Sparse direct solvers and HPC

RTCA 2023, Mathematical Software and High Performance Algebraic Computing

MUMPS group – CERFACS, CNRS, ENS-Lyon, INRIA, INPT, Mumps Technologies, Univ. Bordeaux
(speaker: Jean-Yves L'Excellent, Mumps Technologies)

ENS-Lyon, site Monod, 26-30 June 2023

Outline

Context-main challenges

Reducing asymptotic complexity of direct methods

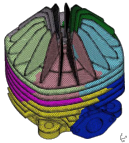
Improving the analysis phase

Computer driven activities
- Parallelism: MPI+multithreading
- Exploiting accelerators

MUMPS a free software supported by industry and academy

Outline

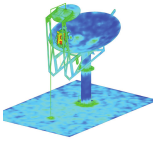
Sparse direct solvers and HPC



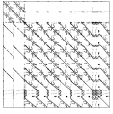
Code Aster (EDF)

Wide range of applications

(e.g. structural analysis, geoscience, electromagnetism, circuit simulation, finite element and optimization ...)



FEKO-EM (Altair)



$\Rightarrow$ Solve $\mathbf{AX} = \mathbf{B}$, with $\mathbf{A}$ a sparse matrix and $\mathbf{B}$ dense or sparse

critical step in HPC simulations

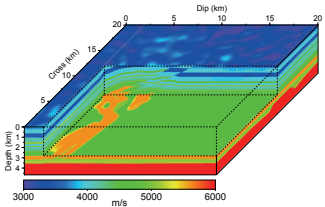
$\Rightarrow$

Sparse direct linear solvers

Factor $\mathbf{A} = \mathbf{LU}$ (or $\mathbf{LDL^T}$ if $\mathbf{A}$ symmetric) ; solve $\mathbf{LY} = \mathbf{B}$, then $\mathbf{UX} = \mathbf{Y}$

Method of choice for its accuracy and robustness

Example from geophysics: Full Waveform Inversion (FWI)



3D EAGE/SEG overthrust model

(credits: SEISCOPE project)

$\Longrightarrow$

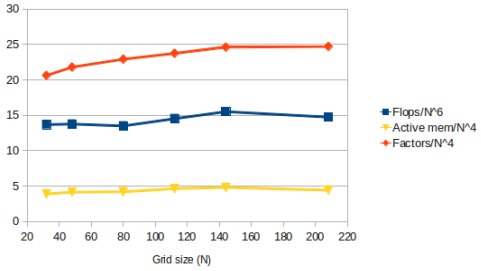
Frequency domain FWI
Helmholtz equations $\Rightarrow$ $\mathbf{AX} = \mathbf{B}$

- **A:** complex unsymmetric sparse matrix
- **B:** multiple (very) sparse right-hand sides
- **required accuracy** $\approx 10^{-4}$

frequency	flops LU	Factor Storage	Peak memory
2 Hz	9.0×10^{11}	3 GB	4 GB
4 Hz	1.6×10^{13}	22 GB	25 GB
8 Hz	5.8×10^{14}	247 GB	283 GB
10 Hz	2.7×10^{15}	728 GB	984 GB

Higher frequency leads to refined model,
non linear increase of flops/memory

Complexity analysis of sparse direct methods

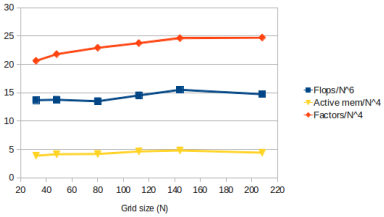


3D example in earth science:
acoustic wave propagation, 27-point finite difference

Extrapolation $n = N^3 = 1000^3$ grid:
55 exaflops, 200 TBytes for factors,
40 TBytes of working memory!

Regular problems (nested dissections)	2D $N \times N$ grid	3D $N \times N \times N$ grid
Nonzeros in original matrix	$\Theta(N^2)$	$\Theta(N^3)$
Nonzeros in factors	$\Theta(N^2 log\ n)$	$\Theta(N^4)$
Floating-point ops	$\Theta(N^3)$	$\Theta(N^6)$

Cost analysis



3D example in earth science:

Extrapolation $N^3 = 1000^3$ grid:
55 exaflops, 200 TBytes for factors,
40 TBytes of working memory!

