPDATE

time step

since version 2018

C

CPU

GPU

PAIR SEARCH

SEND X

RECEIVE X

PME

SHORT-RANGE NON-BONDED

BONDS

RECEIVE F

SEND F

UPDATE

DYN. PRUNING

time step

D

CPU

GPU

PAIR SEARCH

SEND X

RECEIVE X

BONDS

PME

SHORT-RANGE NON-BONDED

RECEIVE F

SEND F

UPDATE

DYNAMIC PRUNING

time step

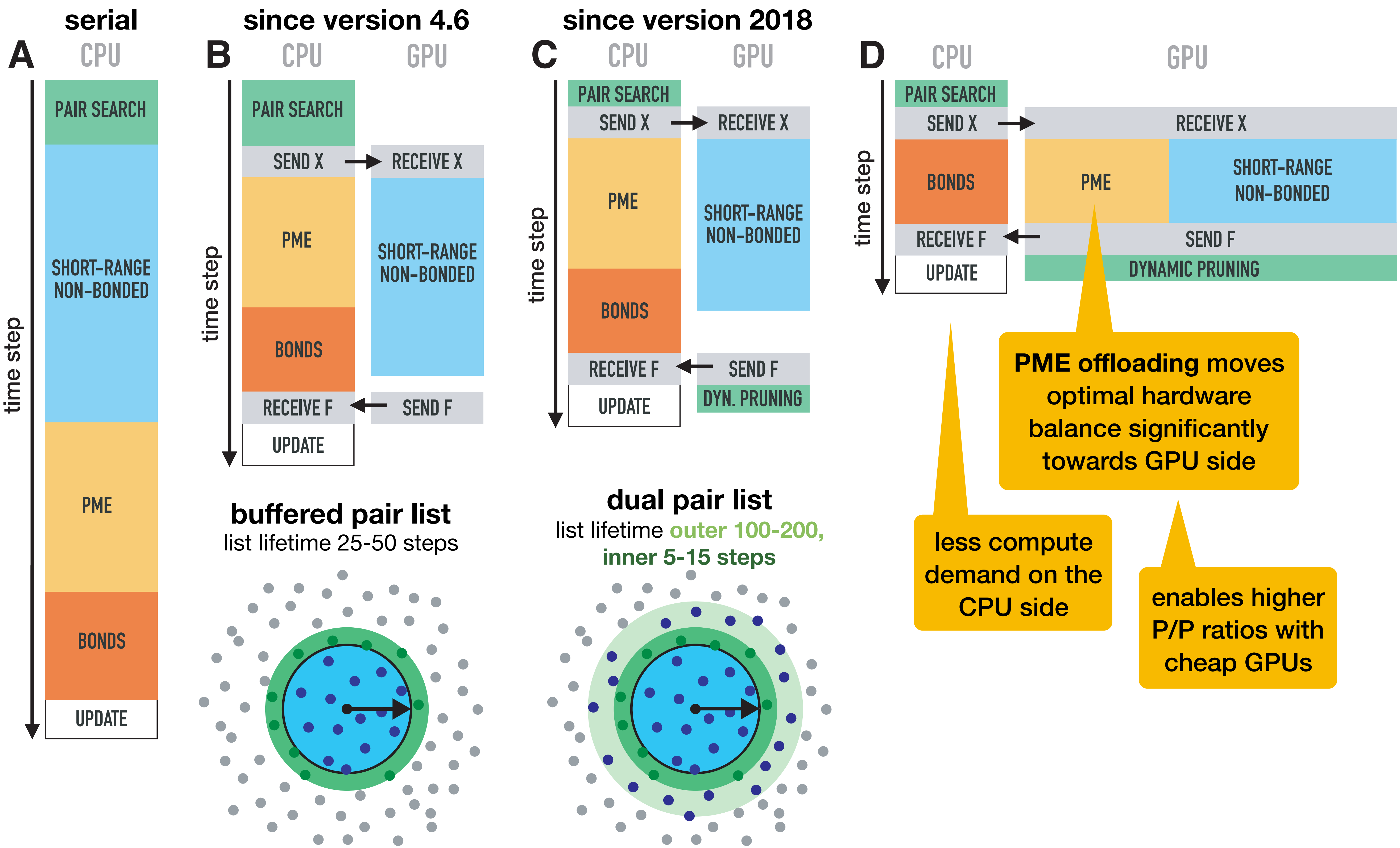
buffered pair list

list lifetime 25-50 steps

dual pair list

list lifetime outer 100-200, inner 5-15 steps

Software Developments



serial

A

CPU

PAIR SEARCH

SHORT-RANGE NON-BONDED

PME

BONDS

UPDATE

time step

since version 4.6

B

CPU	GPU
PAIR SEARCH	
SEND X →	RECEIVE X
PME	SHORT-RANGE NON-BONDED
BONDS	
RECEIVE F ←	SEND F
UPDATE	

time step

since version 2018

C

CPU	GPU
PAIR SEARCH	
SEND X →	RECEIVE X
PME	SHORT-RANGE NON-BONDED
BONDS	
RECEIVE F ←	SEND F
UPDATE	DYN. PRUNING

time step

D

CPU	GPU	
PAIR SEARCH		
SEND X →	RECEIVE X	
BONDS	PME	SHORT-RANGE NON-BONDED
RECEIVE F ←	SEND F	
UPDATE	DYNAMIC PRUNING	

time step

buffered pair list

list lifetime 25-50 steps

dual pair list

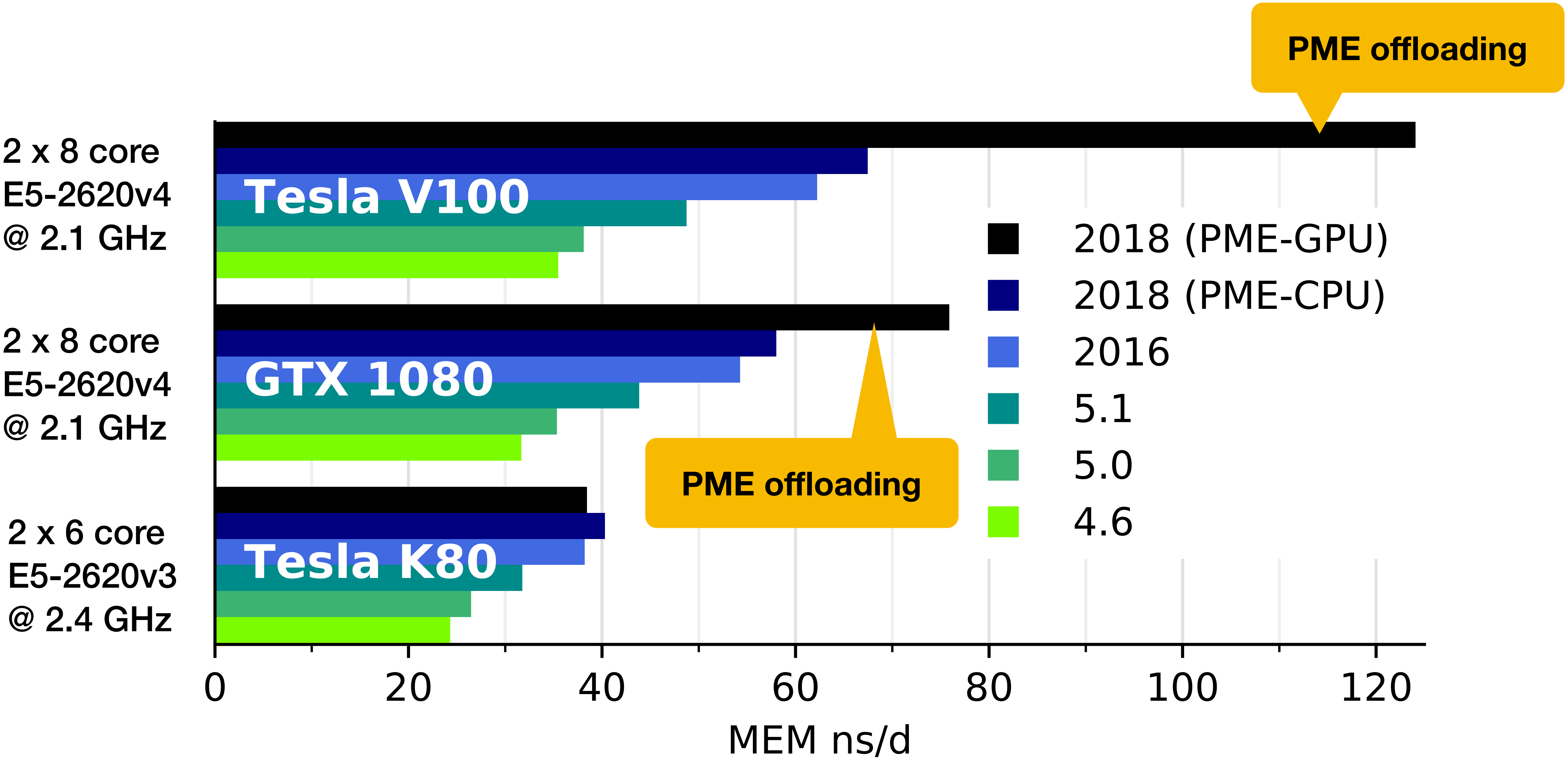
list lifetime outer 100-200, inner 5-15 steps

PME offloading moves optimal hardware balance significantly towards GPU side

less compute demand on the CPU side

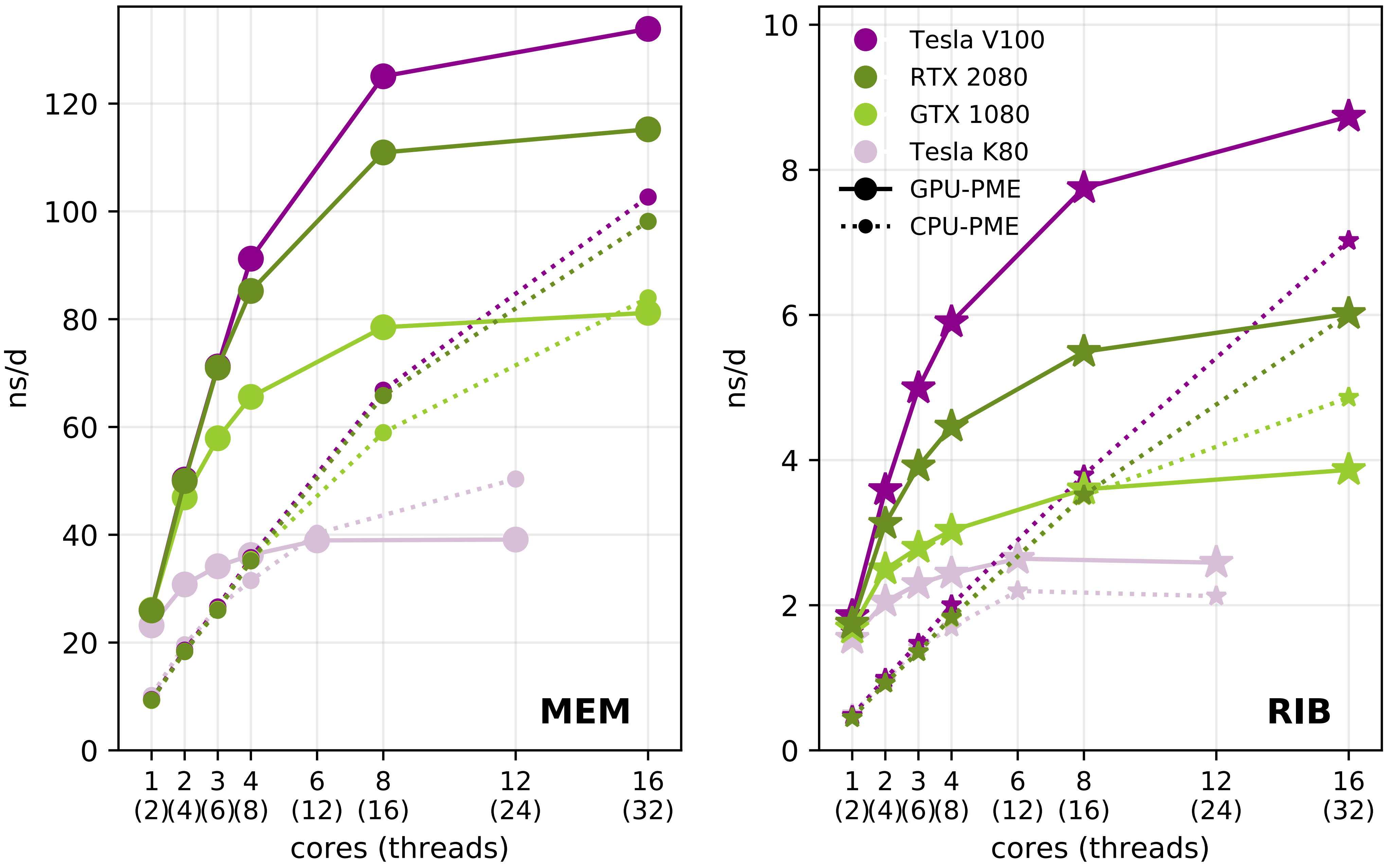
enables higher P/P ratios with cheap GPUs

GROMACS performance evolution on GPU nodes



- most pronounced increase in performance with PME offloading (given a strong enough GPU)

