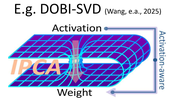
Evelyne Ringoot[1], : Rabab Alomairy[1](advisor), Valentin Churavy[2] (advisor), Alan Edelman[1] (advisor)
([1]Massachusetts Institute of Technology; [2]University of Mainz, [2]University of Augsburg)

Scaling SINGULAR VALUES Beyond GPU Memory Limits:
Out-of-Core, GPU-Accelerated & Unified Across Data Precision and Hardware

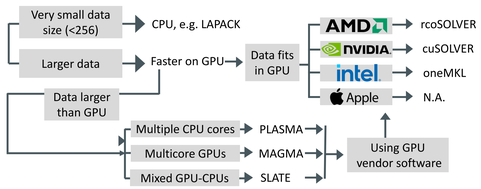
Objective: One SVD for all data sizes & hardware environments

Linear algebra is everywhere: e.g. Low-Rank approximation (LoRa) in Large Language Models (LLMs)

E.g. DOBI-SVD (Wang, e.a., 2025)



Current situation: specialized libraries for every hardware



Complex heterogenous environments, mixed-precision set-ups, New hardware types and data precisions

Alternative: A generic extensible SVD

Tools: abstractions in Julia

Multiple-dispatch provides composability

(Bezanson, e.a., 2019)

Array abstractions: portable CPU - GPU

Kernel abstractions: portable kernels

Include communication in existing algorithm through multiple-dispatch:

```
function banddiag!(A::GPUorLargeMatrix{T},
   Tau::AbstractGPUMatrix{T}, N::Int, backend) where T
   for k in 1:(N-1)
      GETSMQRT!(A,Tau,k, N, backend)
      GETSMQRT!(A',Tau,k, N, backend, "LQ")
   end
   GETSMQRT!(A,Tau,N, N, backend)
   return A
end

function GETSMQRT!(A::AbstractGPUMatrix, ...)
   GETSQRT_fused!(A,k)
   UNTSMQR_fused!(A, k)
end

function GETSMQRT!(A::LargeMatrix, ...)
   GETSMQRT!(A.data[k],...)
   ... #communication CPU-GPU
   KernelAbstractions.@synchronize
end
```

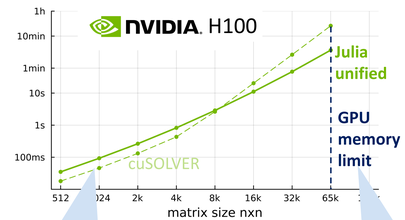
async

Acknowledgements

AIR FORCE RESERVE · ORCD · NSF U.S. National Science Foundation · U.S. DEPARTMENT OF ENERGY · KAUST · DARPA

GPU resident
Data < GPU memory

Unified and performant across hardware and precision (ICPP'25)



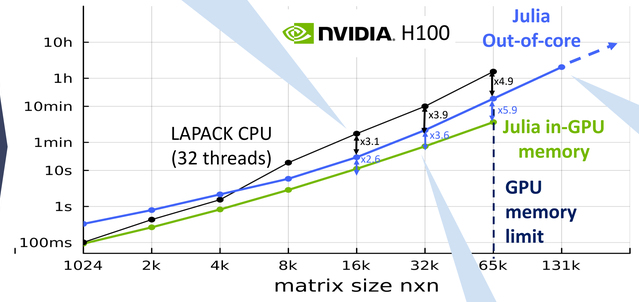
Unified function performs on-par with vendor libraries

Vendor library & unified in-GPU limit

Out-of-Core GPU-accelerated
Data > GPU memory: compute on GPU, store data on CPU

Out-of-core outperforms LAPACK (CPU)

Out-of-core benchmarked under **constrained memory**: GPU assumed to accommodate only quarter of matrix data.

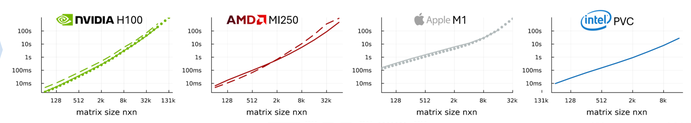


Out-of-core computes data size > GPU memory

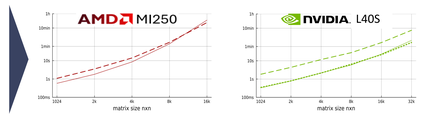
Out-of-core includes CPU-GPU communication & scales similarly to in-GPU

GPU resident

Unified across hardware and precision



Out-of-Core GPU-accelerated

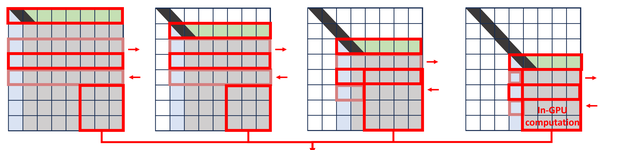


(Ringoot, e.a., 2025)

