

# MAGMA



FIND OUT MORE AT <https://icl.utk.edu/magma>

## MATRIX ALGEBRA ON GPU AND MULTICORE ARCHITECTURES

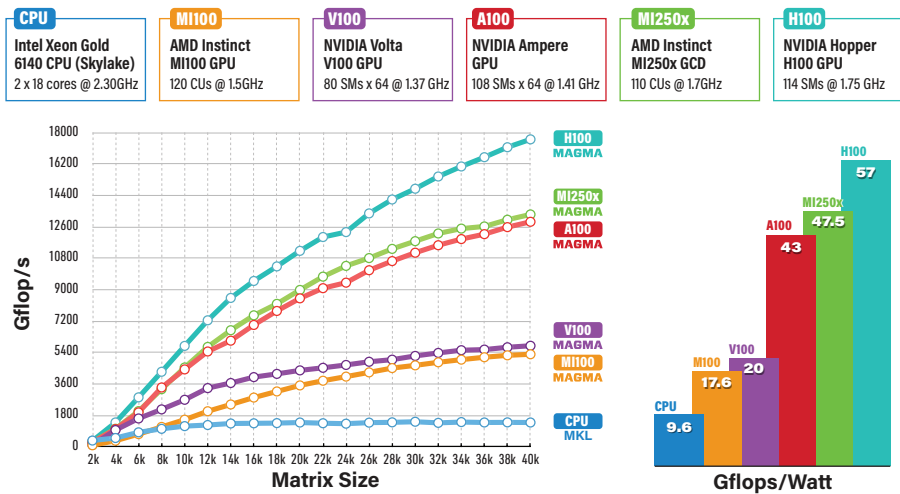
Matrix Algebra on GPU and Multicore Architectures (MAGMA) is a collection of next-generation linear algebra libraries for heterogeneous architectures. MAGMA is designed and implemented by the team that developed LAPACK and ScaLAPACK, incorporating the latest developments in hybrid synchronization-avoiding and communication-avoiding algorithms, as well as dynamic runtime systems. Interfaces for the current LAPACK and BLAS standards are supported to enable computational scientists to seamlessly port any linear algebra-reliant software components to heterogeneous architectures. MAGMA allows applications to fully exploit the power of current heterogeneous systems of multi/many-core CPUs and multiple GPUs to deliver the fastest possible time to accurate solution within given energy constraints.

### HYBRID ALGORITHMS

MAGMA uses a hybridization methodology, where algorithms of interest are split into tasks of varying granularity, and their execution is scheduled over the available hardware components. Scheduling can be static or dynamic. In either case, small non-parallelizable tasks, often on the critical path, are scheduled on the CPU, and larger more parallelizable ones, often Level-3 BLAS, are scheduled on the GPU. When CPU-GPU communication overhead becomes significant, the entire computation might run exclusively on the GPU, leading to what is termed as "GPU-native algorithms".

### PERFORMANCE & ENERGY EFFICIENCY

MAGMA LU factorization in double-precision arithmetic



### FEATURES AND SUPPORT

- ▶ **MAGMA 2.8** for **CUDA** or **HIP**
- ▶ **MAGMA Pre-release** for **SYCL/DPC++**

CUDA	HIP	SYCL Pre-release	Feature
✓	✓	✓	Linear system solvers
✓	✓	✓	Eigenvalue problem solvers
✓	✓	✓	Auxiliary BLAS
✓	✓	✓	Batched LA
✓	✓	✓	Sparse LA
✓	✓	✓	CPU/GPU Interface
✓	✓	✓	Multiple precision support
✓	✓	✓	Mixed precision (including FP16)
✓	✓	✓	Non-GPU-resident factorizations
✓	✓	✓	GPU-only factorizations
✓	✓	✓	Multicore and multi-GPU support
✓	✓	✓	MAGMA DNN v1.4
✓	✓	✓	LAPACK testing
✓	✓	✓	Linux
✓	✓	✓	Windows
✓	✓	✓	macOS

### INDUSTRY COLLABORATION



Long-term collaboration and support on the development of MAGMA.



The objective of the Innovative Computing Laboratory's IPCC is the development and optimization of numerical linear algebra libraries and technologies for applications, while tackling current challenges in heterogeneous Intel® Xeon Phi™ coprocessor-based High Performance Computing.



Long-term collaboration and support on the development of cMAGMA, the OpenCL™ port of MAGMA, and hipMAGMA.