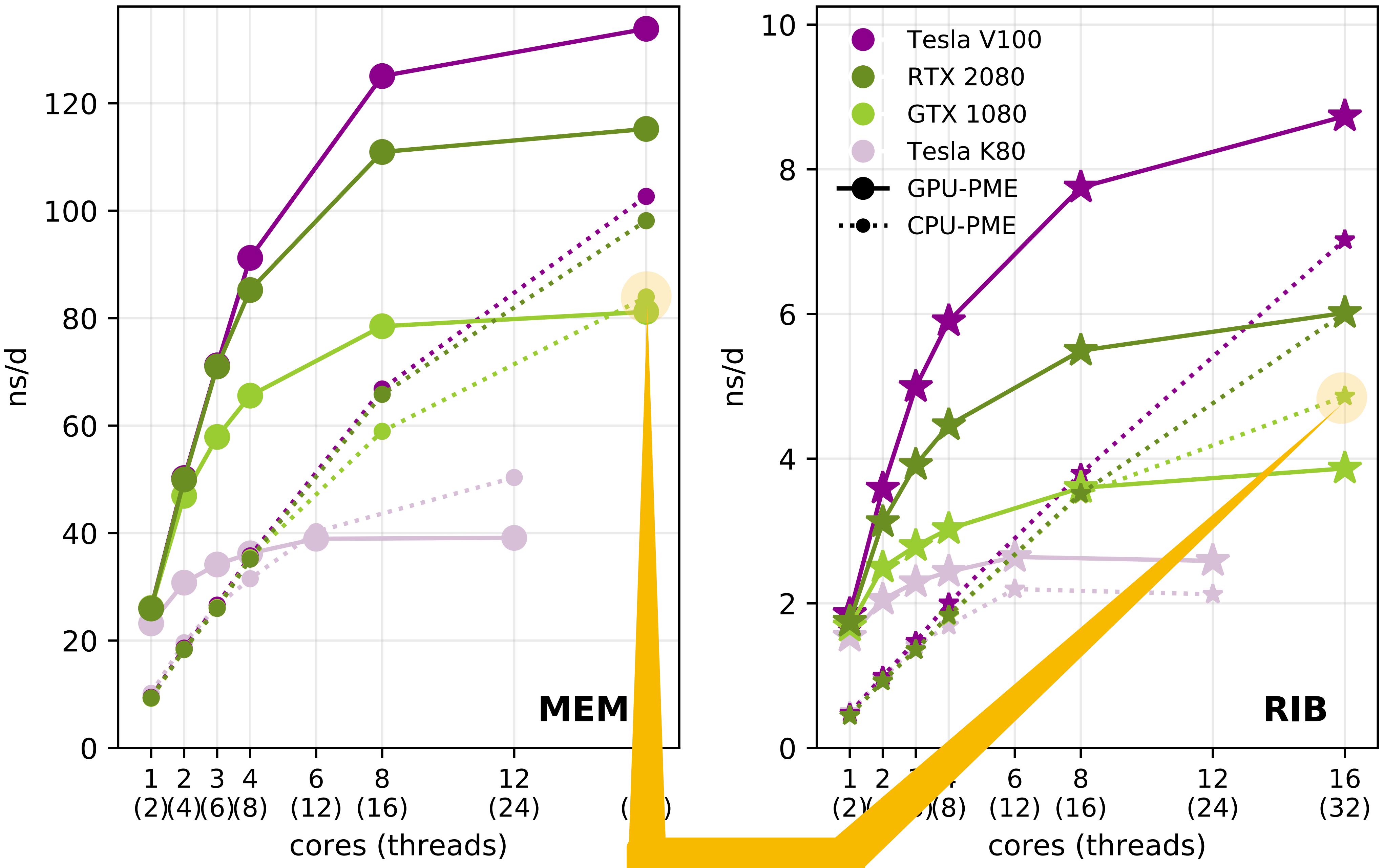
Performance as a function of CPU cores per GPU



2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

Performance as a function of CPU cores per GPU

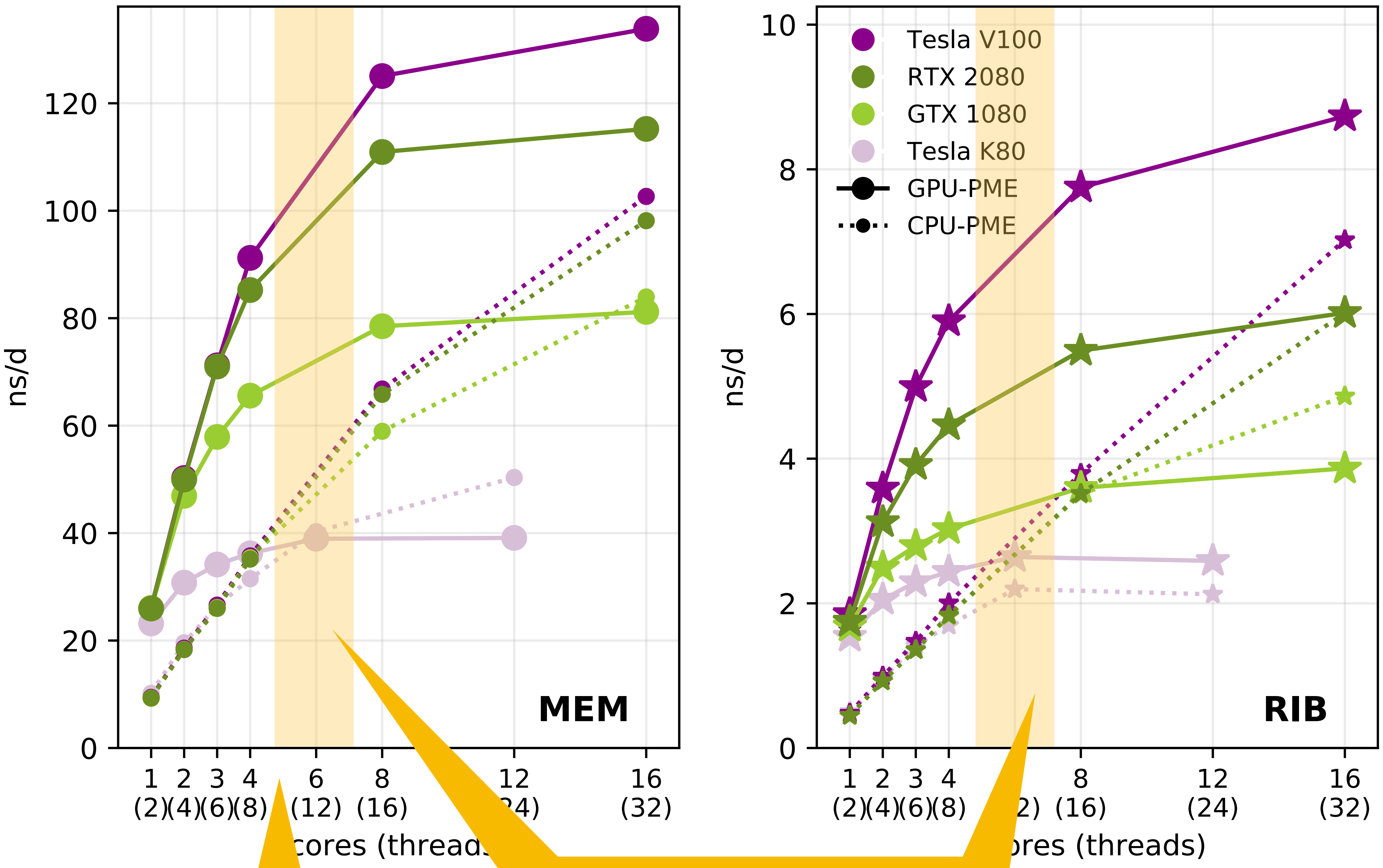


2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

PME on CPU faster only for strong CPU

Performance as a function of CPU cores per GPU



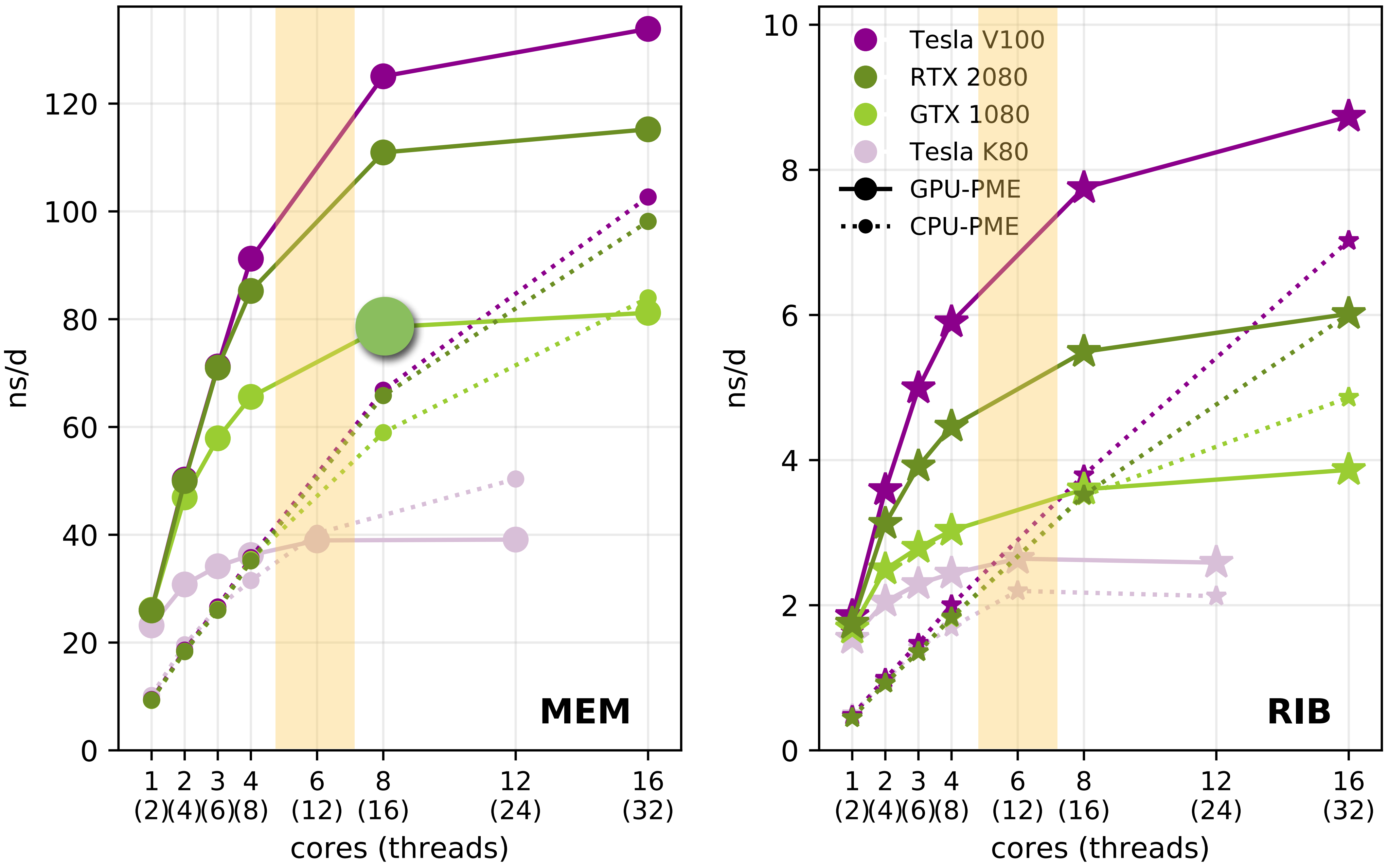
2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

with PME offloading, far less (4–6) cores are needed to reach >80% peak simulation performance

10-15 „core-GHz“ suffice with a mid- to high-end GPU

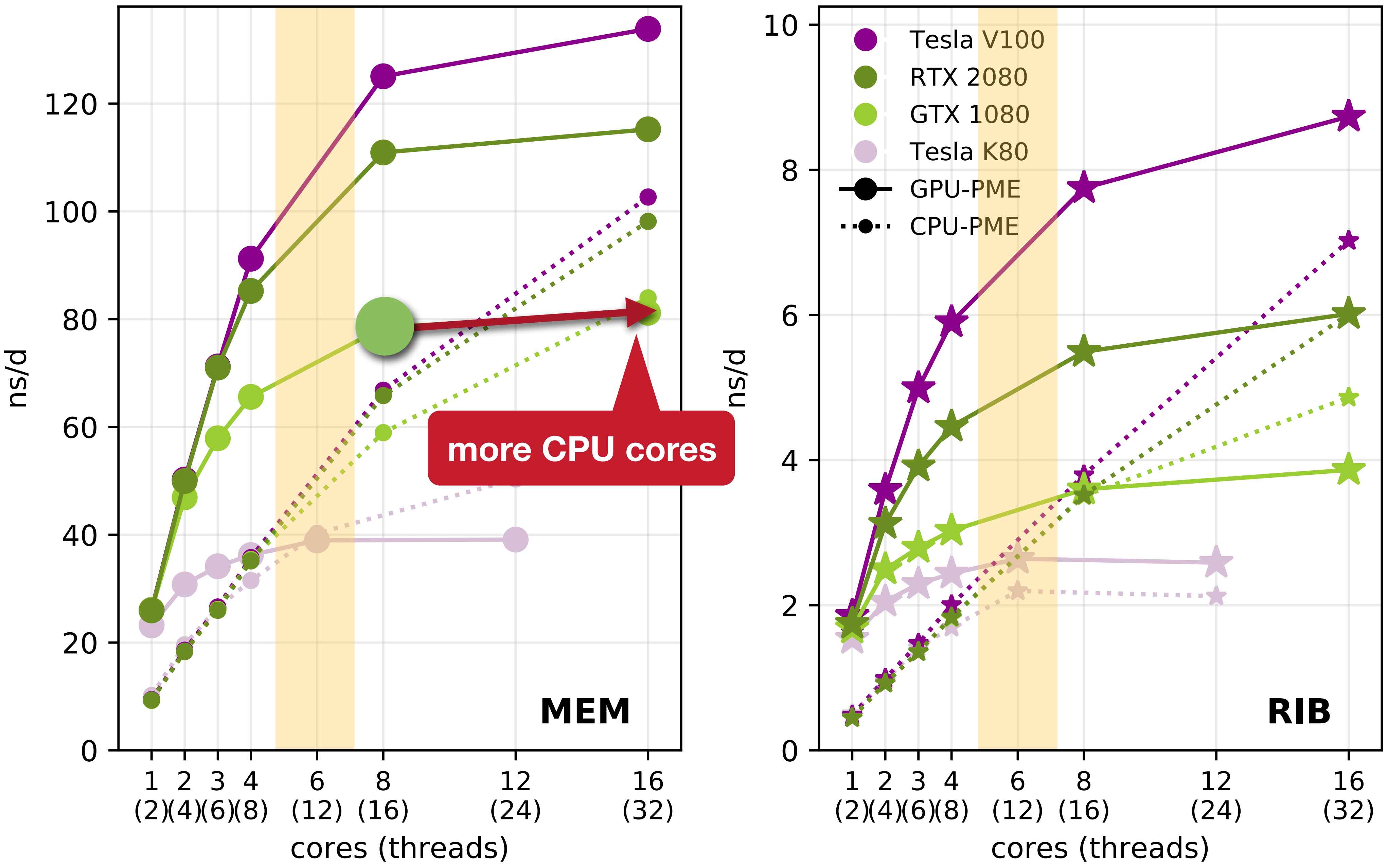
Performance as a function of CPU cores per GPU



