Scaling Tensor Network Algorithms on Massive Multi-Node Multi-GPU Supercomputers

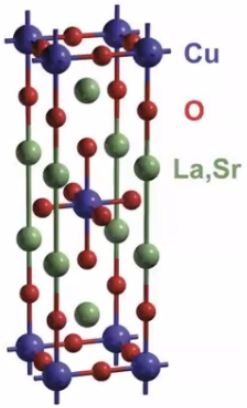
Andor Menczer

Supervisors: Örs Legeza, Tamás Kozsik

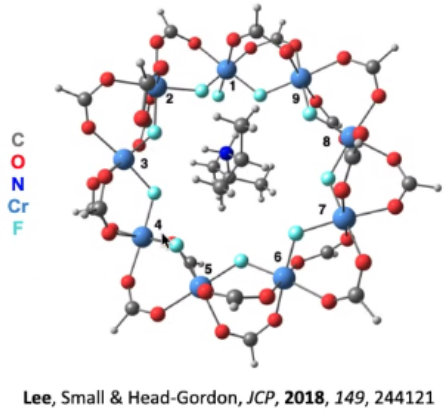
Wigner GPU Day 2024

Budapest, 2024.05.30.

Strong correlations between electrons $\rightarrow$ exotic materials

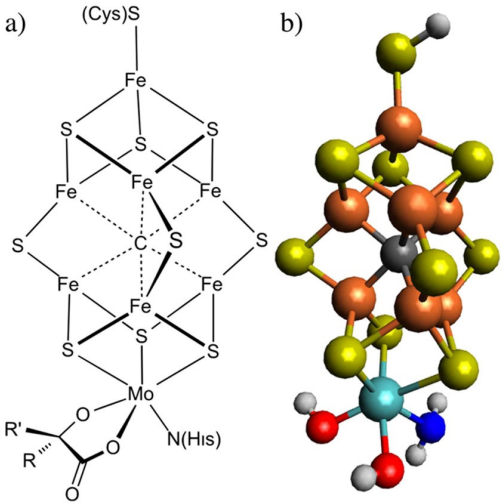


High T_c superconductors

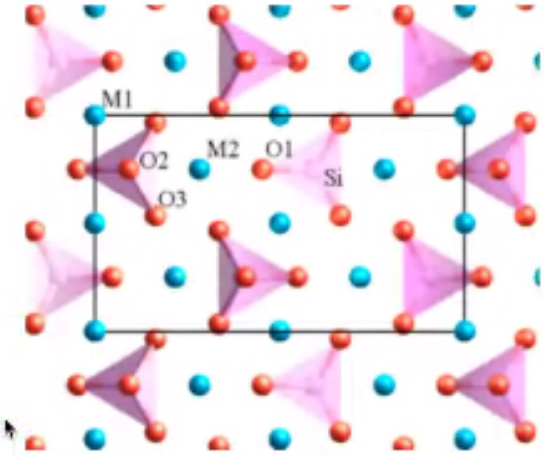


Lee, Small & Head-Gordon, *JCP*, **2018**, *149*, 244121

Single Molecular Magnets

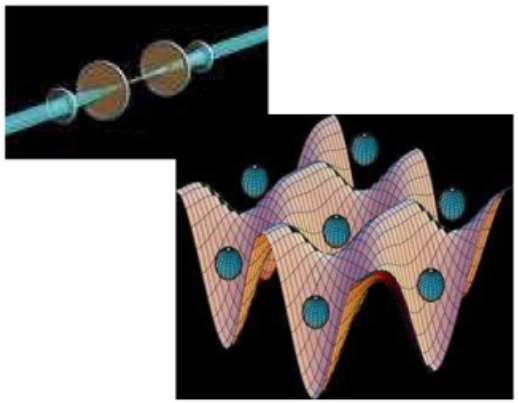


Nitrogenase Cofactor, FeMoco



Battery Technology

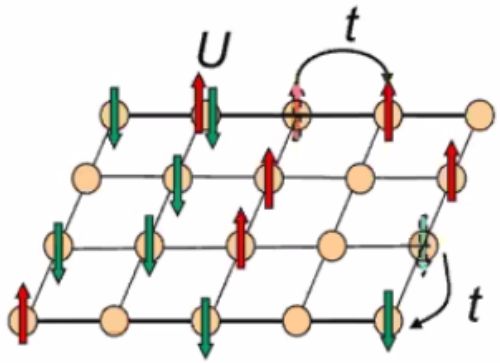
Experimental realizations: optical lattices
Numerical simulations: model systems



- Atoms (represented as blue spheres) pictured in a 2D-optical lattice potential
- Potential depth of the optical lattice can be tuned.
- Periodicity of the optical lattice can be tuned.