Computer related data

- Peak perf.: 2 Pflops/s (power: 1 MW)
- 0.2€/kW.h

Cost of ONE factorization assuming 10% efficiency

- $\sim$ 76 h 10% $\times$ ((55 exaflops / 2 Pflops/s) / 3600)
- 76 MW.h (15200€ for energy)

Critical: energy consumption, parallel efficiency, flops and memory complexity

Challenges for sparse direct solvers

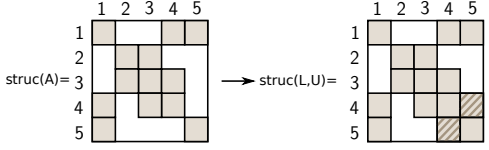
- Complexity of direct methods (flops, memory, energy) in the context of increasingly large applications
- Efficiency and robustness in the context of heterogeneous computers (numerical and algorithmic issues)
- Controlled precision and mixed-precision arithmetic

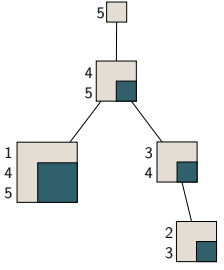
Heterogeneity within application, hardware (processor, memory), arithmetic
$\Rightarrow$ great algorithmic challenges... but also great **opportunities**

Solution of $\mathbf{AX} = \mathbf{B}$ performed in 3 phases:

($\mathbf{A}$ $n \times n$ sparse matrix with NZ non-zeros)

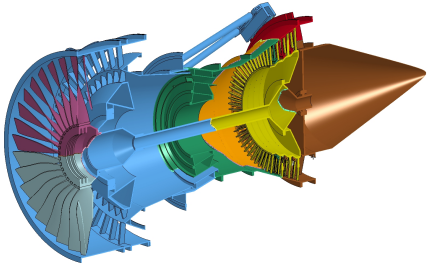
1. **analysis**, on the graph of $\mathbf{A}$
   - build ordering (METIS, SCOTCH, parMETIS, pt-SCOTCH, ...)
   - prepare factorization, symbolic factorization, build elimination tree



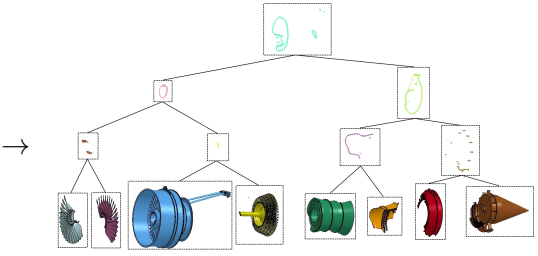


2. **numerical factorization**, decompose $\mathbf{A} = \mathbf{LU}$
   - work on dense matrices following elimination tree
   - stability relies on numerical pivoting
3. **solve**, forward and backward substitutions $\mathbf{LY} = \mathbf{X}$, $\mathbf{UX} = \mathbf{Y}$

Illustration on a real application



Eight-domain partitionning



Tree of separators

Courtesy of LSTC (Livermore Software Technology Corp.) and Rolls-Royce

Outline

Block Low-Rank general context and main features

- Approximate factorization $\mathbf{A} \approx \mathbf{L}_\varepsilon \mathbf{U}_\varepsilon$ at accuracy ε **controlled by the user**

$\Rightarrow$ Many representations: Recursive $\mathcal{H}, \mathcal{H}^2$, HSS, HODLR, BLR ...

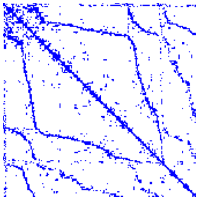
Block Low-Rank[1] (BLR)

- Flat and simple format
  - Robust algebraic solver, compatible with the numerical features of a general solver
  - Backward stability proved by Higham and Mary, IMA J. Num. Ana. (2021)
- Reduced complexity, e.g., for 3D Helmholtz $n = N^3$ 3D mesh
  - Operations during facto. : Full-Rank, $\mathcal{O}\left(N^6\right) \rightarrow$ BLR, $\mathcal{O}\left(N^5\right)$
  - Size of LU factors : Full-Rank, $\mathcal{O}\left(N^4\right) \rightarrow$ BLR, $\mathcal{O}\left(N^{3.5}\right)$

Work supported by PhD theses: C. Weisbecker (2010-2013, EDF grant) T. Mary (2014-2017), B. Vieublé (2019-2022) and M. Gerest (2020-, EDF grant)

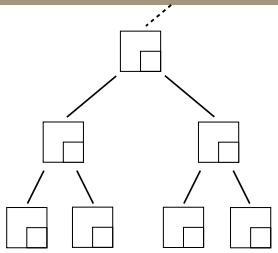
[1] *See publications in SIAM J. Sci. Comput. or ACM Trans. Math. Soft.: "Improving multifrontal methods by means of block low-rank representations" (2015), " On the complexity of the Block Low-Rank multifrontal factorization" (2017), "Bridging the gap between flat and hierarchical low-rank matrix formats: the multilevel BLR format" (2019), "Performance and Scalability of the Block Low-Rank Multifrontal Factorization on Multicore Architectures" (2018)*