Literature: minimize total communication

(Kabir e.a. 2017)

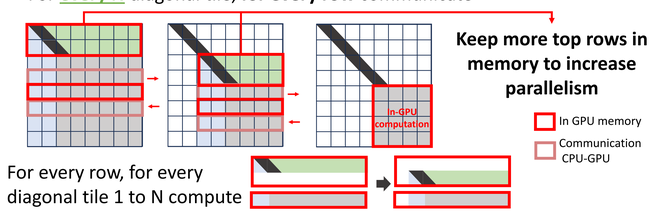
For **every** diagonal tile, **for every row** communicate and compute



Keep lower right part in GPU memory to reduce communication

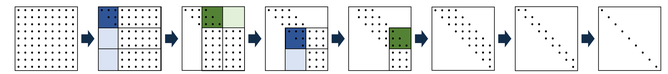
Proposed: balancing computation & communication

For **every N** diagonal tile, **for every row** communicate

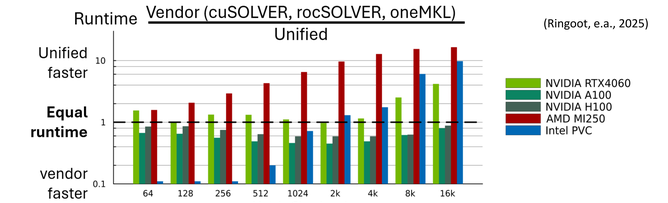


The algorithm: two-stage bandreduction

Algorithm: QR block-bidiagonalization (Haidar e.a.,2013)

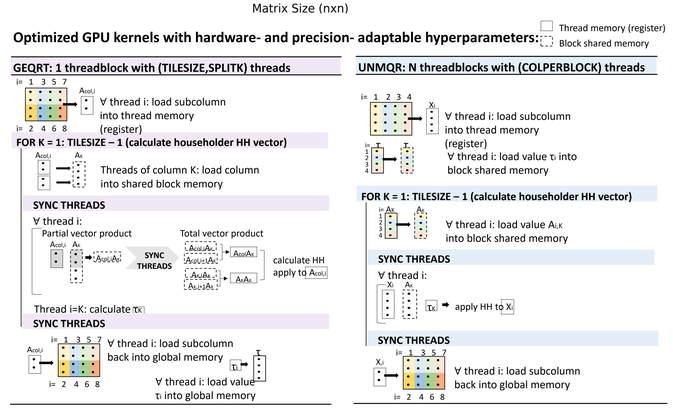


Previous work (ICPP'25): GPU-resident unified singular values with similar performance as vendor libraries



(Ringoot, e.a., 2025)

Optimized GPU kernels with hardware- and precision- adaptable hyperparameters:



Open-source code

Hardware

NVIDIA L40S/H100 — 44GB/80GB GPU, Intel Xeon Platinum 8462Y+/8562Y+

AMD MI250 — 128 GB MI250X, AMD Trento CPU

References

Jeff Bezanson, Jiahao Chen, Benjamin Chung, Stefan Karpinski, Viral B. Shah, Jan Vitek, and Lionel Zoubritzky. 2018. Julia: dynamism and performance reconciled by design. Proc. ACM Program. Lang. 2, OOPSLA, Article 120 (Oct. 2018), 23 pages.doi:10.1145/3276490

Jack Dongarra, Mark Gates, Azzam Haidar, Jakub Kurzak, Piotr Luszczek, Stanimire Tomov, and Ichitaro Yamazaki. 2018. The Singular Value Decomposition: Anatomy of Optimizing an Algorithm for Extreme Scale. SIAM Rev. 60, 4 (2018), 808–865. doi:10.1137/17M1117732

Mark Gates, Ahmad Abdelfattah, Kadir Akbudak, Mohammed Al Farhan, Rabab Alomairy, Daniel Bielich, Treece Burgess, Sébastien Cayrols, Neil Lindquist, Dalal Sukkari, et al. 2025. Evolution of the SLATE Linear Algebra Library. The International Journal of High-Performance Computing Applications 39, 1 (2025), 3–17

Khairul Kabir, Azzam Haidar, Stanimire Tomov, Aurelien Bouteiller, and Jack Dongarra. 2017. A Framework for Out of Memory SVD Algorithms. In High Performance Computing, Julian M. Kunkel, Rio Yokota, Pavan Balaji, and David Keyes (Eds.). Springer International Publishing, Cham, 158–178.

Evelyne Ringoot, Rabab Alomairy, Valentin Churavy, and Alan Edelman. 2025. Performant Unified GPU Kernels for Portable Singular Value Computation Across Hardware and Precision. (2025). arXiv:arXiv:2508.06339 doi:10.1145/3754598.3754667