Hubbard model: lattice model of interacting electron system

$$\mathtt{H} = \mathtt{t} \sum_{\langle \mathtt{i},\mathtt{j}\rangle,\sigma} \mathtt{c}^{\dagger}_{\mathtt{i},\sigma}\mathtt{c}_{\mathtt{j},\sigma} + \frac{\mathtt{U}}{2} \sum_{\sigma\neq\sigma'} \sum_{\mathtt{i}} \mathtt{n}_{\mathtt{i},\sigma}\mathtt{n}_{\mathtt{i},\sigma'}$$



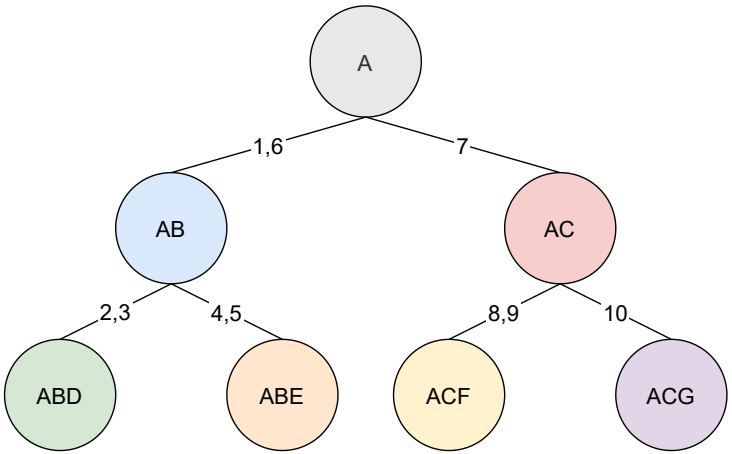
$\mathtt{t}$ hopping amplitude
$\mathtt{U}$ on-site Coulomb interaction
$\sigma \in \uparrow, \downarrow$ spin index

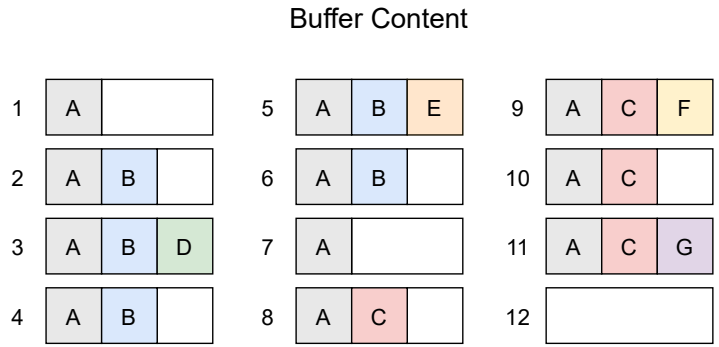
Properties of the TNS/DMRG algorithms

- Key aspect of TNS/DMRG: exponential scaling can be reduced to a polynomial form.
- Underlying tensor and matrix algebra can be organized into several million of independent operations (tasks).
- Dense matrix operations are performed in parallel according to the so-called quantum number decomposed representations (sectors).
- Full matrices, denoted as DMRG bond dimension, $\mathtt{D}$, determines the accuracy of the calculations.
- The overall scaling of the DMRG is $\mathtt{D}^3\mathtt{N}^4$ where $\mathtt{N}$ stands for the system size.
- The memory requirement is proportional to $\mathtt{D}^2\mathtt{N}^2$.
- The iterative diagonalization of the effective Hamiltonian usually accounting for 85% of the total execution time.
- The renormalization step is responsible for 10% of the total execution time.