Block Low-Rank (BLR) multifrontal factorization: principle



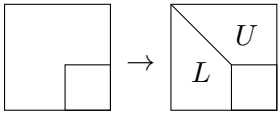
Sparse matrix A

$\Rightarrow$

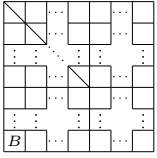


Elimination tree

$\Rightarrow$

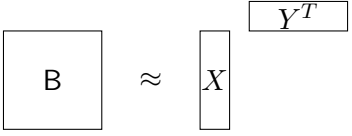


At each node (partial factorization)



Frontal matrix

$\Rightarrow$



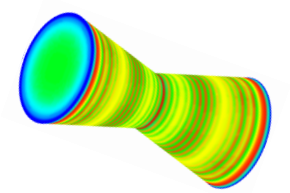
Truncated SVD/RRQR on B

$\Rightarrow \|B - XY^T\| \leq \epsilon \|A\|$

Reduced storage and computations

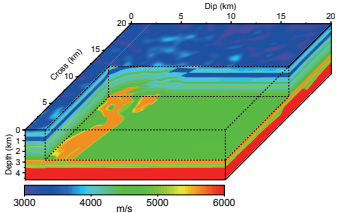
Block Low-Rank: flops reduction versus time performance

Required accuracy: 10^{-9}



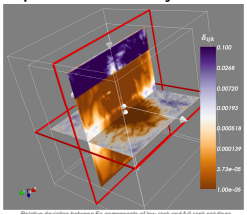
Structural mechanics, $n = 8M$
Flop Ratio=17

Required accuracy: 10^{-3}



Seismic imaging, $n = 17M$
Flop Ratio=27

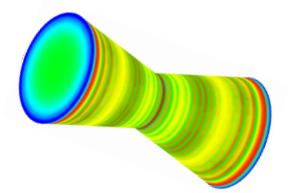
Required accuracy: 10^{-7}



Electromagnetism, $n = 21M$
Flop Ratio=65

Block Low-Rank: flops reduction versus time performance

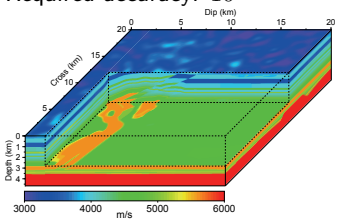
Required accuracy: 10^{-9}



Structural mechanics, $n = 8M$
Flop Ratio=17

→ **Time Ratio= 6**

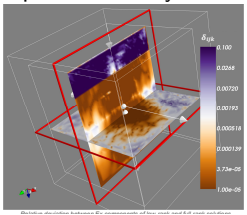
Required accuracy: 10^{-3}



Seismic imaging, $n = 17M$
Flop Ratio=27

→ **Time Ratio= 7**

Required accuracy: 10^{-7}



Electromagnetism, $n = 21M$
Flop Ratio=65

→ **Time Ratio=19**

Converting flop reduction into performance gains is not straightforward[a]

[a] *Amestoy, Buttari, L'Excellent, Mary, Performance and Scalability of the BLR Multifrontal Factorization on Multicores, ACM Trans. on Math. Soft. 2018*

Low precision arithmetics

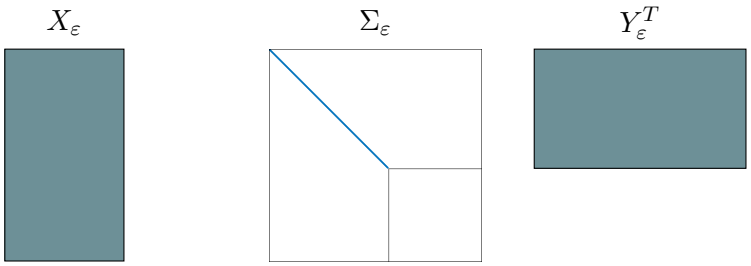
	Signif. bits (t)	Exp.	Rang	$u = 2^{-t}$
fp64	53	11	$10^{\pm 308}$	1×10^{-16}
fp32	24	8	$10^{\pm 38}$	6×10^{-8}
fp16	11	5	$10^{\pm 5}$	5×10^{-4}
bfloat16	8	8	$10^{\pm 38}$	4×10^{-3}
fp8 (e4m3)	4	4	$10^{\pm 2}$	6×10^{-2}
fp8 (e5m2)	3	5	$10^{\pm 5}$	1×10^{-1}