2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

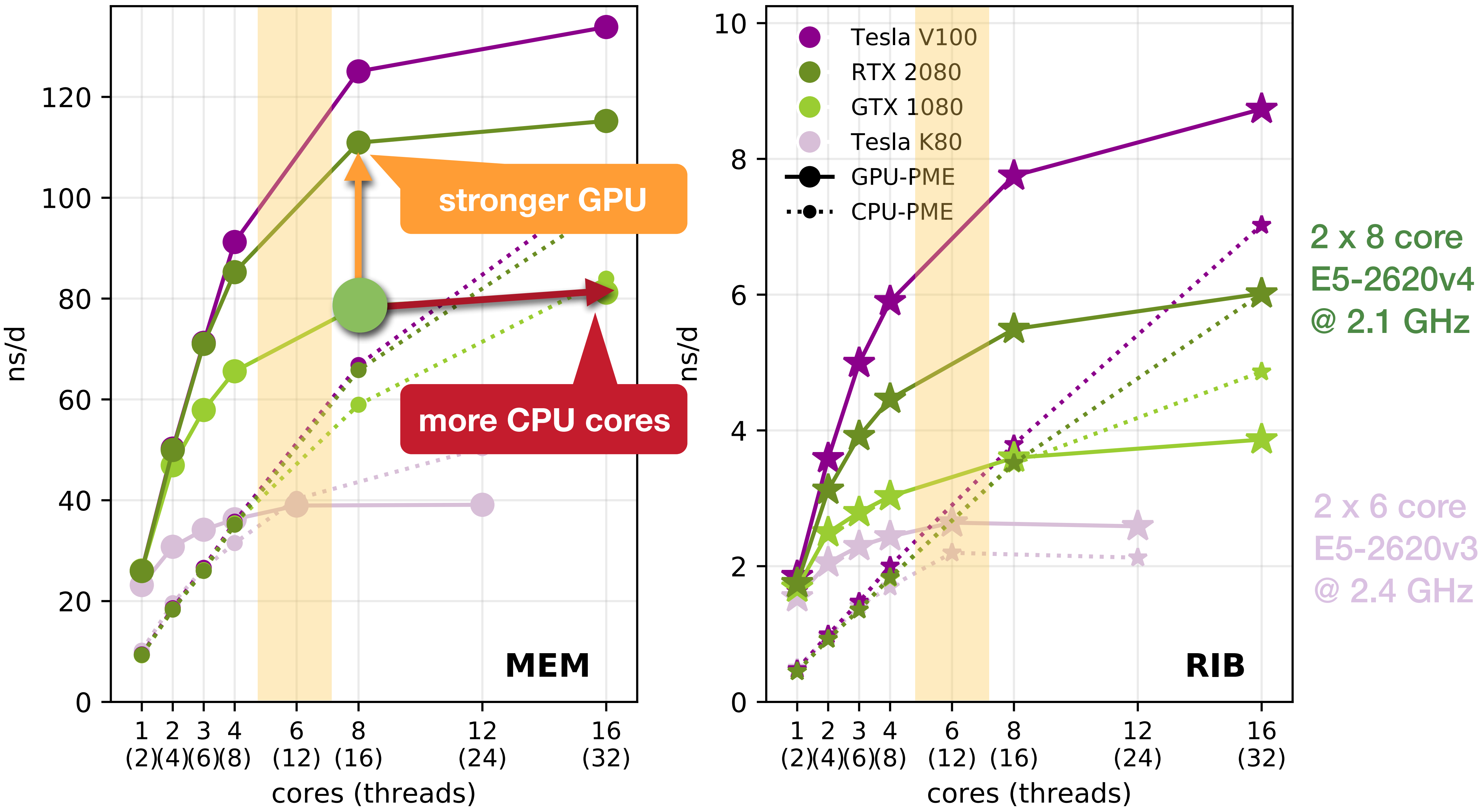
Performance as a function of CPU cores per GPU



2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

Performance as a function of CPU cores per GPU



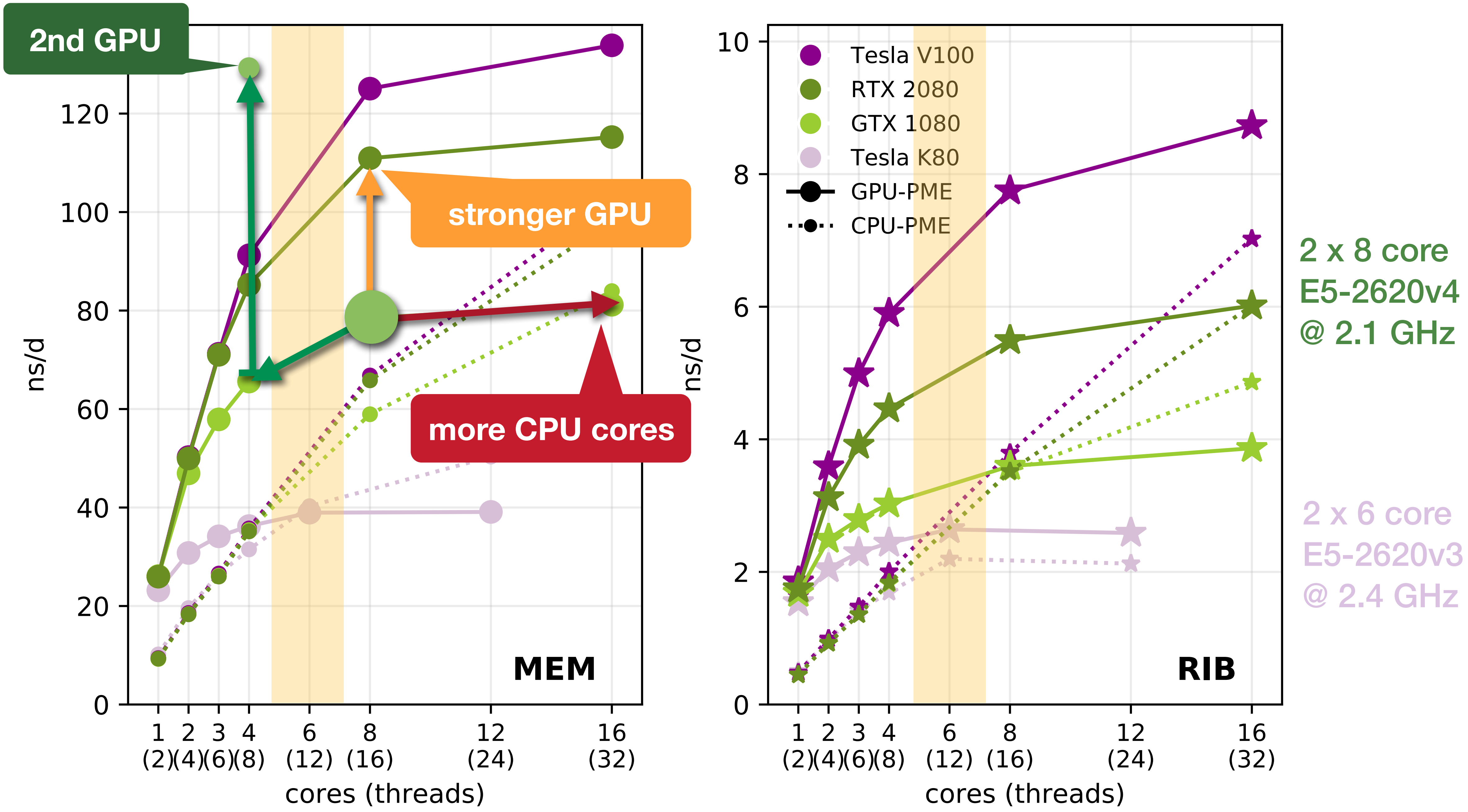
stronger GPU

more CPU cores

2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

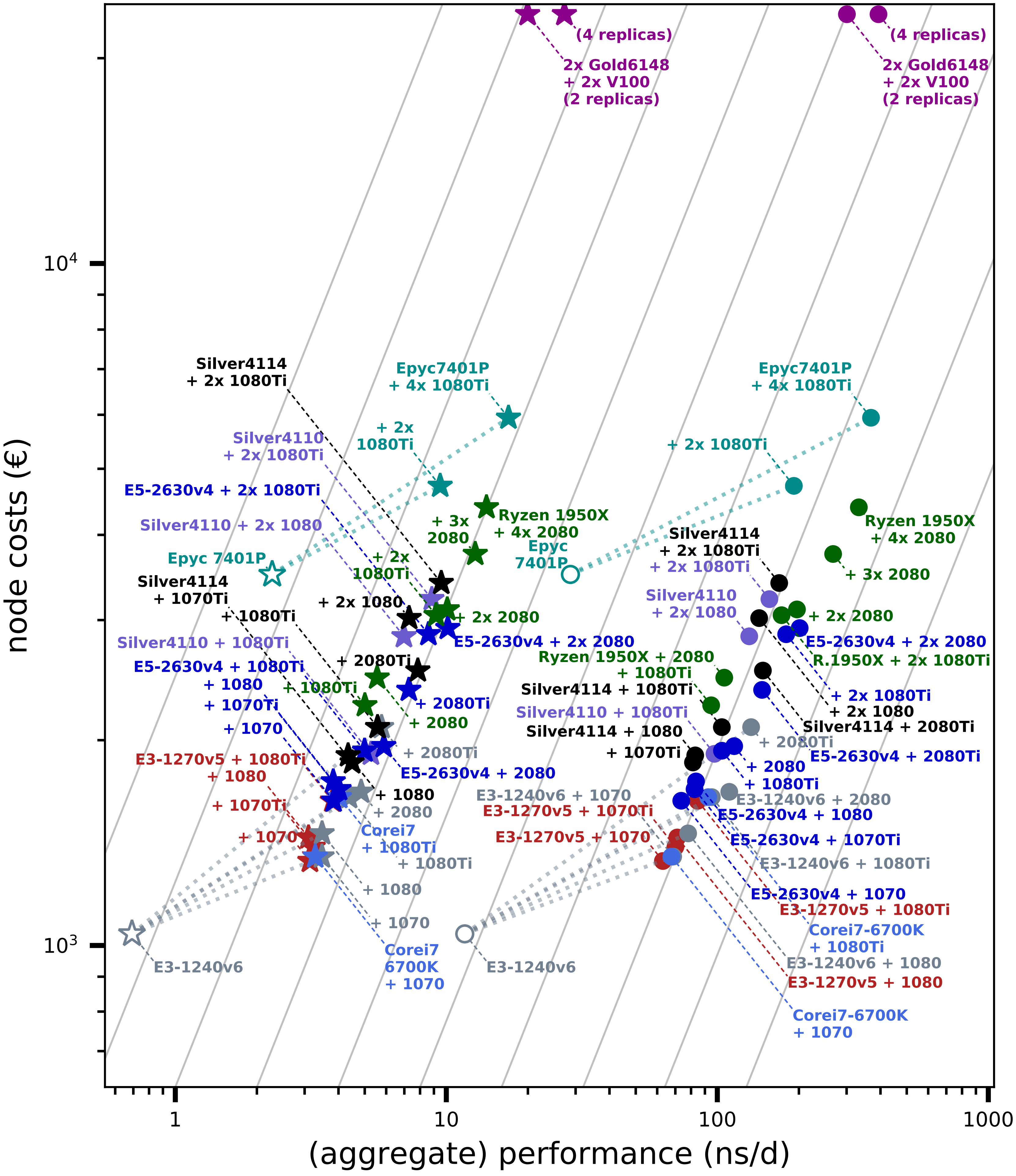
Performance as a function of CPU cores per GPU