Memory management: Data Dependency Trees

- **TTCache** is a model for virtual memory addressing designed to vastly reduce redundant IO operations and eliminate memory fragmentation as well as allocation overhead.
- **TTCache** works by factorizing data into attributes, then hierarchically mapping such attributes to execution blocks. Execution is done by traversing a tree-like structure, in which nodes close to each other depend on largely the same set of attributes

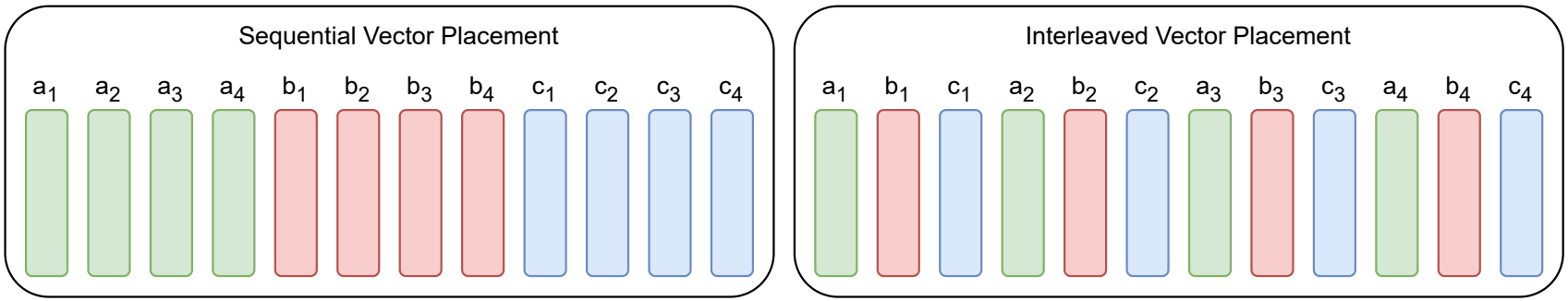




- **Fragmentation-free, sequential write and read operations.**
- **Allocations and deallocations are purely virtual.**

Strided Batched Matrix Multiplication for Summation

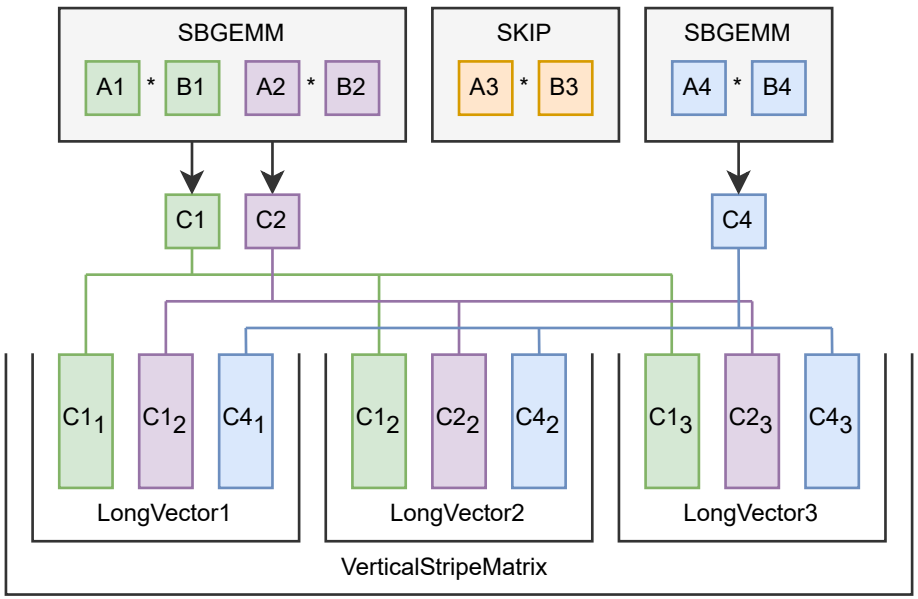
- **SBMM4S** is a batched type matrix multiplication with inherent zero cost sum reduction. Produces a single result by multiplying an entire batch of matrices with concatenated vector arrays of interleaved matrices. Intermediate results of chained matrix multiplications are reached using strided batched type matrix multiplications with specific offset values to enable interleaving.



- A summation such as $\mathtt{B} := \mathtt{B} + \alpha(\sum_{\mathtt{i}=1}^{\mathtt{p}} \mathtt{L_i} * \mathtt{A} * \mathtt{R_i}^{\top})$ can be executed in parallel by first independently calculating the interleaved vectors of each $\mathtt{A} * \mathtt{R_i}^{\top}$, then multiplying $\mathtt{concat}(\mathtt{L_1}, ..., \mathtt{L_p})$ with the matrix holding all previously calculated vectors.

Improved partial execution of SBMM4S

- The multiplication can to be broken up into multiple SBGEMM operations. The leading dimensions and stride values are set in a way that the vectors of the result matrices became interleaved.
- The sequence of such vectors can be viewed as a singular horizontally or vertically very long stripe-like matrix, just as before.