Opportunities to:

- Reduce storage, data movement, and communication
- Increase speed and reduce energy
- However, reduced range and accuracy: low precision $\equiv$ low accuracy

$\rightarrow$ Motivation to use mixed precision algorithms

Mixed precision low-Rank approximation: algorithm[2]

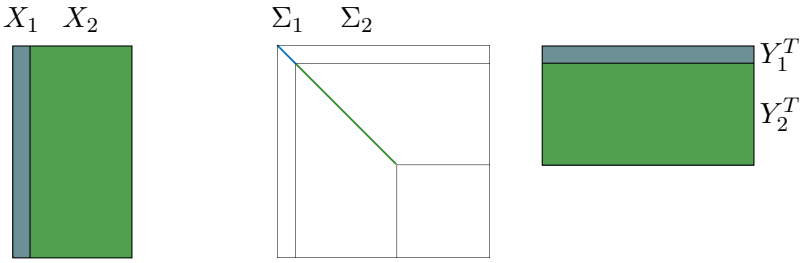


Truncated SVD

- $B \approx \sum_{i=1}^{r} x_i \sigma_i y_i^T$, with r such that
- $\|B - X_\varepsilon \Sigma_\varepsilon Y_\varepsilon^T\| \leq \varepsilon \|A\|$

[2]work supported by EDF

Mixed precision low-Rank approximation: algorithm[3]



Truncated SVD with 2-precision formats (fp64, fp32)

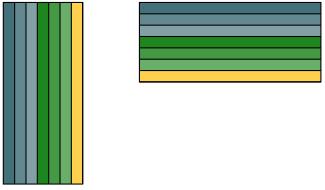
- Can convert X_2 and Y_2 to single precision
- Criterion for storing columns x_i, y_i in precision fp32: $\sigma_i \leq \frac{\varepsilon}{u_s}\|A\|$, with $u_s = 6 \times 10^{-8}$
- $\|B - X\Sigma Y^T\| \leq 3\varepsilon\|A\| \Rightarrow$ preserved accuracy[2]

[2] see Amestoy et al. 2022, IMA JNA https://doi.org/10.1093/imanum/drac037

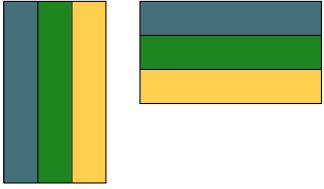
[3] work supported by EDF

Mixed BLR: dissociate storage and compute precisions

Exploiting precisions for computations other than fp64 and fp32 is hardware dependent **but** mathematical theory applies to any number of precisions



Storage precisions:
large number, arbitrary format



Compute precisions:
small number, available in hardware

- Switch between storage and compute representations only once at factorization and solve
- Targeting storage gains:
  - 64-bit computations: 7 precisions for storage: 16, 24, 32, 40, 48, 56, 64 bits
  - 32-bit computations: 3 precisions for storage: 16, 24, 32 bits

Mixed precision low-rank approximation: illustration

thmgaz matrix, provided by EDF
($n = 8M$, 2 MPI $\times$ 18 threads, $\epsilon_{BLR} = 10^{-10}$)

	Factor size(GB)	Total memory(GB)	Factorization time(s)	Solve time(s)	Backward error
BLR double	95	131	85	0.45	6×10^{-14}
BLR mixed	59	105	93	0.35	6×10^{-14}

$\Rightarrow$ Memory and solve time reductions, with preserved accuracy

... ongoing work: use mixed precision to accelerate factorization

Outline

Analysis phase

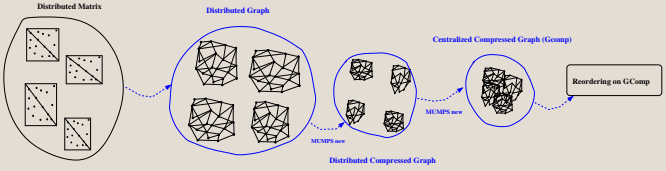
Cost of analysis can be significant with respect to numerical phases

- numerical phases benefit from low rank compression and are highly parallelized
- parallel ordering techniques (parmetis, pt-scotch) sometimes lead to lower quality orderings

Motivated by properties of applications, we illustrate here recent work done on using graph compression during all steps of the analysis phase (ordering, symbolic factorization and BLR clustering)

Objective: Exploit block structure of sparse matrices to reduce time and memory footprint of analysis phase
(k dofs per block $\Rightarrow$ k (k^2) less vertices (edges) to handle!)