2 x 8 core
E5-2620v4
@ 2.1 GHz

2 x 6 core
E5-2620v3
@ 2.4 GHz

Performance in relation to node costs

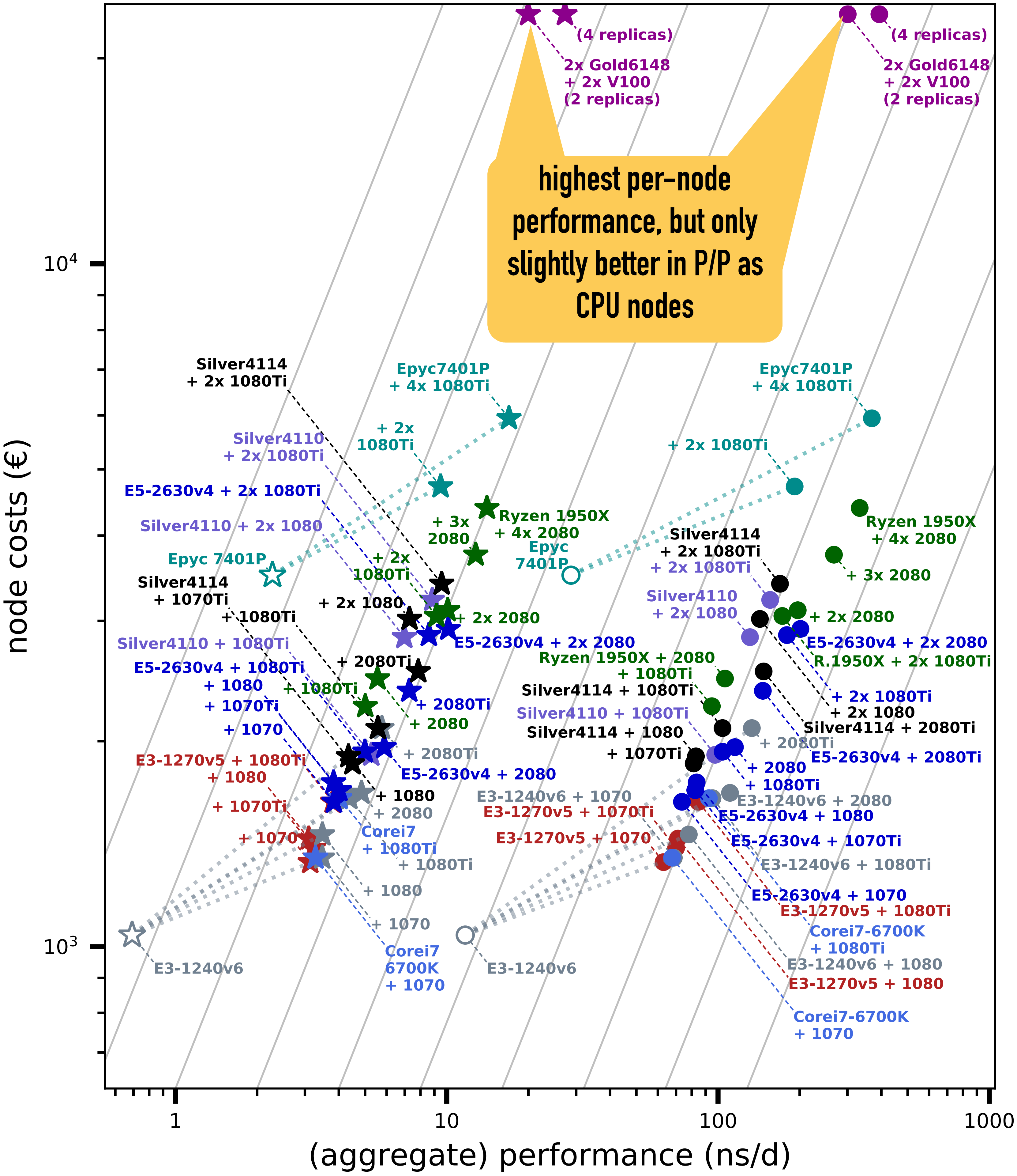


CPU GPU

○···● MEM

☆···★ RIB

Performance in relation to node costs

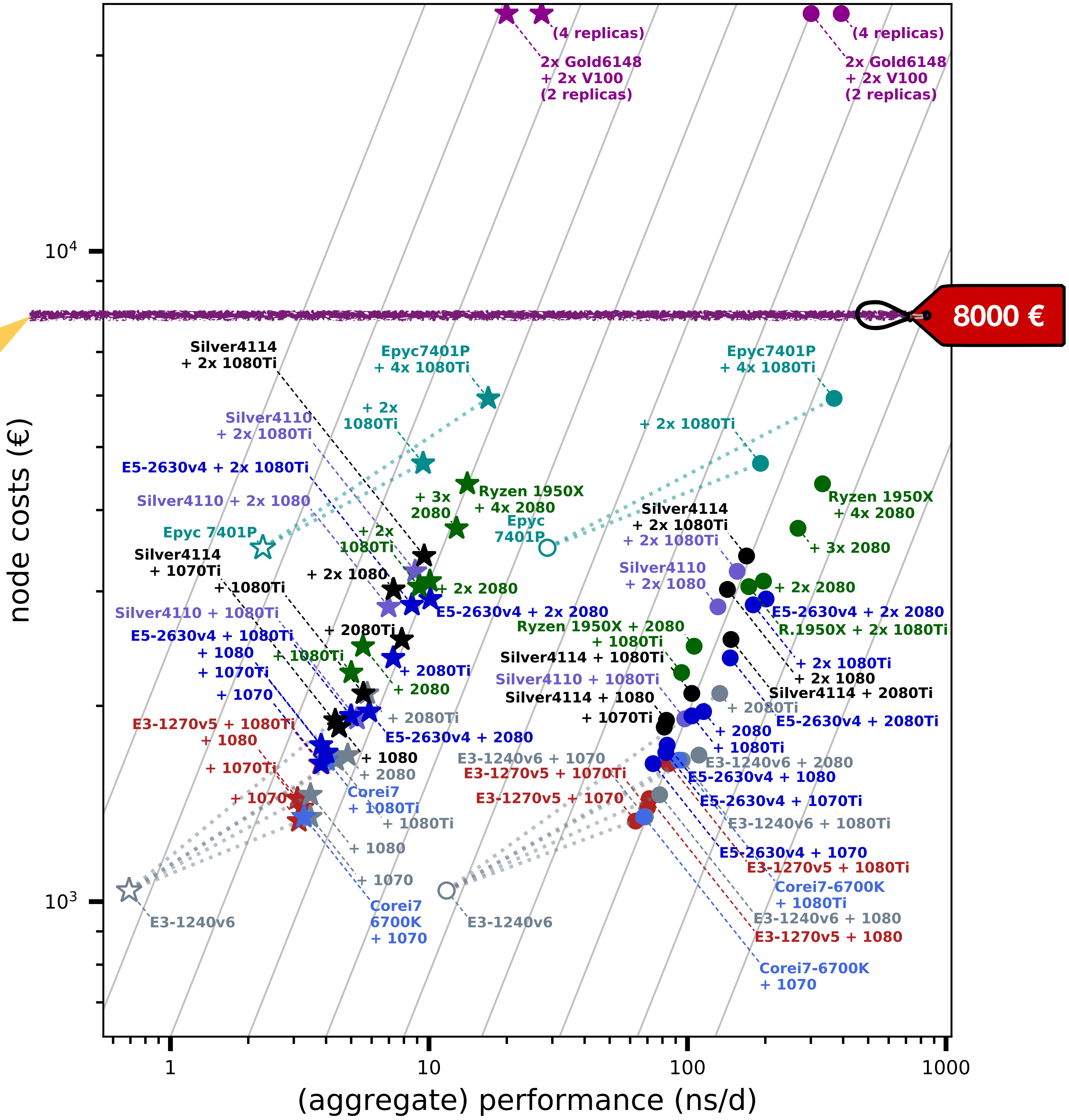


highest per-node performance, but only slightly better in P/P as CPU nodes

CPU		GPU	
○	···	●	MEM
☆	···	★	RIB

Performance in relation to node costs

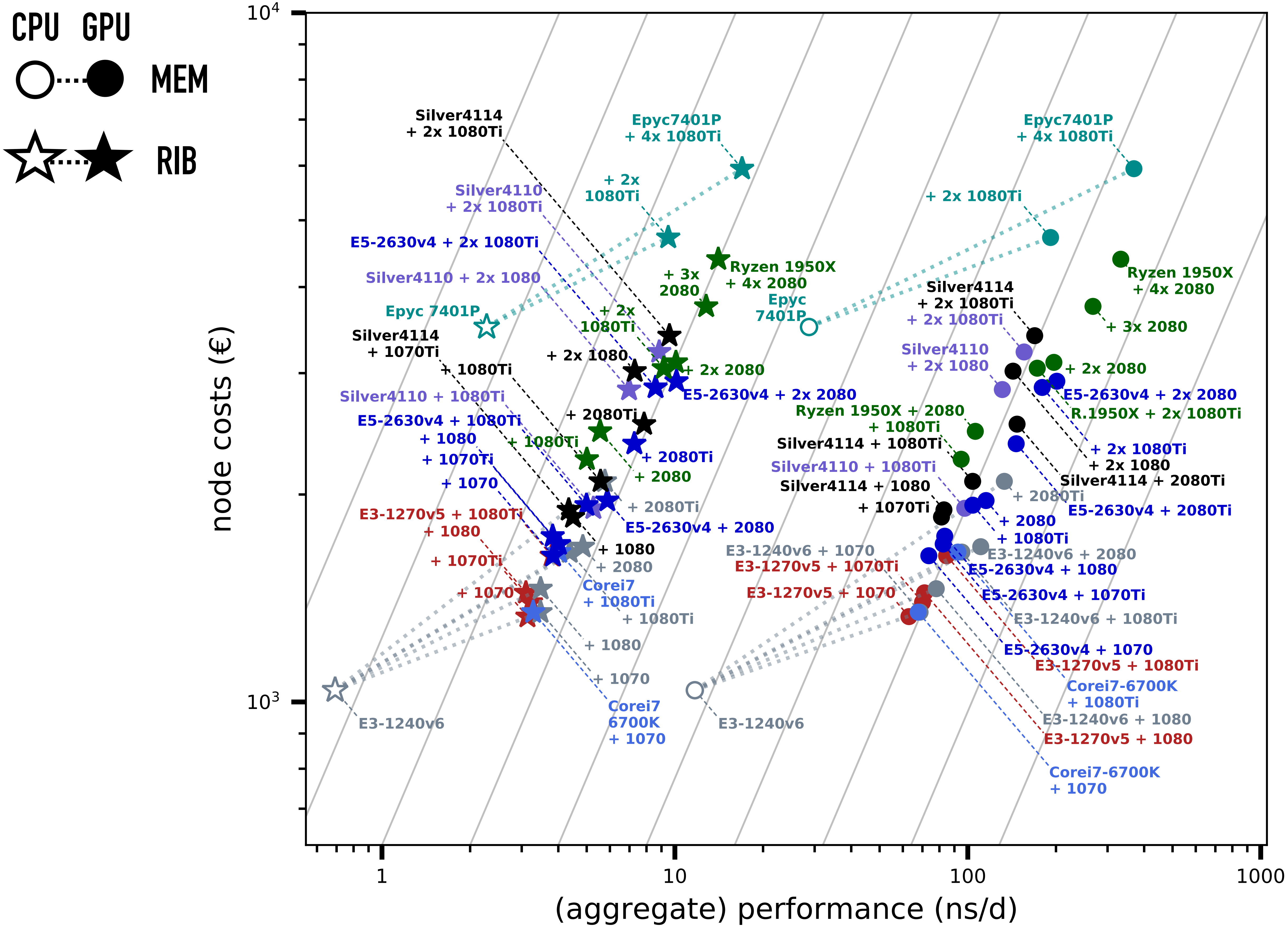
cost of 1 Tesla V100 GPU



CPU GPU

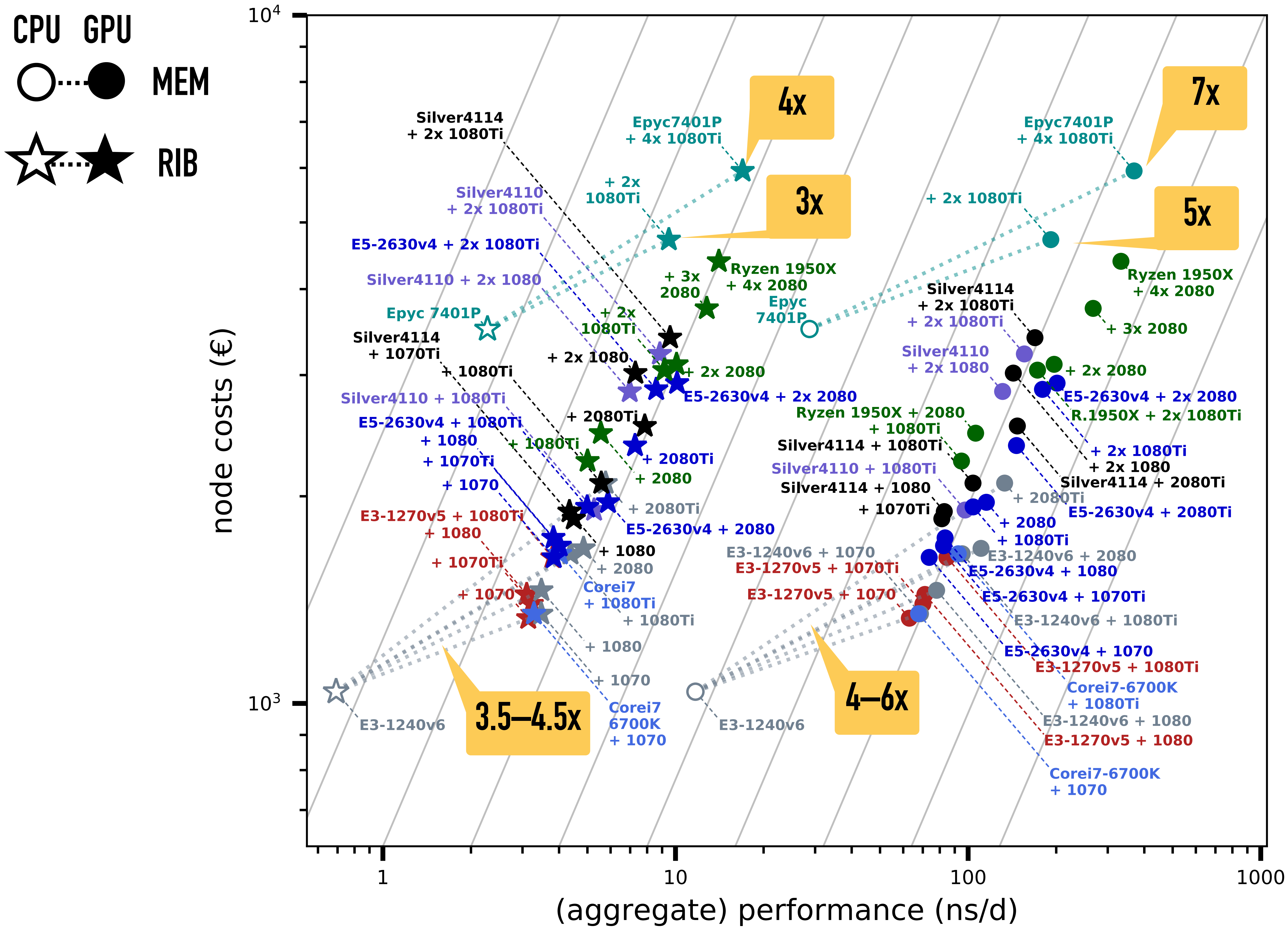
○···● MEM

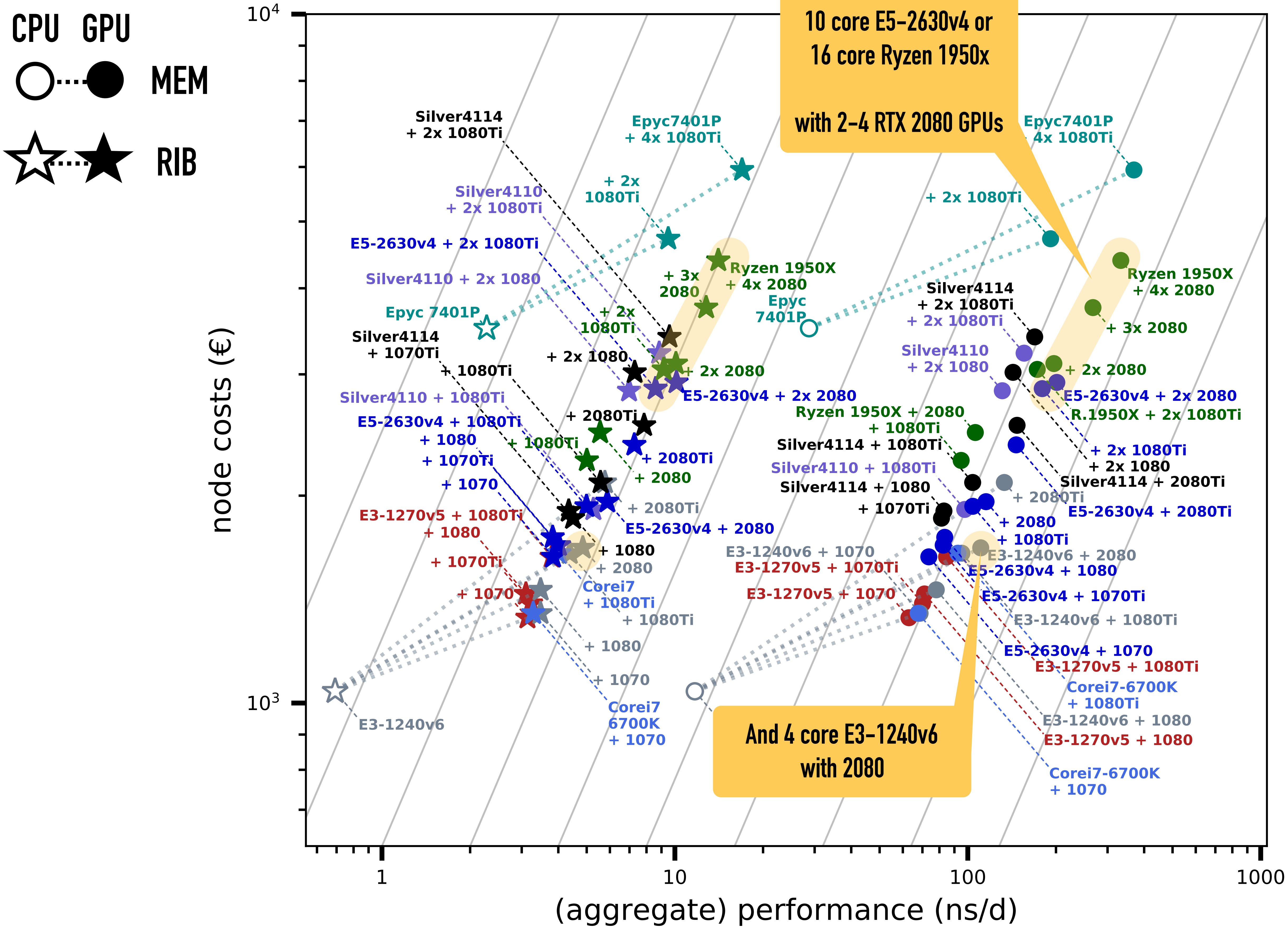
☆···★ RIB



CPU GPU
MEM
RIB
node costs (€)
(aggregate) performance (ns/d)
10^4
10^3
1
10
100
1000
Silver4114
+ 2x 1080Ti
Epyc7401P
+ 4x 1080Ti
+ 2x
1080Ti
Silver4110
+ 2x 1080Ti
E5-2630v4 + 2x 1080Ti
Silver4110 + 2x 1080
+ 3x
2080
Ryzen 1950X
+ 4x 2080
Epyc 7401P
+ 2x
1080Ti
Silver4114
+ 1070Ti
+ 2x 1080
+ 1080Ti
+ 2x 2080
Silver4110 + 1080Ti
E5-2630v4 + 2x 2080
E5-2630v4 + 1080Ti
+ 2080Ti
+ 1080
+ 1080Ti
+ 1070Ti
+ 2080Ti
+ 2080
+ 1070
+ 2080Ti
E3-1270v5 + 1080Ti
+ 1080
E5-2630v4 + 2080
+ 1080
+ 1070Ti
+ 2080
Corei7
+ 1080Ti
+ 1070
+ 1080Ti
+ 1080
+ 1070
Corei7
6700K
+ 1070
E3-1240v6
Epyc7401P
+ 4x 1080Ti
+ 2x 1080Ti
Ryzen 1950X
+ 4x 2080
Epyc
7401P
Silver4114
+ 2x 1080Ti
+ 3x 2080
+ 2x 1080Ti