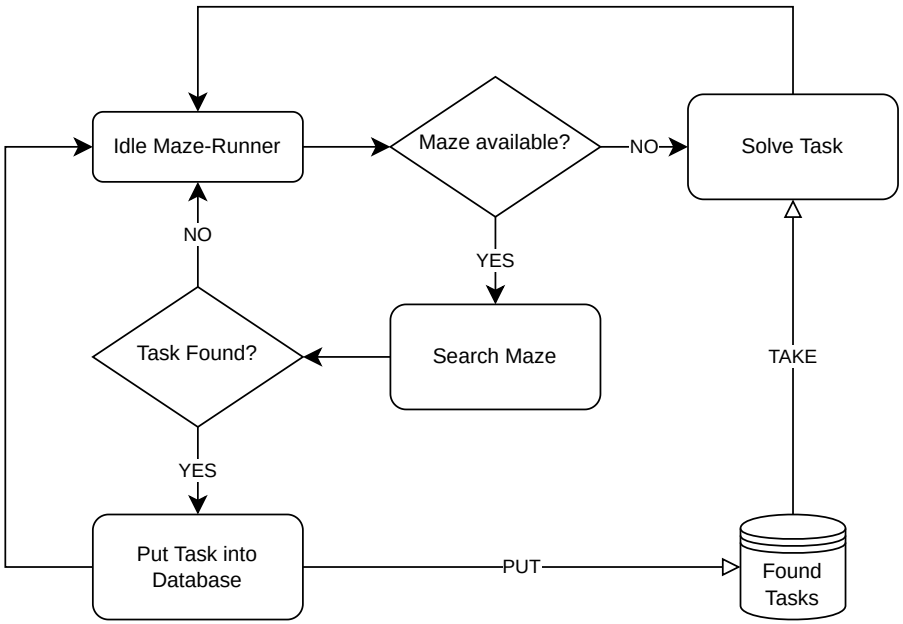
Example for partial SBMM4S

Partial SBMM4S works as a zero overhead drop-in replacement for both GEMM and regular SBMM4S:

- No auxiliary data
- No extra calculations
- Results remain monolithic

Low-latency Self-scheduled threading

- Parallel models relying on inter-thread communication might be ineffective when bombarded with an extreme amount of tiny tasks
- Homogeneous threading with imprinted heuristics as guidance leads to a lightweight, decentralized and communication-free parallel construct.



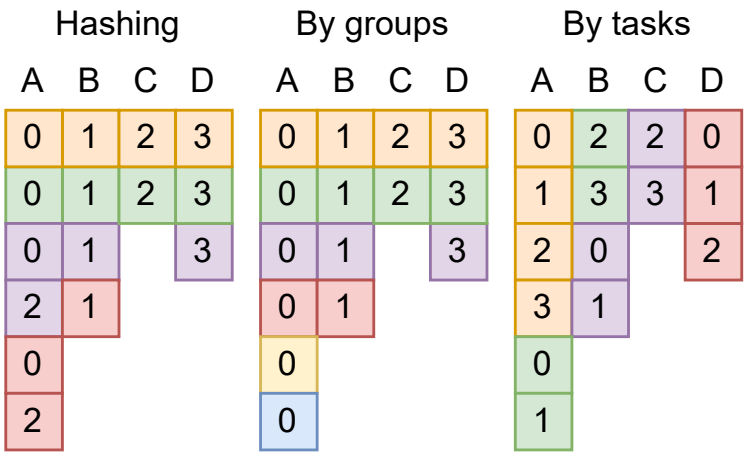
Maze-Runner threads used previously

Contractor threads are self-organized ultra-lightweight constructs:

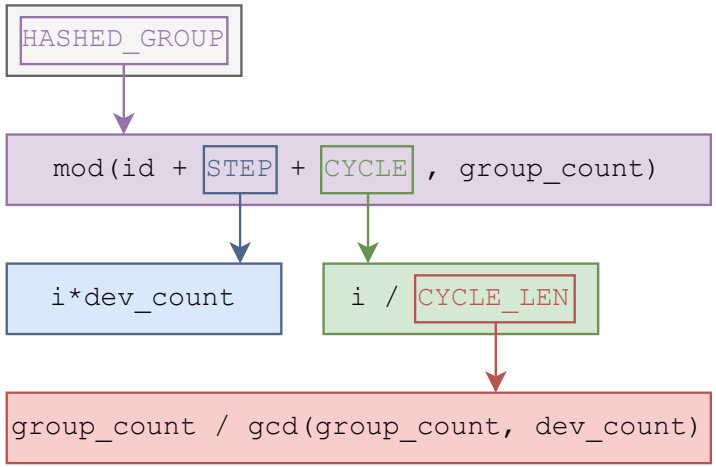
- No external scheduling
- No locking
- No barriers