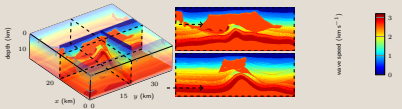
Block format $\rightarrow$ compressed graph G_{comp}



Reduce time+memory: $NZ_G \rightarrow NZ_{Gcomp} = \frac{1}{k^2} NZ_G$ (for k dofs per block)

Analysis by blocks, performance analysis

- **Time-harmonic wave problems** using Hybridizable Discontinuous Galerkin discretization [4] and p-adaptivity HDG Matrix 65k-2hz, general symmetric of order $n = 10M$, $NZ_{G(A+A^T)} = 5\ 566M$, factorization on 360 cores (Olympe)



Elastic SEAM model, S-wave speed model. At 2Hz, the wavelength is between 300 and 1500 m.

- **Application in structural mechanics**, Matrix EngineAssy5M_32 unsymmetric of order $n = 4.6M$, $NZ_{G(A+A^T)} = 94M$, factorization on 64 cores (Olympe)

	HDG Matrix 65k-2hz	EngineAssy5M_32
	Analysis by blocks	
Time (sec) for	OFF	OFF
Analysis	337	90
Factorization	98	38

[4] M. Bonnasse-Gahot, H. Calandra, J. Diaz, and S. Lanteri, Hybridizable discontinuous Galerkin method for the 2D frequency-domain elastic wave equations, Geophysical Journal International, 213 (2017), pp. 637-659.

Analysis by blocks, performance analysis

- **Time-harmonic wave problems** using Hybridizable Discontinuous Galerkin discretization and p-adaptivity HDG Matrix 65k-2hz, general symmetric of order $n = 10M$, $NZ_{G(A+A^T)} = 5\ 566M$, factorization on 360 cores (Olympe)

  dofs on cell faces have identical adjacencies and form blocks in the matrix

$$n/n_{comp} = 24 \longrightarrow NZ_G/NZ_{Gcomp} = 699 > 24^2$$

- **Application in structural mechanics**, Matrix EngineAssy5M_32 unsymmetric of order $n = 4.6M$, $NZ_{G(A+A^T)} = 94M$, factorization on 64 cores (Olympe)

  average number of dofs per grid point is 3

$$n/n_{comp} = 3 \qquad \longrightarrow \qquad NZ_G/NZ_{Gcomp} = 3^2$$

	HDG Matrix 65k-2hz	EngineAssy5M_32
	Analysis by blocks	
Time (sec) for	OFF	OFF
Analysis	337	90
Factorization	98	38

Analysis by blocks, performance analysis

- **Time-harmonic wave problems** using Hybridizable Discontinuous Galerkin discretization and p-adaptivity HDG Matrix 65k-2hz, general symmetric of order $n = 10M$, $NZ_{G(A+A^T)} = 5\ 566M$, factorization on 360 cores (Olympe)

  dofs on cell faces have identical adjacencies and form blocks in the matrix

$$n/n_{comp} = 24 \longrightarrow NZ_G/NZ_{Gcomp} = 699 > 24^2$$

- **Application in structural mechanics**, Matrix EngineAssy5M_32 unsymmetric of order $n = 4.6M$, $NZ_{G(A+A^T)} = 94M$, factorization on 64 cores (Olympe)

  average number of dofs per grid point is 3

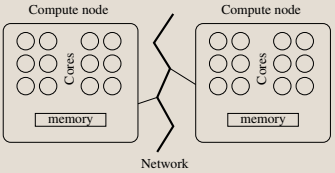
$$n/n_{comp} = 3 \qquad \longrightarrow \qquad NZ_G/NZ_{Gcomp} = 3^2$$

	HDG Matrix 65k-2hz		EngineAssy5M_32	
	Analysis by blocks			
Time (sec) for	OFF	ON	OFF	ON
Analysis	337	**7**	90	**44**
Factorization	98	98	38	38

Outline

Table of Contents

Managing parallelism: multicores



Many cores sharing memory per compute node

Hybrid parallelization

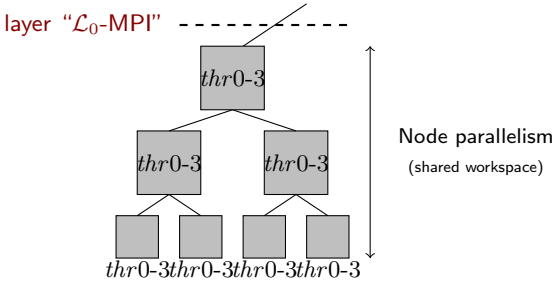
- Distributed memory parallelism (MPI based)

  combined to shared-memory parallelism (multithreading):
  - use of multithreaded BLAS
  - OpenMP directives