Silver4110
+ 2x 1080
+ 2x 2080
E5-2630v4 + 2x 2080
R.1950X + 2x 1080Ti
Ryzen 1950X + 2080
+ 1080Ti
Silver4114 + 1080Ti
+ 2x 1080Ti
+ 2x 1080
Silver4110 + 1080Ti
Silver4114 + 2080Ti
Silver4114 + 1080
+ 2080Ti
+ 1070Ti
E5-2630v4 + 2080Ti
+ 2080
+ 1080Ti
E3-1240v6 + 1070
E3-1240v6 + 2080
E3-1270v5 + 1070Ti
E5-2630v4 + 1080
E3-1270v5 + 1070
E5-2630v4 + 1070Ti
E3-1240v6 + 1080Ti
E5-2630v4 + 1070
E3-1270v5 + 1080Ti
Corei7-6700K
+ 1080Ti
E3-1240v6
E3-1240v6 + 1080
E3-1270v5 + 1080
Corei7-6700K
+ 1070

P/P ratio of consumer GPU nodes w.r.t. CPU nodes





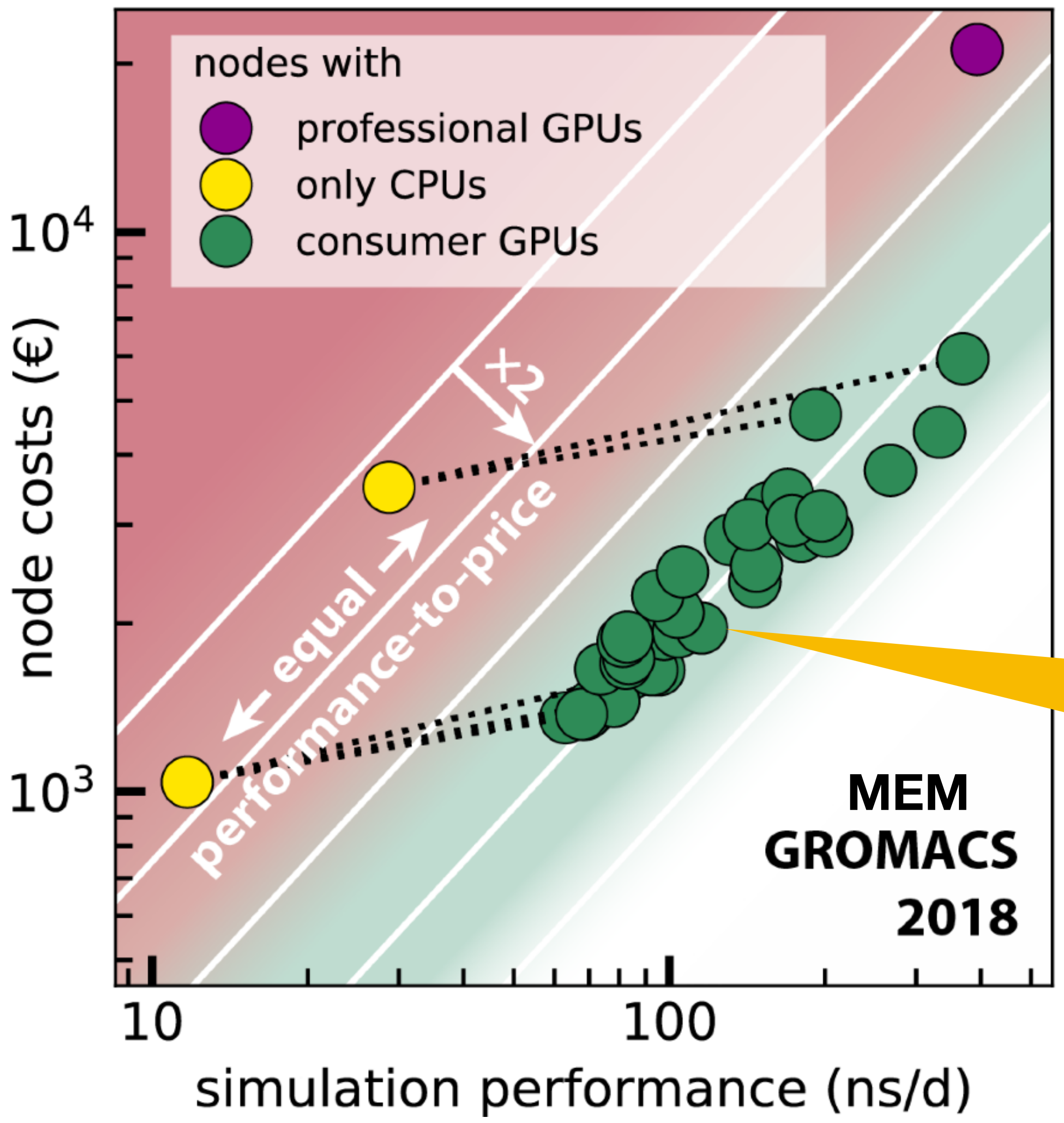
Best P/P ratios:

10 core E5-2630v4 or
16 core Ryzen 1950x

with 2-4 RTX 2080 GPUs

And 4 core E3-1240v6
with 2080

The Gap Widens With GROMACS 2018



- Main 2014 result:

● nodes with GeForce consumer GPUs

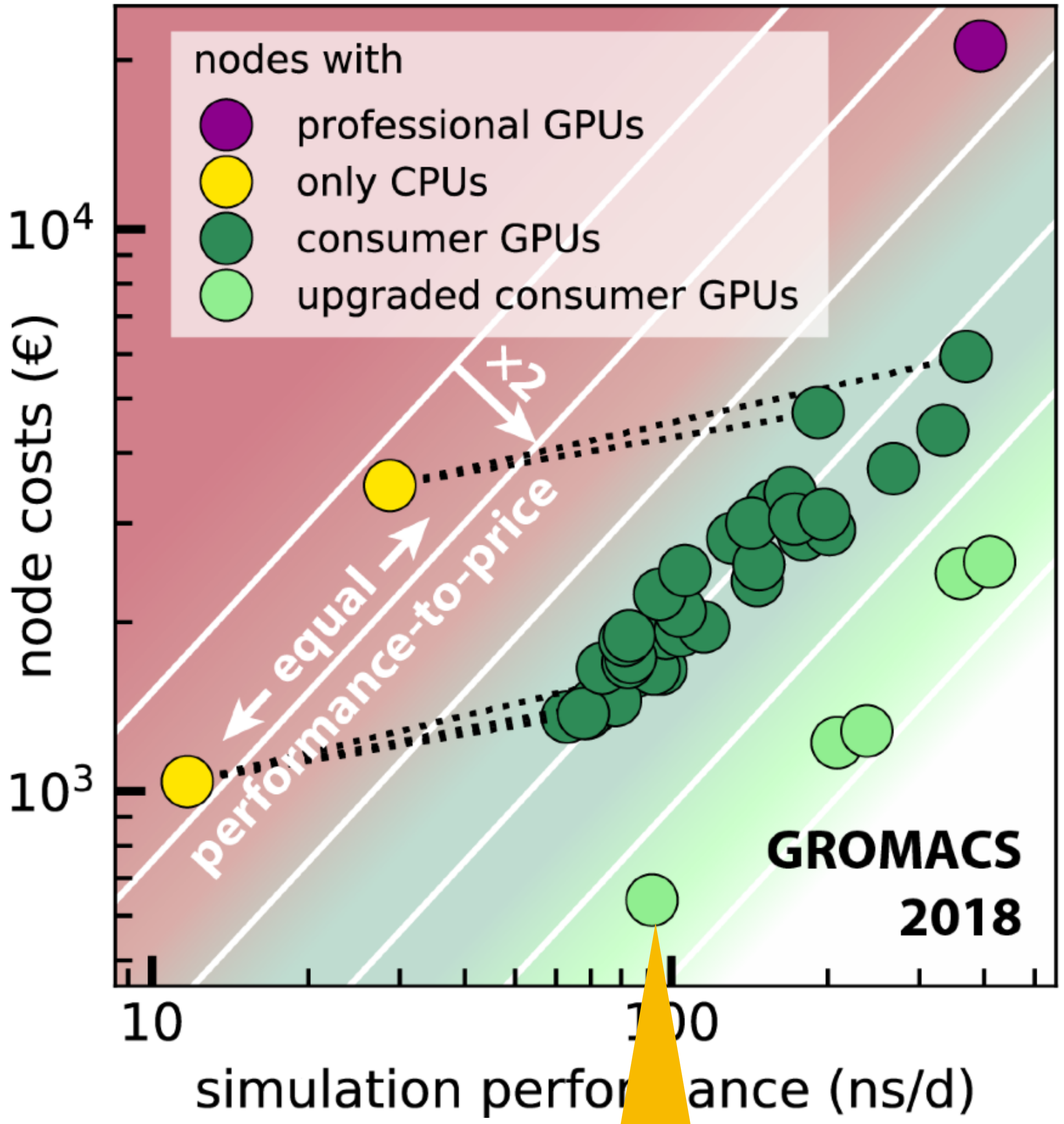
produce **2–3x** as much MD trajectory per invested € as

● CPU nodes

3–7 x with GROMACS 2018

C Kutzner, S Páll, M Fechner, A Esztermann, BL de Groot, H Grubmüller.
More bang for your buck: Improved use of GPU nodes for GROMACS 2018.
JCC 40 (27), pp. 2418-2431 (2019)

Free Lunch! GPU Upgrades

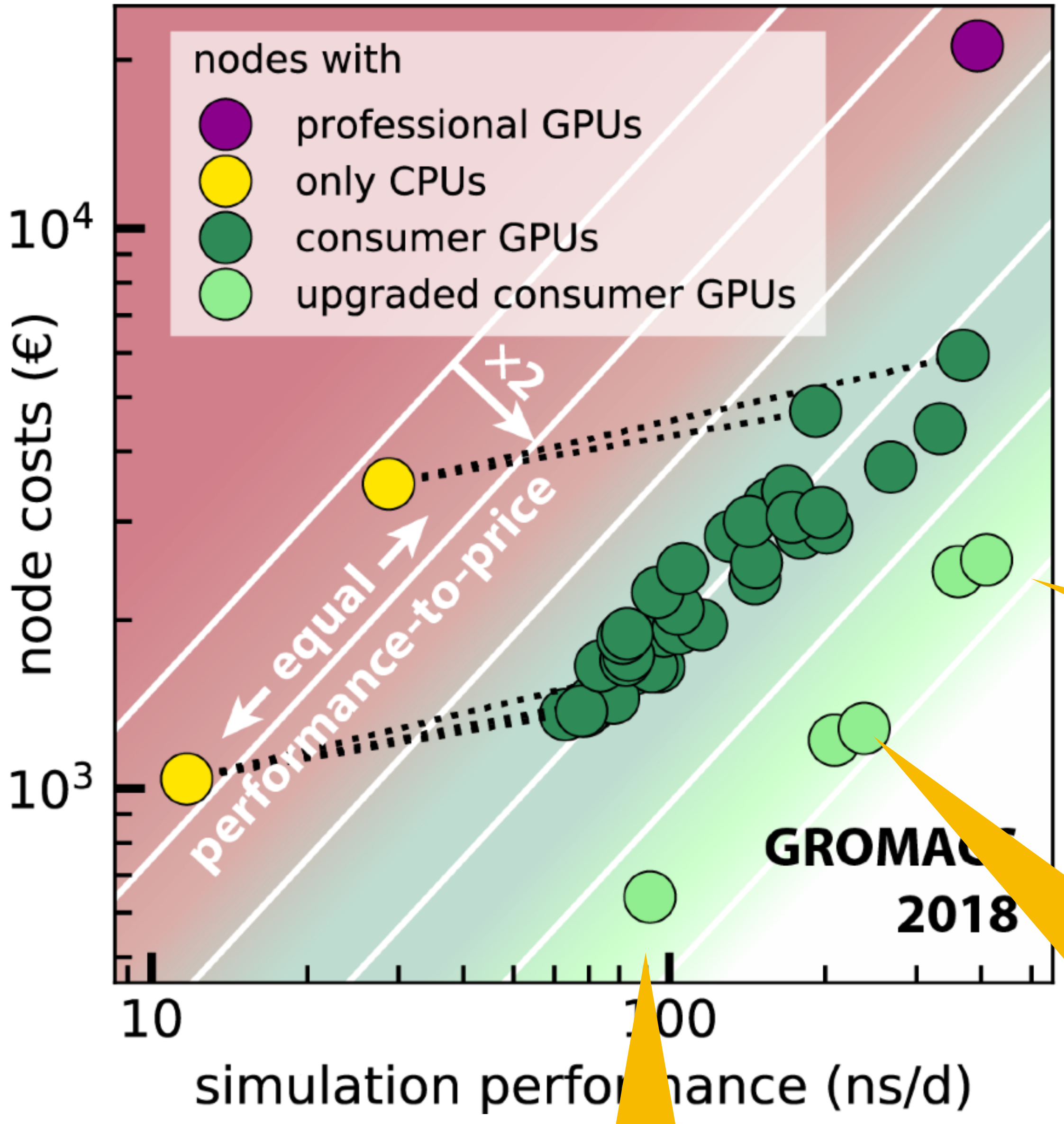


4 core E3-1270v2
RTX 2080

- shift CPU ➔ GPU **allows to upgrade old nodes with recent GPUs**!