Hash based thread scheduling

- Assigning different groups of tasks to different workers can result in unwanted idle time due to size differences, while flattened task queues can result in high IO overhead due to decreased spatial locality.
- Hashing on the other hand assigns different groups to different workers whenever possible, but at the same time allows multiple devices to work within the same group if necessary.



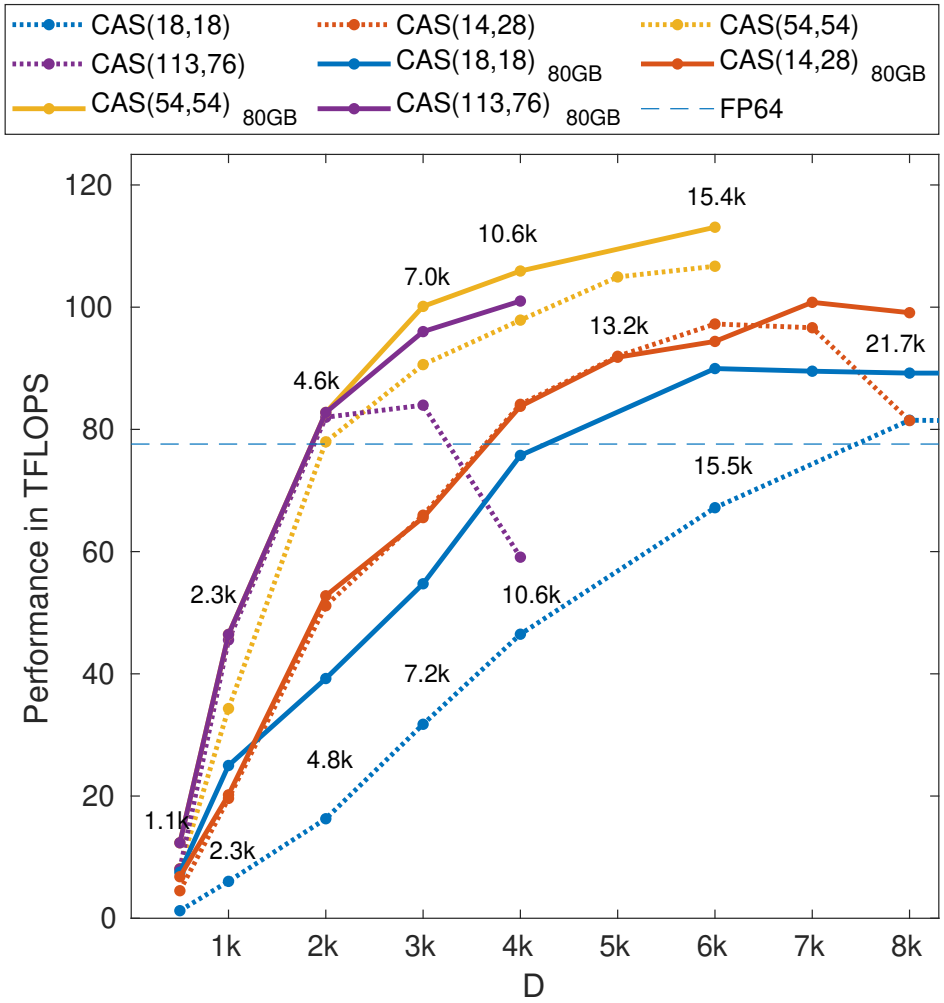
Dynamic granularity



Hash building blocks

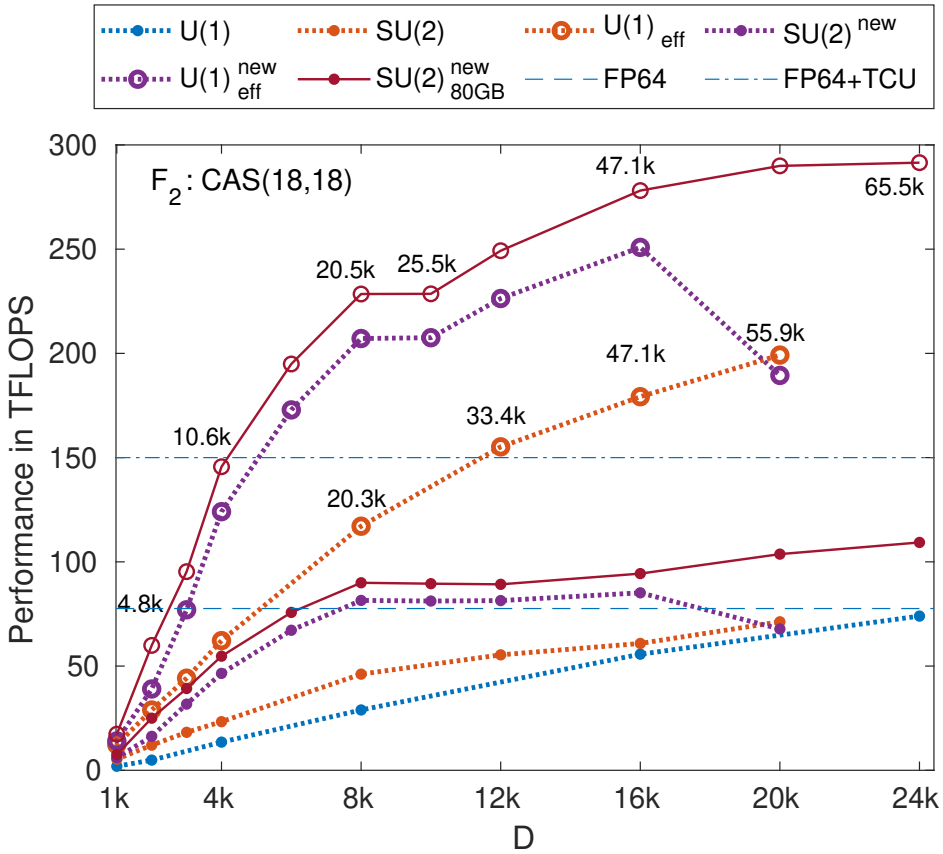
Performance up to CAS(113,76) $\dim \mathcal{H} = 2.88 \times 10^{36}$)

Performance for the F_2 and FeMoco chemical systems for CAS(18,18), CAS(54,54) and CAS(113,76) orbitals spaces, respectively, as a function of the DMRG bond dimension on a dual Intel(R) Xeon(R) Gold 5318Y CPU.



SU2 effective performance

Benchmark results obtained via the SU(2) spin adapted hybrid CPU-multiGPU DMRG for the F_2 molecule for a CAS(18,18) orbital space. Calculations have been performed on a dual AMD EPYC 7702 CPUs with 2×64 cores compiled with eight NVIDIA A100-SXM4-40GB devices.



Power consumption $\rightarrow$ Green DMRG

- The power consumption of the TNS calculations are becoming one of the most important question due to high energy demands and costs.
- The thermal design power (TDP) for 2 $\times$ Intel(R) Xeon Gold 5318Y CPU is 2 $\times$ 165 Watts $\rightarrow$ 2.5 TFLOPS would lead to $\approx$ 7.5 GFLOPS/Watt.