### IN COLLABORATION WITH



### SPONSORED BY



### WITH SUPPORT FROM



# MAGMA

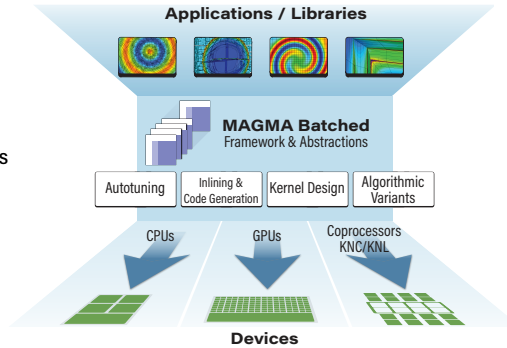


FIND OUT MORE AT <https://icl.utk.edu/magma>

## MAGMA BATCHED Batched factorization of a set of small matrices in parallel

Numerous applications require factorization of many small matrices:

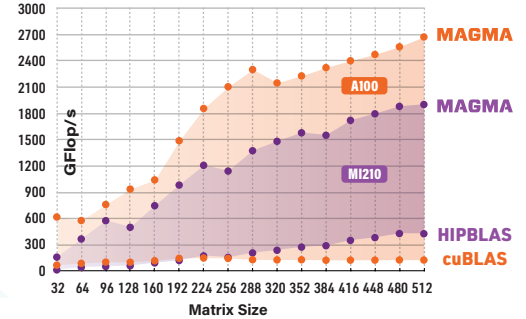
- Deep learning
- Structural mechanics
- Astrophysics
- Sparse direct solvers
- High-order FEM simulations



ROUTINES

- ✓ LU, QR, and Cholesky
- ✓ Solvers and matrix inversion
- ✓ All BLAS 3 (fixed + variable)
- ✓ SYMV, GEMV (fixed + variable)

Performance of Batch QR  
double-precision arithmetic on 1 thousand matrices



## MAGMA 2.8 DRIVER ROUTINES

	MATRIX	OPERATION	ROUTINE	INTERFACES	
				CPU	GPU
LINEAR EQUATIONS	GE	Solve using LU	{sdcz}gesv	✓	✓
		Solve using MP	{zc,ds}gesv		✓
	SPD/HPD	Solve using Cholesky	{sdcz}posv	✓	✓
		Solve using MP	{zc,ds}posv		✓
LEAST SQUARES	GE	Solve LLS using QR	{sdcz}gels		✓
		Solve using MP	{zc,ds}geqrsv		✓
STANDARD EVP	GE	Compute e-values, optionally e-vectors	{sdcz}geev	✓	
		Computes all e-values, optionally e-vectors	{sd}syevd	✓	✓
			{cz}heevd	✓	✓
	SY/HE	Range (D&C)	{cz}heevdx		✓
		Range (B&I It.)	{cz}heevx	✓	✓
		Range (MRRR)	{cz}heevr	✓	✓
STAND. SVP	GE	Compute SVD, optionally s-vectors	{sdcz}gesvd	✓	
			{sdcz}gesdd	✓	
GENERALIZED EVP	SPD/HPD	Compute all e-values, optionally e-vectors	{sd}sygvd	✓	
			{cz}hegvd	✓	
		Range (D&C)	{cz}hegvdv	✓	
	SPD/HPD	Range (B&I It.)	{cz}hegvx	✓	
Range (MRRR)		{cz}hegvr	✓		

### Abbreviations

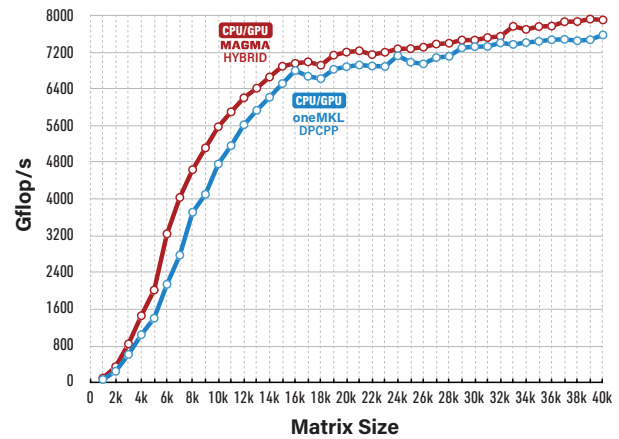
GE	General
SPD/HPD	Symmetric/Hermitian Positive Definite
TR	Triangular
D&C	Divide & Conquer
B&I It	Bisection & Inverse Iteration
MP	Mixed-Precision Iterative Refinement

### Naming Convention

magma\_{routine name}\_{gpu}

## Preliminary Intel GPU results on Aurora

Intel Xeon Max Series CPU + Intel Data Center Max Series GPU (codename PVC)



This work was done on a pre-production supercomputer with early versions of the Aurora software development kit, using Intel oneAPI release 2024.2.1. Aurora is a resource of the Argonne Leadership Computing Facility at Argonne National Laboratory.

## MAGMA SPARSE

**ROUTINES** BiCG, BiCGSTAB, Block-Asynchronous Jacobi, CG, CGS, GMRES, IDR, Iterative refinement, LOBPCG, LSQR, QMR, TFQMR

**PRECONDITIONERS** ILU / IC, Jacobi, ParILU, ParILUT, Block Jacobi, ISAI

**KERNELS** SpMV, SpMM

**DATA FORMATS** CSR, ELL, SELL-P, HYB

IN COLLABORATION WITH



ICL INNOVATIVE COMPUTING LABORATORY