Nb of cores per node increases → more multithreading needs be exposed

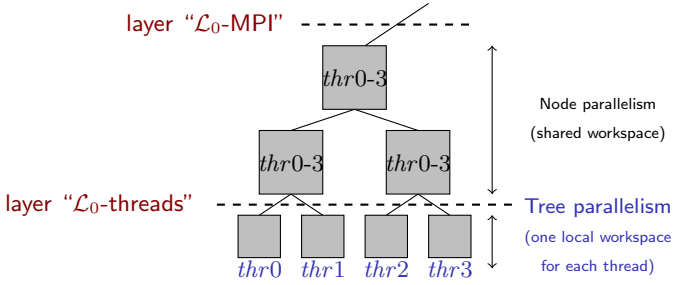
Strategy for hybrid parallelization (case of multiple threads per MPI process):

- parallelism **between nodes** of the elimination tree (MPI only)
- parallelism **within nodes** (MPI and OpenMP)
- under "$\mathcal{L}_0$-MPI": **one** MPI process per subtree (to limit communication)

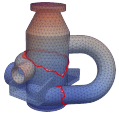


Strategy for hybrid parallelization (case of multiple threads per MPI process):
(work based on W. M. Sid-Lakhdar's PhD thesis, 2014)

- parallelism **between nodes** of the elimination tree (MPI and **OpenMP**)
- parallelism **within nodes** (MPI and OpenMP)
- under "$\mathcal{L}_0$-MPI": **one** MPI process per subtree (to limit communication)
- under "$\mathcal{L}_0$-threads": **one** thread per subtree



$\mathcal{L}_0$-threads feature: impact on BLR and solve phase



Code_Aster RIS pump

Matrix from structural mechanics, real, symmetric:
perf009ar from EDF

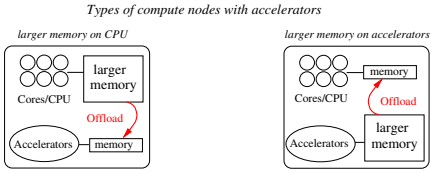
n=5.4M, NZ=208M, flops(LDL^T)= 1.8×10^{13}

pivots: 79k delayed, 37k 2×2, 74k negative

	Time on 36 cores (s)		Memory allocated
	Factorization	Solve (1 RHS)	(GB)
$\mathcal{L}_0$-**threads OFF**	(36 MPI × 1 thread)		
BLR ($\varepsilon_{blr}= 10^{-9}$)	**22.4**	**0.41**	**72**
	(2 MPI × 18 threads)		
BLR ($\varepsilon_{blr}= 10^{-9}$)	**41.3**	**2.58**	**36**
$\mathcal{L}_0$-**thread ON**	(2 MPI × 18 threads)		
BLR ($\varepsilon_{blr}= 10^{-9}$)	**18.7**	**0.50**	**39**

Table of Contents

Exploiting accelerators



Collaborations

- Larger memory on CPU: offload from CPU to GPU (support of Altair), use runtime libraries for BLAS on GPU:
  - cublasXt: provided by Nvidia
  - `XKBlas`: collaboration with T. Gautier (Inria-ENS Lyon), also supports AMD GPU
- Larger memory on accelerator
  - NEC SX-Aurora vector processor (collaboration with NEC): offload scalar parts from vector engine to CPU
  - OpenMP 5.0 approach: target new supercomputer nodes from French national center[4]: 512 GB on four AMD MI250X GPU, 256 GB on CPU

[4] collaboration with GENCI, CINES, HPE and AMD

XKBlas evolutions (T. Gautier, Inria LIP-ENS Lyon)

- Improve performance and robustness, reduce memory transfers
  - chain GPU operations,
  - new kernels (GEMMT, copy-scale algorithm under discussion)
- Portability on AMD GPUs: XKBlas for AMD GPUs

  (available on the GitLab of XKBlas: xkblas-v0.4-rc6-1-gfe1cb265)

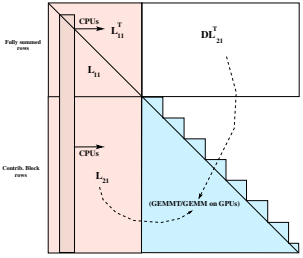
  Preliminary results on MUMPS from T. Gautier:

	LDLT MechaStruct8M		LU 3D Laplacian	
	AMD MI50	V100	AMD MI50	V100
(*)0 GPU	796s	780s	719	749
1 GPU	367s	358s	298	371
2 GPU	292s	295s	240	266

(*) 36 cores, AMD or Intel

- Ongoing: test and tune XKBlas on recent AMD GPU (MI250X) in the context of GENCI-CINES-HPE-AMD collaboration