- For an NVIDIA A100-PCIE-40GB device the TDP is 250 Watts.
- For our 8 card accelerated hybrid algorithm with 110 TFLOPS performance results in $\approx$ 47.2 GFLOPS/Watt.
- For a given calculation the cost of the energy demand arising from the processors can be reduced to 1/6 of the original consumption.
- The energy consumption of the GPU devices fluctuates significantly, thus even a better ratio can be obtained.

Ongoing and future work

- **Published:** GPU accelerated DMRG as shown previously, featuring Maze-Runner, TTcache and SBMM4S
- **In-progress:** Collaboration with NVIDIA, accelerating the simulation of strongly correlated systems using state-of-the-art supercomputing hardware known as DGX-H100. Other prototype super-hardware might also be tested.
- **Published:** Matrix dimensions can be further decreased by exploiting SU(2) symmetries. This leads to higher accuracy at the same dimensions or similar accuracy at much lower dimensions. Improved memory management, thread scheduling and support for a more general SBMM4S with partial execution.
- **Published:** GPU accelerated simulations of quantum lattices.
- **In-progress:** Unbounded scalability through multi-node execution using MPI and InfiniBand. Target: 1 PETAFLOPS.
- **End Goal:** Accurate modelling of strongly correlated subatomic particles at 1+ EXAFLOPS.