- e.g. E3-1270v2 CPU
(4 cores @3.5 GHz)
+ GTX 680 (27 ns/d)
+ (●) RTX 2080 (92 ns/d) ➔ 3.4x perf!

Free Lunch! GPU Upgrades



- shift CPU ➔ GPU **allows to upgrade old nodes with recent GPUs**!

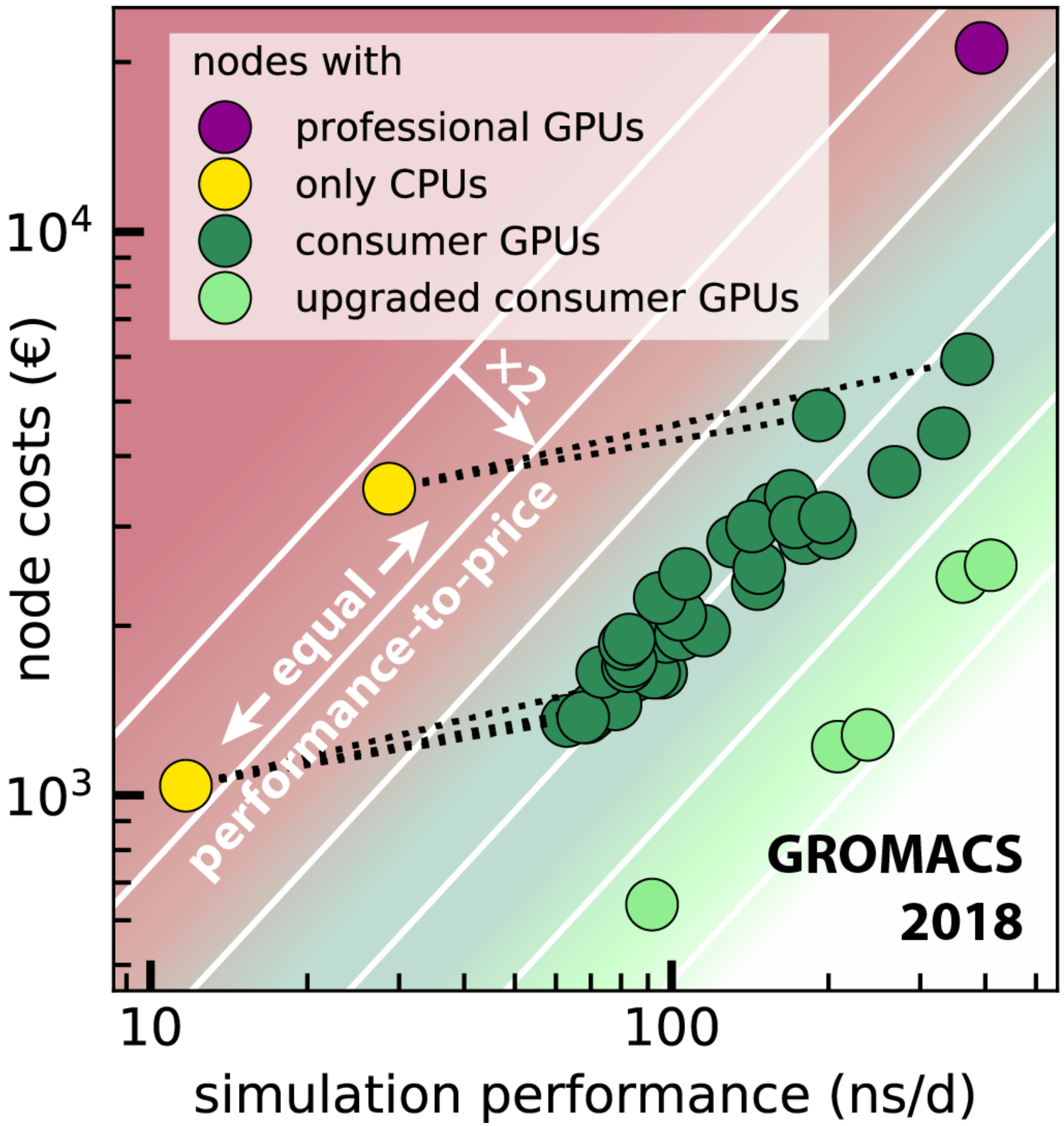
- e.g. E3-1270v2 CPU
  (4 cores @3.5 GHz)
  + GTX 680 (27 ns/d)
  + (●) RTX 2080 (92 ns/d) ➔ 3.4x
  per

2x10 core E5-2680v2
4x 1080Ti or
4x 2080

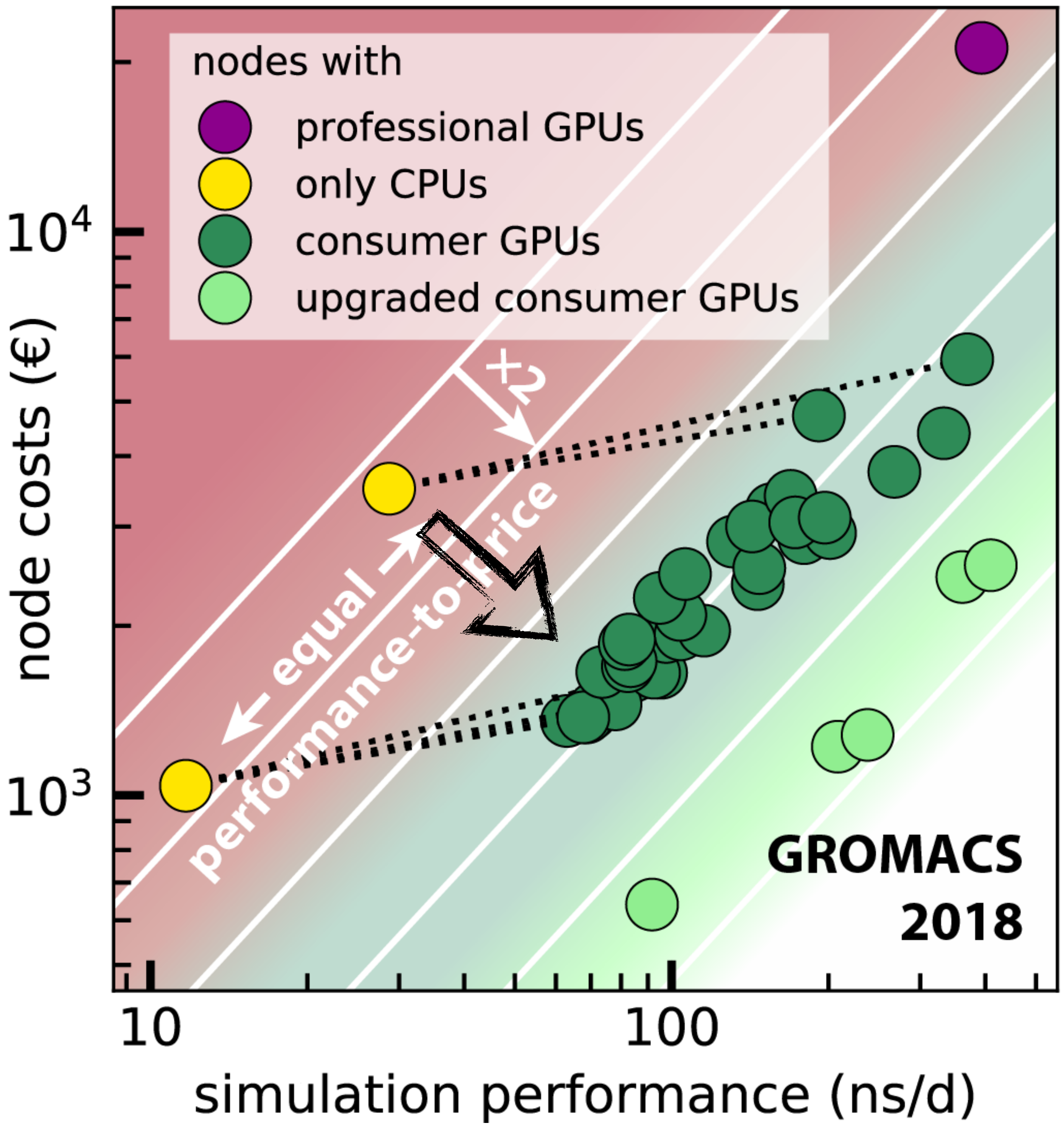
2x10 core E5-2670v2
2x 1080Ti or
2x 2080