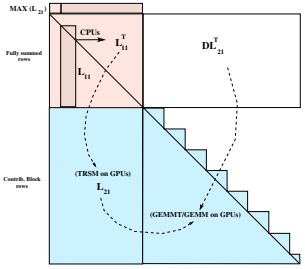
Symmetric indefinite matrices on GPU: influence of numerical pivoting



- Numerical pivoting CNTL(1)=0.01
  - Less BLAS-3 kernels than with CNTL(1)=0.0
  - BLAS-2 update of $\mathbf{L}_{21}$:
    - performance issue on GPUs but also on multicores

CALMIP Olympe computer, time for factorization in seconds 3D Wave equations, 18 cores, XKBlas	
	Factorization time (sec)
Numerical pivoting OFF (CNTL(1)=0)	
18 cores	382
18 cores + 1 GPU	**201**
Numerical pivoting ON (CNTL(1)=0.01)	
18 cores	432
18 cores + 1 GPU	**352**

Symmetric indefinite matrices on GPU: influence of numerical pivoting



- Numerical pivoting CNTL(1)=0.01
  - Less BLAS-3 kernels than with CNTL(1)=0.0
  - BLAS-2 update of $\mathbf{L}_{21}$:
    - performance issue on GPUs but also on multicores
- Relaxed pivoting with CNTL(1)=0.01
  - Vector of column norms used to represent block $\mathbf{L}_{21}$ see Duff, Pralet SIAM SISC (2007)
  - Enable BLAS-3 update of $\mathbf{L}_{21}$:

CALMIP Olympe computer, time for factorization in seconds 3D Wave equations, 18 cores, XKBlas	
	Factorization time (sec)
Numerical pivoting OFF (CNTL(1)=0)	
18 cores	382
18 cores + 1 GPU	**201**
Numerical pivoting ON (CNTL(1)=0.01)	+ Relaxed pivoting
18 cores	~~432~~ → 379
18 cores + 1 GPU	~~352~~ → **216**

Tuning MUMPS for NEC vector engine

Experimental environment

- NEC SX-Aurora 10B, 8 cores, 2.15 Tflops/s peak, 48 GBytes
- 3D frequency-domain FWI, 27-point stencil (Geoazur – S. Operto)

Performance of tuned version for VE (since MUMPS 5.5)

Matrix	N (1E6)	complex flops	factors GBytes	1MPIx8threads Gflops/s	1MPIx8threads LU factorization time
$Geo115^3$	1.52	3.1E13	32.2	717(*)	44 s

(*)corresponds to **2.8 Tflops/s** in single precision, real arithmetic (peak = 4.3 Tflops/s).

Tuning MUMPS for NEC vector engine

Experimental environment

- NEC SX-Aurora 10B, 8 cores, 2.15 Tflops/s peak, 48 GBytes
- 3D frequency-domain FWI, 27-point stencil (Geoazur – S. Operto)

Performance of tuned version for VE (since MUMPS 5.5)

Matrix	N (1E6)	complex flops	factors GBytes	1MPIx8threads Gflops/s	1MPIx8threads LU factorization time
$Geo115^3$	1.52	3.1E13	32.2	717(*)	44 s

(*)corresponds to **2.8 Tflops/s** in single precision, real arithmetic (peak = 4.3 Tflops/s).

Block Low-Rank compression ($\epsilon = 10^{-4}$) on NEC VE (since MUMPS 5.6)

NEC SX-Aurora VE	Memory used (GBytes)		LU factorization time	
1MPIx8threads	Full-Rank	Block Low-Rank	Full-Rank	Block Low-Rank
10B 1.4 GHz	42	38	44 s	39 s
20B 1.6 GHz	42	38	38 s	**27 s**

Outline

MUMPS: brief history of a free software

- Multifrontal approach: Schreiber'82; Duff, Reid'83
- 1996-1999: MUMPS started in Toulouse (EU LTR project PARASOL) inspired from a shared memory research code
- 2000: First "public domain" version of MUMPS
- 2014-2019: Consortium of MUMPS users
  Founding members: *CERFACS, INPT, Inria, ENS-Lyon, Bordeaux University*; Members: EDF, Altair, Michelin, LSTC (USA), SISW-Siemens (Belgium), ESI, Total, FFT/MSC Soft. (Belgium), SAFRAN, Lawrence Berkeley Nat. Lab. (USA)
- 2015: MUMPS 5.0.0, first CeCILL-C version of MUMPS
- 2019: Creation of Mumps Technologies SAS to ensure software sustainability and development of MUMPS solver – `http://mumps-tech.com`
- 2023: Fifth edition of the MUMPS Users Days