4 core E3-1270v2
2080

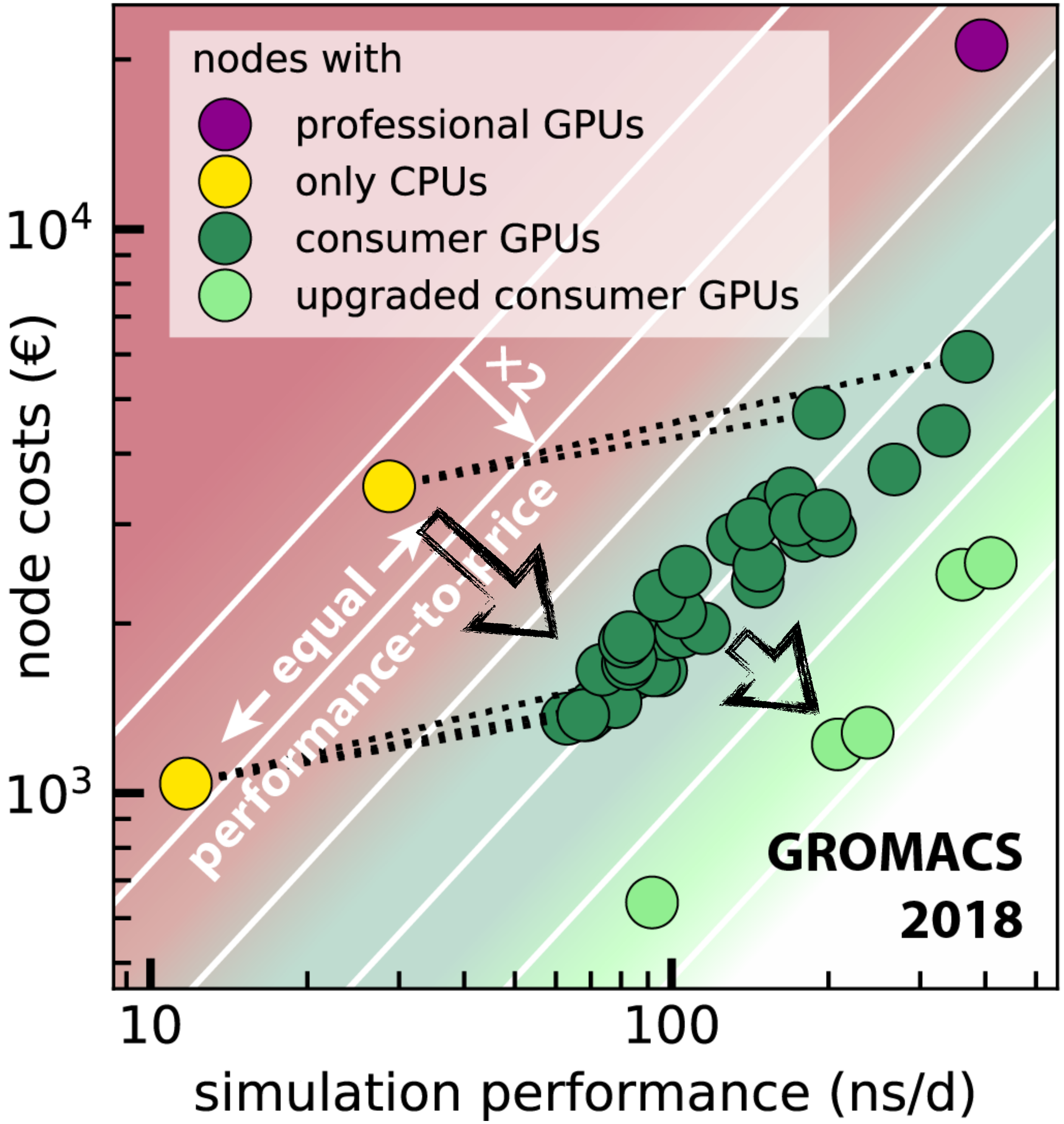
Free Lunch! GPU Upgrades



Free Lunch! GPU Upgrades



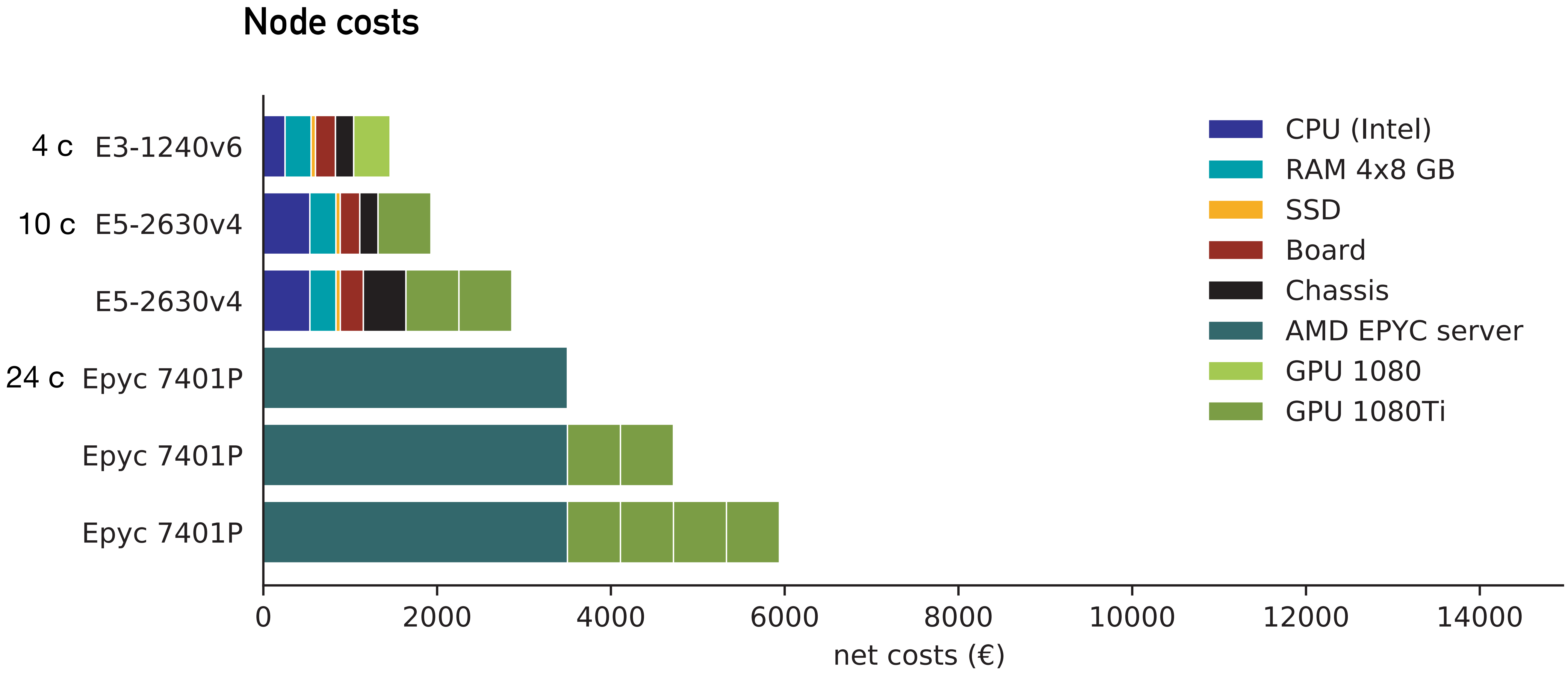
Free Lunch! GPU Upgrades



Energy Efficiency

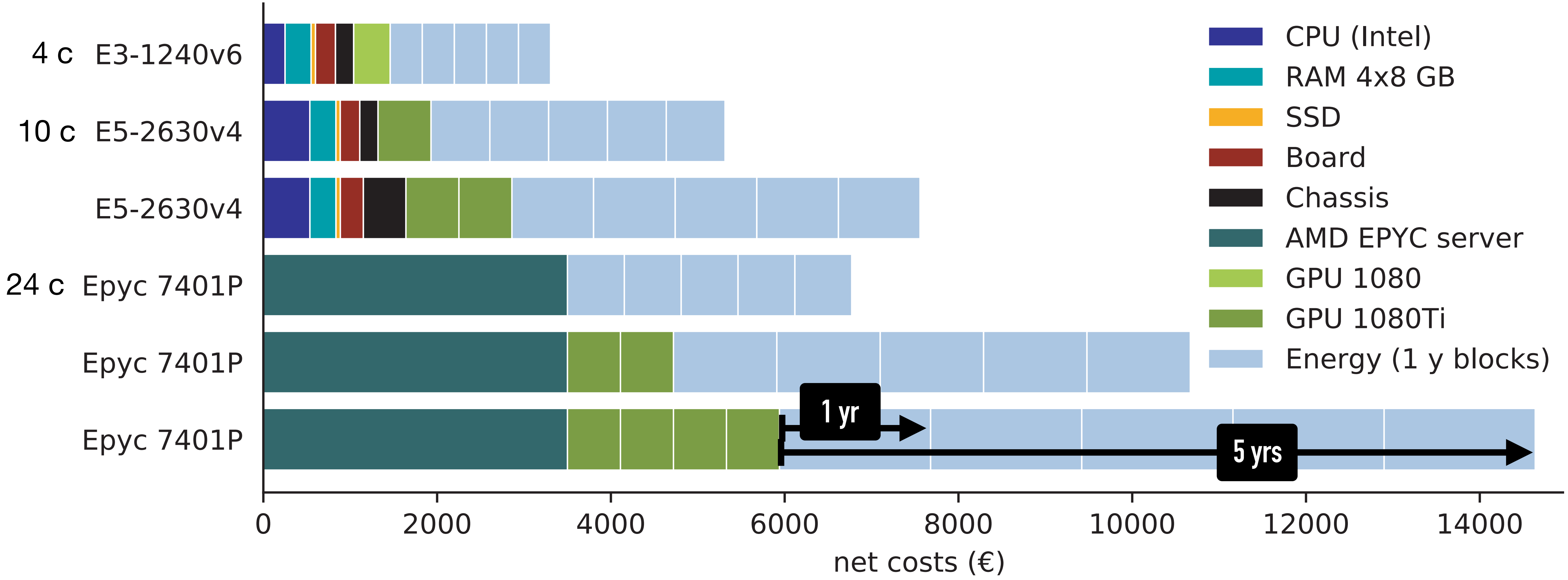
Add energy costs to the bill

Add energy costs to the bill

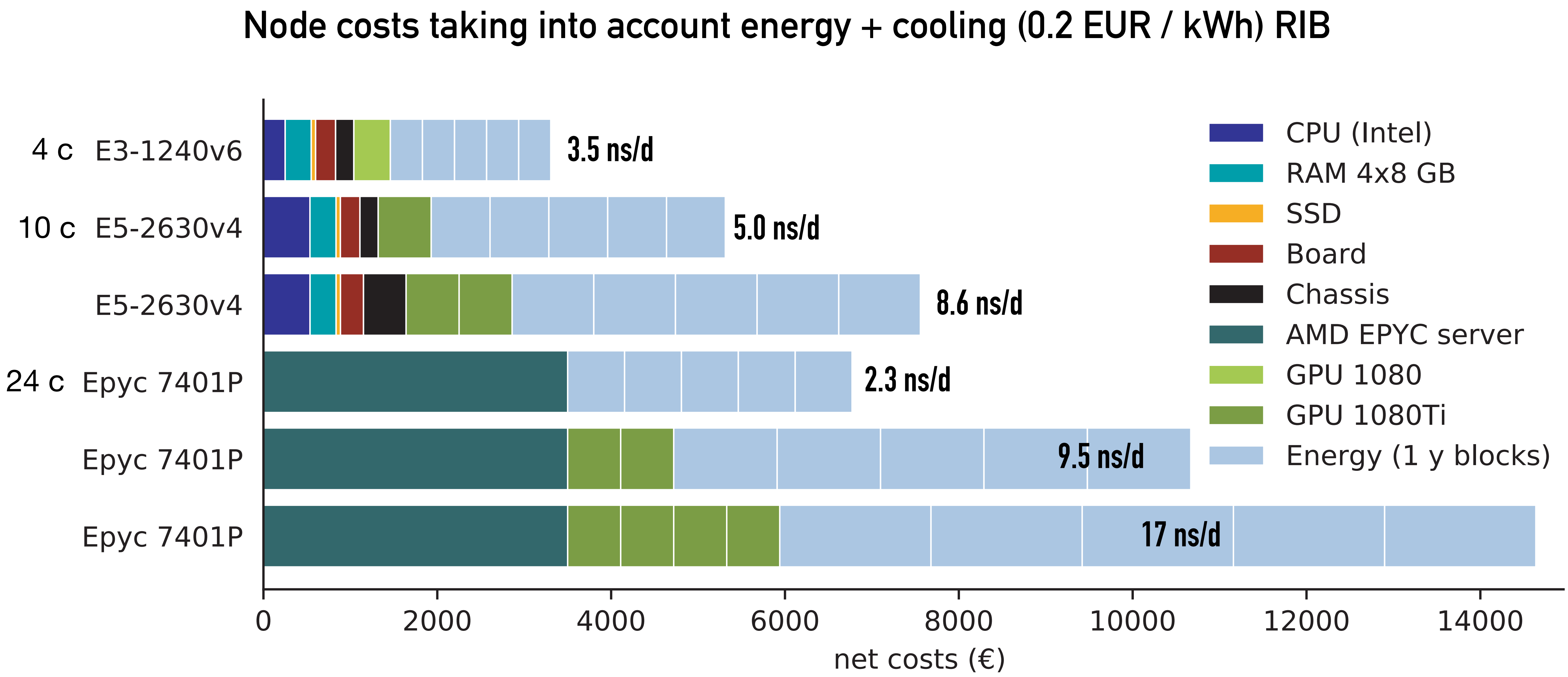


Add energy costs to the bill

Node costs taking into account energy + cooling (0.2 EUR / kWh) RIB

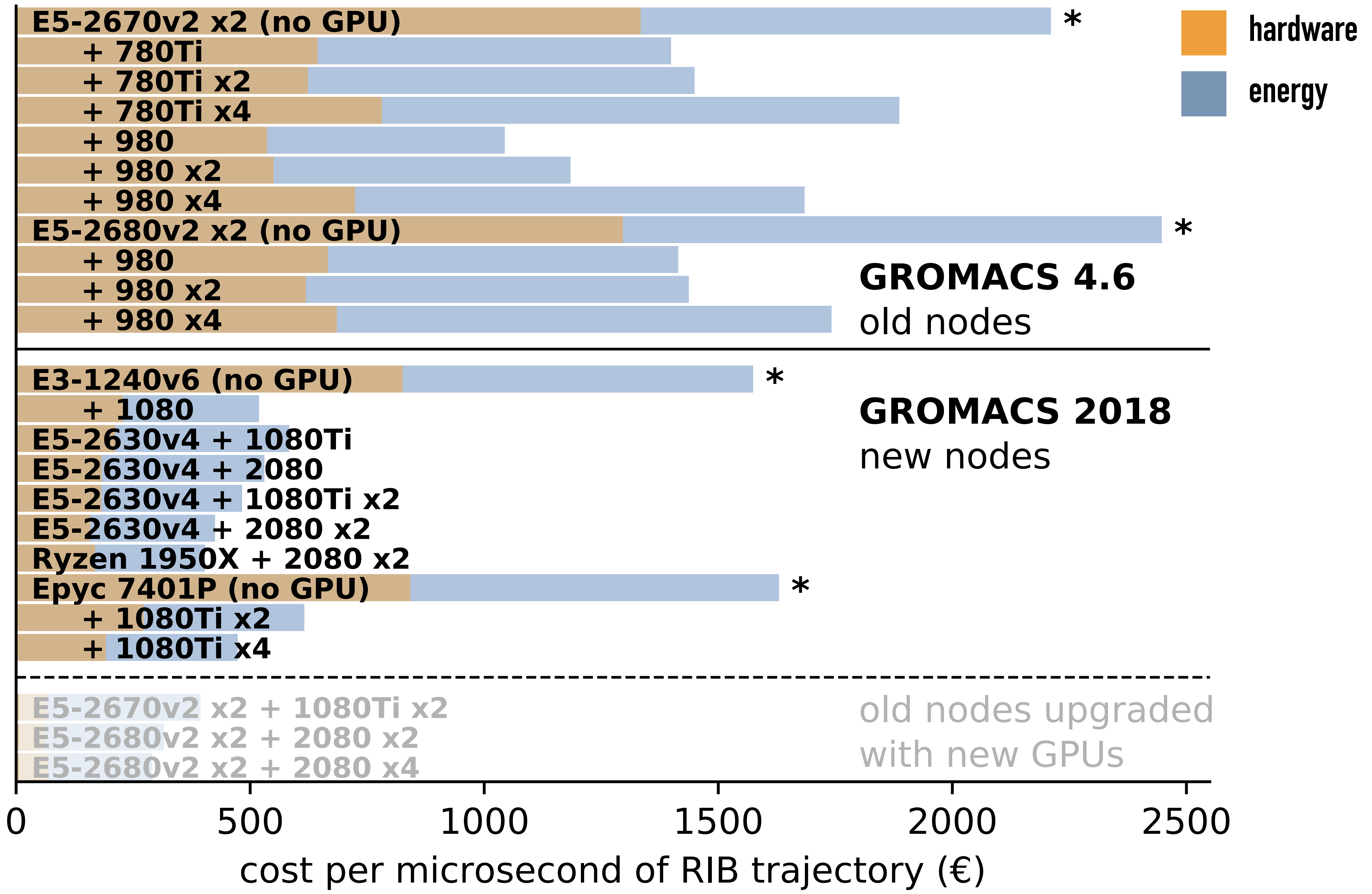


Add energy costs to the bill



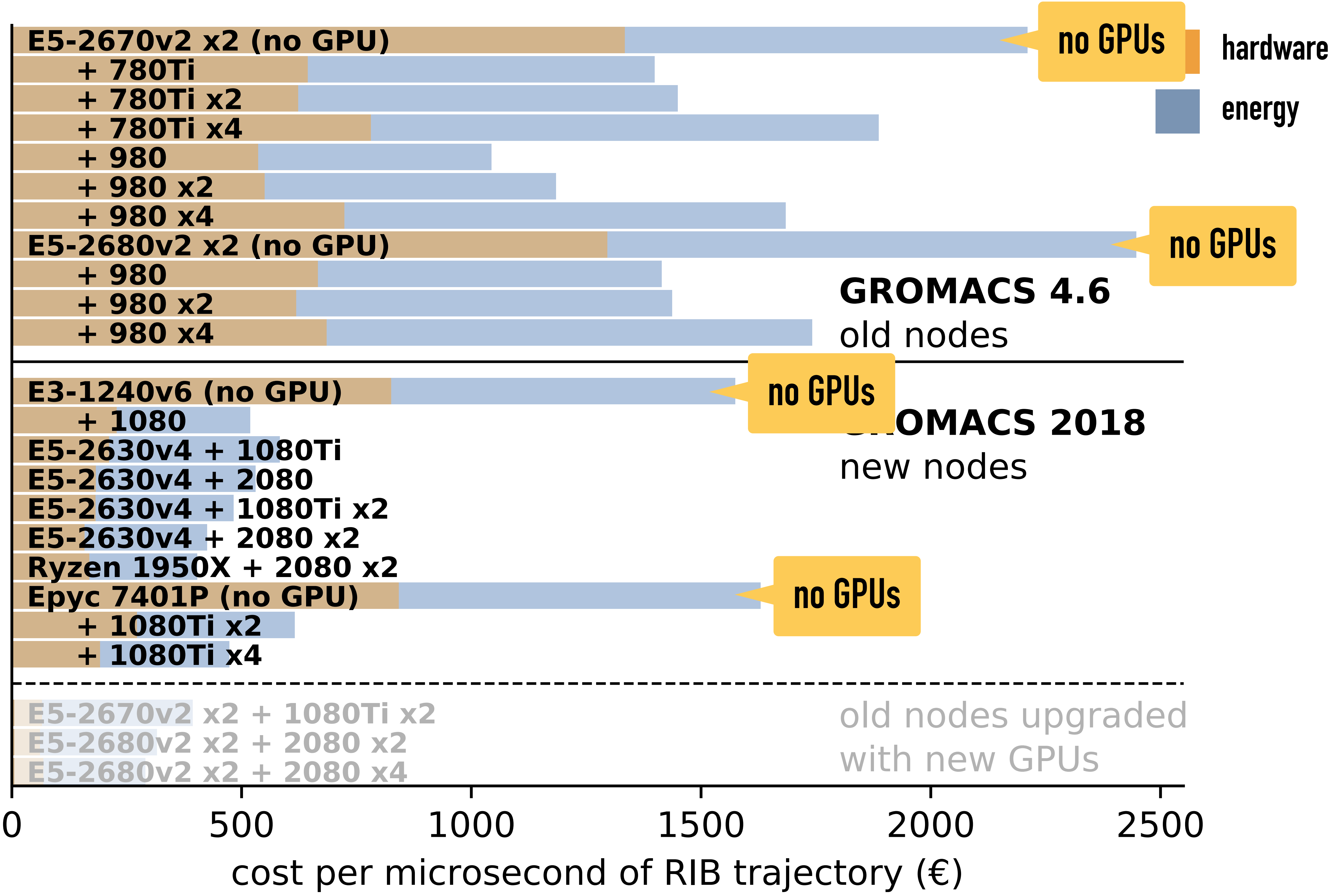
Total trajectory costs hardware+energy

for 5 years of operation



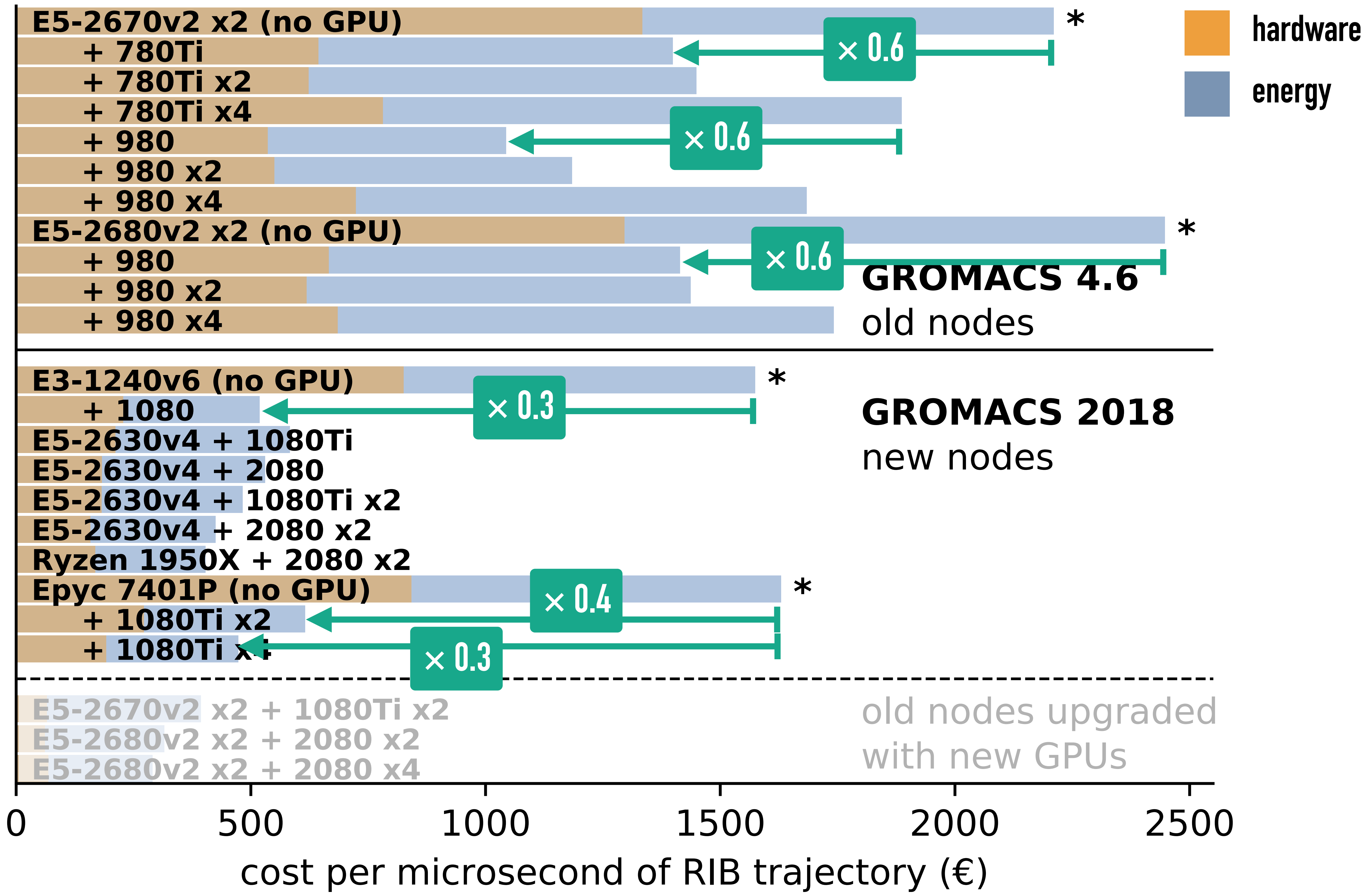
Total trajectory costs hardware+energy

for 5 years of operation



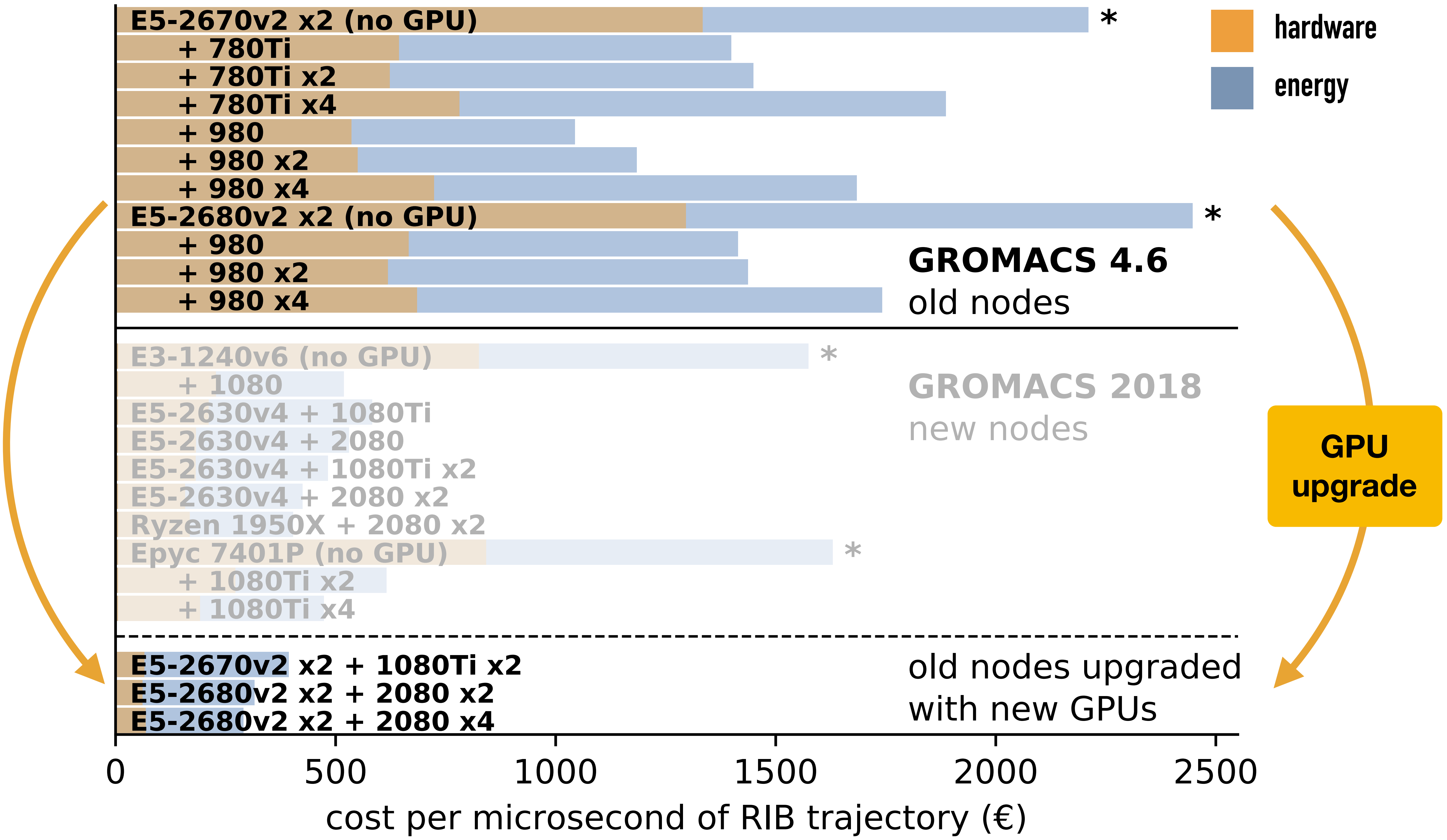
Total trajectory costs hardware+energy

for 5 years of operation



Total trajectory costs hardware+energy

for 5 years of operation



Conclusions

Buying new nodes:

- Consumer GPU nodes have a **much higher performance-to-price ratio** than CPU nodes
  - raw node price: 2–3 x for GROMACS 4.6, and **3–7 x for GROMACS 2018**
  - + energy costs: 2 x for GROMACS 4.6, and **3 x for GROMACS 2018**

Conclusions

Buying new nodes:

- Consumer GPU nodes have a **much higher performance-to-price ratio** than CPU nodes
  - raw node price: 2–3 x for GROMACS 4.6, and **3–7 x for GROMACS 2018**
  - + energy costs: 2 x for GROMACS 4.6, and **3 x for GROMACS 2018**

Recycle old nodes if you can! As a result of CPU ➔ GPU work shifting (PME on GPU)

- **upgrading the GPU** yields large performance increase, whereas
- exchanging the rest of a node (CPU, ..) can be a **waste of money**

Conclusions

Buying new nodes:

- Consumer GPU nodes have a **much higher performance-to-price ratio** than CPU nodes
  - raw node price: 2–3 x for GROMACS 4.6, and **3–7 x for GROMACS 2018**
  - + energy costs: 2 x for GROMACS 4.6, and **3 x for GROMACS 2018**

Recycle old nodes if you can! As a result of CPU ➔ GPU work shifting (PME on GPU)

- **upgrading the GPU** yields large performance increase, whereas