The MUMPS solver – http://mumps-solver.org

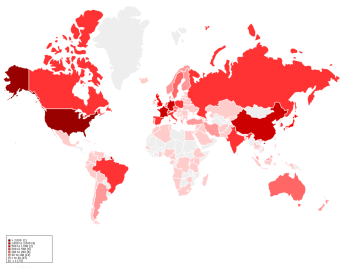
- Free software package, state-of-the-art in its field (fed by the research – 13 theses)
- Co-developed in *Toulouse-Lyon-Paris-Bordeaux* by *CERFACS, CNRS, ENS Lyon, INPT, Inria, Bordeaux Univ.* and, since 2019, *Mumps Technologies*

Users: developers of simulation software

- Used worldwide by industrials/academics
- Part of commercial and open-source packages
- User community (3 explicit software requests/day)
  - Users days every 3-4 years
  - $\simeq$ 800 users emails per year
  - mumps-users mailing list: $\simeq$ 600 subscribers

- Latest release: MUMPS 5.6.1 July 2023, $\approx$ 250 000 lines of C and Fortran code
- License: CeCILL-C

Map of the download requests



Partners supporting the MUMPS solver (Gold subscription to Mumps Technologies):

Summary of recent activities and collaborations

- Mixed precision:
  - PhD thesis of Bastien Vieublé (2019-2022), IRIT (Univ. of Toulouse) on mixed precision iterative refinement (see, *Combining sparse approximate factorizations with mixed precision iterative refinement*, Amestoy, Buttari, Higham, L'Excellent, Mary and Vieublé. *ACM Trans. on Math. Software* [2022])
  - PhD thesis of Matthieu Gerest (2020-) EDF-LIP6 on mixed precision BLR factorization (see, *Mixed precision low-rank approximations and their application to block low-rank LU factorization*, Amestoy, Boiteau, Buttari, Gerest, Jézéquel, L'Excellent, Mary, *IMA J. of Num. Anal. [2023]*)
- Computer driven activities:
  - Compute nodes with GPUs
    - Data on CPU (offload BLAS kernels, with Altair and Inria-LIP-ENS Lyon)
    - Data on GPU (OpenMP 5 related, with CINES and HPE)
  - NEC vector processors (collaboration with NEC)
- Application driven collaborations:
  - PhD thesis of Sébastien Dubois (2022-). Parallel algebraic iterative linear solvers, in collaboration with ONERA (France) and LIP6 (Sorbonne University),
  - Rank-Revealing feature, collaboration with SAFRAN
  - HDG matrices and BLR mixed precision (with TotalEnergies and Inria Makutu)
  - Seismic Imaging by Waveform Inversion with WIND project (see, *Is 3D frequency-domain FWI of full-azimuth/long-offset OBN data feasible? The Gorgon-data FWI case study*, Operto, Amestoy, Aghamiry, Beller, Buttari, Combe, Dolean, Gerest, Guo, Jolivet, L'Excellent, Mamfoumbi, Mary, Puglisi, Ribodetti, Tournier, *The Leading Edge* [2023])

Thank you!

Experimental environment

- CALMIP center of Toulouse (grant number P0989):

  `Olympe nodes`
  - CPU node: Two Intel 18-cores Skylake 6140 @2.3 GHz (Peak/core=73.6 GF/s, Peak/node=2.6 TFlops/s DP), 192 GB memory per node
  - GPU node: Two Intel 18-cores Skylake 6140 @2.3 GHz (Peak/core=73.6 GF/s, Peak/node=2.6 TFlops/s DP), 384 GB memory per node, 4 GP-GPU Nvidia Volta (V100 - 7.8 TFlops/s DP)

  `TURPAN from MésoNET project, experimental computer: 15 nodes with`
  - Ampere Altra Max Q80-30 (ARM version 8.2) 80 cores @3 GHz (peak 24Gflops/s/core), peak 1.9 Tflops/s DP, 512 GB memory
  - 2 GPU Nvidia A100-80, 19.5 Tflops/s DP per GPU, 2x80 GB memory

- GENCI-CINES, ADASTRA supercomputer: HPE Cray EX235a
  - 61.6 PFlops/s peak, 46 PFlops/s (Linpack); 50 GFlops/Watt
  - accelerated nodes based on AMD Optimized 3rd Generation EPYC 64C 2.4 GHz, 512 GB on four AMD Instinct MI250X GPU, 256 GB on CPU

- NEC SX-Aurora Type 10B with 8 cores, 1.4GHz, 2.15 TFlops/s peak, 48 GB and NEC SX-Aurora Type 20B, 1.6GHz