- exchanging the rest of a node (CPU, ..) can be a **waste of money**

- optimal hardware balance: ~15 core-GHz per 2080 GPU

Conclusions

Buying new nodes:

- Consumer GPU nodes have a **much higher performance-to-price ratio** than CPU nodes
  - raw node price: 2–3 x for GROMACS 4.6, and **3–7 x for GROMACS 2018**
  - + energy costs: 2 x for GROMACS 4.6, and **3 x for GROMACS 2018**

Recycle old nodes if you can! As a result of CPU ➔ GPU work shifting (PME on GPU)

- **upgrading the GPU** yields large performance increase, whereas
- exchanging the rest of a node (CPU, ..) can be a **waste of money**

- optimal hardware balance: ~15 core-GHz per 2080 GPU

- results transfer to GROMACS 2019 as well
  - bonded interactions ➔ CUDA GPU
  - PME offload with OpenCL ➔ AMD GPUs

Additional Material

- want to compare your own hardware and contribute to benchmarking? https://www.mpibpc.mpg.de/grubmueller/bench has various **benchmark .tprs for download** (CC licensed)

- Related publications:
  - GROMACS 2018/2019:
    More Bang for Your Buck: Improved use of GPU Nodes for GROMACS 2018
    - JCC https://onlinelibrary.wiley.com/doi/10.1002/jcc.26011
    - arXiv https://arxiv.org/abs/1903.05918
  - Summary **poster:** https://www.mpibpc.mpg.de/grubmueller/kutzner/posters
  - GROMACS 4.6/5.0:
    Best bang for your buck: GPU nodes for GROMACS biomolecular simulations
    - JCC https://onlinelibrary.wiley.com/doi/full/10.1002/jcc.24030
    - arXiv https://arxiv.org/abs/1507.00898

Acknowledgments

The Department of Theoretical & Computational Biophysics
@ MPI for Biophysical Chemistry Göttingen

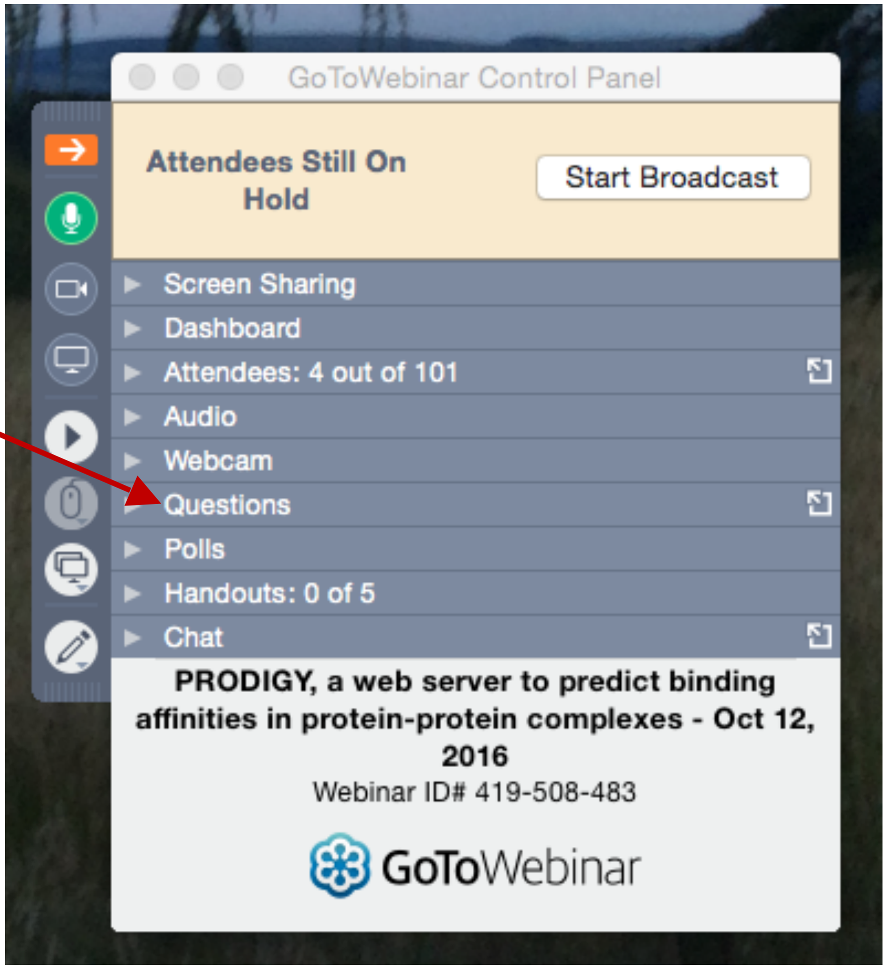
Markus Rampp, Hermann Lederer (Max Planck Computing & Data Facility)

bioexcel

SPPEXA

Audience Q&A session

- Please use the Questions function in GoToWebinar application
- Any other questions or points to discuss after the live webinar? Join the discussions at http://ask.bioexcel.eu.



PRODIGY, a web server to predict binding affinities in protein-protein complexes - Oct 12, 2016

Webinar ID# 419-508-483

Coming up next!

12 Sept 2019

Enhanced molecular simulations with PLUMED